Identity management on the web with Open ID

ID CHECK

OpenID offers an open standard for logging on to closed-door websites.

BY NILS MAGNUS

Web 2.0 is not everybody's idea of a good thing. One problem is the multitude of password-protected websites. Personal blogs, virtual communities from Xing to Facebook, and sites that manage workflow, expenses, and vacation planning often rely on web-based applications with private user accounts. This overload of passwords and login boxes is causing some hapless users to lose track of all the options. As long as users work on a single, local, physical device (or can access a server in the vicinity), tools such as the password managers offered by most web browsers, as well as alternatives such as KDE wallet, are useful aids. But the Web 2.0 paradigm assumes the user can move about and log in from different locations.

Community Approach

Identity management solutions provide a more mobile and flexible solution for simplifying web login. These tools often employ the principle of a trusted third party. A few large global players have stepped up with services that offer single-source, trusted third party login solutions. Microsoft's Passport system was created in line with this belief. Today, Microsoft markets Passport as "Windows Live ID" [1]. Many users, however, are wary of becoming dependent on proprietary applications.

An early alternative known as the Liberty Alliance Project [2] offered a more open approach, but it was widely regarded as an overspecified dinosaur, and the Liberty Alliance still has not found widespread acceptance despite a seven-year effort. The OpenID project, under the auspices of the OpenID Foundation [3], relies on simpler functionality that can be integrated more easily into online authentication systems.

Users who choose the OpenID alternative do not enter a user name but, instead, identify themselves with a URI (Uniform Resource Identificator) that can be displayed in a web browser. The URI can be a web address offered by an

OpenID service, such as *http://nilsmagnus.myopenid.com* with Myopenid [4]. The type of identity is not important as long as a browser can access the page. The page needs to add a tag that points to the service provider:

```
<link rel="openid.server" href=➥
"http://www.myopenid.com/server" />

<link rel="openid.delegate" href= ➥
"http://nilsmagnus.myopenid.com/" />
```

The provider-side server specifies the first relation; the second restates the name of the identity. The provider will typically set up a page to make this information available to websites that request login information. However, a user could alternatively integrate the necessary details into a personal homepage or blog. In that case, your own address would serve as your OpenID.

Logging In

An application that supports OpenID will display an OpenID login field in addition to the traditional login page. For instance, Figure 1 shows the OpenID authentication page for the Amarok wiki.

When a user enters the URI-based ID, the web application, which is known as a *consumer* in OpenID speak, retrieves the server portion of the URI. OpenID refers to the server as the *identity provider*.

The consumer (the Amarok wiki, in this case) asks the provider for the name associated with the URI. To do so, it redirects to the provider's website, and the provider indicates who has issued the request. The user then agrees to the request by entering a password. The provider redirects the browser back to the consumer site, where the user is now logged in.

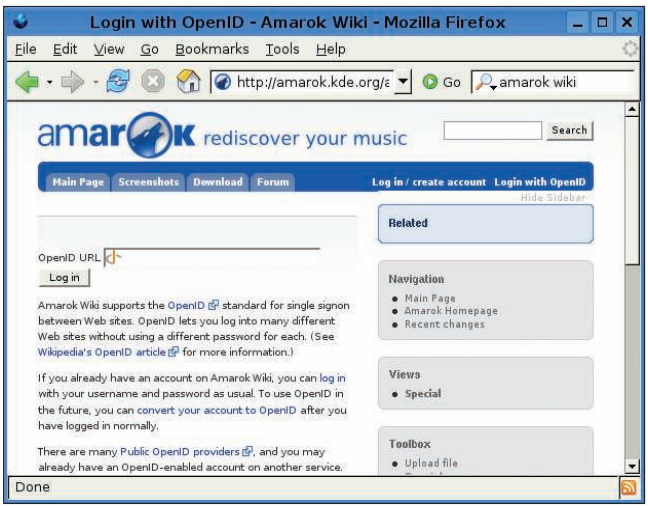

Figure 1: The Amarok wiki lets users login with OpenID.

One practical feature of OpenID is that you can store several attributes for a single identity with a provider, for example, your full name, your preferred language, or your date of birth (see Figure 2). A user who receives a consumer request can specify which data the provider should disclose to the consumer and which data to keep secret.

This approval process is important for preventing the misuse of a critical parameter, such as a bank PIN, which you can actually store along with your ID. Some providers let the user create multiple personae, each with a separate set of attributes.

This approach is also known as User-Centric Identity Management, in that every user can individually define the information the provider supplies to inquiring consumers.

Some identity providers issue identities for free. It is up to the user to decide which provider to trust. In contrast to the centralized approach used by Passport, a decentralized collection of OpenID providers compete with each other to offer services. Users are even free to set up their own provider.

If you are interested in developing your own solution, open source packages are available in programming languages such as Perl, PHP, Ruby, Python, and Java [7].

Security Concerns

You might be asking yourself how secure OpenID is if anybody can act as a provider. Can an attacker spoof or hijack an identity? The first question points to a classic security issue: If you can manipulate a third party website, you can redirect people to your own identity provider or write one to fit the bill.

In other words, security is in the hands of the people running the hosting site. Considering the code quality of many sites written in popular scripting

Identity Management and Federation

OpenID is not the only identity management project. Feder ID [9], for example, is an open source project from France. One of the project's contributors, Clément Oudot, underlined the importance of digital identities for access to web resources in a recent interview with Linux Magazine.

According to Oudot, many users possess a separate identity for each website. This is major issue for large enterprises and organizations, as users need to memorize multiple passwords. Feder ID provides tools for synchronizing identity repositories. These attributes are not only available to a single local organization; they can be shared by trusted partners.

The Feder ID tools are open source licensed and comply with the IETF (Internet Engineering Task Force), OASIS (Organization for the Advancement of Structured Information Standards), and Liberty Alliance standards for identity management.

INFO

[1] Microsoft Passport: *http://www.passport.net*

[2] Liberty Alliance Project: *http://www.projectliberty.org*

[3] OpenID project: *http://openid.net*

[4] Myopenid (provider): *http://myopenid.com*

[5] Mediawiki extension for OpenID: *http://www.mediawiki.org/wiki/Extension:OpenID*

[6] Drupal support for OpenID: *http://drupal.org/project/openid*

[7] Open Source libraries for OpenID: *http://wiki.openid.net/Libraries*

[8] "Keeping Customers and Merchants Secure", Whitepaper, Secure Computing: *http://www.securecomputing.com/webform.cfm?id=289&ref=pci*

[9] Feder ID: *http://federid.objectweb.org*

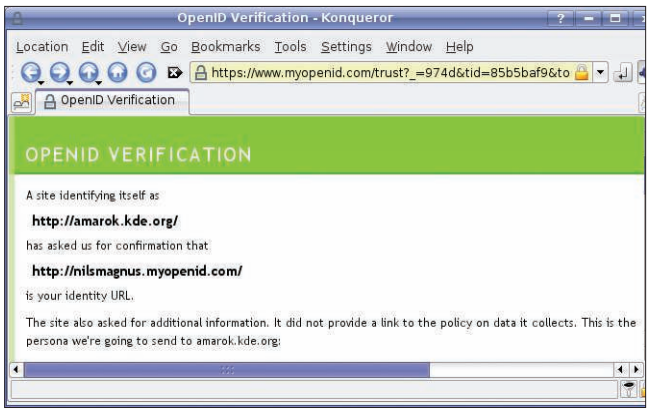

Figure 2: The provider specifies the details it intends to send to the consumer.

languages, this is a concern, but not a fundamental argument against OpenID.

The second question is trickier: Can a hacker sniff the communications between the consumer and identity provider and store the sessions? After all, the provider sends a confirmation message in case of successful authentication. An attacker could try to present a recorded session as credentials for a new login. However, this can be prevented by enabling OpenID SSL/TLS to secure the connection and adding a challenge to each request. This approach means that any response will be valid once only, which rules out trivial recycling.

Despite this, it is not a good idea to underestimate the complexity of stateful session management in a stateless protocol like http, which is the basis for OpenID. The fact that various web applications have been compromised is a clear indication of the dangers, assuming you believe surveys and whitepapers [8].

Practical and Open

OpenID is a step in the right direction for identity management. Because OpenID implements single sign-on, it becomes more convenient for users by reducing the number of passwords you need to remember. The ability to manage attributes is far more powerful than it appears at first glance.

The number of websites using OpenID continues to skyrocket, but some really big applications will still need to prove whether they fulfill all of the operative and conceptional requirements with respect to trust and availability. ■

Table 1: Identity Providers for Web Apps

Provider	OpenID
AOL	*http://openid.aol.com/Screenname*
Blogger	*http://Blogname.blogspot.com*
Flickr	*http://www.flickr.com/photos/Username*
Livedoor	*http://profile.livedoor.com/Username*
Livejournal	*http://Username.livejournal.com*
Technorati	*http://technorati.com/people/technorati/Username*
Wordpress	*http://Username.wordpress.com*
Yahoo	*http://openid.yahoo.com*