

The Future of Federated Identity: **IVORY TOWER OR IN THE TRENCHES?**

The path to developing federated identity technology hasn't been straight or smooth. Technology which began with a narrow focus is now being seen as having a much larger purpose in networked computing. Multiple "standard" protocols have emerged, creating concerns that a protocol war could break out just as customers realize how much they need this technology. The protocols we see today originated from different experiences, and their stories foreshadow what is to come.

The development of federated identity began when customers of various "web access management" vendors began calling for single sign on between enterprises. Several vendors, realizing that a one-off, proprietary solution could not solve the problem, started talking about collaboration. One group, led by Securant (later acquired by RSA) began working on a standard they called AuthXML to link access control systems. Simultaneously, a second group, led by Netegrity and Verisign, began work on a similar effort known as S2ML. Even at this early stage, customers could see interoperability problems looming. Reacting to those concerns Burton Group analyst Phil Schacter began urging a combined effort. In response, roughly ten vendors gathered at the DeAnza Hotel in San Jose to see if they could agree on how they would enable cross-enterprise single sign on.

SAML

The "DeAnza group," as they came to call themselves, eventually decided to do this work within OASIS. They donated their work on AuthXML and S2ML to a collective effort that came to be called SAML (the "Security Assertion Markup Language"). The first version of the SAML standard was drafted on January 9, 2001 and reflected its mission to enable enterprise SSO. After a year and a half of work, SAML 1.0 was ratified as an OASIS standard in November 2002.

The Liberty Alliance

Steve Ballmer first hinted at Microsoft's Passport and Hailstorm initiatives at a 1999 press conference. By March 2001, they had become official. Fears that Microsoft might be gathering too much personal information about individuals via Passport (and Hailstorm) combined with security concerns to focus criticism on the "centralized" nature of Microsoft's first identity based service.



Behind the scenes, companies were also concerned. Senior management of VISA, in particular, was disturbed by the potential competition of Passport. On a phone call between Scott McNealy of Sun Microsystems and VISA USA senior management, it was decided that they would attempt to form an organization that could address some of the “network identity” issues that Microsoft had raised. A phone call or two later and an organization named “The Liberty Alliance” was formed.

Federated Identity

With the formation of the Liberty Alliance, the federated identity standards game was officially underway – in fact it was the Liberty Alliance that caused the term “federated identity” to become popularized. Prompted by its formation as a reaction to the centralized Passport, Liberty soon found itself pushing identity linkage in the most decentralized, privacy enhancing fashion possible. In the process they came to see the potential of what was first called

“network identity” and ultimately “federated identity.”

Liberty built its specifications on two draft protocols, WS-Security and SAML. They published their first specification just seven months after formation, igniting the federated identity race. Within eighteen months that standards race created a flurry of specifications either directly related to or secondarily related to federation: SAML1.0, SAML1.1, Liberty1.0,

Liberty1.1, Liberty1.2 (also known as ID-FF 1.2), Liberty ID-WSF (web services framework), SPML, etc. It also created a much larger group of people with a much larger picture of federated identity.

Missions Differ

SAML had been formulated by vendors to address a very specific customer problem (cross-enterprise single sign on.) Liberty was an organization largely run by end-user companies. That distinction led Liberty to become an organization focused on business process problems first, subsequently developing technical specifications to address them.

This led SAML and Liberty to address similar problems, but with different outlooks. As a result, their approaches to solving problems were different in orientation and motivation. Indeed, it has been asserted in some circles that SAML was the vendors' way of being able to claim interoperability without ever actually achieving it. This statement arose from the fact that SAML 1.0 is such a broad specification that any implementation of it would require custom extensions – thus, rendering every implementation devoid of the very “standards-based interoperability” that SAML was supposed to achieve.

In that context, Liberty can be seen as an extension of SAML that seeks to solve the interoperability problems by specifying use cases more completely. The evolution of Liberty and SAML has continued as vendors hear more clearly every day that they must create interoperable products in the federated identity space. But as these standards evolve, difficulties appear. SAML 1.1 was incompatible with SAML 1.0 and both were incompatible with Liberty. The cry for convergence is growing, and Liberty has contributed the ID-

FF portion of its work to SAML for incorporation into SAML 2.0

Coming versions of the SAML specification are expected to address convergence, but it is difficult to see how this can happen without again becoming incompatible with present versions. SAML 2.0 will certainly clear the air a lot, but it will still not address the depth of federated identity that the current Liberty specifications do.

A Star Filled Horizon?

The history of Liberty and SAML protocols reflects their heritage of trying to solve problems tactically, while under pressure to deploy rapidly and relieve the pain in the marketplace. Microsoft and IBM have taken a very different approach to federated identity, trying to integrate it into the fabric of a web services infrastructure. Because of that approach, their efforts, commonly referred to as the “WS-*” (pronounced “WS-star”) specifications, are developing more slowly.

The history of the WS-* specifications goes back some time as well. “The whole effort got started back in when we were finishing up the first round of our .NET application server technologies,” said Microsoft Software Architect, John Shewchuk. “They were really the first commercial implementation of a service oriented architecture based on web services. The first part of that effort was taking a look at some of the kinds of requirements we saw around security.”

In Service of the Architecture

“The Liberty approach and some of the early work that had gone on in Microsoft around Passport were focused around the notion of single sign on,” said Shewchuk. “With [WS-*] we were starting from a fairly different point, which was how do we make it possible for people to build very sophisticated service oriented applications running the whole gamut – everywhere from things like device drivers all the way up to the largest scale cross enterprise sce-

narios – with things like off line, with things like reliable messaging integration with things like transaction integration. While we think what’s going on with the Liberty approach is great and it gives us a good single sign on solution, it doesn’t provide what we are looking for in terms of that entire stack of integration that we needed.”

So Microsoft went to work on designing security architecture for SOAP protocol web services. “We’ve spent a bunch of time here at Microsoft thinking about what that would look like,” said Shewchuk. “We came up with a model that, in particular, supported both public key technologies as well as symmetric key technologies, as well as some of the more advanced DRM style technologies. And we put together a spec that encompassed all of those things.”

The Chicago Meeting

Early on, Shewchuk indicates that Microsoft reached out to IBM on this effort. “Back in 1997, early in the year, I proposed that we talk to the folks at IBM because I thought that they would have very similar kinds of requirements,” said Shewchuk. “After a long series of internal meetings here at Microsoft, we finally arranged a meeting where I got everybody from the executive staff on our side and on the IBM side to fly out to Chicago and meet with me and the IBM technical people. And we asked this kind of interesting question, which is ‘couldn’t we make Websphere and .NET talk to each other through open public protocols?’ This was a pretty radical way of thinking about interoperability. It was going to be based on a common run time – like the Java approach – it was going to be based on protocols which could interoperate with completely different stacks on either side. We walked through the details and we said ‘you know, we could do this.’ And everybody agreed that this would be good for our customers and good for the industry. So we launched the effort to go start building the WS-* stack which we have been working on for the last several years.”

The foundation of the WS-* stack, the WS-Security protocol, was recently ratified as an OASIS standard. "That WS security process kind of set in motion the overall game plan," said Shewchuk. "Get a spec out, do a rev, do interoperability, send it to standards bodies after you've proven the specification, submit it royalty-free, and then grow the collection of people working on it each time you move forward one of those steps."

"That's what we've been doing with not just WS-security but all of the subsequent specs," said Shewchuk. "We have even formalized it a little bit further. We now have these formal processes called workshops where the first kinds are feedback workshops. People come, they read over the specs, they give us feedback on it [and] typically, we revise the specs based on that feedback. Then we do an interoperability workshop where we actually make sure that the things work. Then [the specs] go through a rev. Then we go submit them to the standards body. A good example of that is a couple of weeks ago we had a number of vendors come in for an interoperability workshop around WS-Federation."

Cross-Vendor Compatibility

Microsoft has been promoting interoperable WS-* adoption. The first of these interop workshops happened this spring. "These workshops that we've been going through that [IBM, Netegrity, Oblix, Open Network, Ping ID, RSA Security] and others have participated in, are going to be very important," said Shewchuk. "Because they are establishing the way that multiple organizations that are providing these single sign on solutions will be able to integrate with that overall proposal. I think we are heading very rapidly toward a world where most organizations will be able to do federations between not just IBM and Microsoft, but all of the major vendors are going to be able to hook these things together in the upcoming year or two."

Tactical vs. Strategic

The development of SAML, Liberty, and WS-* shapes up as a difference in outlooks on the mission at hand. One approach gets you out of the blocks fast to solve known problems. The other intends to create a framework in which both known and unknown problems can be solved.

With a new technology that is as poorly understood as federated identity, both approaches are needed. Only by being in the real world, deploying technology, can we learn the real implications of a new technology. Tactical approaches implement about as much as we really understand how to do, and in the process show us what things look like just over the next hill. Some amount of iteration at this level is required to learn what concepts as large as federated identity really mean. But such an approach also creates a trail of deployed "legacy protocol" solutions.

Strategic approaches on the other hand, can suffer from "ivory tower" syndrome. Because so much thought and study go into the generalities of them, they can become isolated from the needs of the real world and create solutions for problems that don't exist while missing problems that do exist. Those working on WS-* are trying to address this by bringing in multiple vendors to workshops on a regular basis which will also force the real world into the project.

Which will win? It depends largely on whether the use cases we now see and understand for federated identity have generally the same structures as what we learn we need in the future. If so, then some version of SAML or Liberty will solve a sufficient number of problems that the strategic capabilities won't be in much demand. If not, then by the time the WS-* solutions start to appear, the world will eagerly look to them to "clean up the mess" and create true cross-vendor interoperability over a much larger set of capabilities. In any case, all of these protocols will likely have to coexist for a long time." ■