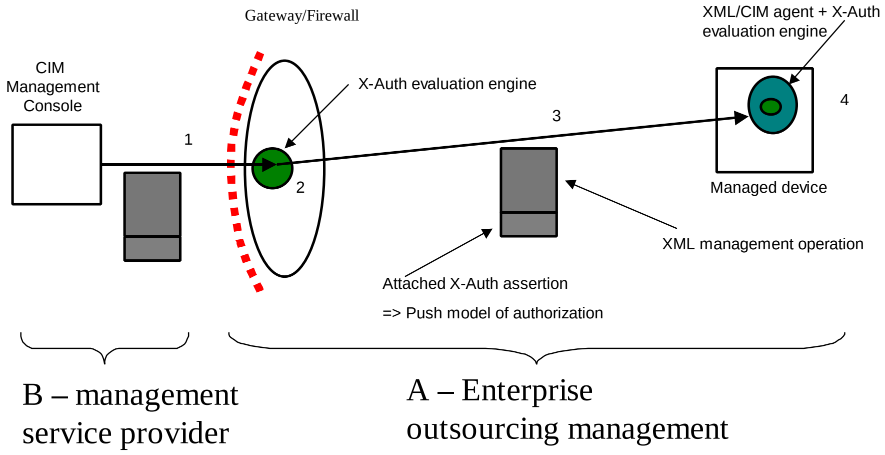
XML authorization and authentication scenarios

Nigel Edwards, Hewlett-Packard

This note is intended as input to the Oasis Security Services Technical Committee group considering use cases and requirements. It describes two scenarios in which XML security assertions are used, but exchanged in different ways. The term X-Auth is used as a code word to be replaced by whatever name the Oasis Security Services Technical Committee chooses for its output.

Outsourced Management


Gateway/Firewall
XML/CIM agent + X-Auth evaluation engine
CIM Management Console
X-Auth evaluation engine
1
2
3
4
Managed device
XML management operation
Attached X-Auth assertion
=> Push model of authorization
B – management service provider
A – Enterprise outsourcing management

The above figure shows an enterprise A that has outsourced the management of its network devices to a management service provider B. Management messages are exchanged using CIM/XML over HTTP. (CIM or Common Information Model, is a management standard being developed by the Distributed Management Task Force - http://www.dmtf.org/, an XML DTD for CIM has been defined.)

Suppose the operator, Joe, wants to invoke the StopService method. This will be executed by the XML/CIM agent on the managed device, if authorized. The following steps occur.

1. The CIM management console generates the XML content and attaches an X-Auth assertion. This X-Auth assertion has been generated by B's attribute authority (or Policy Decision Point) and confers the role "System Manager for A" to Joe. The CIM management console signs the request and sends it as an HTTP request.
2. The request now has to traverse A's firewall or the boundary into A's network. The gateway at this boundary uses its X-Auth evaluation engine (or Policy Enforcement Point) to verify that this incoming message is allowed. It does this, by verifying the signature and discovering the request is from Joe. Next it uses two assertions to authorize the incoming message: the assertion issued by B's attribute authority that is attached to the message (conferring the role "System Manager for A" on Joe); an assertion issued by A's attribute authority granting "Gateway Access" to any entity that has a valid "System Manager for A" assertion issued by B's attribute authority.

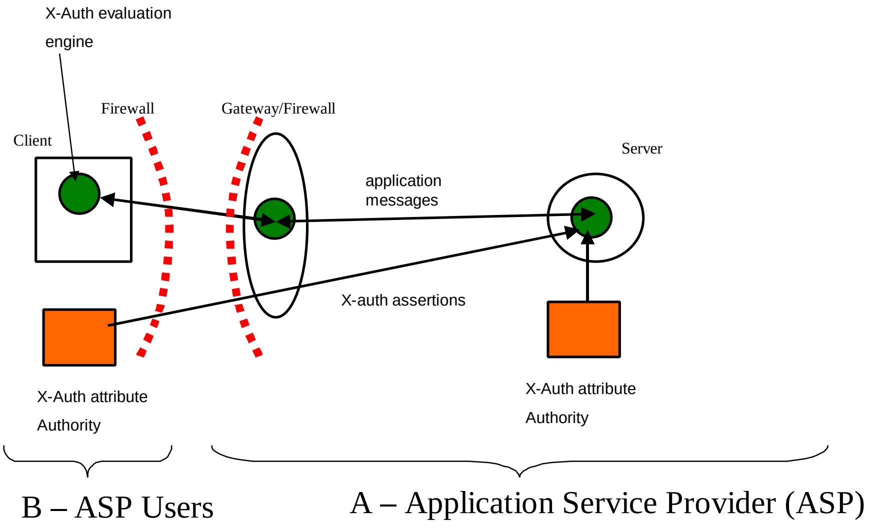
   Note that the second assertion can be pushed to the gateway (part of its configuration), or retrieved dynamically from a repository (or indeed the issuer).

3. The request is forwarded by the gateway to the managed device.
4. The X-Auth evaluation engine on the managed device needs to determine if a “StopService” request from Joe is allowed. It does this by using two assertions: the “System Manager for A” assertion issued by A’s attribute authority; an assertion issued by B’s attribute authority granting “Full Management Rights” to any entity with a valid “System Manager for A” assertion issued by B’s attribute authority.

This scenario shows an X-Auth assertion being pushed, as part of the application request. It is possible that fine-grain, method-level access control might be needed. So specific methods may be authorized instead of “Full Management Rights” as discussed above.

Application Service Provider

An Application Service Provider scenario is shown below. In this scenario an ASP, A, is providing an application (possible examples could be a word processor or an ERP application) to users in another enterprise, B. A VPN (for example IPSEC) is used to provide a secure end-to-end tunnel between the client and server.



Some aspects of this scenario are similar to the above “outsourced management” scenario. The gateway needs to know that incoming packets from a client in B are allowed. It needs an assertion from B’s attribute authority that the client is a valid user, and an assertion from A’s attribute authority that entities issued “valid user” assertions from B are allowed access. The X-Auth evaluation engine in the server will perform a similar check.

It is also important that the client check that the application is valid. This avoids problems such as an attacker spoofing the service provider and providing a word processor service that silently emails copies of all documents generated by the client to the attacker. This might be done by the client X-auth evaluation engine checking two assertions: one from A granting

“ Approved Application” status to the server; one from B granting the attribute “execute” to any entity with “ Approved Application” status issued by A.

A major difference between this scenario and the outsource management service scenario is that all assertions are “pulled” in this scenario. This means the assertions are not attached to application messages; instead they must be retrieved either directly from the attribute authority, or a repository. For example, once the client has been authenticated, the X-auth evaluation engine in the server needs to retrieve the X-Auth assertions issued by A and B. This will involve making a request to a repository inside B, traversing both A and B’s firewall as shown in the diagram. Similarly the X-Auth engines in the gateway and client will have to retrieve assertions issued by both authorities (these interactions are not shown in the diagram).