Internet Single Sign-On Systems

Radovan SEMANČÍK*

nLight, s.r.o.
Súľovská 34, 812 05 Bratislava, Slovak Republic
semancik@nlight.sk

Abstract. This document describes the requirements and general principles of Internet Single Sign-On systems. The general model of Internet SSO system is described. The Liberty ID-FF, WS-Federation, Shibboleth, SXIP and LID specifications are considered and of these specifications suitability for implementing an Internet SSO system is evaluated.

1 Introduction

Applications based on the HTTP and HTML are the most commonly used mechanisms for providing live content on the Internet, but the use of independent strong authentication systems in each Internet application directly is inefficient. The solution may be the outsourcing of authentication services to trusted third parties. This document provides overview of systems, that allow user to authenticate on one Internet site and use services on the other Internet site. These systems are referred to as *Single Sign-On (SSO)* systems, because they allow single authentication for multiple services.

1.1 Requirements

The requirements for web application Simplified Sign-On system for the usage on the Internet are defined as follows:

- It must be based on open protocols and standards.
- It must support cross-organization operation.
- It must provide a mechanism to securely share user attributes across organizational boundaries.
- It must support privacy features.

* Supervisor: Doc. Ing. Margaréta Kotočová, CSc., Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

M.Bieliková (Ed.), IIT-SRC 2005, April 27, 2005 pp. 116-123.

- It must support standard web browser, that implements current versions of HTTP, HTML and accompanying standards.

2 Internet SSO Systems Principles

The goal of Simplified Sign-On system is to securely transfer user's identity, attributes and current authentication status of a user from source site (Identity Provider) to the destination site (Service Provider). The trust relationship must be established between source and target sites for a source site to trust the target site's requests and for a target site to trust the source site's identity statements. Establishing and maintaining this trust relationship is out-of-band for most SSO systems. The described Internet SSO systems follow the proxy-based true SSO model according to [1].

The representation of user in computer system will be called in this work *persona*. The persona contains *attributes*. The physical data structure that stores the persona attributes is called *account*.

2.1 User Identifiers and Attributes

The persona identifier may be presented to the target site in different ways:

- *Direct Linking*: Provide the same persona identifier (e.g. username) as the user has established with the source site.
- *Indirect Linking*: Provide a pseudonym for a persona. The pseudonym an identifier that is different as the primary persona identifier established with the source site, but is fixed in time for the same persona and the same target site. Indirect linking may be used to implement pseudonymity [2].
- *No linking*: Do not provide identifier or provide an anonymous handle valid for a single session or a part of session. No linking is used in anonymity scenarios [2].

2.2 Message Exchange

All considered SSO systems employs similar mechanism to transfer authentication status from source to destination site. In all these cases, browser redirect or form processing capabilities are used to transfer security tokens between sites. The process is illustrated on Figure 1 and it consists of following steps:

1. The user **requests resource** on target system (service provider).
2. Target site does not recognize the user (has no valid session for the user/persona). The target site constructs the **authentication request** and returns it to the user's browser in the response. The response is returned in the form of HTTP redirect or HTML form, that will redirect user interaction to the source (identity provider) site.

3. The **authentication request** is received by source site (identity provider). The source site processes the request, and applies any relevant policy.
4. The source site may **authenticate** the user, if not already authenticated or if any policy or the request requires re-authentication.
5. The source site constructs the **authentication response**, which contains the results of persona identity evaluation. The authentication response may contain a security token, that will prove persona identifier and/or attributes to the target site. The authentication response is returned in the HTTP response to the user's browser in a form of HTTP redirect or HTML form, that will redirect user interaction back to the target (service provider) site.
6. The authentication response is received by the target site. The response is processed and the security token is evaluated. For the response and token processing it may be necessary to contact source site directly (6a), for example to resolve references in the response. Note that the token itself may be passed by reference in the authentication response and it may be needed to dereference it by direct communication to the source site. After the response and any related security tokens and processed the persona identifier and/or attributes are determined.
7. The target site applies any relevant policies to the original access request (step 1) combined with the information determined in step 6. If the request is allowed, the target site will in most cases establish a local session with the user's browser. The local session will help avoid quite significant overhead of future re-authentications.

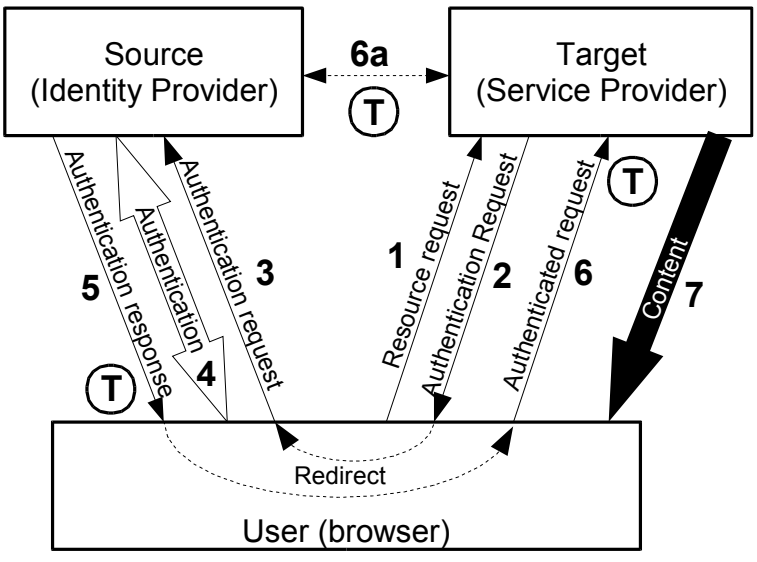

Figure 1 Generic SSO message exchange

3 SSO Systems Overview

The results of Internet SSO systems evaluation are summarized in the following sections.

3.1 Liberty Identity Federation Framework (Liberty ID-FF)

The Liberty ID-FF system [3] is build on top of SAML and uses SAML assertions as a security tokens, therefore it is dependent on SAML. The modification of Liberty specification for other, non-SAML security tokens may be difficult.

The Liberty specifications mandates the use of pseudonyms by default. This requirement may help to enforce good privacy features to all Liberty-compliant implementations.

3.2 Web Services Federation Language (WS-Federation)

The WS-Federation specification [4] are built on top of WS-Trust and WSS: SOAP Message Security (WS-Security) specifications. The WS-Security specifications leaves a lot of details to the implementers decisions and to be defined by the service policy. Although that is good for flexibility, it brings additional degree of complexity to the system. The implementing systems may not be interoperable by implementing different subsets of specifications and/or using non-compatible policies.

The privacy decisions (e.g. use of pseudonyms) is left to the implementers. This may in practice lead to the implementations, that will not adhere to the best practice and the level of privacy in WS-Federation-compliant systems may be lower (in average).

3.3 Shibboleth

The Shibboleth system [5] is built on top of SAML specifications. The system implemented using Shibboleth specifications will need to specify a lot of local details, e.g. name identifier types, linking policies, etc. This type of flexibility may lead to situation that two shibboleth-compliant implementations will not interoperate.

The shibboleth will depend on other specification to define account linking, if such will be needed. The use of transient name identifiers allow good degree of privacy, but for any practical purpose will require a solid attribute service.

3.4 SXIP Network

The SXIP protocol [6] has two variations: simple and XML commands. The variation using simple commands provides only minimal security. Even the use of HTTPS does not add any real security to the simple command exchange.

Some of the SXIP XML commands include XML digital signature element, that should protect the integrity of SxipML message. However, no method or guidelines are documented for creation and validation of these signatures.

The storex and fetchx command messages are not authenticated, which may lead to the implementations that may allow anyone reading or setting arbitrary persona attributes.

The use of globally unique GUPI at several membersites makes it easy for the membersites to collude and correlate persona activities at several sites. This is partially mitigated by the use of different personae for different membersites, but in practice this approach may be inconvenient or maybe even unfeasible.

3.5 Lightweight Identity (LID)

The LID system [7] uses URL as a persona identifier and GPG signature as a security token. LID URL as an identifier may leak information, especially in self-hosting scenario as proposed by LID documentation. For detailed explanations see section .

GPG public key validation is left on simple "callback" method. No other method is mandated by the LID documentation. The described simple method can be dangerous when using HTTP protocol, for example due to the DNS attacks [8]. While using HTTPS method to get public key, using SSL/TLS brings a dependency on X.509 PKI. The result is that LID uses two different PKI systems (X.509 and GPG), that are in principle and features nearly the same, but not compatible.

4 Discussion

The considered web SSO systems are similar in the generic SSO mechanism, but are different in the following areas:

- The *method of identifying* personae and accounts, the way of generating and assigning identifiers. Global identifiers are better suited for tightly-coupled systems that are same organizational control or share common policies. Local identifiers are better suited for loose-coupled systems that cross organizational boundaries.
- The *method of linking* accounts and personae in different target and source systems. The direct linking is desirable only when user privacy is not a concern, it is not well suited for the Internet environment. The pseudonymity of indirect linking or anonymity of no linking is better suited to privacy-sensitive environments. The "no linking" SSO case will in practice require secure and interoperable attribute service.
- The *level of detail* that is specified in the documents and the freedom that is left for system implementers.

The Table 1 summarizes features of considered SSO systems and the next subsections provides discussion on some aspects of SSO system's architecture and design.

Tab. 1 Summary of SSO system's features

System	Version	Security Tokens	Linking Method	Persona Identifier	Extensible to Web Services
Liberty ID-FF	1.2	SAML	Indirect	Local	Yes
WS-Federation	1	WS-Trust	Not specified	Not specified	Yes
Shibboleth	Working Draft 09	SAML	None, other	Transient, other	No
LID	Jan 3, 2005	GPG signature	Direct	Global (URL)	No
SXIP	1.0.4	None, XML signature	Direct	Global (GUPI)	No

4.1 Persona identifiers

The Single Sign-On systems need a way to link several account. The linking is implemented by associating persona or account identifiers on different systems. There are two approaches to the management of persona identifiers:

- *Global persona identifiers.* The persona identifiers are allocated by central authority that guarantees global uniqueness of the identifier. The global uniqueness of the identifier allows direct linking of accounts on the global (Internet) scale.
- *Local persona identifiers.* The persona identifiers are allocated by the system, where the persona originated. These identifiers are unique only in the scope of the source system. For the purposes of persona or account linking, the target site must accept the identifier in this form or (more frequently) apply appropriate identifier mapping.

While the direct linking and global persona identifiers may be the easiest scenario, global identifiers shared by many sites may be used to correlate user activities on several systems and thus reveal personal information without user consent. To overcome this problem, lower level virtual personae (with different identifiers) may be used as pseudonyms. If this approach is deployed in the Internet scale, the persona management may become difficult and may need automation. The automatic pseudonym persona management is technically close to the indirect linking scenario, and the indirect linking may be considered as better approach for the Internet environment.

4.2 Self Hosting of Source Sites

One way of storing identity information is to host a source site on a system, that is under user's sole control. As this concept may seem attractive from the privacy point of view, it in fact may be undesirable in the practice:

- URLs of self-hosted source site may leak information.

- The maintaining of site security on the operating system and application by an non-expert user may be a security hazard.
- The trust relation between source site and target site may not be unidirectional. The trust to the source site in the self-hosting scenario may be questionable, and the high number of self-hosting sites with whom to establish trust may make the process unfeasible.

The self-hosting scenario is technically proxy-based true SSO system [1], but may be regarded a local true SSO system from the organizational control point of view. The self-hosting of source sites brings only one advantage: control over the stored data. But the control over data is lost when transmitted to other sites and even control of the stored data itself may be questionable. The self-hosting scenario will in most cases likely lower the privacy level.

5 Conclusion

This document described the generic model for Internet Single Sign-On mechanisms and provided an overview of existing Internet SSO systems. The Liberty Alliance ID-FF and the WS-Federation were found as the most advanced and flexible Internet SSO systems, suitable for general use. The WS-Federation specifications are quite generic, lack a considerable amount of details and the early WS-Federation implementations may have interoperability problems. However, the WS-Federation may become a good platform for SSO services in the future, extended to the web services area as well. The Liberty Alliance ID-FF specifies an practical SSO system built on top of SAML specifications. The level of detail is sufficient for good interoperability, the dependency on SAML is reasonable in the Internet environment. The Liberty Alliance also specifies extensions to ID-FF for web services environments (ID-WSF). The Shibboleth specifications also depends in SAML, but the level of details is considerable lower compared the the Liberty case. The Shibboleth system is suitable for large communities, that are mostly composed of independent organizations (e.g. academic community). The SXIP and LID SSO systems in it's current state are not well suitable for the Internet environment. They provide only minimal privacy features, use global identifiers and feature limited standards support. These systems may be suitable for closed communities or for the environments where security and privacy is not an concern.

While all these systems use similar mechanisms, their properties vary considerably. Especially the use of persona identifiers and pseudonyms as well as the use of attribute services will require further study for the SSO systems to be deployed in secure and privacy-supporting manner.

6 References

1. Pashalidis, A., Mitchell, C.: A taxonomy of single sign-on systems, Proceedings, volume 2727 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, July 2003, pp. 249-264.

2. Pfitzmann, A., Köhntopp, M.: Anonymity, Unobservability, Pseudonymity, and Identity Management A Proposal for Terminology, vol. 2009 of Lecture Notes in Computer Science, Springer-Verlag, 2000, pp. 1-9.
3. Cantor, S., Kemp, J.: Liberty Bindings and Profiles Specification, Liberty Alliance Project Specification, 2003.
4. Bajaj, S., et.al.: Web Services Federation Language (WSFederation), BEA, IBM, Microsoft, RSA Security, Verisign, 2003.
5. Shibboleth Architecture Protocols and Profiles, http://shibboleth.internet2.edu/shibboleth-documents.html.
6. The Simple eXtensible Identity Protocol (SXIP) Reference, https://sxip.net/archive/specs/sxip-reference.pdf.
7. Ernst, J.: Light-Weight Identity, NetMesh Inc., 2005.
8. Bellovin, S. M.,: Using the Domain Name System for System Breakin, 5th USENIX UNIX Security Symposium, 1995, pp. 199-208.