Replacing the NASCAR experience with the identities you actually have

Mike Jones – Microsoft

April 6, 2010

Background

- NASCAR experience has sites guess what identities you may have
  - Works for large OPs in NASCAR page
  - Poor experience when using vast majority of Ops
    - Hosted sites, like asu.edu
    - Your company OP
  - Likely identities very different in US, FR, JP, CN

Solution

- Enable you to bring the identities you actually have to the sites you use

- Requires support for an optional active client

Example (from Nov '09 summit demo)
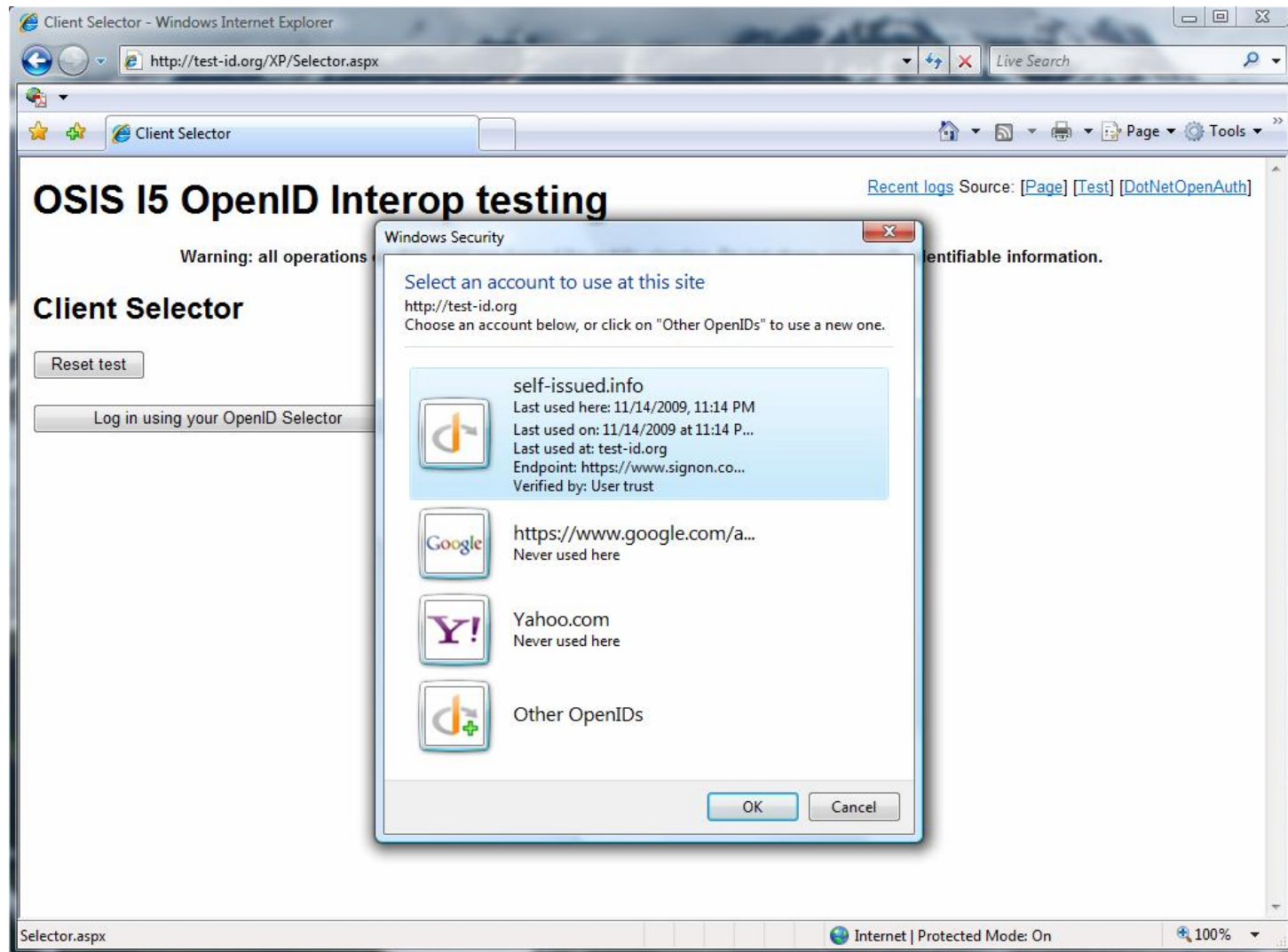
Benefits

- Practical user choice of OPs enhanced
  - Aligns with OpenID vision
- Client can provide phishing protection
- More consistent user experience

Requirements

- Way for RP to publish requirements to client
- Way for RP to detect presence of client
- Way for RP to invoke client

Example Publishing RP Requirements

```
<object type="application/x-informationCard" id="infoCardObjectTag">
    <param name="protocol" value="http://specs.openid.net/auth/2.0" />
    <param name="tokenType" value="http://specs.openid.net/auth/2.0" />
    <param name="issuer" value="Google.com/accounts/o8/id Yahoo.com myOpenID.com" />
    <param name="issuerExclusive" value="false" />
    <param name="OpenIDAuthParameters" value=
"openid.ns:http://specs.openid.net/auth/2.0
openid.return_to:http://www.plaxo.com/openid?actionType=complete
openid.realm:http://*.plaxo.com/
openid.ns.sreg:http://openid.net/extensions/sreg/1.1
openid.sreg.required:email
openid.sreg.optional:fullname,nickname,dob,gender,postcode,country,language,timezone
openid.sreg.policy_url:http://www.plaxo.com/about/privacy_policy
" />
</object>
```

Example RP Detecting Active Client

```html
<head>
    <script type="text/javascript">
        function onLoad()
        {
            var elemObjectTag = document.getElementById( "objectTag" );
            if (elemObjectTag.hasCapability &&
                elemObjectTag.hasCapability("openIdExperimental"))
            {
                // selector exists
            }
            else
            {
                // selector does not exist
            }
        }
    </script>
</head>

<body onload='onLoad();'>
    <object type='application/x-informationCard' id='objectTag'>
        <!-- necessary object tag parameters for OpenID here -->
    </object>
</body>
```

Example Ways for RP to Invoke Client

- Form submit for object tag in form
- Invoking .value method on object tag

For v.Next

- Syntax could be completely different
  - Possibly integrated with discovery
- I believe the requirements stand

What about Rich Client Applications?

- API could invoke active client
  - Triggers interaction with OP
  - Delivers security token to application
- Protocol details TBD for v.Next
  - Possible relationship to WRAP rich client profile

Closing Thought: Personal OPs

- A personal OP in your active client
- No conceptual reason a 3rd party OP needed
- Ultimate example of bringing your identities with you to the site

Contact Info

- Mike Jones
- mbj@microsoft.com
- http://self-issued.info/

Backup Slides

Prototype OpenID Selector Diagram

Legend:
- ★ New component
- ☆ Changed component

RP
- Web
- OID Library ☆

OP
- Web
- OID Library

Browser
- Browser Extension ★

Selector ★
- Known OPs, OP/RP pairs, recent usage
- UI

1. Sign in page with auth request parameters in Object Tag
2. User chooses to sign in with OpenID
3. Trigger Selector and pass request parameters
4. Build list of potential OPs
5. Perform discovery for new OPs
6. Categorize OPs as known or unknown
7. Display OPs to user
8. Select OP or enter OpenID identifier
9. Utilize trusted or untrusted UX flow
10. Record pertinent data
11. Construct authentication request
12. Send auth request to OP on behalf of the RP via browser extension
13. Selector closes
14. Auth request to OP on behalf of the RP
15. Potential user interaction
16. Auth response from OP to RP (exactly as OpenID does today)

User