Distributed Identity Case Studies - Part 2: The Microsoft/IBM Web Services (WS) Security Framework and Privacy (November 2003)

Francis Vierboom, Galexia Consulting

This paper is the second part in a series of case studies from Galexia Consulting on **distributed identity**.

Part One[1] considered the model and its privacy impacts generally, as well as assessing two case studies – the Liberty Alliance standard and the Irish REACH e-government system.

This paper considers another option, the Web Services (WS) Security framework developed by Microsoft and IBM (also together with Verisign, RSA and BEA).

This paper is available in the following formats from <http://consult.galexia.com>:

- Distributed identity – Case studies – Part 2 – HTML[2]
- Distributed identity – Case studies – Part 2 – PDF[3]

Contents

[1] Galexia Consulting, *Distributed Identity Case Studies - Part 1*, September 2003, <http://consult.galexia.com/public/research/articles/research_articles-pa02.html>.

[2] HTML version of this paper: <http://consult.galexia.com/public/research/articles/research_articles-pa03.html>.

[3] PDF version of this paper: <http://consult.galexia.com/public/research/assets/gc_distributed_identity_paper_part2_200311.pdf>.

1. WS (Web Services) overview

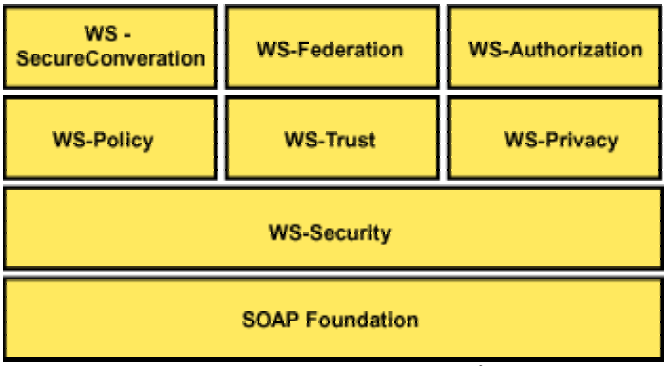
In April 2002, IBM and Microsoft jointly published the whitepaper *Security in a Web Services World*[4], dubbed the 'roadmap' of a secure web services framework, together with its founding WS-Security specification. The roadmap set out a body of protocols that would use XML[5] messages as a standard way for computers to communicate service requests to each other, regardless of the software or hardware they used – the web services model.

For now, the WS framework provides an advanced security infrastructure for integrating enterprise IT systems, and only for a small number of ambitious companies. Eventually it aims to provide security for a new generation of distributed applications for both consumers and businesses.

WS is, at best, a skeleton distributed identity system. The nature and content of the information exchanged is not dealt with by the WS specifications. Other IBM/Microsoft specifications associated this framework (WS-Transactions and WS-ReliableMessaging) enhance the technical soundness of the framework rather than address specific business transactions or information flows. Indeed, only WS-Federation and WS-Privacy really contemplate the identity federation applications of the WS system. But WS provides a highly advanced 'infrastructure' necessary for such information exchange.

1.1. Technical overview

The web services security roadmap sets out the seven standards that were planned to form the WS security architecture, as well as noting the SOAP standard on which it is based.



The initial framework plan[6]

SOAP[7] (Simple Object Access Protocol) is a recognised standard for passing messages between web applications. It enables different programs running on different operating systems in different countries to pass each other data and request operations.

[4] IBM Corporation and Microsoft Corporation, *Security in a Web Services World*, April 2002, <http://www-106.ibm.com/developerworks/webservices/library/ws-secmap/>.

[5] Extensible Markup Language (XML): <http://www.w3.org/XML/>

[6] Image from <http://www-106.ibm.com/developerworks/webservices/library/ws-secmap/>.

[7] More information: *SOAP Version 1.2 Part 0: Primer*, W3C, June 2003 <http://www.w3.org/TR/2003/REC-soap12-part0-20030624/>.

WS-Security[8] defines extensions to the SOAP standard to include encryption information such as PKI certificates and Kerberos tickets in a 'security token' attached to the SOAP message. Since its release in April 2002, WS-Security has become a standard adopted by OASIS[9](the Organisation for the Advancement of Structured Information Standards). WS-Security, as an extension of SOAP, provides the foundation layer for the IBM/Microsoft WS framework and has also been adopted by the Liberty Alliance project.[10]

The security provided at this level (the 'message layer') is independent of 'transport layer' encryption like SSL, so it could be as readily used on an internal corporate network as a 'normal' internet HTTP connection.

WS-Policy[11] defines a framework for exchanging 'policy assertions' between web services systems. For example, one application may have a requirement or preference that a requestor of information uses PKI certificate encryption. However, WS-Policy is not restricted to any particular type of rule or preference. It could be used to require that it is a pleasant sunny day at the locale of the requestor. It simply defines how such questions are asked and answered.

Since the initial publication of the roadmap, WS Policy has been renamed[12] **WS-PolicyFramework** and extended to include

- **WS-PolicyAttachments**[13] which specifies how to attach a policy to an actual web service;
- **WS-PolicyAssertions**[14] which defines a number of general purpose assertions – for example, language preference or character-set preference; and
- **WS-SecurityPolicy**[15] which defines security assertions for use by WS-Security tokens.

WS-Trust[16] provides a model for establishing 'trust relationships' using WS-Security tokens. It describes how WS-Security tokens can be assessed and managed over multiple systems.

WS-SecureConversation[17] defines how to establish a security context for a WS session among two or more systems for the duration of a conversation.

WS-Federation[18] brings together the four standards defined above to describe a 'federated' web services model, and details the use of identifiers and pseudonyms across service providers and requestors. It also considers the types of transactions that could occur and some of the privacy and security precautions applied to a federated system.

WS-Privacy is not yet published but it is described in the roadmap document. It may use WS-Security (for basic security), WS-Policy (as a structured way to ask privacy questions) and WS-Trust (as a way to manage privacy across several transactions) to provide for privacy controls in web services networks. Systems can use WS-Privacy to make assertions about their privacy practices – for example, they can promise not to pass the data on to any third parties.

---

[8] WS-Security: <http://www-106.ibm.com/developerworks/library/ws-secure/>.

[9] OASIS: <http://www.oasis-open.org/>.

[10] Liberty Alliance, *Liberty Alliance Releases New Specifications, Privacy and Security Guidelines to Drive Development of Identity-Based Web Services*, 15 April 2003, <http://www.projectliberty.org/press/releases/2003-04-15-Phase2.html>.

[11] WS-PolicyFramework: <http://www-106.ibm.com/developerworks/library/ws-polfram/>.

[12] M Hondo, D Melgar, A Nadalin, *Web Services Security: Moving up the stack*, IBM, 1 December 2002, <http://www-106.ibm.com/developerworks/library/ws-secroad/>.

[13] WS-PolicyAttachments: <http://www-106.ibm.com/developerworks/library/ws-polatt/>.

[14] WS-PolicyAssertions: <http://www-106.ibm.com/developerworks/library/ws-polas/>.

[15] WS-SecurityPolicy: <http://www-106.ibm.com/developerworks/library/ws-secpol/>.

[16] WS-Trust: <http://www-106.ibm.com/developerworks/library/ws-trust/>.

[17] WS-SecureConversation: <http://www-106.ibm.com/developerworks/library/ws-secon/>.

[18] WS-Federation: <http://www-106.ibm.com/developerworks/library/ws-fed/>.

WS-Authorisation is also yet to be published. It will determine how security token claims are interpreted and assessed to permit access to web services.

1.2. WS applications

By keeping their specifications at such a broad and low level, only a very small group of very large companies can afford to adopt an IBM web services system in the near future. This is bound to change, as the next version of Microsoft Windows (codenamed 'Longhorn') due out in 2005 includes the WS framework as a major feature[19].

For the immediate future, WS is faced with the classical paradox of distributed computing open standards; they only become useful when everyone has them. Accentuating this paradox for IBM and Microsoft is the fact that the WS project is more abstract than other sector initiatives like Liberty Alliance[20] or MS Passport. In most cases there is no instant gratification from integrating IT systems with WS like there is from integrating identities and customer profiles using Liberty or Passport. Rather, the web services concept is a new way of using distributed systems. WS is only a means, not an end.

However it is conceivably a means to many powerful ends. Implementations of WS can simplify and automate many varied business transactions.

Of course, most practical applications that WS could manage have already been automated to a degree. It is no great feat in 2003 to have an inventory database communicate with a supplier to order parts. But many features of such systems have to be set up within each individual relationship for the purposes of security, privacy and basic compatibility. In fact, usually they need to be running the same software on the same operating system to communicate in any significant way.

A part of the WS goal is to create a new standard layer of automated information flows and business transactions that allows entities like suppliers and inventory databases to communicate easier. However, in doing so, it will need to address privacy - a highly significant area of legislative compliance and the customer relationship that has, until now, been managed by humans and implemented in IT systems on an ad hoc basis.

[19] Mike Ricciuti, *Gates gambles on Longhorn*, CNET News.com, 28 October 2003, <http://news.com.com/2008-1016-5098285.html>.

[20] The Liberty Alliance Project: <http://www.projectliberty.org/>.

1.3. WS-Privacy

The WS framework has been developed with such issues in mind. Thus the WS security architecture (in marked contrast to the Liberty Alliance project) actually includes specific functionality for managing information privacy. The WS-Privacy module will provide a way for WS systems to inform each other of their privacy practices.

As mentioned above, the WS-Privacy specification release is expected late this year. However, we gain some hints about the possible shape of WS-Privacy from a diploma thesis[21] written at the IBM Zürich Research Laboratory. Judging from the thesis, it is possible that WS-Privacy will add a new set of headers for use in a SOAP message to communicating privacy policies that look something like this:

```
<soap-pr:Privacy xmlns:soap-pr="http://schemas/soap/privacy"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <soap-pr:Promise> </soap-pr:Promise>
  <soap-pr:Promise-Ack> </soap-pr:Promise-Ack>
  <soap-pr:Requirement> </soap-pr:Requirement>
  <soap-pr:Requirement-Ack> </soap-pr:Requirement-Ack>
  <soap-pr:Binding> </soap-pr:Binding>
  <ds:Signature> hSK7aasjd89A#)NF983hr9Q*#h(3839h</ds:Signature>
</soap-pr:Privacy>
```

The `soap-pr:Promise` element can be used by a data requester to communicate a privacy assertion – for example, "Information will only be disclosed for purposes related to accounting." Such an assertion is acknowledged by the data provider with the `soap-pr:Promise-Ack` element.

The data provider can also impose privacy requirements on the data recipient using the `soap-pr:Requirement` element – for example, "Information must only be used for purposes related to subject's cheque account." The recipient can then choose to acknowledge that request with `soap-pr:Requirement-Ack` or abandon the transaction.

The `soap-pr:Binding` element is used to translate the vocabulary used in the actual data request (contained in the SOAP body element which is not shown) and vocabulary used in the privacy policy that is included or referred to in a Promise or Requirement.

The `ds:Signature` is a digital signature the provides message integrity (proof the message has not been tampered with) and non-repudiation (proof that the sender actually sent the message).

The contractual nature of such a transaction gives rise to a degree of legal uncertainty. The complexities of contract law developing around electronic transactions is not in the scope of this paper, but it will be important to consider the legal status and precedence of WS-Privacy exchanges in the future. For example, where a malfunctioning WS-Privacy-managed transaction violates a prior agreement between the two parties, which contract will remain effective?

At any rate, before any such legal uncertainty arises, WS systems need to be able to communicate their privacy practices in a standard and machine-readable way.

For consumers, the Platform for Privacy Preferences (P3P)[22] is already an established way for web sites to communicate privacy policies to users in general fashion. For some applications this may be good enough to provide consumers with notice and choice about how their information will be used. P3P is being developed and reformatted into XML[23] for specific use in WS applications. However, P3P cannot realistically be used by enterprise IT systems for high-volume, complex data transactions.

---

[21] Walid Bagga, *Privacy-enabled Application Scenarios for Web Services*, Network Security & Cryptography Group, IBM Zurich Research Laboratory, September 2003, <http://rangiroa.essi.fr/DEA-RSD/2002-03/03-dea-bagga.pdf>.

[22] The Platform for Privacy Preferences (P3P) Project <http://www.w3.org/P3P/>.

[23] An editor's draft schema is publicly available at <http://www.w3.org/P3P/2003/09-Schema.html> .

For these more demanding situations the Enterprise Privacy Authorisation Language (EPAL) is another proposed standard that IBM has recently published[24]. EPAL is a language for expressing privacy rules in terms of the data user, data type, the action to be performed on the data, the purpose of the action, and other context variables (eg data subject's consent, current time, local security environment) stored in 'containers'. By regulating privacy transactions using an EPAL policy layered over a WS-Privacy system, the privacy policy is easy to modify, even by non-technical staff, and a company's privacy policy compliance is simple to audit.

IBM anticipates the use of EPAL in conjunction with WS-Privacy as can be seen in one of its presentations, *Enterprise Privacy and Federated Identity Management*[25] (a presentation which explains the EPAL model well and is recommended to all those interested in this article).

[24] The Enterprise Privacy Authorization Language (EPAL 1.1) <http://www.zurich.ibm.com/security/enterprise-privacy/epal/>.

[25] Ashley et al, *Enterprise Privacy and Identity Management,* IBM Zurich Research Lab and IBM Privacy Research Institute. Presented by Michael Waidner at Almaden Institute 10 April 2003, <http://www.almaden.ibm.com/institute/pdf/2003/MichaelWaidner.pdf> at p12.

2. Privacy and the WS framework

At the time of publishing, prior to the release of WS-Privacy, the WS framework provides only minor privacy features in addition to its security basis. WS-Federation allows for the optional use of 'opaque identifiers' so that no identifying information need be shared for basic identity federation functions. Further, WS-Federation supports single-session identifiers for federation, which is aimed at preventing the long-term tracking of a user's navigation across sites, although such information could still be deduced by a company determined to do so.

WS-Privacy is an encouraging step towards making privacy a functional part of IT systems. But implementations of the specification will not do any better at protecting privacy than they are designed to. As noted in Galexia Consulting's previous paper, *Distributed Identity Case Studies - Part 1*[26], distributed identity systems have a serious impact on privacy. WS-Privacy may provide a useful tool for enforcing the rules but good privacy, just like good business, is based on treating customers fairly and with respect, not just complying with the legislation, as may be the temptation with a WS-Privacy system.

Of course, before criticising WS for privacy shortcomings, it should also be remembered that the WS security framework is essentially a technically focused initiative. At the moment, the WS framework is not used for federated identity and it is only one of several applications for which the framework will eventually be useful. Microsoft and IBM may well deserve the congratulations and support of privacy advocates for treating the issue with such respect at this early stage.

But they must continue to take the responsibility for building privacy solutions into WS. The other notable feature of the WS framework is that while its actual applications are still fairly abstract, it does not generally assume that the two parties are already in some kind of relationship. WS systems will interoperate with other WS systems almost fresh out of the box[27]. The obvious consequence is that at some time in the future, if the use of the WS framework becomes widespread – as it seems bound to with the eventual release of the WS-enabled Longhorn Windows operating system – web services will be able to conduct transactions with 'foreign', unknown systems. The security and privacy risks of such a structure are, naturally, the focus of much of IBM and Microsoft's efforts but they will always be risks, especially considering the fondness hackers have for targeting Microsoft security vulnerabilities.

[26] Galexia Consulting, *Distributed Identity Case Studies - Part 1*, September 2003, <http://consult.galexia.com/public/research/articles/research_articles-pa02.html>.
[27] Lee, Yvonne, *Web Services Federation Now Defined*, SD Times, 1 August 2003, <http://www.sdtimes.com/news/083/story1.htm>.

3. WS and Liberty Alliance – the future of distributed identity

There is growing tension between the WS initiative and the Liberty Alliance[28] federated identity project. The most recent WS specification, jointly published by Microsoft, IBM, Verisign, BEA and RSA Security, WS-Federation, steps sorely on the toes of a large part of the work done by the Liberty consortium (made up of over 160 various companies and organisations). Despite the earlier co-operation shown by Liberty's adoption of the WS-Security standard into their own specifications, the simmering rivalry between Microsoft and Sun Microsystems (one of Liberty's founders and a leading proponent) may yet poison efforts to have the two work together.

Liberty Alliance is a commercially funded and oriented project. Much of its research has been aimed at specific commercial goals related to identity management and federation, in contrast to the more technical motivation of much of the WS work. Thus, Liberty has made far more progress in defining how to manage and exchange personal information across Liberty networks – although this is an area where it will be in competition with MS Passport (which is being reconceived to work on top of WS systems) rather than a WS standard. At this 'application level', Liberty's open approach may well be favoured over the widespread but much maligned Passport.

The deployment of the WS standards within the so-called 'Indigo' software in the next release of Microsoft Windows, however, will likely supplant the more basic infrastructure aspects of the Liberty Alliance specifications. Admittedly, Microsoft is simply providing the WS functionality and developers are free to use other web services software[29], perhaps as much out of fear of antitrust lawsuits as for developer convenience. But without judging the technical merits of either system, it would seem the WS framework, through the muscle of Windows, will render Liberty's low-level technical specifications still-born, especially considering Liberty's focus on consumer-facing applications pitted against Windows' stranglehold on that market.

Liberty's likeliest eventual path appears to be to retreat to higher-level applications above the WS framework. However, depending on the scope of the WS-Privacy specification, there may still be room for more standards in privacy management at this higher level. It remains to be seen if EPAL[30] will gain traction as a privacy language, and if it (or something similar) will be widely implemented in Liberty-style federated identity systems.

Liberty's perspective on WS-Federation is expressed in a recent whitepaper, *Liberty Alliance & WS-Federation: A Comparative Overview*[31]. It is unsurprisingly unkind to the WS-Federation specification and the overall WS framework. It points to weaknesses in the WS framework's information-sharing capabilities and champions Liberty's privacy efforts. This position may be weakened by the eventual public release of WS-Privacy. Curiously, as part of the comparison on sharing privacy policies, Liberty refers to a 'Privacy Policy Expression Language'. At the time of writing Galexia Consulting has been unable to find further public references to this[32]. Additionally this comparison does not acknowledge IBM's EPAL specification.

---

[28] The Liberty Alliance Project: <http://www.projectliberty.org/>. For a more detailed consideration of Liberty Alliance by Galexia Consulting see *Distributed Identity Case Studies - Part 1*, September 2003, <http://consult.galexia.com/public/research/articles/research_articles-pa02.html>.

[29] Elizabeth Montalbano, *Advanced Web Services To Find Home In Microsoft's Indigo*, CRN, 28 October 2003 <http://www.crn.com/sections/BreakingNews/dailyarchives.asp?ArticleID=45569>.

[30] The Enterprise Privacy Authorization Language (EPAL 1.1) <http://www.zurich.ibm.com/security/enterprise-privacy/epal/>.

[31] Liberty Alliance Project: *Liberty Alliance & WS-Federation: A Comparative Overview*, 14 October 2003, <http://www.projectliberty.org/resources/whitepapers/wsfed-liberty-overview-10-13-03.pdf >.

[32] See for example: <http://www.google.com/search?q=%22Privacy+Policy+Expression+Language%22>.

4. Conclusion

It is likely Microsoft and IBM's WS framework will gain a foothold in providing basic interoperable web services. But there may yet be a battle over how privacy and personal information will be managed in a web services enabled world. WS-Privacy and EPAL are promising initiatives that could provide the answer.

They are not simple or cheap solutions. But finding a solution is important. Companies need to be able to comply with regulatory frameworks. And a future hard-wired for thoughtless exchanges of personal information is good news for nobody.