            Subject Identifiers for Security Event Tokens
              draft-ietf-secevent-subject-identifiers-07

Abstract

   Security events communicated within Security Event Tokens may support
   a variety of identifiers to identify subjects related to the event.
   This specification formalizes the notion of subject identifiers as
   structured information that describe a subject, and named formats
   that define the syntax and semantics for encoding subject identifiers
   as JSON objects. It also defines a registry for defining and
   allocating names for such formats, as well as the "sub_id" JSON Web
   Token (JWT) claim.

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 September 9, 2021.

to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

   As described in Section 1.2 of SET [RFC8417], subjects related to
   security events may take a variety of forms, including but not
   limited to a JWT [RFC7519] principal, an IP address, a URL, etc.

Different types of subjects may need to be identified in different
ways. (e.g., a host might be identified by an IP or MAC address,
while a user might be identified by an email address) Furthermore,
even in the case where the type of the subject is known, there may be
multiple ways by which a given subject may be identified. For
example, an account may be identified by an opaque identifier, an
email address, a phone number, a JWT "iss" claim and "sub" claim,
etc., depending on the nature and needs of the transmitter and
receiver. Even within the context of a given transmitter and
receiver relationship, it may be appropriate to identify different
accounts in different ways, for example if some accounts only have
email addresses associated with them while others only have phone
numbers. Therefore it can be necessary to indicate within a SET the
mechanism by which a subject is being identified.

To address this problem, this specification defines Subject
Identifiers - JSON [RFC7159] objects containing information
identifying a subject - and Identifier Formats - named sets of rules
describing how to encode different kinds of subject identifying
information (e.g., an email address, or an issuer and subject pair)
as a Subject Identifier.

Below is a non-normative example of a Subject Identifier that
identifies a subject by email address, using the Email Identifier
Format.

```
{
  "format": "email",
  "email": "user@example.com",
}
```

   Figure 1: Example: Subject Identifier using the Email Identifier
                                 Format

Subject Identifiers are intended to be a general purpose mechanism
for identifying subjects within JSON objects and their usage need not
be limited to SETs. Below is a non-normative example of a JWT that
uses a Subject Identifier in the "sub_id" claim (defined in this
specification) to identify the JWT Subject.

```
   {
     "iss": "issuer.example.com",
     "sub_id": {
       "format": "phone_number",
       "phone_number": "+12065550100",
     },
   }
```

Figure 2: Example: JWT using a Subject Identifier with the sub_id
claim

Usage of Subject Identifiers also need not be limited to identifying
JWT Subjects. They are intended as a general purpose means of
expressing identifying information in an unambiguous manner. Below
is a non-normative example of a SET containing a hypothetical
security event describing the interception of a message, using
Subject Identifiers to identify the sender, intended recipient, and
interceptor.

```
   {
     "iss": "issuer.example.com",
     "iat": 1508184845,
     "aud": "aud.example.com",
     "events": {
       "https://secevent.example.com/events/message-interception": {
         "from": {
           "format": "email",
           "email": "alice@example.com",
         },
         "to": {
           "format": "email",
           "email": "bob@example.com",
         },
         "interceptor": {
           "format": "email",
           "email": "eve@example.com",
         },
       },
     },
   }
```

Figure 3: Example: SET with an event payload containing multiple
Subject Identifiers

2. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

2.1. Definitions

This specification utilizes terminology defined in [RFC7159],
[RFC7519], and [RFC8417].

Within this specification, the terms "Subject" and "subject" refer
generically to anything being identified via one or more pieces of
information. The term "JWT Subject" refers specifically to the to
the subject of a JWT. (i.e., the subject that the JWT asserts claims
about)

3. Subject Identifiers

A Subject Identifier is a JSON [RFC7159] object whose contents may be
used to identify a subject within some context. An Identifier Format
is a named definition of a set of information that may be used to
identify a subject, and the rules for encoding that information as a
Subject Identifier; they define the syntax and semantics of Subject
Identifiers. A Subject Identifier MUST conform to a specific
Identifier Format, and MUST contain a "format" member whose value is
the name of that Identifier Format.

Every Identifier Format MUST have a unique name registered in the
IANA "Security Event Identifier Formats" registry established by
Section 8.1, or a Collision-Resistant Name as defined in [RFC7519].
Identifier Formats that are expected to be used broadly by a variety
of parties SHOULD be registered in the "Security Event Identifier
Formats" registry.

An Identifier Format MAY describe more members than are strictly
necessary to identify a subject, and MAY describe conditions under
which those members are required, optional, or prohibited. The
"format" member is reserved for use as described in this
specification; Identifier Formats MUST NOT declare any rules
regarding the "format" member.

Every member within a Subject Identifier MUST match the rules
specified for that member by this specification or by Subject
Identifier's Identifier Format. A Subject Identifier MUST NOT
contain any members prohibited or not described by its Identifier
Format, and MUST contain all members required by its Identifier
Format.

3.1. Identifier Formats versus Principal Types

   Identifier Formats define how to encode identifying information for a
   subject. They do not define the type or nature of the subject
   itself. E.g., While the "email" Identifier Format declares that the
   value of the "email" member is an email address, a subject in a
   Security Event that is identified by an "email" Subject Identifier
   could be an end user who controls that email address, the mailbox
   itself, or anything else that the transmitter and receiver both
   understand to be associated with that email address. Consequently
   Subject Identifiers remove ambiguity around how a subject is being
   identified, and how to parse an identifying structure, but do not
   remove ambiguity around how to resolve that identifier to a subject.
   For example, consider a directory management API that allows callers
   to identify users and groups through both opaque unique identifiers
   and email addresses. Such an API could use Subject Identifiers to
   disambiguate between which of these two types of identifiers is in
   use. However, the API would have to determine whether the subject is
   a user or group via some other means, such as by querying a database,
   interpreting other parameters in the request, or inferring the type
   from the API contract.

3.2. Identifier Format Definitions

   The following Identifier Formats are registered in the IANA "Security
   Event Identifier Formats" registry established by Section 8.1.

3.2.1. Account Identifier Format

   The Account Identifier Format identifies a subject using an account
   at a service provider, identified with an "acct" URI as defined in
   [RFC7565]. Subject Identifiers in this format MUST contain a "uri"
   member whose value is the "acct" URI for the subject. The "uri"
   member is REQUIRED and MUST NOT be null or empty. The Account
   Identifier Format is identified by the name "account".

   Below is a non-normative example Subject Identifier for the Account
   Identifier Format:

```
{
  "format": "account",
  "uri": "acct:example.user@service.example.com",
}
```

      Figure 4: Example: Subject Identifier for the Account Identifier
                                 Format

3.2.2. Email Identifier Format

The Email Identifier Format identifies a subject using an email
address. Subject Identifiers in this format MUST contain an "email"
member whose value is a string containing the email address of the
subject, formatted as an "addr-spec" as defined in Section 3.4.1 of
[RFC5322]. The "email" member is REQUIRED and MUST NOT be null or
empty. The value of the "email" member SHOULD identify a mailbox to
which email may be delivered, in accordance with [RFC5321]. The
Email Identifier Format is identified by the name "email".

Below is a non-normative example Subject Identifier in the Email
Identifier Format:

```
{
  "format": "email",
  "email": "user@example.com",
}
```

Figure 5: Example: Subject Identifier in the Email Identifier Format

3.2.2.1. Email Canonicalization

Many email providers will treat multiple email addresses as
equivalent. While the domain portion of an [RFC5322] email address
is consistently treated as case-insensitive per [RFC1034], some
providers treat the local part of the email address as case-
insensitive as well, and consider "user@example.com",
"User@example.com", and "USER@example.com" as the same email address.
This has led users to view these strings as equivalent, driving
service providers to implement proprietary email canonicalization
algorithms to ensure that email addresses entered by users resolve to
the same canonical string. When receiving an Email Subject
Identifier, the recipient SHOULD use their implementation's
canonicalization algorithm to resolve the email address to the same
string used in their system.

3.2.3. Phone Number Identifier Format

The Phone Number Identifier Format identifies a subject using a
telephone number. Subject Identifiers in this format MUST contain a
"phone_number" member whose value is a string containing the full
telephone number of the subject, including international dialing
prefix, formatted according to E.164 [E164]. The "phone_number"
member is REQUIRED and MUST NOT be null or empty. The Phone Number
Identifier Format is identified by the name "phone_number".

Below is a non-normative example Subject Identifier in the Email
Identifier Format:

```
{
  "format": "phone_number",
  "phone_number": "+12065550100",
}
```

Figure 6: Example: Subject Identifier in the Phone Number Identifier
Format.

[3.2.4](#). Issuer and Subject Identifier Format

The Issuer and Subject Identifier Format identifies a subject using a
pair of "iss" and "sub" members, analagous to how subjects are
identified using the "iss" and "sub" claims in OpenID Connect
[OpenID.Core] ID Tokens. These members MUST follow the formats of
the "iss" member and "sub" member defined by [RFC7519], respectively.
Both the "iss" member and the "sub" member are REQUIRED and MUST NOT
be null or empty. The Issuer and Subject Identifier Format is
identified by the name "iss_sub".

Below is a non-normative example Subject Identifier in the Issuer and
Subject Identifier Format:

```
{
  "format": "iss_sub",
  "iss": "http://issuer.example.com/",
  "sub": "145234573",
}
```

Figure 7: Example: Subject Identifier in the Issuer and Subject
Identifier Format

[3.2.5](#). Aliases Identifier Format

The Aliases Identifier Format describes a subject that is identified
with a list of different Subject Identifiers. It is intended for use
when a variety of identifiers have been shared with the party that
will be interpreting the Subject Identifier, and it is unknown which
of those identifiers they will recognize or support. Subject
Identifiers in this format MUST contain an "identifiers" member whose
value is a JSON array containing one or more Subject Identifiers.
Each Subject Identifier in the array MUST identify the same entity.
The "identifiers" member is REQUIRED and MUST NOT be null or empty.
It MAY contain multiple instances of the same Identifier Format
(e.g., multiple Email Subject Identifiers), but SHOULD NOT contain
exact duplicates. This type is identified by the name "aliases".

"alias" Subject Identifiers MUST NOT be nested; i.e., the
"identifiers" member of an "alias" Subject Identifier MUST NOT
contain a Subject Identifier of type "aliases".

Below is a non-normative example Subject Identifier in the Aliases
Identifier Format:

```
{
  "format": "aliases",
  "identifiers": [
    {
      "format": "email",
      "email": "user@example.com",
    },
    {
      "format": "phone_number",
      "phone_number": "+12065550100",
    },
    {
      "format": "email",
      "email": "user+qualifier@example.com",
    }
  ],
}
```

        Figure 8: Example: Subject Identifier in the Aliases Identifier
                                  Format

3.2.6. Opaque Identifier Format

The Opaque Identifier Format describes a subject that is identified
with a string with no semantics asserted beyond its usage as an
identifier for the subject, such as a UUID or hash used as a
surrogate identifier for a record in a database. Subject Identifiers
in this format MUST contain an "id" member whose value is a JSON
string containing the opaque string identifier for the subject. The
"id" member is REQUIRED and MUST NOT be null or empty. The Opaque
Identifier Format is identified by the name "opaque".

Below is a non-normative example Subject Identifier in the Opaque
Identifier Format:

```
{
  "format": "opaque",
  "id": "11112222333344445555",
}
```

Figure 9: Example: Subject Identifier in the Opaque Identifier Format

4. Subject Identifiers in JWTs

4.1. "sub_id" Claim

The "sub" JWT Claim is defined in Section 4.1.2 of [RFC7519] as
containing a string value, and therefore cannot contain a Subject
Identifier (which is a JSON object) as its value. This document
defines the "sub_id" JWT Claim, in accordance with Section 4.2 of
[RFC7519], as a common claim that identifies the JWT Subject using a
Subject Identifier. When present, the value of this claim MUST be a
Subject Identifier that identifies the subject of the JWT. The
"sub_id" claim MAY be included in a JWT, whether or not the "sub"
claim is present. When both the "sub" and "sub_id" claims are
present in a JWT, they MUST identify the same subject, as a JWT has
one and only one JWT Subject.

When processing a JWT with both "sub" and "sub_id" claims,
implementations MUST NOT rely on both claims to determine the JWT
Subject. An implementation MAY attempt to determine the JWT Subject
from one claim and fall back to using the other if it determines it
does not understand the format of the first claim. For example, an
implementation may attempt to use "sub_id", and fall back to using
"sub" upon finding that "sub_id" contains a Subject Identifier whose
type is not recognized by the implementation.

Below are non-normative examples of JWTs containing the "sub_id"
claim:

```
{
  "iss": "issuer.example.com",
  "sub_id": {
    "format": "email",
    "email": "user@example.com",
  },
}
```

Figure 10: Example: JWT containing a `sub_id` claim and no `sub`
                              claim

```
{
  "iss": "issuer.example.com",
  "sub": "user@example.com",
  "sub_id": {
    "format": "email",
    "email": "user@example.com",
  },
}
```

Figure 11: Example: JWT where both the `sub` and `sub_id` claims
        identify the JWT Subject using the same identifier

```
{
  "iss": "issuer.example.com",
  "sub": "user@example.com",
  "sub_id": {
    "format": "email",
    "email": "elizabeth@example.com",
  },
}
```

Figure 12: Example: JWT where both the `sub` and `sub_id` claims
    identify the JWT Subject using different values of the same
                        identifier type

```
{
  "iss": "issuer.example.com",
  "sub": "user@example.com",
  "sub_id": {
    "format": "account",
    "uri": "acct:example.user@service.example.com",
  },
}
```

Figure 13: Example: JWT where the `sub` and `sub_id` claims identify
        the JWT Subject via different types of identifiers

4.2. "sub_id" and "iss_sub" Subject Identifiers

The "sub_id" claim MAY contain an "iss_sub" Subject Identifier. In
this case, the JWT's "iss" claim and the Subject Identifier's "iss"
member MAY be different. For example, in OpenID Connect
[OpenID.Core] client may construct such a JWT when sending JWTs back
to its OpenID Connect Identity Provider, in order to identify the JWT
Subject using an identifier known to be understood by both parties.
Similarly, the JWT's "sub" claim and the Subject Identifier's "sub"
member MAY be different. For example, this may be used by an OpenID

Connect client to communicate the JWT Subject's local identifier at
the client back to its Identity Provider.

Below are non-normative examples of a JWT where the "iss" claim and
"iss" member within the "sub_id" claim are the same, and a JWT where
they are different.

```
{
  "iss": "issuer.example.com",
  "sub_id": {
    "format": "iss_sub",
    "iss": "issuer.example.com",
    "sub": "example_user",
  },
}
```

Figure 14: Example: JWT with a `iss_sub` Subject Identifier where JWT
              issuer and JWT Subject issuer are the same

```
{
  "iss": "client.example.com",
  "sub_id": {
    "format": "iss_sub",
    "iss": "issuer.example.com",
    "sub": "example_user",
  },
}
```

 Figure 15: Example: JWT with an `iss_sub` Subject Identifier where
         the JWT issuer and JWT Subject issuer are different

```
{
  "iss": "client.example.com",
  "sub": "client_user",
  "sub_id": {
    "format": "iss_sub",
    "iss": "issuer.example.com",
    "sub": "example_user",
  },
}
```

 Figure 16: Example: JWT with an `iss_sub` Subject Identifier where
  the JWT `iss` and `sub` claims differ from the JWT Subject's `iss`
                        and `sub` members

5. Considerations for Specifications that Define Identifier Formats

   Identifier Format definitions MUST NOT make assertions or
   declarations regarding the subject being identified by the Subject
   Identifier (e.g., an Identifier Format cannot be defined as
   specifically identifying human end users), as such statements are
   outside the scope of Identifier Formats and Subject Identifiers, and
   expanding that scope for some Identifier Formats but not others would
   harm interoperability, as applications that depend on this expanded
   scope to disambiguate the subject type would be unable to use
   Identifier Formats that do not provide such rules.

6. Privacy Considerations

6.1. Identifier Correlation

   The act of presenting two or more identifiers for a single subject
   together (e.g., within an "aliases" Subject Identifier, or via the
   "sub" and "sub_id" JWT claims) may communicate more information about
   the subject than was intended. For example, the entity to which the
   identifiers are presented now knows that both identifiers relate to
   the same subject, and may be able to correlate additional data based
   on that. When transmitting Subject Identifiers, the transmitter
   SHOULD take care that they are only transmitting multiple identifiers
   together when it is known that the recipient already knows that the
   identifiers are related (e.g., because they were previously sent to
   the recipient as claims in an OpenID Connect ID Token), or when
   correlation is essential to the use case.

   The considerations described in Section 6 of [RFC8417] also apply
   when Subject Identifiers are used within SETs. The considerations
   described in Section 12 of [RFC7519] also apply when Subject
   Identifiers are used within JWTs.

7. Security Considerations

7.1. Confidentiality and Integrity

   This specification does not define any mechanism for ensuring the
   confidentiality or integrityi of a Subject Identifier. Where such
   properties are required, implementations MUST use mechanisms provided
   by the containing format (e.g., integrity protecting SETs or JWTs
   using JWS [RFC7515]), or at the transport layer or other layer in the
   application stack (e.g., using TLS [RFC8446]).

   Further considerations regarding confidentiality and integrity of
   SETs can be found in Section 5.1 of [RFC8417].

8. IANA Considerations

8.1. Security Event Identifier Formats Registry

   This document defines Identifier Formats, for which IANA is asked to
   create and maintain a new registry titled "Security Event Identifier
   Formats". Initial values for the Security Event Identifier Formats
   registry are given in Section 3. Future assignments are to be made
   through the Expert Review registration policy [BCP26] and shall
   follow the template presented in Section 8.1.2.

   It is suggested that multiple Designated Experts be appointed who are
   able to represent the perspectives of different applications using
   this specification, in order to enable broadly informed review of
   registration decisions. In cases where a registration decision could
   be perceived as creating a conflict of interest for a particular
   Expert, that Expert should defer to the judgment of the other
   Experts.

8.1.1. Registry Location

   (This section to be removed by the RFC Editor before publication as
   an RFC.)

   The authors recommend that the Identifier Formats registry be located
   at "https://www.iana.org/assignments/secevent/".

8.1.2. Registration Template

   Type Name
      The name of the Identifier Format, as described in Section 3. The
      name MUST be an ASCII string consisting only of lower-case
      characters ("a" - "z"), digits ("0" - "9"), underscores ("_"), and
      hyphens ("-"), and SHOULD NOT exceed 20 characters in length.

   Type Description
      A brief description of the Identifier Format.

   Change Controller
      For types defined in documents published by the IETF or its
      working groups, list "IETF". For all other types, list the name
      of the party responsible for the registration. Contact
      information such as mailing address, email address, or phone
      number may also be provided.

   Defining Document(s)
      A reference to the document or documents that define the
      Identifier Format. The definition MUST specify the name, format,

and meaning of each member that may occur within a Subject
Identifier of the defined type, as well as whether each member is
optional, required, prohibited, or the circumstances under which
the member may be optional, required, or prohibited. URIs that
can be used to retrieve copies of each document SHOULD be
included.

8.1.3. Initial Registry Contents

8.1.3.1. Account Identifier Format

o Type Name: "account"

o Type Description: Subject identifier based on "acct" URI.

o Change Controller: IETF

o Defining Document(s): Section 3 of this document.

8.1.3.2. Email Identifier Format

o Type Name: "email"

o Type Description: Subject identifier based on email address.

o Change Controller: IETF

o Defining Document(s): Section 3 of this document.

8.1.3.3. Issuer and Subject Identifier Format

o Type Name: "iss_sub"

o Type Description: Subject identifier based on an issuer and
   subject.

o Change Controller: IETF

o Defining Document(s): Section 3 of this document.

8.1.3.4. Phone Number Identifier Format

o Type Name: "phone_number"

o Type Description: Subject identifier based on an phone number.

o Change Controller: IETF

   o Defining Document(s): Section 3 of this document.

8.1.3.5. Aliases Identifier Format

   o Type Name: "aliases"

   o Type Description: Subject identifier that groups together multiple
      different subject identifiers for the same subject.

   o Change Controller: IETF

   o Defining Document(s): Section 3 of this document.

8.1.3.6. Opaque Identifier Format

   o Type Name: "opaque"

   o Type Description: Subject identifier based on an opaque string.

   o Change Controller: IETF

   o Defining Document(s): Section 3 of this document.

8.1.4. Guidance for Expert Reviewers

   The Expert Reviewer is expected to review the documentation
   referenced in a registration request to verify its completeness. The
   Expert Reviewer must base their decision to accept or reject the
   request on a fair and impartial assessment of the request. If the
   Expert Reviewer has a conflict of interest, such as being an author
   of a defining document referenced by the request, they must recuse
   themselves from the approval process for that request. In the case
   where a request is rejected, the Expert Reviewer should provide the
   requesting party with a written statement expressing the reason for
   rejection, and be prepared to cite any sources of information that
   went into that decision.

   Identifier Formats need not be generally applicable and may be highly
   specific to a particular domain; it is expected that types may be
   registered for niche or industry-specific use cases. The Expert
   Reviewer should focus on whether the type is thoroughly documented,
   and whether its registration will promote or harm interoperability.
   In most cases, the Expert Reviewer should not approve a request if
   the registration would contribute to confusion, or amount to a
   synonym for an existing type.

8.2. JSON Web Token Claims Registration

   This document defines the "sub_id" JWT Claim, which IANA is asked to
   register in the "JSON Web Token Claims" registry IANA JSON Web Token
   Claims Registry [IANA.JWT.Claims] established by [RFC7519].

8.2.1. Registry Contents

   o Claim Name: "sub_id"

   o Claim Description: Subject Identifier

   o Change Controller: IESG

   o Specification Document(s): Section 4.1 of this document.

9. References

9.1. Normative References

   [BCP26]   Cotton, M., Leiba, B., and T. Narten, "Guidelines for
              Writing an IANA Considerations Section in RFCs", BCP 26,
              RFC 8126, DOI 10.17487/RFC8126, June 2017,
              <https://www.rfc-editor.org/info/rfc8126>.

   [E164]   International Telecommunication Union, "The international
              public telecommunication numbering plan", 2010,
              <http://www.itu.int/rec/T-REC-E.164-201011-I/en>.

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

   [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>.

   [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321,
              DOI 10.17487/RFC5321, October 2008,
              <https://www.rfc-editor.org/info/rfc5321>.

   [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322,
              DOI 10.17487/RFC5322, October 2008,
              <https://www.rfc-editor.org/info/rfc5322>.

   [RFC7159]   Bray, T., Ed., "The JavaScript Object Notation (JSON) Data
               Interchange Format", RFC 7159, DOI 10.17487/RFC7159, March
               2014, <https://www.rfc-editor.org/info/rfc7159>.

   [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>.

   [RFC7565]   Saint-Andre, P., "The 'acct' URI Scheme", RFC 7565,
               DOI 10.17487/RFC7565, May 2015,
               <https://www.rfc-editor.org/info/rfc7565>.

   [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>.

9.2. Informative References

   [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>.

   [RFC1034]   Mockapetris, P., "Domain names - concepts and facilities",
               STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987,
               <https://www.rfc-editor.org/info/rfc1034>.

   [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>.

   [RFC8446]   Rescorla, E., "The Transport Layer Security (TLS) Protocol
               Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018,
               <https://www.rfc-editor.org/info/rfc8446>.

Acknowledgements

   The authors would like to thank the members of the IETF Security
   Events working group, as well as those of the OpenID Shared Signals
   and Events Working Group, whose work provided the original basis for
   this document.

Change Log

   (This section to be removed by the RFC Editor before publication as
   an RFC.)

   Draft 00 - AB - First draft

   Draft 01 - AB:

   o   Added reference to RFC 5322 for format of "email" claim.

   o   Renamed "iss_sub" type to "iss-sub".

   o   Renamed "id_token_claims" type to "id-token-claims".

   o   Added text specifying the nature of the subjects described by each
       type.

   Draft 02 - AB:

   o   Corrected format of phone numbers in examples.

   o   Updated author info.

   Draft 03 - AB:

   o   Added "account" type for "acct" URIs.

   o   Replaced "id-token-claims" type with "aliases" type.

   o   Added email canonicalization guidance.

   o   Updated semantics for "email", "phone", and "iss-sub" types.

   Draft 04 - AB:

   o   Added "sub_id" JWT Claim definition, guidance, examples.

   o   Added text prohibiting "aliases" nesting.

   o   Added privacy considerations for identifier correlation.

   Draft 05 - AB:

   o   Renamed the "phone" type to "phone-number" and its "phone" claim
       to "phone_number".

   Draft 06 - AB:

   o   Replaced usage of the word "claim" to describe members of a
       Subject Identifier with the word "member", in accordance with
       terminology in RFC7159.

o Renamed the "phone-number" type to "phone_number" and "iss-sub" to
   "iss_sub".

o Added normative requirements limiting the use of both "sub" and
   "sub_id" claims together when processing a JWT.

o Clarified that identifier correlation may be acceptable when it is
   a core part of the use case.

o Replaced references to OIDF with IETF in IANA Considerations.

o Recommended the appointment of multiple Designated Experts, and a
   location for the Subject Identifier Types registry.

o Added "_" to list of allowed characters in the Type Name for
   Subject Identifier Types.

o Clarified that Subject Identifiers don't provide confidentiality
   or integrity protection.

o Added references to SET, JWT privacy and security considerations.

o Added section describing the difference between subject identifier
   type and principal type that hopefully clarifies things and
   doesn't just muddy the water further.

Draft 07 - AB:

o Emphasized that the spec is about identifiers, not the things they
   identify:

   * Renamed "Subject Identifier Type" to "Identifier Format".

   * Renamed "subject_type" to "format".

   * Renamed "Security Event Subject Identifier Type Registry" to
      "Security Event Identifier Format Registry".

   * Added new section with guidance for specs defining Identifier
      Formats, with normative prohibition on formats that describe
      the subject itself, rather than the identifier.

o Clarified the meaning of "subject":

   * Defined "subject" as applying generically and "JWT Subject" as
      applying specifically to the subject of a JWT.

   * Replaced most instances of the word "principal" with "subject".

   o Added "opaque" Identifier Format

Authors' Addresses

   Annabelle Backman (editor)
   Amazon

   Email: richanna@amazon.com

   Marius Scurtescu
   Coinbase

   Email: marius.scurtescu@coinbase.com