A minimal NSI AAI

OGF40, Oxford, January 16, 2014

Hans Trompert
SURFnet

Control plane security

To ensure the integrity and confidentiality of the NSI control plane messages:

- NSAs will mutually authenticate themselves
- Control plane traffic between two peering NSAs will be encrypted

This is implemented by client authenticated TLS:

- With a set of certificates exchanged at the time the peering was established
- Allowing self signed certificates
- Without the need of a central certificate authority

User authentication

Every NSA that allows a so-called “originating user” to enter a NSI messages into the NSI control plane:

- must authenticate the identity of the user using any authentication scheme that is supported locally
- Add the user identity information to the NSI message header
- Store the use identify information together with the entered NSI message for future reference

Furthermore:

- Not every NSA along the way needs to be able to interpret the user identify information as long as it is possible to trace back the message to the originating NSA

Traceability

To keep track of the path traveled by a NSI message every NSA in the control plane path will:

- Add a connection URN to the NSI header
- locally store enough information to trace back the message to the originating NSA if needed

This connection trace allows for:

- In case of misconduct, if a NSA cannot interpret the user identity information itself, the user identity can still be determined by following the trace back to the originating NSA
- Detecting loops in the path finding process
- Debugging purposes

Transit authorization

- It is the responsibility of two peering partners (E-NNI to ENNI) to agree on a use policy and possible associated costs
- Agreements on policies and associated costs for user traffic entering a domain (UNI) are the responsibility of that domain
- Once user traffic has entered the NSI data plane it will be transported without any additional cost for the user as the inter domain costs are covered by the before mentioned peering agreements
- The amount of traffic that is allowed to be transported is however subject to peering point policies if any
- Additional peering point authorization can be done based on the identity of the requesting user.

Endpoints authorization

- UNI authorization is done by the domain associated with that UNI based on local policy
- It is the responsibility of the user to supply the needed endpoint authorization tokens
- Endpoint authorization tokens will be sent with the users request as part of the NSI message header
- The kind of authorization token needed for a UNI is based on the supported authorization mechanisms of that domain

NSI message header

The information in the NSI header will be extended with:

- Identity information of the originating user (mandatory)
  - Added as sessionSecurityAttr in the NSI header
- Endpoint authorization tokens (optional)
  - Added as sessionSecurityAttr in the NSI header
- Connection trace (mandatory)
  - Added using the “any” to the bottom of the NSI header

Originating user identity information

The originating user identity information consists of:

- Exactly one User ID
- Optionally one or more Group ID

Example:

```
<ns1:sessionSecurityAttr>
   <Attribute Name="user">
      <ns1:AttributeValue>htj@nordu.net</ns1:AttributeValue>
   </Attribute>
   <Attribute Name="group">
      <ns1:AttributeValue>nordu.net</ns1:AttributeValue>
      <ns1:AttributeValue>dev.nordu.net</ns1:AttributeValue>
   </Attribute>
</ns1:sessionSecurityAttr>
```

Endpoint authorization tokens

Example:

```
<ns1:sessionSecurityAttr>
   <Attribute Name="token">
      <ns1:AttributeValue>
         237e2d04-67cc-11e3-a633-f0def14b5d43
      </ns1:AttributeValue>
   </Attribute>
   <Attribute Name="token">
      <ns1:AttributeValue>
         614fc135-faf7-4e0c-a8b0-fbd982e3b015
      </ns1:AttributeValue>
   </Attribute>
</ns1:sessionSecurityAttr>
```

Connection trace

A connection trace consists of a list of connection URNs and is added to the bottom of the NSI header

Connection URN = NSA URN + ':' + connection ID

Example:

NSA:	urn:ogf:network:aruba.net:nsa
Connection Id:	AR-Tfe07c58e3fff
Connection URN:	urn:ogf:network:aruba.net:nsa:AR-Tfe07c58e3fff
Namespace:	http://nordu.net/namespaces/2013/12/gnsbod

```
<gns:ConnectionTrace xmlnx:gns="http://nordu.net/namespaces/2013/12/gnsbod">
    <gns:Connection>urn:ogf:network:aruba:2013:nsa:AR-Tfe07c58e3fff</gns:Connection>
    <gns:Connection>urn:ogf:network:bonaire:2013:nsa:BO-s7780</gns:Connection>
    <gns:Connection>urn:ogf:network:curacao:2013:nsa:CU-1234</gns:Connection>
</gns:ConnectionTrace>
```

NSI message header example

```
<soapenv:Envelope xmlns:type="http://schemas.ogf.org/nsi/2013/07/connection/types" xmlns:head=http://schemas.ogf.org/nsi/2013/07/framework/
headers xmlns:assertion="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
   <soapenv:Header>
      <head:nsiHeader>
         <protocolVersion>application/vnd.ogf.nsi.cs.v2.provider+soap</protocolVersion>
         <correlationId>urn:uuid:2863dcdf-7c3c-4f61-8f53-dac93c9ba9fd</correlationId>
         <requesterNSA>urn:ogf:network:nsa:surfnet-nsi-requester</requesterNSA>
         <providerNSA>urn:ogf:network:surfnet.nl:1990:nsa:bod</providerNSA>
         <replyTo>http://nsi-requester.host.domain/reply</replyTo>
         <assertion:sessionSecurityAttr>
            <assertion:Attribute Name="user">
               <assertion:AttributeValue>user@domain</ns1:AttributeValue>
            </assertion:Attribute>
            <assertion:Attribute Name="group">
               <assertion:AttributeValue>urn:collab:group:surfteams.nl:nl:surfnet:diensten:bod_test_team</assertion:AttributeValue>
               <assertion:AttributeValue>dev.surf.net</assertion:AttributeValue>
            </assertion:Attribute>
            <assertion:Attribute Name="token">
               <assertion:AttributeValue>237e2d04-67cc-11e3-a633-f0def14b5d43</assertion:AttributeValue>
               <assertion:AttributeValue>abac477b-9159-4da3-9c0f-6aaa099735e2</assertion:AttributeValue>
            </assertion:Attribute>
         </assertion:sessionSecurityAttr>
         <gns:ConnectionTrace xmlnx:gns="http://nordu.net/namespaces/2013/12/gnsbod">
            <gns:Connection>urn:ogf:network:aruba:2013:nsa:AR-e07c3</gns:Connection>
            <gns:Connection>urn:ogf:network:bonaire:2013:nsa:BO-s7780</gns:Connection>
            <gns:Connection>urn:ogf:network:curacao:2013:nsa:CU-12345</gns:Connection>
         </gns:ConnectionTrace>
      </head:nsiHeader>
   </soapenv:Header>
   <soapenv:Body>
   </soapenv:Body>
</soapenv:Envelope>
```

hans.trompert[at]surfnet.nl

www.surfnet.nl