OGF NSI CS v2.0 Security Proposals and Issues Summary

Requirements (agreement so far)

1) Confidentiality
2) Integrity
3) Mitigation of Third-Party Replay Attacks
4) Mutual Authentication
5) Authorization support

Historic Issues

Two approaches proposed:

1. Message level Integrity + transport layer security
    a. Message Level Integrity: WS-Security
    b. Transport Layer Security: TLS

2. Transport layer security, no specific message level security.
    a. TLS only

Original concerns:

Figure 1. Inter-Domain NSA-to-NSA (with proxy) Communication

The main point of disagreement has been if TLS is good enough especially in the diagram described in Fig. 1, and do we need WS-Security to secure the message end-to-end.

Big Question: We need to have a security model that production networks are comfortable with deploying in production, and not to satisfy near-term demonstration needs.

New Proposal

1. Transport layer as the security model, no end-to-end message security. The premise is that a secure point-to-point transport layer implies a level of message integrity (i.e. the message cannot be tampered with in transit). In the example outlined in Fig 1., where an NSA (i.e. NSA B) communicates with NSAs in other domains (i.e. NSA A) via a proxy, the following assertions are held:
    a. The transport layer between two directly communicating entities in different domains (i.e. NSA A, and TLS Proxy B) is secure.
    b. The transport layer between two directly communicating entities within the same domain (i.e. TLS Proxy B, and NSA B) is secure.

   The above two assertions places some restrictions on how NSAs can talk to one another. Specifically, NSAs cannot talk "thru" other NSAs. For example, NSA A cannot send a message to NSA B via NSA X (see Fig 2.) as the point-to-point transport security between NSA A and NSA B is violated by NSA X (man-in-the-middle).

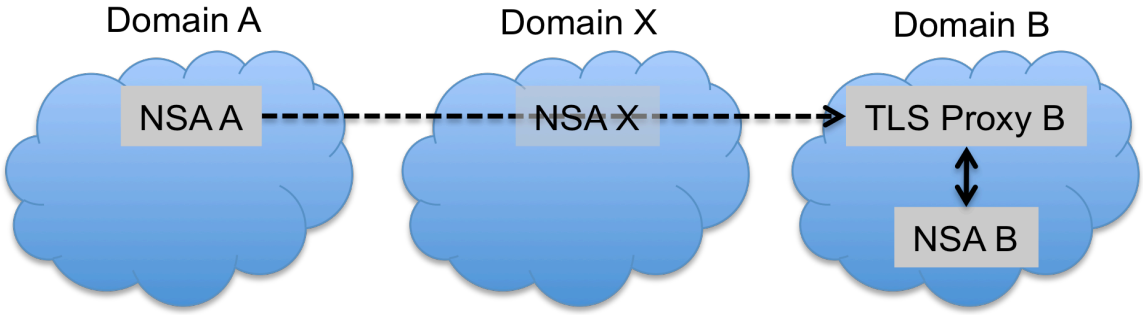

Figure 2. Disallowed NSA "transit" communications

   For CS v2.0 it is proposed that the inter-domain transport layer security utilize TLS with client/server public key certificates. This enables:
    a. Confidentiality (using TLS)
    b. Integrity (implicit with point-to-point transport security)
    c. Mitigation of Third-Party Replay Attacks (implicit with point-to-point transport security)
    d. Mutual Authentication (client/server public key certificates)

2. Security token within the default SAML security profile (as defined by a Best and Current Practices (BCP) document) for authorization purposes. The consideration of the token is to provide the flexibility to decouple the authorization Policy Decision Point (PDP) and the Policy Enforcement Point (PEP). The PEP is performed by the provider NSA, but the PDP may be executed by a trusted third party, or the provider NSA itself. An example of a token could be a web services cookie or an OAuth bear token. The specifications of the token are flexible and can be unique to each requestor/provider NSA peering.

For CS v2.0 it is proposed that the default token would be the client public key certificate. This can be used to facilitate authorization functions.