WIKIPEDIA

OpenID

OpenID is an open standard and decentralized authentication protocol. Promoted by the non-profit OpenID Foundation, it allows users to be authenticated by co-operating sites (known as relying parties, or RP) using a third-party service, eliminating the need for webmasters to provide their own ad hoc login systems, and allowing users to log into multiple unrelated websites without having to have a separate identity and password for each.[1] Users create accounts by selecting an OpenID identity provider[1] and then use those accounts to sign onto any website that accepts OpenID authentication. Several large organizations either issue or accept OpenIDs on their websites, according to the OpenID Foundation.[2]

The OpenID logo

The OpenID standard provides a framework for the communication that must take place between the identity provider and the OpenID acceptor (the "relying party").[3] An extension to the standard (the OpenID Attribute Exchange) facilitates the transfer of user attributes, such as name and gender, from the OpenID identity provider to the relying party (each relying party may request a different set of attributes, depending on its requirements).[4] The OpenID protocol does not rely on a central authority to authenticate a user's identity. Moreover, neither services nor the OpenID standard may mandate a specific means by which to authenticate users, allowing for approaches ranging from the common (such as passwords) to the novel (such as smart cards or biometrics).

The final version of OpenID is OpenID 2.0, finalized and published in December 2007.[5] The term *OpenID* may also refer to an identifier as specified in the OpenID standard; these identifiers take the form of a unique Uniform Resource Identifier (URI), and are managed by some "OpenID provider" that handles authentication.[1]

Contents

Adoption

Technical overview
- Logging in
- Identifiers

OpenID Foundation
- People
- Chapters
- Intellectual property and contribution agreements
- Legal issues

Security
- Authentication bugs
- Phishing
- Privacy and trust issues
- Authentication hijacking in unsecured connection
- Covert Redirect

History

OpenID vs. pseudo-authentication using OAuth
- Attack against pseudo-authentication
  - Verifying the letter

OpenID Connect

See also

References

External links

Adoption

As of March 2016, there are over 1 billion OpenID-enabled accounts on the Internet (see below) and approximately 1,100,934 sites have integrated OpenID consumer support:[6] AOL, Flickr, France Telecom, Google, Amazon.com, Canonical (provider name Ubuntu One), LiveJournal, Microsoft (provider name Microsoft account), Mixi, Myspace, Novell, OpenStreetMap, Orange, Sears, Sun, Telecom Italia, Universal Music Group, VeriSign, WordPress, Yahoo!, the BBC,[7] IBM,[8] PayPal,[9] and Steam,[10] although some of those organizations also have their own authentication management.

Many if not all of the larger organizations require users to provide authentication in the form of an existing email account or mobile phone number in order to sign up for an account (which then can be used as an OpenID identity). There are several smaller entities that accept sign-ups with no extra identity details required.

Facebook did use OpenID in the past, but moved to Facebook Connect.[11] Blogger also used OpenID, but since May 2018 no longer supports it.[12]

Technical overview

An *end-user* is the entity that wants to assert a particular identity. A *relying party* (RP) is a web site or application that wants to verify the end-user's identifier. Other terms for this party include "service provider" or the now obsolete "consumer". An identity provider, or *OpenID provider* (OP) is a service that specializes in registering OpenID URLs or XRIs. OpenID enables an end-user to communicate with a relying party. This communication is done through the exchange of an identifier or *OpenID*, which is the URL or XRI chosen by the end-user to name the end-user's identity. An identity provider provides the OpenID authentication (and possibly other identity services). The exchange is enabled by a *user-agent*, which is the program (such as a browser) used by the end-user to communicate with the relying party and OpenID provider.

Logging in

The end-user interacts with a relying party (such as a website) that provides an option to specify an OpenID for the purposes of authentication; an end-user typically has previously registered an OpenID (e.g. `alice.openid.example.org`) with an OpenID provider (e.g. `openid.example.org`).[1]

The relying party typically transforms the OpenID into a canonical URL form (e.g. `http://alice.openid.example.org/`).

- With OpenID 1.0, the relying party then requests the HTML resource identified by the URL and reads an HTML link tag to discover the OpenID provider's URL (e.g. `http://openid.example.org/openid-auth.php`). The relying party also discovers whether to use a *delegated identity* (see below).
- With OpenID 2.0, the relying party discovers the OpenID provider URL by requesting the *XRDS document* (also called the *Yadis document*) with the content type `application/xrds+xml`; this document may be available at the target URL and is always available for a target XRI.

There are two modes in which the relying party may communicate with the OpenID provider:

- `checkid_immediate`, in which the relying party requests that the OpenID provider not interact with the end-user. All communication is relayed through the end-user's user-agent without explicitly notifying the end-user.
- `checkid_setup`, in which the end-user communicates with the OpenID provider via the same user-agent used to access the relying party.

The `checkid_immediate` mode can fall back to the `checkid_setup` mode if the operation cannot be automated.

First, the relying party and the OpenID provider (optionally) establish a shared secret, referenced by an *associate handle*, which the relying party then stores. If using the `checkid_setup` mode, the relying party redirects the end-user's user-agent to the OpenID provider so the end-user can authenticate directly with the OpenID provider.

The method of authentication may vary, but typically, an OpenID provider prompts the end-user for a password or some cryptographic token, and then asks whether the end-user trusts the relying party to receive the necessary identity details.

If the end-user declines the OpenID provider's request to trust the relying party, then the user-agent is redirected back to the relying party with a message indicating that authentication was rejected; the relying party in turn refuses to authenticate the end-user.

If the end-user accepts the OpenID provider's request to trust the relying party, then the user-agent is redirected back to the relying party along with the end-user's credentials. That relying party must then confirm that the credentials really came from the OpenID provider. If the relying party and OpenID provider had previously established a shared secret, then the relying party can validate the identity of the OpenID provider by comparing its copy of the shared secret against the one received along with the end-user's credentials; such a relying party is called *stateful* because it stores the shared secret between sessions. In contrast, a *stateless* or *dumb* relying party must make one more background request (`check_authentication`) to ensure that the data indeed came from the OpenID provider.

After the OpenID has been verified, authentication is considered successful and the end-user is considered logged into the relying party under the identity specified by the given OpenID (e.g. `alice.openid.example.org`). The relying party typically then stores the end-user's OpenID along with the end-user's other session information.

Identifiers

To obtain an OpenID-enabled URL that can be used to log into OpenID-enabled websites, a user registers an OpenID identifier with an identity provider. Identity providers offer the ability to register a URL (typically a third-level domain, e.g. username.example.com) that will automatically be configured with OpenID authentication service.

Once they have registered an OpenID, a user can also use an existing URL under their own control (such as a blog or home page) as an alias or "delegated identity". They simply insert the appropriate OpenID tags in the HTML[13] or serve a Yadis document.[14]

Starting with OpenID Authentication 2.0 (and some 1.1 implementations), there are two types of identifiers that can be used with OpenID: URLs and XRIs.

XRIs are a new form of Internet identifier designed specifically for cross-domain digital identity. For example, XRIs come in two forms—i-names and i-numbers—that are usually registered simultaneously as synonyms. I-names are reassignable (like domain names), while i-numbers are never reassigned. When an XRI i-name is used as an OpenID identifier, it is immediately resolved to the synonymous i-number (the CanonicalID element of the XRDS document). This i-number is the OpenID identifier stored by the relying party. In this way, both the user and the relying party are protected from the end-user's OpenID identity ever being taken over by another party as can happen with a URL based on a reassignable DNS name.

OpenID Foundation

The OpenID Foundation (OIDF) promotes and enhances the OpenID community and technologies. The OIDF is a non-profit international standards development organization of individual developers, government agencies and companies who wish to promote and protect OpenID. The OpenID Foundation was formed in June 2007 and serves as a public trust organization representing an open community of developers, vendors and users. OIDF assists the community by providing needed infrastructure and help in promoting and supporting adoption of OpenID. This includes managing intellectual property and trade marks as well a fostering viral growth and global participation in OpenID.

People

The OpenID Foundation's board of directors has four community members and eight corporate members:[15]

Community Board Members

- Chairman: Nat Sakimura (Nomura Research Institute)
- Treasurer: John Bradley (Ping Identity)
- Secretary: Mike Jones (Microsoft)
- Community Representative: George Fletcher (AOL)

Corporate Board Members

- Google – Adam Dawes
- Janrain – Jim Kaskade
- Microsoft – Anthony Nadalin
- Oracle – Prateek Mishra
- Ping Identity – Pam Dingle
- Symantec – Brian Berliner
- US Department of Health & Human Services, Office of the National Coordinator – Debbie Bucci
- Verizon – Bjorn Hjelm
- VMware – Ashish Jain

Chapters

OIDF is a global organization to promote digital identity and to encourage the further adoption of OpenID, the OIDF has encouraged the creation of member chapters. Member chapters are officially part of the Foundation and work within their own constituency to support the development and adoption of OpenID as a framework for user-centric identity on the internet.

Intellectual property and contribution agreements

The OIDF ensures that OpenID specifications are freely implementable therefore the OIDF requires all contributors to sign a contribution agreement. This agreement both grants a copyright license to the Foundation to publish the collective specifications and includes a patent non-assertion agreement. The non-assertion agreement states that the contributor will not sue someone for implementing OpenID specifications.

Legal issues

The OpenID trademark in the United States was assigned to the OpenID Foundation in March 2008.[16] It had been registered by NetMesh Inc. before the OpenID Foundation was operational.[17][18] In Europe, as of August 31, 2007, the OpenID trademark is registered to the OpenID Europe Foundation.[19]

The OpenID logo was designed by Randy "ydnar" Reddig, who in 2005 had expressed plans to transfer the rights to an OpenID organization.[20]

Since the original announcement of OpenID, the official site has stated:[21]

> Nobody should own this. Nobody's planning on making any money from this. The goal is to release every part of this under the most liberal licenses possible, so there's no money or licensing or registering required to play. It benefits the community as a whole if something like this exists, and we're all a part of the community.

Sun Microsystems, VeriSign and a number of smaller companies involved in OpenID have issued patent non-assertion covenants covering OpenID 1.1 specifications. The covenants state that the companies will not assert any of their patents against OpenID implementations and will revoke their promises from anyone who threatens, or asserts, patents against OpenID implementors.[22][23]

Security

Authentication bugs

In March, 2012, a research paper[24] reported two generic security issues in OpenID. Both issues allow an attacker to sign into a victim's relying party accounts. For the first issue, OpenID and Google (an Identity Provider of OpenID) both published security advisories to address it.[25][26] Google's advisory says "An attacker could forge an OpenID request that doesn't ask for the user's email address, and then insert an unsigned email address into the IDPs response. If the attacker relays this response to a website that doesn't notice that this attribute is unsigned, the website may be tricked into logging the attacker in to any local account." The research paper claims that many popular websites have been confirmed vulnerable, including Yahoo! Mail, smartsheet.com, Zoho, manymoon.com, diigo.com. The researchers have notified the affected parties, who have then fixed their vulnerable code.

For the second issue, the paper called it "Data Type Confusion Logic Flaw", which also allows attackers to sign into victim's RP accounts. Google and PayPal were initially confirmed vulnerable. OpenID published a vulnerability report[27] on the flaw. The report says Google and PayPal have applied fixes, and suggest other OpenID vendors to check their implementations.

Phishing

Some observers have suggested that OpenID has security weaknesses and may prove vulnerable to phishing attacks.[28][29][30] For example, a malicious relaying party may forward the end-user to a bogus identity provider authentication page asking that end-user to input their credentials. On completion of this, the malicious party (who in this case also controls the bogus authentication page) could then have access to the end-user's account with the identity provider, and then use that end-user's OpenID to log into other services.

In an attempt to combat possible phishing attacks, some OpenID providers mandate that the end-user needs to be authenticated with them prior to an attempt to authenticate with the relying party.[31] This relies on the end-user knowing the policy of the identity provider. In December 2008, the OpenID Foundation approved version 1.0 of the Provider Authentication Policy Extension (PAPE), which "enables Relying Parties to request that OpenID Providers employ specified authentication policies when authenticating users and for OpenID Providers to inform the Relying Parties which policies were actually used."[32]

Privacy and trust issues

Other security issues identified with OpenID involve lack of privacy and failure to address the trust problem.[33] However, this problem is not unique to OpenID and is simply the state of the Internet as commonly used.

The Identity Provider does, however, get a log of your OpenID logins; they know when you logged into what website, making cross-site tracking much easier. A compromised OpenID account is also likely to be a more serious breach of privacy than a compromised account on a single site.

Authentication hijacking in unsecured connection

Another important vulnerability is present in the last step in the authentication scheme when TLS/SSL are not used: the redirect-URL from the identity provider to the relying party. The problem with this redirect is the fact that anyone who can obtain this URL (e.g. by sniffing the wire) can replay it and get logged into the site as the victim user. Some of the identity providers use nonces (number used once) to allow a user to log into the site once and fail all the consecutive attempts. The nonce solution works if the user is the first one to use the URL. However, a fast attacker who is sniffing the wire can obtain the URL and immediately reset a user's TCP connection (as an attacker is sniffing the wire and knows the required TCP sequence numbers) and then execute the replay attack as described above. Thus nonces only protect against passive attackers, but cannot prevent active attackers from executing the replay attack.[34] Use of TLS/SSL in the authentication process can significantly reduce this risk.

This can be restated as:

```
IF (Both RP1 and RP2 have Bob as a client) AND       // a common case
   (Bob uses the same IDP with both RP1 and RP2) AND // a common case
```

```
       (RP1 does not use VPN/SSL/TLS to secure their connection with the client) // preventable!
    THEN
      RP2 could obtain credentials sufficient to impersonate Bob with RP1
    END-IF
```

Covert Redirect

On May 1, 2014, a bug dubbed "Covert Redirect related to OAuth 2.0 and OpenID" was disclosed.[35][36] It was discovered by mathematics doctoral student Wang Jing at the School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore.[37][38][39]

The announcement of OpenID is: "'Covert Redirect', publicized in May 2014, is an instance of attackers using open redirectors – a well-known threat, with well-known means of prevention. The OpenID Connect protocol mandates strict measures that preclude open redirectors to prevent this vulnerability."[40]

"The general consensus, so far, is that Covert Redirect is not as bad, but still a threat. Understanding what makes it dangerous requires a basic understanding of Open Redirect, and how it can be exploited."[41]

A patch was not immediately made available. Ori Eisen, founder, chairman and chief innovation officer at 41st Parameter told Sue Marquette Poremba, "In any distributed system, we are counting of the good nature of the participants to do the right thing. In cases like OAuth and OpenID, the distribution is so vast that it is unreasonable to expect each and every website to patch up in the near future".[42]

History

The original OpenID authentication protocol was developed in May 2005[43] by Brad Fitzpatrick, creator of popular community website LiveJournal, while working at Six Apart.[44] Initially referred to as Yadis (an acronym for "Yet another distributed identity system"),[45] it was named OpenID after the openid.net domain name was given to Six Apart to use for the project.[46] OpenID support was soon implemented on LiveJournal and fellow LiveJournal engine community DeadJournal for blog post comments and quickly gained attention in the digital identity community.[47][48] Web developer JanRain was an early supporter of OpenID, providing OpenID software libraries and expanding its business around OpenID-based services.

In late June, discussions started between OpenID users and developers from enterprise software company NetMesh, leading to collaboration on interoperability between OpenID and NetMesh's similar Light-Weight Identity (LID) protocol. The direct result of the collaboration was the Yadis discovery protocol, adopting the name originally used for OpenID. The new Yadis was announced on October 24, 2005.[49] After a discussion at the 2005 Internet Identity Workshop (http://iiw.idcommons.net) a few days later, XRI/i-names developers joined the Yadis project,[50] contributing their Extensible Resource Descriptor Sequence (XRDS) format for utilization in the protocol.[51]

In December, developers at Sxip Identity began discussions with the OpenID/Yadis community[52] after announcing a shift in the development of version 2.0 of its Simple Extensible Identity Protocol (SXIP) to URL-based identities like LID and OpenID.[53] In March 2006, JanRain developed a Simple Registration (SREG) extension for OpenID enabling primitive profile-exchange[54] and in April submitted a proposal to formalize extensions to OpenID. The same month, work had also begun on incorporating full XRI support into OpenID.[55] Around early May, key OpenID developer David Recordon left Six Apart, joining VeriSign to focus more on digital identity and guidance for the OpenID spec.[48][56] By early June, the major differences between the SXIP 2.0 and OpenID projects were resolved with the agreement to support multiple personas in OpenID by submission of an identity provider URL rather than a full identity URL. With this, as well as the addition of extensions and XRI support underway, OpenID was evolving into a full-fledged digital identity framework, with Recordon proclaiming "We see OpenID as being an umbrella for the framework that encompasses the layers for identifiers, discovery, authentication and a messaging services layer that sits atop and this entire thing has sort of been dubbed 'OpenID 2.0'.[57] " In late July, Sxip began to merge its Digital Identity Exchange (DIX) protocol into OpenID, submitting initial drafts of the OpenID Attribute Exchange (AX) extension in August. Late in 2006, a ZDNet opinion piece made the case for OpenID to users, web site operators and entrepreneurs.[58]

On January 31, 2007, Symantec announced support for OpenID in its Identity Initiative products and services.[59] A week later, on February 6 Microsoft made a joint announcement with JanRain, Sxip, and VeriSign to collaborate on interoperability between OpenID and Microsoft's Windows CardSpace digital identity platform, with particular focus on developing a phishing-resistant authentication solution for OpenID. As part of the collaboration, Microsoft pledged to support OpenID in its future identity server products and JanRain, Sxip, and VeriSign pledged to add support for Microsoft's Information Card profile to their future identity solutions.[60] In mid-February, AOL announced that an experimental OpenID provider service was functional for all AOL and AOL Instant Messenger (AIM) accounts.[61]

In May, Sun Microsystems began working with the OpenID community, announcing an OpenID program,[62] as well as entering a non-assertion covenant with the OpenID community, pledging not to assert any of its patents against implementations of OpenID.[22] In June, OpenID leadership formed the OpenID Foundation, an Oregon-based public benefit corporation for managing the OpenID brand and property.[63] The same month, an independent OpenID Europe Foundation was formed in Belgium[64] by Snorri Giorgetti. By early December, non-assertion agreements were collected by the major contributors to the protocol and the final OpenID Authentication 2.0 and OpenID Attribute Exchange 1.0 specifications were ratified on December 5.[65]

In mid-January 2008, Yahoo! announced initial OpenID 2.0 support, both as a provider and as a relying party, releasing the provider service by the end of the month.[66] In early February, Google, IBM, Microsoft, VeriSign and Yahoo! joined the OpenID Foundation as corporate board members.[67] Around early May, SourceForge, Inc. introduced OpenID provider and relying party support to leading open source software development website SourceForge.net.[68] In late July, popular social network service MySpace announced support for OpenID as a provider.[69] In late October, Google launched support as an OpenID provider and Microsoft announced that Windows Live ID would support OpenID.[70] In November, JanRain announced a free hosted service, RPX Basic, that allows websites to begin accepting OpenIDs for registration and login without having to install, integrate and configure the OpenID open source libraries.[71]

In January 2009, PayPal joined the OpenID Foundation as a corporate member, followed shortly by Facebook in February. The OpenID Foundation formed an executive committee and appointed Don Thibeau as executive director. In March, MySpace launched their previously announced OpenID provider service, enabling all MySpace users to use their MySpace URL as an OpenID. In May, Facebook launched their relying party functionality,[72][73] letting users use an automatic login-enabled OpenID account (e.g. Google) to log into Facebook.[74]

In September 2013, Janrain announced that MyOpenID.com would be shut down on February 1, 2014; a pie chart showed Facebook and Google dominate the social login space as of Q2 2013.[75] Facebook has since left OpenID; it is no longer a sponsor, represented on the board, or permitting OpenID logins.[15][76]

In May 2016, Symantec announced that they would be discontinuing their pip.verisignlabs.com OpenID personal identity portal service.[77][78]

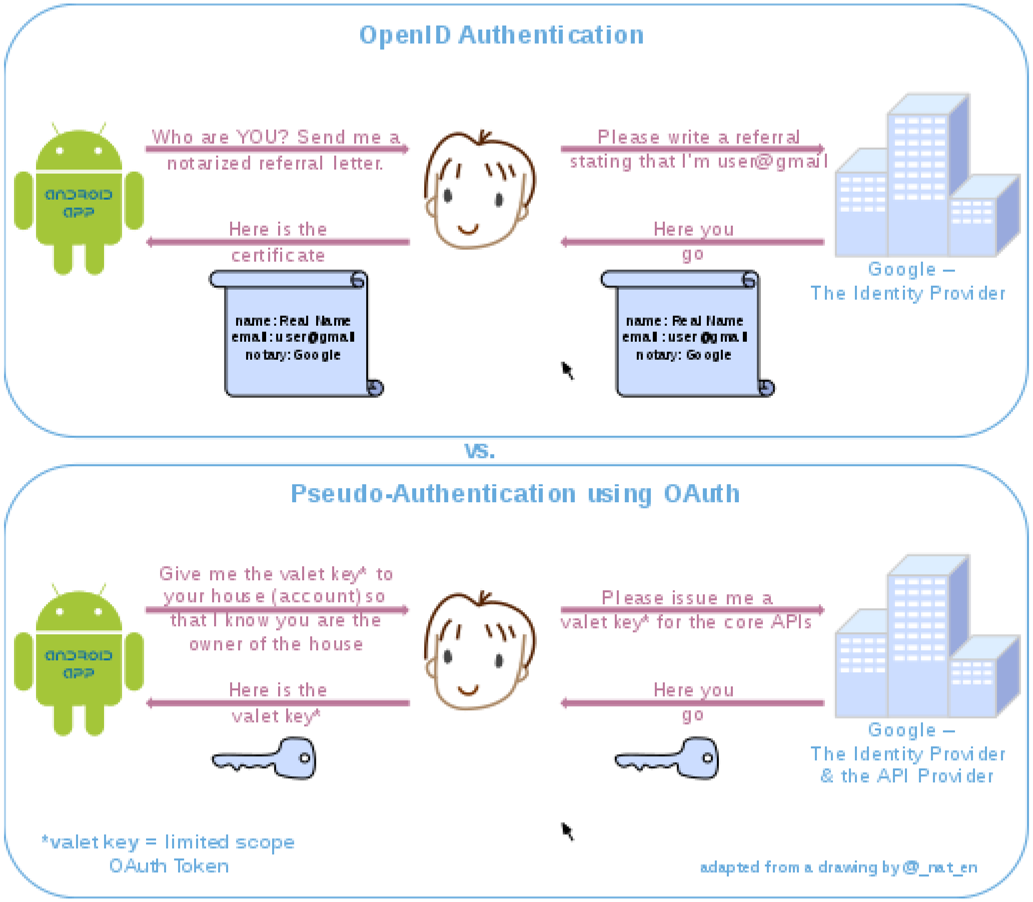
In March 2018, Stack Overflow announced an end to OpenID support, citing insufficient usage to justify the cost. In the announcement, it was stated that based on activity, users strongly preferred Facebook, Google, and e-mail/password based account authentication.[79]

OpenID vs. pseudo-authentication using OAuth

OpenID is a way to use a single set of user credentials to access multiple sites, while OAuth facilitates the authorization of one site to access and use information related to the user's account on another site. Although OAuth is not an authentication protocol, it can be used as part of one.

> Authentication in the context of a user accessing an application tells an application who the current user is and whether or not they're present.[...]Authentication is all about the user and their presence with the application, and an internet-scale authentication protocol needs to be able to do this across network and security boundaries.
>
> However, OAuth tells the application none of that. OAuth says absolutely nothing about the user, nor does it say how the user proved their presence or even if they're still there. As far as an OAuth client is concerned, it asked for a token, got a token, and eventually used that token to access some API. It doesn't know anything about who authorized the application or if there was even a user there at all. In fact, much of the point of OAuth is about giving this delegated access for use in situations where the user is not present on the connection between the client and the resource being accessed. This is great for client authorization, but it's really bad for authentication where the whole point is figuring out if the user is there or not (and who they are).[80]

The following drawing highlights the differences between using OpenID vs. OAuth for authentication. Note that with OpenID, the process starts with the application asking the user for their identity (typically an OpenID URI), whereas in the case of OAuth, the application directly requests a limited access OAuth Token (valet key) to access the APIs (enter the house) on user's behalf. If the user can grant that access, the application can retrieve the unique identifier for establishing the profile (identity) using the APIs.


OpenID Authentication
android app
Who are YOU? Send me a notarized referral letter.
Here is the certificate
name: Real Name
email: user@gmail
notary: Google
Please write a referral stating that I'm user@gmail
Here you go
name: Real Name
email: user@gmail
notary: Google
Google – The Identity Provider
vs.
Pseudo-Authentication using OAuth
android app
Give me the valet key* to your house (account) so that I know you are the owner of the house
Here is the valet key*
Please issue me a valet key* for the core APIs
Here you go
Google – The Identity Provider & the API Provider
*valet key = limited scope OAuth Token
adapted from a drawing by @_nat_en

Attack against pseudo-authentication

OpenID provides a cryptographic verification mechanism that prevents the attack below against users who misuse OAuth for authentication.

Note that the valet key does not describe the user in any way, it only provides limited access rights, to some house (which isn't even necessarily the user's, they just had a key). Therefore if the key becomes compromised (the user is malicious and managed to steal the key to someone else's house), then the user can impersonate the house owner to the application who requested their authenticity. If the key is compromised by any point in the chain of trust, a malicious user may intercept it and use it to impersonate user X for any application relying on OAuth2 for pseudo authentication against the same OAuth authorization server. Conversely, the notarized letter contains the user's signature, which can be checked by the requesting application against the user, so this attack is not viable. [81]

Verifying the letter

The letter can use public-key cryptography to be authenticated.

- The requesting application provides its encryption public key to the user, which provides it to the Authentication server.
- The authentication server encrypts a document containing an encryption key which corresponds to a one way hash of a secret the user knows (e.g. passphrase) for challenge response using the application's public key.
- The user passes the encrypted document back to the application, which decrypts it.
- The application encrypts a random phrase using the received encryption key, and asks that the user do the same, then compares the results, if they match, the user is authentic.

OpenID Connect

Published in February 2014 by the OpenID Foundation, the third generation of OpenID technology, OpenID Connect, is an authentication layer that sits on top of the OAuth 2.0 authorization framework.[82] It allows computing clients to verify the identity of an end-user based on the authentication performed by an authorization server, as well as to obtain the basic profile information about the end-user in an interoperable and REST-like manner. In technical terms, OpenID Connect specifies a RESTful HTTP API, using JSON as a data format. OpenID Connect allows a range of parties, including web-based, mobile and JavaScript clients, to request and receive information about authenticated sessions and end-users. The OpenID Connect specification is extensible, supporting optional features such as encryption of identity data, discovery of OpenID providers, and session management.

See also

- Authorization
- Athens access and identity management
- BrowserID
- Central Authentication Service
- IndieAuth
- Information Card
- Liberty Alliance
- Light-Weight Identity
- SAML
- Shibboleth (Shibboleth Consortium)
- Single sign-on
- SQRL
- WebFinger
- WebID
- WS-Federation

References

1. Eldon, Eric (2009-04-14). "Single sign-on service OpenID getting more usage" (https://venturebeat.com/2009/04/14/single-sign-on-service-openid-getting-more-usage/). venturebeat.com. Retrieved 2009-04-25.
2. "What is an OpenID?" (http://openid.net/get-an-openid/what-is-openid/). Retrieved 19 June 2014.
3. "OpenID Authentication 2.0 specification – Final" (http://openid.net/specs/openid-authentication-2_0.html). Retrieved 2011-10-24.

4. "OpenID Attribute Exchange 1.0 – Final" (http://openid.net/specs/openid-attribute-exchange-1_0.html). Retrieved 2011-10-24.
5. "OpenID Authentication 2.0 - Final" (http://openid.net/specs/openid-authentication-2_0.html). 2007-12-05. Retrieved 2014-05-18.
6. "OpenID Usage Statistics" (https://trends.builtwith.com/docinfo/OpenID).
7. bashburn, bill (2008-04-22). "BBC Joins OpenID Foundation" (http://openid.net/2008/04/22/british-broadcasting-corp-bbc-joins-openid-foundation/).
8. "Technology Leaders Join OpenID Foundation to Promote Open Identity Management on the Web" (http://www-03.ibm.com/press/us/en/pressrelease/23461.wss). 2008-02-07.
9. "PayPal Access Uses OpenID 2.0" (http://openid.net/2011/10/19/paypal-access-uses-openid-2-0/). OpenID ·. Retrieved 19 June 2014.
10. "Steam Community :: Steam Web API Documentation" (http://steamcommunity.com/dev). Retrieved 2012-02-10.
11. Perez, Juan Carlos. "Facebook, Google launch data portability programs to all" (http://www.networkworld.com/article/2270803/data-center/facebook--google-launch-data-portability-programs-to-all.html). Network World, Inc. Retrieved 19 June 2014.
12. "It's spring cleaning time for Blogger" (https://blogger.googleblog.com/2018/05/its-spring-cleaning-time-for-blogger.html). Blogger team. Retrieved 10 September 2019.
13. "OpenID Authentication 1.1#Delegation" (http://openid.net/specs/openid-authentication-1_1.html#delegating_authentication).
14. Paul Tarjan. "Easy OpenID Delegation with Yadis" (https://web.archive.org/web/20090704234010/http://blog.paulisageek.com/2009/06/easy-openid-delegation-with-yadis.html). Archived from the original (http://blog.paulisageek.com/2009/06/easy-openid-delegation-with-yadis.html) on 2009-07-04. Retrieved 2009-06-30.
15. "Leadership" (http://openid.net/foundation/leadership/). openID Foundation. Retrieved 19 June 2014.
16. "Trademark Assignment, Serial #: 78899244" (http://assignments.uspto.gov/assignments/q?db=tm&sno=78899244). United States Patent and Trademark Office. 2008-05-06. Retrieved 2008-05-19. "Exec Dt: 03/27/2008"
17. "Latest Status Info" (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=78899244). United States Patent and Trademark Office. 2006-03-27. Retrieved 2008-03-20.
18. "NetMesh: Company / Management" (https://web.archive.org/web/20070830095651/http://netmesh.us/company/management/). *NetMesh*. Archived from the original (http://netmesh.us/company/management/) on 2007-08-30. Retrieved 2008-03-20.
19. "OpenID Europe Trademark & Logo Policy" (https://web.archive.org/web/20080309221914/http://www.openideurope.eu/policies/openid-trademark-policy/). *OpenID Europe Foundation*. Archived from the original (http://www.openideurope.eu/policies/openid-trademark-policy/) on 2008-03-09. Retrieved 2008-03-20.
20. Reddig, Randy (2005-06-29). "OpenID Logo" (http://lists.danga.com/pipermail/yadis/2005-June/000990.html). *Danga Interactive*. Retrieved 2008-03-20.
21. Fitzpatrick, Brad. "Intellectual Property" (http://openid.net/intellectual-property/).
22. "Sun OpenID: Non-Assertion Covenant" (http://www.sun.com/software/standards/persistent/openid/nac.xml). *Sun Microsystems*. Retrieved 2008-03-20.
23. "VeriSign's OpenID Non-Assertion Patent Covenant" (https://web.archive.org/web/20080415211645/http://www.verisign.com/research/Consumer_Identity_and_Profile_Management/042160.html). *VeriSign*. Archived from the original (http://www.verisign.com/research/Consumer_Identity_and_Profile_Management/042160.html) on 2008-04-15. Retrieved 2008-03-20.
24. Rui Wang; Shuo Chen & XiaoFeng Wang. "Signing Me onto Your Accounts through Facebook and Google: a Traffic-Guided Security Study of Commercially Deployed Single-Sign-On Web Services" (http://research.microsoft.com/apps/pubs/default.aspx?id=160659).
25. "Attribute Exchange Security Alert" (http://openid.net/2011/05/05/attribute-exchange-security-alert/).

26. "Security advisory to websites using OpenID Attribute Exchange" (http://googlecode.blogspot.com/2011/05/security-advisory-to-websites-using.html).
27. "Vulnerability report: Data confusion" (http://openid.net/2012/03/14/vulnerability-report-data-confusion/).
28. Crowley, Paul (2005-06-01). "Phishing attacks on OpenID" (http://lists.danga.com/pipermail/yadis/2005-June/000470.html). *Danga Interactive*. Retrieved 2008-03-20.
29. Anderson, Tim (2007-03-05). "OpenID still open to abuse" (http://www.itweek.co.uk/2184695). IT Week. Retrieved 2007-03-13.
30. Slot, Marco. "Beginner's guide to OpenID phishing" (http://openid.marcoslot.net/). Retrieved 2007-07-31.
31. "Verisign PIP FAQ" (https://web.archive.org/web/20081113065317/https://pip.verisignlabs.com/faq.do#faq5). Archived from the original (https://pip.verisignlabs.com/faq.do#faq5) on 2008-11-13. Retrieved 2008-11-13.
32. Jones, Mike. "PAPE Approved as an OpenID Specification" (http://openid.net/2008/12/31/pape-approved-as-an-openid-specification/). OpenID Foundation.
33. Stefan Brands (2007-08-22). "The problem(s) with OpenID" (https://web.archive.org/web/20110516013258/http://www.untrusted.ca/cache/openid.html). Archived from the original (http://www.untrusted.ca/cache/openid.html) on 2011-05-16. Retrieved 2010-12-12. (originally published on The Identity Corner at www.idcorner.org/?p=161)
34. Tsyrklevich, Eugene. "Single Sign-On for the Internet: A Security Story" (https://www.blackhat.com/presentations/bh-usa-07/Tsyrklevich/Whitepaper/bh-usa-07-tsyrklevich-WP.pdf) (PDF). Blackhat USA. Retrieved 2012-04-19.
35. "Serious security flaw in OAuth, OpenID discovered" (https://www.cnet.com/news/serious-security-flaw-in-oauth-and-openid-discovered/). CNET. 2 May 2014. Retrieved 10 November 2014.
36. "Covert Redirect" (http://tetraph.com/covert_redirect/). Tetraph. 1 May 2014. Retrieved 10 November 2014.
37. "Facebook, Google Users Threatened by New Security Flaw" (https://news.yahoo.com/facebook-google-users-threatened-security-192547549.html). Yahoo. 2 May 2014. Retrieved 10 November 2014.
38. "Nasty Covert Redirect Vulnerability found in OAuth and OpenID" (http://thehackernews.com/2014/05/nasty-covert-redirect-vulnerability.html). The Hacker News. 3 May 2014. Retrieved 10 November 2014.
39. "Math student detects OAuth, OpenID security vulnerability" (http://techxplore.com/news/2014-05-math-student-oauth-openid-vulnerability.html). Tech Xplore. 3 May 2014. Retrieved 10 November 2014.
40. "Covert Redirect" (http://openid.net/2014/05/15/covert-redirect/). OpenID. 15 May 2014. Retrieved 10 November 2014.
41. " 'Covert Redirect' vulnerability impacts OAuth 2.0, OpenID" (http://www.scmagazine.com/covert-redirect-vulnerability-impacts-oauth-20-openid/article/345407/). SC Magazine. 2 May 2014. Retrieved 10 November 2014.
42. "Lessons to be Learned from Covert Redirect" (http://www.the41.com/buzz/press/lessons-be-learned-covert-redirect). 41st Parameter. 5 May 2014. Retrieved 10 November 2014.
43. Fitzpatrick, Brad (2005-05-16). "Distributed Identity: Yadis" (https://web.archive.org/web/20060504054201/http://community.livejournal.com/lj_dev/683939.html). *LiveJournal*. Archived from the original (http://community.livejournal.com/lj_dev/683939.html) on 2006-05-04. Retrieved 2008-03-20.
44. Waters, John K (2007-12-01). "OpenID Updates Identity Spec" (https://web.archive.org/web/20080208155322/http://reddevnews.com/news/devnews/article.aspx?editorialsid=913). *Redmond Developer News*. Archived from the original (http://reddevnews.com/news/devnews/article.aspx?editorialsid=913) on 2008-02-08. Retrieved 2008-03-20.
45. "Glossary" (http://www.livejournal.com/doc/server/appx.glossary.html). LiveJournal Server: Technical Info. Retrieved 13 October 2009.

46. Lehn, David I. (18 May 2005). "18 May 2005" (https://web.archive.org/web/20101221200243/http://advogato.org/person/dlehn/diary/5.html). *Advogato blog for dlehn*. Advogato. Archived from the original (http://www.advogato.org/person/dlehn/diary/5.html) on 21 December 2010. Retrieved 13 October 2009. "They were looking for a name and managed to email me about openid.net right before I was going to offer it to them. So I gave it to them for the new and improved OpenID project."

47. "OpenID: an actually distributed identity system" (https://web.archive.org/web/20050924033518/http://www.danga.com/openid/). 2005-09-24. Archived from the original (http://www.danga.com/openid/) on 2005-09-24. Retrieved 2008-03-20.

48. Fitzpatrick, Brad (2006-05-30). "brad's life – OpenID and SixApart" (https://web.archive.org/web/20070425033329/http://brad.livejournal.com/2226738.html). *LiveJournal*. Archived from the original (http://brad.livejournal.com/2226738.html) on 2007-04-25. Retrieved 2008-03-20.

49. Recordon, David (2005-12-24). "Announcing YADIS...again" (http://lists.danga.com/pipermail/yadis/2005-October/001511.html). *Danga Interactive*. Retrieved 2008-03-20.

50. Reed, Dummond (2005-12-31). "Implementing YADIS with no new software" (http://lists.danga.com/pipermail/yadis/2005-October/001544.html). *Danga Interactive*. Retrieved 2008-03-20.

51. Reed, Drummond (2008-11-30). "XRD Begins" (http://www.equalsdrummond.name/?p=172). *Equals Drummond*. Retrieved 5 January 2009.

52. Hardt, Dick (2005-12-18). "Sxip concerns with YADIS" (http://lists.danga.com/pipermail/yadis/2005-December/001873.html). *Danga Interactive*. Retrieved 2008-03-20.

53. Hardt, Dick (2005-12-10). "SXIP 2.0 Teaser" (https://web.archive.org/web/20070814212337/http://identity20.com/?p=44). *Identity 2.0*. Archived from the original (http://identity20.com/?p=44) on 2007-08-14. Retrieved 2008-03-20.

54. Hoyt, Josh (2006-03-15). "OpenID + Simple Registration Information Exchange" (http://lists.danga.com/pipermail/yadis/2006-March/002304.html). *Danga Interactive*. Retrieved 2008-03-20.

55. Grey, Victor (2006-04-02). "Proposal for an XRI (i-name) profile for OpenID" (http://lists.danga.com/pipermail/yadis/2006-April/002388.html). *Danga Interactive*. Retrieved 2008-03-20.

56. Recordon, David (2006-04-29). "Movin' On..." (https://web.archive.org/web/20061020010916/http://daveman692.livejournal.com/251286.html) *LiveJournal*. Archived from the original (http://daveman692.livejournal.com/251286.html) on 2006-10-20. Retrieved 2008-03-20.

57. Recordon, David (2006-06-16). "Moving OpenID Forward" (http://lists.danga.com/pipermail/yadis/2006-June/002631.html). *Danga Interactive*. Retrieved 2008-05-19.

58. Johannes Ernst and David Recordon. Editor:Phil Becker (2006-12-04). "The case for OpenID" (https://www.zdnet.com/blog/digitalid/the-case-for-openid/78). *ZDNet*. Retrieved 2010-12-12.

59. "Symantec Unveils Security 2.0 Identity Initiative at DEMO 07 Conference" (http://www.symantec.com/about/news/release/article.jsp?prid=20070131_01). *Symantec*. 2007-01-31. Retrieved 2008-03-20.

60. Graves, Michael (2007-02-06). "VeriSign, Microsoft & Partners to Work together on OpenID + Cardspace" (https://web.archive.org/web/20080503001116/http://blogs.verisign.com/infrablog/2007/02/verisign_microsoft_partners_to_1.php). *VeriSign*. Archived from the original (http://blogs.verisign.com/infrablog/2007/02/verisign_microsoft_partners_to_1.php) on 2008-05-03. Retrieved 2008-03-20.

61. Panzer, John (2007-02-16). "AOL and 63 Million OpenIDs" (https://web.archive.org/web/20080511162600/http://dev.aol.com/aol-and-63-million-openids). *AOL Developer Network*. Archived from the original (http://dev.aol.com/aol-and-63-million-openids) on 2008-05-11. Retrieved 2008-03-20.

62. "Sun Microsystems Announces OpenID Program" (http://www.prnewswire.com/cgi-bin/stories.pl?ACCT=104&STORY=/www/story/05-07-2007/0004582105&EDATE=). *PR Newswire*. 2007-05-07. Retrieved 2008-03-20.

63. OpenID Board of Directors (2007-06-01). "OpenID Foundation" (http://openid.net/foundation/). Retrieved 2008-03-20.

64. OpenID Europe Foundation (http://www.openideurope.eu/foundation/)

65. "OpenID 2.0...Final(ly)!" (http://openid.net/2007/12/05/openid-2_0-final-ly/). *OpenID Foundation*. 2007-12-05. Retrieved 2008-03-20.
66. "Yahoo! Announces Support for OpenID; Users Able to Access Multiple Internet Sites with Their Yahoo! ID" (https://web.archive.org/web/20080304014817/http://biz.yahoo.com/bw/080117/20080117005332.html). *Yahoo!*. 2008-01-17. Archived from the original (http://biz.yahoo.com/bw/080117/20080117005332.html) on 2008-03-04. Retrieved 2008-03-20.
67. "Technology Leaders Join OpenID Foundation to Promote Open Identity Management on the Web" (http://www.marketwire.com/mw/release.do?id=818650). *OpenID Foundation*. Marketwire. 2008-02-07. Retrieved 2008-03-20.
68. "SourceForge Implements OpenID Technology" (https://web.archive.org/web/20080513100231/http://www.primenewswire.com/newsroom/news.html?d=142213) (Press release). SourceForge, Inc. May 7, 2008. Archived from the original (http://www.primenewswire.com/newsroom/news.html?d=142213) on May 13, 2008. Retrieved 2008-05-21.
69. "MySpace Announces Support for "OpenID" and Introduces New Data Availability Implementations" (http://www.businesswire.com/news/home/20080722006024/en). *Business Wire*. MySpace. 2008-07-22. p. 2. Retrieved 2008-07-23.
70. "Microsoft and Google announce OpenID support" (http://openid.net/2008/10/30/microsoft-and-google-announce-openid-support/). OpenID Foundation. 2008-10-30.
71. "JanRain Releases Free Version of Industry Leading OpenID Solution" (https://web.archive.org/web/20081218054500/http://www.janrain.com/press/2008/rpxnow) (Press release). JanRain, Inc. November 14, 2008. Archived from the original (http://www.janrain.com/press/2008/rpxnow) on December 18, 2008. Retrieved 2008-11-14.
72. "Facebook Developers | Facebook Developers News" (https://web.archive.org/web/20091223072159/http://developers.facebook.com/news.php?blog=1&story=246). Developers.facebook.com. 2009-05-18. Archived from the original (https://developers.facebook.com/news.php?blog=1&story=246) on 2009-12-23. Retrieved 2009-07-28.
73. "Facebook now accepts Google account logins" (http://www.pocket-lint.com/news/news.phtml/24185/facebook-accepting-google-login-openid.phtml). Pocket-lint.com. 2009-05-19. Retrieved 2009-07-28.
74. "OpenID Requirements – Facebook Developer Wiki" (https://web.archive.org/web/20091223094413/http://wiki.developers.facebook.com/index.php/OpenID_Requirements). Wiki.developers.facebook.com. 2009-06-26. Archived from the original (http://wiki.developers.facebook.com/index.php/OpenID_Requirements) on 2009-12-23. Retrieved 2009-07-28.
75. Kane, Zee M (4 September 2013). "MyOpenID to shut down. Will be turned off on February 1, 2014" (https://thenextweb.com/insider/2013/09/04/myopenid-to-shut-down/). *The Next Web*. Retrieved 5 September 2013.
76. "OpenID Sponsoring Members" (http://openid.net/foundation/sponsoring-members/). Retrieved 17 April 2014.
77. "Symantec Personal Identification Portal banner indicates service will be discontinued on 12 September 2016" (https://web.archive.org/web/20160611151621/https://pip.verisignlabs.com/login.do). Archived from the original (https://pip.verisignlabs.com/login.do) on 11 June 2016. Retrieved 17 May 2016.
78. "Is Symantec failing hard at being Google?" (https://what.thedailywtf.com/topic/19880/is-symantec-failing-hard-at-being-google/12). 7 May 2016. Retrieved 17 May 2016.
79. "Support for OpenID ended on July 25, 2018" (https://meta.stackexchange.com/questions/307647/support-for-openid-ends-on-july-1-2018).
80. "User Authentication with OAuth 2.0" (http://oauth.net/articles/authentication/). *OAuth.net*. Retrieved 19 March 2015.
81. "Why is it a bad idea to use plain oauth2 for authentication?" (https://security.stackexchange.com/questions/133065/why-is-it-a-bad-idea-to-use-plain-oauth2-for-authentication/134280#134280). *Information Security Stack Exchange*. Retrieved 7 July 2018.
82. "OpenID Connect FAQ and Q&As" (http://openid.net/connect/faq/). Retrieved 25 August 2014.

External links

- Official website (https://openid.net/)
- OpenID (https://curlie.org/Computers/Security/Authentication/Single_Sign-On/OpenID) at Curlie

Retrieved from "https://en.wikipedia.org/w/index.php?title=OpenID&oldid=995914471"