
# SAML 技术手册

# SAML 技术手册

安全是所有 Web 项目在设计时都要考虑的一个重要因素。无论是选择最短口令，决定何时使用 SSL 加密 HTTP 会话，还是通过自动登录 cookie 来识别用户，都经常要付出重大的设计努力，以保护用户的身份信息和他们可能存放于 Web 站点的其他资料。糟糕的安全性可能带来公关灾难。当最终用户努力保持对其个人信息的控制时，他们要面临令人迷惑的隐私政策，需要牢记众多站点的不同口令，以及遭遇“钓鱼式攻击”事件。下面我们就来具体看一下 SAML 在这里所起到的重要作用。

## SAML 定义及特点

安全断言标记语言（SAML，Security Assertion Markup Language）是一种可扩展标识语言（XML）标准，它允许用户登录一次相关但单独的网站。SAML 设计用于商户对商户（B2B）以及商户对顾客（B2C）的事务处理。

- ❖ 安全断言标记语言 SAML

## SAML：不仅仅是为网络服务而定制

在这部分中，我们将来讲讲 SAML 的具体内容，我们知道 SAML 是为了解决 Web 浏览器单点登陆的问题而产生的。但是，SAML1.1 就这个目的而言是有局限的，事实上，SAML1.1 通过使用 SAML 作为 WS-Security 令牌，更有效地解决的问题是 SOAP Web 服务的身份认证和授权。

- ❖ SAML：不仅仅是为网络服务而定制
- ❖ SAML：企业级的 IdP

- SAML：IdP 和 SP 用户存储库
- SAML 断言的说明
- SAML2.0 特性分析

## SAML 的应用

我们试图在一个巨大的电信级产品上执行 Web 服务。数据需要在线路上传输是很关键的，并且要保证数据安全。你能就我的安全框架使用作出建议么？执行和响应时间也是非常重要的。我被这些 WS－安全规格搞得焦头烂额。我想要知道普遍使用的安全框架是什么？

- 安全声明标记语言（SAML）的应用
- 如何让 SAML 适应你的 SOA 安全方案

# 安全断言标记语言 SAML

安全断言标记语言（SAML，Security Assertion Markup Language）是一种可扩展标识语言（XML）标准，它允许用户登录一次相关但单独的网站。SAML 设计用于商户对商户（B2B）以及商户对顾客（B2C）的事务处理。

SAML 定义三种元件：断言、协议以及约束。存在验证、标志、授权三种断言。验证断言确认用户的身份，标志断言包含特定的用户信息，授权断言确认用户得到授权。

协议定义 SAML 如何请求和接收断言。约束定义如何将 SAML 消息交换映射成简单对象访问协议（SOAP）交换。SAML 与多个协议一起工作，包括超文本传输协议（HTTP）、简单邮件传输协议（SMTP）、文件传输协议（FTP），它还支持 SOAP、BizTalk 以及电子商务 XML（ebXML）。结构化信息标准促进组织（OASIS）是 SAML 的标准组织。

*（作者： Gerard Enter　　来源：TechTarget 中国）*

原文链接：http://www.whatis.com.cn/word_4182.htm

# SAML：不仅仅是为网络服务而定制

## 简介

SAML（安全断言标记语言）是一种用来在安全域之间交换身份认证和授权数据的基于 XML 的标准。SAML 是 OASIS 安全服务技术委员会的一个产品。最重要的一个问题是 SAML 是为了解决 Web 浏览器单点登陆的问题而产生的。但是，SAML1.1 就这个目的而言是有局限的，事实上，SAML1.1 通过使用 SAML 作为 WS-Security 令牌，更有效地解决的问题是 SOAP Web 服务的身份认证和授权。有用的技术会找到相应的办法，以一些自然的方式适应广泛的技术前景。例如，Kerberos 曾是苦命的 DCE（分布式计算环境）不可分割的一部分，虽然 DCE 已经明显地沉寂了很久，但 Kerberos 还是成为了微软的核心安全组件。

SAML 2 明确包含了用于解决 Web 浏览器进行多网站验证问题的新特性。现在，关于 SAML，是否保持 1.版本还是转到 2.0 版本，很多组织都正处于“左右为难的处境”。

新近一个项目的准备阶段将涉及保证门户应用安全和使用 SAML 来实现 SSO 的相关 Web 应用，在与客户接洽前我通常做得是——向我的个人专家列表发邮件，并对 SAML 进行调查。对我来说迎面而来的问题是关于这个主题缺乏实质性文档。

## 调查，结论和建议

一些同行推荐了一篇由 Vikrant Sawant 发表在 http://www.oracle.com/technology/pub/articles/dev2arch/2006/12/sso-with-saml.html 上
的名为《用 WebLogic Server 9.2 中的 SAML 来配置单点登陆》的文章。虽然这篇文章有点过时，但它对这个有几分神秘色彩的技术领域仍然具有开创性的贡献。我第一次读它是在一年半前，当时我正参与一个在 SOA 环境内，使用 SAML 实现 SSO 的项目。

虽然本文是一个 SAML 入门有效例子，但基于下面这些原因，它没有为现实的实施提供指导：

- 一个用作 IdP（身份提供者）的 Oracle WebLogic 实例，而不是更倾向于企业级访问的管理应用。
- 既用作 IdP 又用作服务提供商（SP）的本地 LDAP 实例，而不是“虚拟用户”。
- 由此产生的 SAML 断言不包括“组”的属性，它需要提供 RBAC（基于角色的访问控制）。
- 问题检测需要的不仅仅是对源服务器和目标服务器的日志分析，而且要支持应用服务器的调试和 SAML 信息的过滤。
- 可信任伙伴密钥和证书的配置不只是需要配置一个私钥。
- 要在一个独立环境下配置安全模型，而不是在可感知的集群，和生产型的环境下。

本文的目的是提供一个如何在企业范围的平台上构架和实现 SAML 安全的概览。

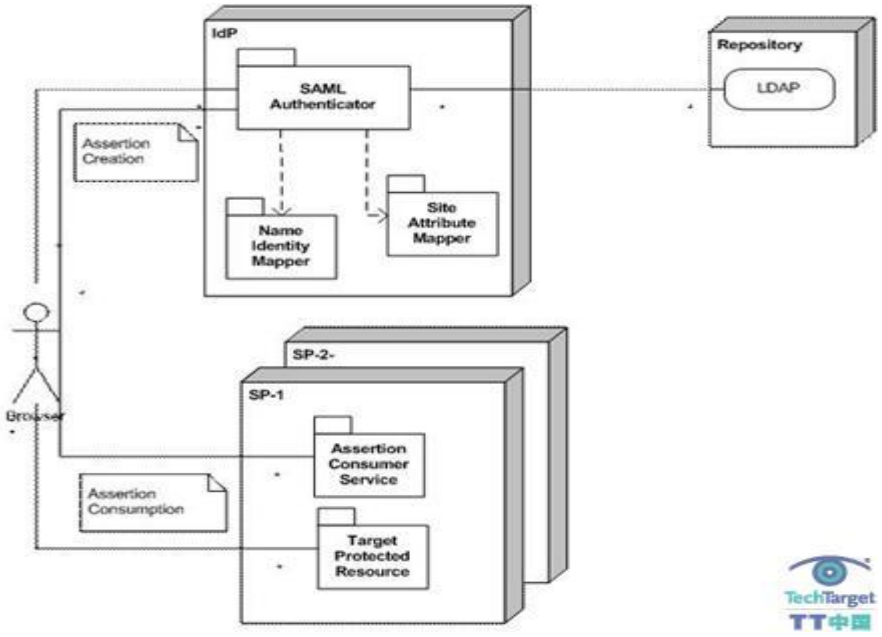
*（作者： Frank Teti 译者：杨晓明 来源：TechTarget 中国）*

原文链接：http://www.searchsoa.com.cn/showcontent_30027.htm

## SAML：企业级的 IdP

虽然使用 Oracle WebLogic 作为 IdP 是显然可行的，但它没有一个真实访问管理环境的特征，诸如 Sun 访问管理器，Tivoli 访问管理器，Oracle 访问管理器等。作为 IdP 的 Oracle WebLogic 服务器不提供，类似像 Sun 访问管理器所提供的会话管理功能。众多厂商支持的 SAML 提供了一个优于企业内部网的 SSO 解决方案。图一描述了一个使用以下技术的参考架构：

- 作为 IdP 的 Sun 访问管理器 7.1；
- 作为 SP 的 WebLogic 服务器 10.3。




图一 理论上 SAML 架构 在此之上包含了一个 LDAP 存储库

虽然很多 IdP 厂商同时支持格式和协议，但因为不同厂商实现方式各不相同，并且都包含了他们自身内部的构造，导致配置 SAML 变得有点像映射工作。随着 Oracle 收购了 Sun， Sun 访问管理器是否会退役而有利于 Oracle 访问管理器还有待观察，而 Oracle 访问管理器本身就是 Oracle 以前从 Oblix 收购的。虽然通过许多了解 SAML 的技术专家在 Sun 访问管理器中实现作为 Idp 的 SAML，类似于在 WebLogic 中实现作为 IdP 的 SAML，但我可以保证配置并不是对称的，可以这么说，正如我所料。


图 2 Sun 访问管理器 7.1 控制台的 SAML 可信任伙伴配置视图

Sun 访问管理器的源 ID 是一个难以理解的数据类型，它就是以 SP 站点“protocol://hostname:port”字符串的 Base64 编码来表示 SHA1，尽管 SP 不需要知道这个 ID。更具体地说，可信任伙伴的配置窗口捕捉了以下信息：

| 屏幕文字/参数 | 属性值 |
|---|---|
| 源 ID | 一个经过编码的，可信任的唯一伙伴 ID |
| 目标 | SP 主机名和端口 |
| 公布 URL | SP 断言用户的 Servlet，WebLogic ITS（站点间传输服务）的一部分，它可以通过 SSL 访问。 |
| 站点属性映射 | 在 IdP 中用来格式化 SP SAML 断言命名空间的 XML 元素 |
| 名称身份映射 | 在 IdP 中用来格式化 SP SAML 断言名称标识符的 XML 元素 |
| 版本 | SAML 1.1 |

*（作者： Frank Teti 译者：杨晓明 来源：TechTarget 中国）*

原文链接：http://www.searchsoa.com.cn/showcontent_30029.htm

## SAML：IdP 和 SP 用户存储库

在 SAML 的身份验证模型里，IdP 需要一个作为身份验证用户的“记录系统”的本地存储库。通常，用户存在于本地 LDAP 库中。反过来说，当在 SP 上调用受保护的资源时，SP 可以使用同一个本地 LDAP 库来验证用户。不过，虽然这种模式可能适用于 Intranet 应用，但跨站点的网络应用上是不可行的，并且实际上没有利用 SAML SSO 断言模型。

另外，通过配置 SP 就可以“虚拟用户”。图 3 窗口描述了在 WebLogic 控制台上创建一个声明方来授权虚拟用户。基于传入的断言，SP 中这个可配置的选项允许 SAML 身份断言器实例化用户和组的 Principal(s)。这个配置也要求 SAML 验证提供者为安全界限做一些配置。这个配置使用户以虚拟用户身份登录这个虚拟用户，和任何本地的已知用户都没有对应关系。

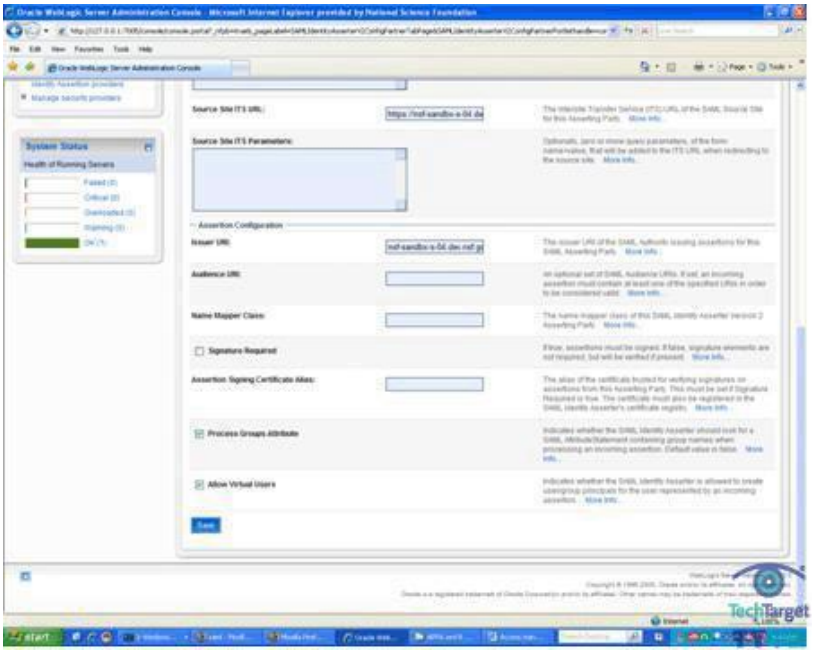


图 3 在 WebLogic 服务器 10.3 控制台上创建一个断言方的视图

确切地说，创建断言方窗口捕获了下述信息：

| Screen Literal/Parameter | Value |
|---|---|
| Issuer URI: | The URI of the SAML Assertion issuer |
| Signature Require | If the certificate is signed, this optional attribute can be set to true. |
| Process Groups | Parse and store SAML Assertion Group Attribute into the JAAS objects of the SP |
| Allow Virtual Users | Trust the identity of the incoming assertion for all authentication and authorization (group) information. |

*(作者： Frank Teti 译者：杨晓明 来源：TechTarget 中国)*

原文链接：http://www.searchsoa.com.cn/showcontent_30031.htm

## SAML 断言的说明

通常设计和开发包括门户的 JEE 应用来提供 RBAC 功能。这需要组通过结合应用服务器配置的应用来完成角色映射。在 SAML 断言模型内，SP 需要注明它是否会处理受保护应用中的组行为（见图 3），用户需要成为 IdP 的本地（LDAP）库中的组员之意。通过 SP 验证器映射到 JAASPrincipal 对象的“组”属性将使断言被处理（见清单 1）。

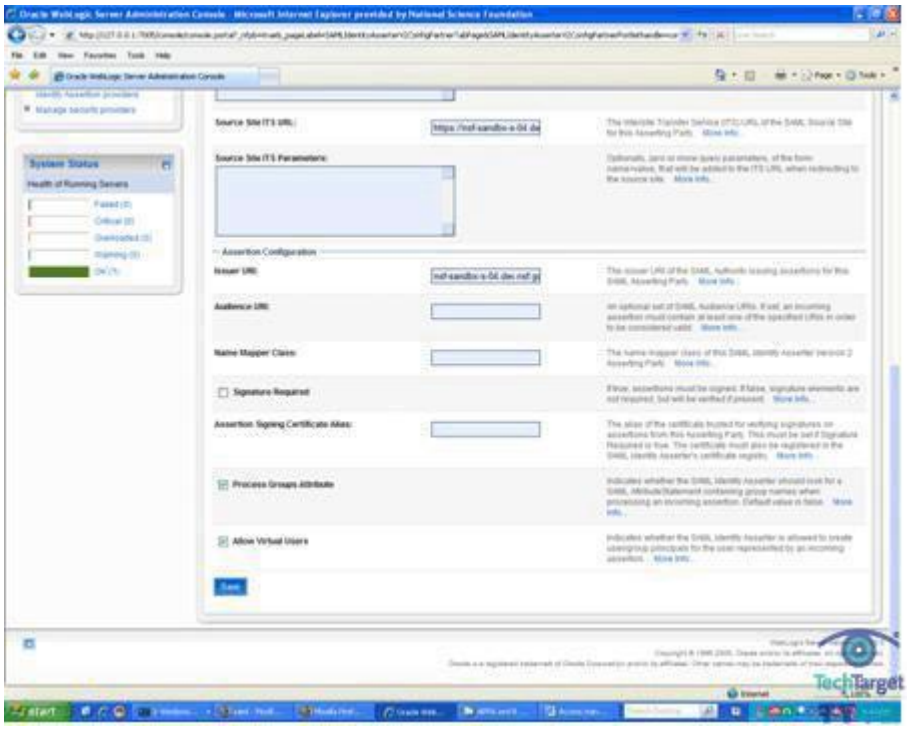


图 3

清单 1 有关传入用户的 SAML 断言元素

以下是引用片段：

```
<Subject>
<NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">auser</NameIdentifier>
<SubjectConfirmation>
<ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer</ConfirmationMetho
d>
</SubjectConfirmation>
</Subject>
<Attribute AttributeName="Groups" AttributeNamespace="urn:bea:security:saml:gr
oups">
<AttributeValue>atlantis</AttributeValue> </Attribute>
```

图 3 当处理一个传入断言时，有一个标志用于指示 SAML 身份断言器是否应该寻找一个包含组名的 SAML 属性声明。这需要应用能够允许 RBAC 分别使用清单 2 和清单 3 中 Web.xml 和相关联的 WebLogic.xml。在该应用中，“工作”目录下的资源需要传入的用户是“atlantis”组的成员之一。注意这一点很重要，在 JAAS 和 JEE 中一个 Principal 可以是某个特定的用户或组。

清单 2　SP 的 Web.xml

以下是引用片段：

```
<security-constraint>
<web-resource-collection>
<web-resource-name>TheSecurePages</web-resource-name>
<description>Pages accessible by authorized users.</description>
<url-pattern>/working/*</url-pattern>
<http-method>GET</http-method>
</web-resource-collection>
```

```
<auth-constraint>
<description>Roles who have access.</description>
<role-name> atlantis </role-name>
</auth-constraint>
</security-constraint>
<login-config>
<auth-method>CLIENT-CERT</auth-method>
<realm-name>myrealm</realm-name>
</login-config>
<security-role>
<description>These are the roles who have access.</description>
<role-name> atlantis </role-name>
</security-role>
</web-app>
```

清单 3 - SP 的 WebLogic.xml

**以下是引用片段：**

```
...
<security-role-assignment>
<role-name> atlantis </role-name>
<principal-name> atlantis </principal-name>
</security-role-assignment>
<context-root>TargetApplication</context-root>
</weblogic-web-app>
```

*(作者： Frank Teti 译者：杨晓明 来源：TechTarget 中国)*

原文链接：http://www.searchsoa.com.cn/showcontent_30091.htm

# SAML2.0 特性分析

### IdP 和 SP 错误的有效解决方案

在 Oracle WebLogic 或者任何其他应用服务器中，中间件的错误检测和解决方案，允许同时为 IdP 和 SP 进行 SAML 安全调试和/或直接“跟踪”日志是一个标准测试过程。为 Web 应用浏览器认证使用 SAML 时，另一种工具是 tcpmon。tcpmon 是一个在 TCP 连接上监视数据流动的开源工具，在客户端和服务器之内与两者之间通过配置来使用。

如果使用 Firefox 为 Live HTTP 头进行单元测试，tcpmon 是一个很好的插件，用来测试要求的 SAML 身份验证的网站重定向。在 HTTP 协议层，为了调试 SP 和 Idp 如何在验证和断言处理期间重定向浏览器，Live HTTP 标题是很有用的。

### 建立伙伴信任关系

如参考文章所言，关键配置要求不只配置一个仅有的私钥。这种处理类型只对开发模式有好处。我们强烈推荐从一个如 VeriSign 这样有效的证书颁发机构获得一个完全可信的证书。

- 为可信任伙伴的 SAML 集成，至少需要两条关键因素：
- 用于 SP 网站上的断言消费服务的私钥为 IdP(见图一)提供了 SSL 客户端身份验证功能（见图一）。

从断言方来的断言用来验证签名公钥或可信任的证书。这个证书必须在 SAML 身份断言器的证书注册表中注册过，必须为浏览器/POST 注册信息所配置。

### 生产环境与独立

需要 SSO 或跨域集成的中间件应用通常要求中间件环境下的高度可扩展性或容错复制。就中间件 SAML 安全模型而言，除了要求源站点和目标网站 URLs 的模型不受到影响，还要反映虚拟地址的负载均衡。

**启示：SAML2.0 特性**

对很多组织来说，最有可能是根据新特性是否具有足够的吸引力迫使软件组织决定转向 SAML2.0：

- 升级访问管理器的应用；
- 重新设计 SP 配置，并且；
- 重写应用逻辑。

SAML2.0 包含一个单点退出协议，它几乎同时支持 Web SSO 参与者会话的注销。对于 SAML1.1，这种“普遍注销”功能必须结合 IdP 内在的 Cookie 管理行为来设计。例如，在验证和实现单点登陆到多个服务的提供者后，用户可以依照身份提供者的要求自动退出所有服务提供者。<AuthenticationStatement>元素已更名为<AuthnStatement>。<AuthnStatement>元素现在支持了会话的概念，以便支持单点注销和其他会话管理要求。

SAML 模式的可扩展性机制已经更新了。因为有利于类型扩展，XSD 元素的替代者已被封掉。已经选择性地将<xs:anyAttribute> 通配符添加进结构中，就是被视为有价值的地方，因为可以添加任意属性而无需创建一个模式扩展，如，对象确认数据和 SAML 属性。这是一个有用的修改，因为现实世界的实现有时是要求瞬变的，传输中的信息需要包含在断言中，就像传入用户登陆的上下文。例如，一个通过了身份验证用户可能有能力从多个位置验证应用，并且该应用可能对非常有兴趣了解这个信息。

这篇文章曾是基于浏览器/POST 配置文件。在 SAML1.1 中这两个原始的浏览器配置文件（Browser/Artifact 和 Browser/POST）已经合并到了单一的浏览器 SSO 的配置文件中了。虽然增强的用户和代理 SSO 配置已经加入了，但又产生了两种不同的配置。

建议越过 SAML 1.1 而使用 SAML 2.0 不只是跟上当前技术问题，SAML 2 提供了一个完整的分布式 SSO 体验，然而，SAML1.1 有一些严重的缺陷。最后，送给持有反对意见的采纳人一句话，改变既不见得是好事，也不一定是坏事，如此而已。

Frank Teti 是 Ironworks 的一位架构师，以前属于 Oracle/BEA Systems SOA/BPM 的实施。你可以用 fteti@ironworks.com 和他联系。

*(作者： Frank Teti 译者：杨晓明 来源：TechTarget 中国)*

原文链接：http://www.searchsoa.com.cn/showcontent_30092.htm

# 安全声明标记语言（SAML）的应用

**问：**我们试图在一个巨大的电信级产品上执行 Web 服务。数据需要在线路上传输是很关键的，并且要保证数据安全。你能就我的安全框架使用作出建议么？执行和响应时间也是非常重要的。我被这些 WS－安全规格搞得焦头烂额。我想要知道普遍使用的安全框架是什么？

**答：**安全声明标记语言（SAML）是现在最好的解决方法，并且支持一致性管理的概念，对保护 Web 服务起重要的作用。因此，有眼光的卖主和工具包都支持安全声明标记语言（SAML）。

*（作者：David Linthicum 来源：TechTarget 中国）*

原文链接：http://www.searchsoa.com.cn/showcontent_7516.htm

# 如何让 SAML 适应你的 SOA 安全方案

由于作为服务软件的越来越多，更多的机构开始倾向于面向服务架构（SOA），这种环境往往是攻击者的主要目标。这种趋势要求特别设计的技术解决安全性问题，接下来我将描述由安全性断言标志语言（SAML）阐释的方法。

SAML 是在 OASIS 指导下开发的标准。OASIS 管理一系列广泛的标准，这些标准和 XML 以及像 WS-*和 SGML 这样的 Web 服务有关。SAML 这如其首字母缩略词所显示的那样，通过 XML 或者 HTML 标识语言强制执行安全性，使其成为广泛安全方案的一部分。

使用 SAML 的最新领域包括由自由联盟计划开发的单一登录计划，本质上和开发 ID 项目很相似——例如通过 WS-Security 在基于 SOAP 服务中使用的通用领域。理解 SAML 的关键是要理解其断言机制，该断言机制是语言的基石。列表 1-1 展示了一个 SAML 断言。

```
<saml:Assertion xmlns:saml=â  urn:oasis:names:tc:SAML:2.0:assertionâ
   Version="2.0"
   IssueInstant="2008-10-15T12:00:00Z">
   <saml:Issuer Format=urn:oasis:names:SAML:2.0:nameid-format:entity>
     http://www.acme.org
   </saml:Issuer>
   <saml:Subject>
     <saml:NameID
       Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
     j.smith@acme.org
  </saml:NameID>
</saml:Subject>
<saml:Conditions
```

```
    NotBefore="2008-10-15T12:00:00Z"
    NotOnOrAfter="2008-10-15T12:10:00Z">
  </saml:Conditions>
  <saml:AuthnStatement
    AuthnInstant="2008-10-15T12:00:00Z" SessionIndex="67775277772">
    <saml:AuthnContext>
      <saml:AuthnContextClassRef>
        urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
      </saml:AuthnContextClassRef>
    </saml:AuthnContext>
  </saml:AuthnStatement>
</saml:Assertion>
```

在最后一个列表中，你可以看到，SAML 断言包含了很多安全语句，这些语句都和一个主题有关。一个断言的价值可以体现在各个方面，范围从基本要素（例如就某个主题做出断言的人）延伸到更为细致的话题（例如一个断言的时间有效的和日期，甚至还包括令断言更为严格的附加条件）。

SAML 当局â（提供断言方）和 SAML 依赖方（要求断言）之间交换断言。依照交换的性质，第三方或者“被断言方”也会参与进来。第三方满足单一登录方案，在该方案中“被断言方”是一个企图访问资源的终端用户，主管 SAML 当局和 SAML 依赖方交换。对于其它 SAML 交换来说，“被断言方”不过是组成 SAML 依赖方的应用逻辑。

要想将 SAML 融入到 SOA 安全方案，主要取决于其平台中立性。例如早期 SAML 断言<NameSubject>中的<NameID>要素。在这个实例中，断言是通过电子邮箱做出的，但是这只是其中的一种情况。SAML 同样也支持像 X.509 或者 Kerberos 这样的身份，这在企业设置中是很常见的，因此这些身份可以在一个特定的平台中屏蔽应用程序。

把这个相同的情况看做是单一注册方法；而不是将程序 â“和用户 â”锁定到电子邮件地址，基于 SAML 的架构准许“被断言方”提供凭证，例如数字签名证书（X.509）或者 Kerberos，准许安全策略建立在更为广阔的或者已经存在的安全基础设施中。

实际上最主要的安全性问题“这一方真是其所宣称的那样吗？”，是通过 SAML 来回答的，而不是通过某一个特定平台来回答的，这些平台总有特有的方法断言（例如超时值设定的，失败的条件）。SAML 会将这个断言过程标准化。

我们再看一下含有 SOAP 服务的案例场景，列表 1-2 展示了一个嵌入在 SOAP 请求中的 SAML 代码片段。

```
<? xml version="1.0" encoding="UTF-8"? >
<env:Envelope
  xmlns:env=â  http://www.w3.org/2003/05/soap/envelope/â  >
  <env:Body>
<samlp:AttributeQuery
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  ID="aaf23196-1773-2113-474a-fe114412ab72"
  Version="2.0"
  IssueInstant="2008-10-15T20:31:40Z">
  <saml:Issuer>http://example.sp.com</saml:Issuer>
  <saml:Subject>
     <saml:NameID
     Format="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName">
      C=US,  O=TECHTARGET-TEST,  OU=User,  CN=jsmith@acme.org
     </saml:NameID>
  </saml:Subject>
 <saml:Attribute
```

```
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
            Name="urn:oid:2.5.4.42"
            FriendlyName="givenName">
        </saml:Attribute>
      </samlp:AttributeQuery>
    </env:Body>
  </env:Envelope>
```

注意<NameID>值现在指向了 X.509 值。此外在客户服务器 SOAP 交换过程中还要执行安全性检查，但是因为断言是建立在 SAML 基础之上的，这需要松耦合和切换邮箱地址或者 kerberos 的功能（如果需要的话），这也是 SOA 最主要的目标。

那么在哪里 SAML 会融入到 Open-ID，自由联盟和 Web 服务中呢？在每个应用程序栈â“markup â”底部，都会有 SAML 运行。因此你不可能在这样的项目中看到有明确提到这方面的内容，这些项目都是整体的办法解决安全性问题。但是 SAML 的颗粒度使其功能变得更为强大，可以处理各种各样涉及云计算的安全隐患。

目前有许多开放源 SAML 实施，进一步扩展了 SAML 的用途，远远超出了之前我们所说的那个项目。所以如果你的 SOA 项目也需要具有松耦合特性的安全检查，考虑一下 SAML 吧，它和 XML 一样，可以解决分布式系统安全问题。

*（作者：Daniel Rubio　译者：杨君 来源：TechTarget 中国）*

原文链接：http://www.searchsoa.com.cn/showcontent_17996.htm