|  |
| --- |
| [**TOC**](#toc) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | draft | N. Sakimura | |  | NRI | |  | J. Bradley | |  | Ping Identity | |  | N. Agarwal | |  | Google | |  | E. Jay | |  | Illumila | |  | August 16, 2014 | |

# OpenID 2.0 to OpenID Connect Migration 1.0 - draft 04

### Abstract

This specification defines how an OpenID Authentication 2.0 relying party can migrate the user from OpenID 2.0 identifier to OpenID Connect Identifier by using an ID Token that includes the OpenID 2.0 verified claimed ID. In this specification, the method to request such an additional claim and the method for the verification of the resulting ID Token is specified.

### Table of Contents

[**1.**](#Introduction)  Introduction  
    [**1.1.**](#rnc)  Requirements Notation and Conventions  
    [**1.2.**](#Terminology)  Terminology  
[**2.**](#RequestOpenid2Id)  Requesting the OpenID 2.0 Identifier and OpenID Connect Identifier Together  
[**3.**](#VerifyRP)  Verification of the Relying Party by the OpenID Provider  
[**4.**](#ReturnOpenID2ID)  Returning the OpenID 2.0 Identifier  
    [**4.1.**](#ErrorResponses)  Error Responses  
        [**4.1.1.**](#Openid2NotSupported)  Scope "openid2" Not Supported  
        [**4.1.2.**](#NoOpenid2.0Association)  No Associated OpenID 2.0 Identifier Found  
[**5.**](#VerifyIDToken)  Verification of the ID Token  
[**6.**](#VerifyOPAuthority)  Verification that the OpenID Connect OP is Authoritative  
[**7.**](#AssociateIdentifiers)  Associating the Existing OpenID 2.0 Account with the OpenID Connect Identifier  
[**8.**](#ImplementationConsiderations)  Implementation Considerations  
    [**8.1.**](#AfterEOL)  After End-Of-Life of the OpenID 2.0 OP  
[**9.**](#PrivacyConsiderations)  Privacy Considerations  
    [**9.1.**](#Correlation)  Correlation  
    [**9.2.**](#IdByOthers)  Identification by Other Parties  
    [**9.3.**](#SecondaryUse)  Secondary Use  
    [**9.4.**](#Disclosure)  Disclosure  
    [**9.5.**](#Exclusion)  Exclusion  
[**10.**](#Security)  Security Considerations  
[**11.**](#rfc.references1)  References  
    [**11.1.**](#rfc.references1)  Normative References  
    [**11.2.**](#rfc.references2)  Informative References  
[**Appendix A.**](#SequenceDiagram1)  Sequence Diagrams  
[**Appendix B.**](#GoogleDifference)  Differences from Google’s Migration Guide as of June 3, 2014   
[**Appendix C.**](#Acknowledgements)  Acknowledgements  
[**Appendix D.**](#Notices)  Notices  
[**§**](#rfc.authors)  Authors' Addresses

|  |
| --- |
| [**TOC**](#toc) |

### 1.  Introduction

OpenID Authentication 2.0 is a popular authentication federation protocol through which the relying party can obtain the user’s verified identifier from the OpenID Provider (OP) to which the user was authenticated. OpenID Connect is a new version of OpenID Authentication but the identifier format is different and thus relying parties need to migrate those user identifiers to continue serving these users.

In this specification, a standard method for this kind of migration on a per-user basis is described.

|  |
| --- |
| [**TOC**](#toc) |

### 1.1.  Requirements Notation and 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 [RFC 2119 (Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels,” March 1997.)](#RFC2119) [RFC2119].

In the .txt version of this document, values are quoted to indicate that they are to be taken literally. When using these values in protocol messages, the quotes MUST NOT be used as part of the value. In the HTML version of this document, values to be taken literally are indicated by the use of this fixed-width font.

|  |
| --- |
| [**TOC**](#toc) |

### 1.2.  Terminology

The terms defined in [OpenID Connect Core 1.0 (Sakimura, N., Bradley, J., Jones, M., de Medeiros, B., and C. Mortimore, “OpenID Connect Core 1.0,” February 2014.)](#OpenID.Core) [OpenID.Core] and [OpenID Authentication 2.0 (OpenID Foundation, “OpenID Authentication 2.0,” December 2007.)](#OpenID.2.0) [OpenID.2.0] are used by this specification. Where a same term is defined in both specifications, the term defined in OpenID Connect Core takes precedence.

This specification also defines the following terms:

OpenID 2.0 Identifier

Verified user identifier as specified by OpenID Authentication 2.0.

Connect OP

OpenID Connect OP

OpenID Connect Identifier

OpenID Connect issuer and subject pair

|  |
| --- |
| [**TOC**](#toc) |

### 2.  Requesting the OpenID 2.0 Identifier and OpenID Connect Identifier Together

To obtain the OpenID 2.0 Identifier, the RP sends a modified OpenID Connect Authentication Request by adding openid2 as an additional scope value.

If PPID was used to obtain the OpenID 2.0 Identifier, openid.realm has to be sent to the OP with the request. For this purpose, a new authentication request parameter openid2\_realm is defined.

openid2\_realm

OPTIONAL. The openid.realm value as defined in Section 9.1 of [OpenID 2.0 (OpenID Foundation, “OpenID Authentication 2.0,” December 2007.)](#OpenID.2.0) [OpenID.2.0]

If the authority section of Authorization Endpoint URI is different from the authority section of the OpenID 2.0 OP’s OP Endpoint URL, the client MUST issue a GET request to the OpenID 2.0 Identifier obtained through the ID Token, i.e., the value of openid2\_id, with an Accept header set to application/json to obtain the value of iss claim in it. The value of the iss claim obtained this way and the value of the iss claim in the ID Token MUST exactly match.

**Note**: This is similar to YADIS. In case of YADIS, it is using Accept header with its value set to application/xml+xrds.

The following is a non-normative example of an authentication request to request the OpenID 2.0 Identifier (with line wraps within values for display purposes only). NOTE: This example assumes that the OpenID 2.0 OP Identifier is https://openid2.example.com.

GET /authorize?response\_type=id\_token

&scope=openid%20openid2

&client\_id=s6BhdRkqt3

&state=af0ifjsldkj

&nonce=n-0S6\_WzA2Mj

&openid2\_realm=https%3A%2F%2Fopenid2.example.com

&redirect\_uri=https%3A%2F%2Fclient.example.org%2Fcb HTTP/1.1

Host: server.example.com

The End-User performs authentication and authorization at the Connect OP which then returns the authentication response:

HTTP /1.1 200 OK

Location: https://client.example.com/cb#

id\_token=eyJhbGciOiJSUzI1NiIsImtpZCI6IktleTAwMSJ9.ew0KIC

Jpc3MiOiAiaHR0cDovL3NlcnZlci5leGFtcGxlLmNvbSIsDQogInN1Yi

I6ICIyNDgyODk3NjEwMDEiLA0KICJhdWQiOiAiczZCaGRSa3F0MyIsDQ

ogIm5vbmNlIjogIm4tMFM2X1d6QTJNaiIsDQogImV4cCI6IDEzMTEyOD

E5NzAsDQogImlhdCI6IDEzMTEyODA5NzAsDQogIm9wZW5pZDJfaWQiOi

AiaHR0cHM6Ly9vcGVuaWQyLmV4YW1wbGUuY29tL3VzZXIzNTkzOTA4Nz

IxMTIiDQp9.rSo68AZGeJY15WxTtDxHrGlpJu2S7jIHsd\_lBrBB20uva

UbbEvJyMJFuQVUeqH-b8XiyUFtHHynXxYq6P8SpMw7UX2y4BGg0Ky-5z

KeGJkT8-Cfkx8eLzKMVE-qsB31NhS3bZ4Wp3mHTsUCOUhbHfEeDRJaCJ

G3NlNEc2QLKBcmfzzdvVw98XuMySFIE0r9ekqx8h0IMvxRQgJENEDQ1q

70v5oR4YcEO1lcbT3a9wAA-0N27zAP0OUURXSMQaIfpmo8kDIaj7oRd8

36PowpRodp7VHKO0RoyhOFGFrDDA4z\_mCE4Yopx-tWPZFPO8sekrz-H9

670UAZUOBux2CYGEw&

state=af0ifjsldkj

The contents of the ID Token after decoding are:

{

"iss": "http://server.example.com",

"sub": "248289761001",

"aud": "s6BhdRkqt3",

"nonce": "n-0S6\_WzA2Mj",

"exp": 1311281970,

"iat": 1311280970,

"openid2\_id": "https://openid2.example.com/user359390872112"

}

To verify the issuer in the ID Token is authoritative for openid2\_id, get the issuer from the OpenID 2.0 Identifier URI.

GET /user359390872112 HTTP/1.1

Host: openid2.example.com

Accept: application/json

HTTP /1.1 200 OK

Content-Type: application/json

{

"iss": "http://server.example.com"

}

Verify the value of iss claim of ID Token exactly matches the value of iss claim of this response.

|  |
| --- |
| [**TOC**](#toc) |

### 3.  Verification of the Relying Party by the OpenID Provider

There could be an attack by a malicious RP to obtain the user’s PPID for another RP to perform identity correlation. To mitigate the risk, the OP MUST verify that the realm and RP’s Redirect URI matches as per Section 9.2 of [OpenID 2.0 (OpenID Foundation, “OpenID Authentication 2.0,” December 2007.)](#OpenID.2.0) [OpenID.2.0].

|  |
| --- |
| [**TOC**](#toc) |

### 4.  Returning the OpenID 2.0 Identifier

If the verification of the Relying Party was successful and an associated OpenID 2.0 Identifier for the user is found, then the OP MUST include the OpenID 2.0 Identifier in the asymmetrically signed ID Token with the following claim name:

openid2\_id

REQUIRED. OpenID 2.0 Identifier. It MUST be represented as a JSON string.

For XRI, OpenID 2.0 Identifier MUST be created as https://xri.net/ concatenated with the user’s verified XRI without the xri:// scheme.

The following is a non-normative example of an ID Token with an OpenID 2.0 Identifier claim (with line wraps within values for display purposes only)

{

"iss": "http://server.example.com",

"sub": "248289761001",

"aud": "s6BhdRkqt3",

"nonce": "n-0S6\_WzA2Mj",

"exp": 1311281970,

"iat": 1311280970,

"openid2\_id": "https://openid2.example.com/user359390872112"

}

|  |
| --- |
| [**TOC**](#toc) |

### 4.1.  Error Responses

In addition to the error conditions defined in [OpenID Connect Core 1.0 (Sakimura, N., Bradley, J., Jones, M., de Medeiros, B., and C. Mortimore, “OpenID Connect Core 1.0,” February 2014.)](#OpenID.Core) [OpenID.Core], the following error conditions are defined in this standard.

|  |
| --- |
| [**TOC**](#toc) |

### 4.1.1.  Scope "openid2" Not Supported

If the openid2 scope is not supported, the error invalid\_scope as defined in 4.1.2.1 of [OAuth (Hardt, D., “The OAuth 2.0 Authorization Framework,” October 2012.)](#RFC6749) [RFC6749] SHOULD be returned.

|  |
| --- |
| [**TOC**](#toc) |

### 4.1.2.  No Associated OpenID 2.0 Identifier Found

If a corresponding OpenID 2.0 Identifier is not found for the authenticated user, the openid2\_id claim in the ID Token MUST have the value NOT FOUND.

NOTE: Even if the openid2\_id claim value is NOT FOUND, the overall ID Token can still be valid.

|  |
| --- |
| [**TOC**](#toc) |

### 5.  Verification of the ID Token

The RP MUST verify the ID Token as specified in 3.1.3.7 of [OpenID Connect Core 1.0 (Sakimura, N., Bradley, J., Jones, M., de Medeiros, B., and C. Mortimore, “OpenID Connect Core 1.0,” February 2014.)](#OpenID.Core) [OpenID.Core].

|  |
| --- |
| [**TOC**](#toc) |

### 6.  Verification that the OpenID Connect OP is Authoritative

A malicious OP may try to impersonate the user by returning the OpenID 2.0 Identifier that it is not authoritative for. Therefore, verifying that the Connect OP is indeed authoritative for the OpenID 2.0 Identifier is imperative. To verify that the Connect OP is authoritative for the OpenID 2.0 Identifier, the RP MUST verify that one of the following verification rules hold:

1. If the RP a priori knows that the authority hosted only one OpenID 2.0 OP and OpenID Connect OP each, the authority section of Authorization Endpoint URI is the same as the authority section of the OpenID 2.0 OP’s OP Endpoint URL.
2. If they are not (or when a higher confidence is sought), RP MUST make a GET call to the obtained verified claimed ID with an Accept header set to application/json. The server SHOULD return a JSON with iss as its top level member. The value of this member MUST exactly match the iss in the ID Token.

If both fails, it is a failure and the RP MUST NOT accept the OpenID 2.0 Identifier.

|  |
| --- |
| [**TOC**](#toc) |

### 7.  Associating the Existing OpenID 2.0 Account with the OpenID Connect Identifier

As the association between OpenID Connect Identifier and OpenID 2.0 Identifier has been verified, the RP SHOULD associate the existing OpenID 2.0 account with the OpenID Connect account.

**NOTE**: At some point in the future, the OpenID Connect server may drop the support for openid2 scope. In this case, the OP will return the invalid\_scope in the error as defined in [Section 4.1 (Error Responses)](#ErrorResponses).

|  |
| --- |
| [**TOC**](#toc) |

### 8.  Implementation Considerations

|  |
| --- |
| [**TOC**](#toc) |

### 8.1.  After End-Of-Life of the OpenID 2.0 OP

This standard allows the RP to verify the authenticity of the OpenID 2.0 Identifier through ID Token even after the OpenID 2.0 OP is taken down. To enable this, the OP MUST publish the public keys that were used to sign the ID Token with openid2\_id claim at the URI that this OpenID 2.0 Identifier points to.

NOTE: The OpenID 2.0 Identifiers can be mapped to a static file containing the keys, so maintaining such can require minimal overhead compared to maintaining the full OpenID 2.0 OP.

|  |
| --- |
| [**TOC**](#toc) |

### 9.  Privacy Considerations

This section considers the Privacy Specific Threats described in Section 5.2 of [RFC 6973 (Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, “Privacy Considerations for Internet Protocols,” July 2013.)](" \l "RFC6973) [RFC6973].

|  |
| --- |
| [**TOC**](#toc) |

### 9.1.  Correlation

This standard essentially is a correlation specification. It correlates the OpenID Connect identifier with OpenID 2.0 Identifier. In the usual case where the user has only one account and the Connect and OpenID 2.0 OPs look similar, then the user probably would be expecting that those identifiers to be correlated silently. However, if the OPs looks very different, then some users may prefer not to be correlated. As such, the OP SHOULD make sure that to ask the user if the user wants to correlate.

When multiple accounts are available for the user, then the OP MUST make sure that the user picks the intended identity.

|  |
| --- |
| [**TOC**](#toc) |

### 9.2.  Identification by Other Parties

Since the channel is encrypted, this risk is low. If the channel was vulnerable, then user identifiers and other attributes will be exposed and thus allows the attacker to identify the user. To avoid it, the parties can employ ID Token encryption as well.

|  |
| --- |
| [**TOC**](#toc) |

### 9.3.  Secondary Use

While there is no technical control in this standard as to the secondary use is concerned, RP is strongly advised to announce its policy against secondary use in its privacy policy. Secondary use usually is associated with privacy impact, so its legitimacy should be carefully evaluated.

|  |
| --- |
| [**TOC**](#toc) |

### 9.4.  Disclosure

Since the channel is encrypted, this risk is low. If the channel was vulnerable, then user identifiers and other attributes will be exposed and thus allows the attacker to identify the user. To avoid it, the parties can employ ID Token encryption as well.

|  |
| --- |
| [**TOC**](#toc) |

### 9.5.  Exclusion

To avoid Exclusion in this case, make sure to ask the user if he wants the identifiers to be correlated.

|  |
| --- |
| [**TOC**](#toc) |

### 10.  Security Considerations

In addition to correctly implementing the usual OpenID Connect security measures, the RP MUST carefully follow and correctly implementing [Section 6 (Verification that the OpenID Connect OP is Authoritative)](#VerifyOPAuthority). If in doubt, skipping step 1 and just doing step 2 is safer.

|  |
| --- |
| [**TOC**](#toc) |

### 11.  References

|  |
| --- |
| [**TOC**](#toc) |

### 11.1. Normative References

|  |  |
| --- | --- |
| **[OpenID.2.0]** | OpenID Foundation, “OpenID Authentication 2.0,” December 2007 ([TXT](http://openid.net/specs/openid-authentication-2_0.txt), [HTML](http://openid.net/specs/openid-authentication-2_0.html)). |
| **[OpenID.Core]** | Sakimura, N., Bradley, J., Jones, M., de Medeiros, B., and C. Mortimore, “[OpenID Connect Core 1.0](http://openid.net/specs/openid-connect-core-1_0.html),” February 2014. |
| **[RFC2119]** | [Bradner, S.](mailto:sob@harvard.edu), “[Key words for use in RFCs to Indicate Requirement Levels](http://tools.ietf.org/html/rfc2119),” BCP 14, RFC 2119, March 1997 ([TXT](http://www.rfc-editor.org/rfc/rfc2119.txt), [HTML](http://xml.resource.org/public/rfc/html/rfc2119.html), [XML](http://xml.resource.org/public/rfc/xml/rfc2119.xml)). |
| **[RFC6749]** | Hardt, D., “[The OAuth 2.0 Authorization Framework](http://tools.ietf.org/html/rfc6749),” RFC 6749, October 2012 ([TXT](http://www.rfc-editor.org/rfc/rfc6749.txt)). |

|  |
| --- |
| [**TOC**](#toc) |

### 11.2. Informative References

|  |  |
| --- | --- |
| **[RFC6973]** | Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, “[Privacy Considerations for Internet Protocols](http://tools.ietf.org/html/rfc6973),” RFC 6973, July 2013 ([TXT](http://www.rfc-editor.org/rfc/rfc6973.txt)). |

|  |
| --- |
| [**TOC**](#toc) |

### Appendix A.  Sequence Diagrams

Migration Sequence Diagram for Implicit Flow

+----+ +----------+ +--------------+ +----------+ +----------+

| UA | | Resource | | Redirect URI | | AuthZ EP | |OpenID2URI|

+-+--+ +----+-----+ +-----+--------+ +---+------+ +-----+----+

Click|AuthN Link| | | |

+--------> | | | |

|AuthN Req | | | |

| <--------+ | | |

| | AuthN Req | | |

+---------------------------------------> | |

+----+-----------------------------------------------------------------+

|OPT | | | AuthN Page | | | |

+----+ | <---------------------------------------+ | |

| | | Credential | | | |

| +---------------------------------------> | | |

+----------------------------------------------------------------------+

| |302 to RedirectURI | |

| <------------------------+--------------+ |

| |ID Token | | |

+------------------------> | | |

| | |------+ | |

| |Get OpenID2URI | | | |

| |from ID Token | <----+ | |

| | | GET w/Accept: application/json

| | +----------------------------> |

| | | iss in JSON |

| | | <----------------------------+

| | | | |

+-+--+ +----+-----+ +------+-------+ +----+-----+ +------+---+

| UA | | Resource | | Redirect URI | | AuthZ EP | |OpenID2URI|

+----+ +----------+ +--------------+ +----------+ +----------+

|  |
| --- |
| [**TOC**](#toc) |

### Appendix B.  Difference from Google’s Migration Guide as of June 3, 2014

In this appendix, the differences between this spec and Google’s migration guide as of June 3, 2014 are explained. The differences are categorized in accordance with the section number of this specification. Google's migration guide is available at [Migrating to OAuth 2.0 login (OpenID Connect)](https://developers.google.com/accounts/docs/OpenID#openid-connect).

[2 (Requesting the OpenID 2.0 Identifier and OpenID Connect Identifier Together)](#RequestOpenid2Id). [Requesting the OpenID 2.0 Identifier and OpenID Connect Identifier Together (Requesting the OpenID 2.0 Identifier and OpenID Connect Identifier Together)](#RequestOpenid2Id)

Google uses openid.realm instead. Since OpenID Connect uses param\_name style instead of param.name, as well as the name openid.realm may mislead the user that it is a Connect parameter proper, it has been changed to openid2\_realm.

Google uses the existence of openid.realm parameter to switch the behavior at the Connect OP. New scope value openid2 has been introduced in this spec to make it more explicit and semantically in-line that it is asking for a resource.

[3 (Verification of the Relying Party by the OpenID Provider)](#VerifyRP). [Verification of the Relying Party by the OpenID Provider (Verification of the Relying Party by the OpenID Provider)](#VerifyRP)

Google does not perform RP verification.

[4 (Returning the OpenID 2.0 Identifier)](#ReturnOpenID2ID). [Returning the OpenID 2.0 Identifier (Returning the OpenID 2.0 Identifier)](#ReturnOpenID2ID)

Google uses the claim name openid\_id instead of openid2\_id . It was changed to openid2\_id because openid\_id may cause confusion among people that it is the Connect identifier. Since this spec allows providing openid2\_id even after the OpenID 2.0 OP has been taken down, this claim may persists much longer than the OpenID 2.0 OP. Thus, the chance of confusion should be minimized.

Google does not take care of XRI while this standard does.

[6 (Verification that the OpenID Connect OP is Authoritative)](#VerifyOPAuthority). [Verification that the OpenID Connect OP is Authoritative (Verification that the OpenID Connect OP is Authoritative)](#VerifyOPAuthority)

Google does not perform authority verification.

|  |
| --- |
| [**TOC**](#toc) |

### Appendix C.  Acknowledgements

In addition to the authors, the OpenID Community would like to thank the following people for their contributions to this specification:

Breno de Medeiros (breno@google.com), Google

Ryo Ito (ryo.ito@mixi.co.jp), mixi, Inc.

Michael B. Jones (mbj@microsoft.com), Microsoft

Nov Matake (nov@matake.jp), Independent

Allan Foster (allan.foster@forgerock.com), ForgeRock

Chuck Mortimore (cmortimore@salesforce.com), Salesforce

Torsten Lodderstedt (torsten@lodderstedt.net), Deutsche Telekom

Justin Richer (jricher@mitre.org), MITRE Corporation

|  |
| --- |
| [**TOC**](#toc) |

### Appendix D.  Notices

Copyright (c) 2014 The OpenID Foundation.

The OpenID Foundation (OIDF) grants to any Contributor, developer, implementer, or other interested party a non-exclusive, royalty free, worldwide copyright license to reproduce, prepare derivative works from, distribute, perform and display, this Implementers Draft or Final Specification solely for the purposes of (i) developing specifications, and (ii) implementing Implementers Drafts and Final Specifications based on such documents, provided that attribution be made to the OIDF as the source of the material, but that such attribution does not indicate an endorsement by the OIDF.

The technology described in this specification was made available from contributions from various sources, including members of the OpenID Foundation and others. Although the OpenID Foundation has taken steps to help ensure that the technology is available for distribution, it takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this specification or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any independent effort to identify any such rights. The OpenID Foundation and the contributors to this specification make no (and hereby expressly disclaim any) warranties (express, implied, or otherwise), including implied warranties of merchantability, non-infringement, fitness for a particular purpose, or title, related to this specification, and the entire risk as to implementing this specification is assumed by the implementer. The OpenID Intellectual Property Rights policy requires contributors to offer a patent promise not to assert certain patent claims against other contributors and against implementers. The OpenID Foundation invites any interested party to bring to its attention any copyrights, patents, patent applications, or other proprietary rights that may cover technology that may be required to practice this specification.

|  |
| --- |
| [**TOC**](#toc) |

### Authors' Addresses

|  |  |
| --- | --- |
|  | Nat Sakimura |
|  | Nomura Research Institute, Ltd. |
| **Email:** | [n-sakimura@nri.co.jp](mailto:n-sakimura@nri.co.jp) |
| **URI:** | <http://nat.sakimura.org/> |
|  |  |
|  | John Bradley |
|  | Ping Identity |
| **Email:** | [ve7jtb@ve7jtb.com](mailto:ve7jtb@ve7jtb.com) |
| **URI:** | <http://www.thread-safe.com/> |
|  |  |
|  | Naveen Agarwal |
|  | Google |
| **Email:** | [naa@google.com](mailto:naa@google.com) |
| **URI:** | <http://www.google.com> |
|  |  |
|  | Edmund Jay |
|  | Illumila |
| **Email:** | [ejay@illumi.la](mailto:ejay@illumi.la) |
| **URI:** | <http://illumi.la> |