DTLS over SNMP

Wes Hardaker

14 November 2008
1 Background

2 SNMP over DTLS: Details

3 Issues
Motivation

- Support X.509 Certificate Authentication
- Support for a UDP based security solution
  - TCP vs UDP performance in bad networks is still a problem
The Resulting Document

- draft-hardaker-isms-dtls-tm-01
- Closely aligns in structure the SSH document
- Compliant with the TSM security model expectations
DTLS

- Defined in RFC4347
- DTLS is functionally identically to TLS
- Uses the same on-the-wire format
  - X.509 certificates for authentication.

(Editor's Note: I'm not a DTLS expert; hopefully Eric is here!)
DTLS Architecture Overview

Client

ClientHello --------> ServerHello
Certificate*
ServerKeyExchange*
CertificateRequest*
<-------- ServerHelloDone
Certificate*
ClientKeyExchange
CertificateVerify*
[ChangeCipherSpec]
Finished --------> [ChangeCipherSpec]
Finished
Application Data <-------- Application Data
DTLS Considerations

- TLS relies on TCP for session demultiplexing
  - Does not contain an in-protocol session identifier
- UDP doesn’t provide session demultiplexing

Result: we have to define how to demultiplex multiple connections
- Need a unique key to latch to a DTLS session
- Key: src addr, srt port, dst addr, dst port
X509 Certificates to securityName

- X509 Certificates use a very different naming structure
- The **Issuer**: field identifies who handed out the certificate
- The **Subject**: field typically identifies a user and contains:
  - Location information (C: Country, ST: State)
  - Organization information (O: Name, OU: Unit)
  - Personal Information (CN: Common Name)
Client X.509 Certificate Examples

Example: My Fedora User Certificate
- Subject: C=US, ST=North Carolina, O=Fedora Project, OU=Fedora User Cert, CN=hardaker/emailAddress=wjhns174@hardakers.net
- Issuer: C=US, ST=North Carolina, L=Raleigh, O=Fedora Project, OU=Fedora Project CA, CN=Fedora Project CA/emailAddress=admin@fedoraproject.org

Example: Fedora CA
- XXX...
Server X.509 Certificate Examples

Example: The Fedora Server Certificate

- **Subject:** C=US, ST=North Carolina, L=Raleigh, O=Fedora Project, OU=Fedora Project CA, **CN=Fedora Project CA/emailAddress=admin@fedoraproject.org**
- **Issuer:** C=US, ST=North Carolina, L=Raleigh, O=Fedora Project, OU=Fedora Project CA, CN=Fedora Project CA/emailAddress=admin@fedoraproject.org

Example: The www.ietf.org HTTPS Certificate

- **Subject:** O = *.ietf.org, OU = Domain Control Validated, CN = *.ietf.org
- **Issuer:** CN = Starfield Secure Certification Authority, OU = http://certificates.starfieldtech.com/repository, O = Starfield Technologies, Inc., L = Scottsdale, ST = Arizona, C = US
X.509 Subject to securityName Mapping

- The *Subject* field is the identifying field.
- The *Common Name (CN)* tag within it is typically the *account name*.
- It is paired with the *Issuer* field to be unique.
- Potential ways to be mapped into a securityName:
  - Take the **CN** in raw form
  - Map the **CN** to a securityName
- This mapping is configured through DTLSTM-MIB tables.
DTLSTM-MIB

- Domain and Address definitions
- Counters
- Configuration
  - dtIstmCertificateToSNTable
  - dtIstmParamsTable
- Conformance statements
Incoming securityName Selection

- The dtIstmCertificateToSNTable maps incoming certificates to securityNames.
- Two modes:
  - Accept the CN directly from trusted CAs
  - Map a Subject to specific securityName

**dtIstmCertificateToSNTable**

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>dtIstmCertID(1)</td>
<td>1</td>
</tr>
<tr>
<td>dtIstmCertIssuerDN(2)</td>
<td>Fedora... specified (or byCN)</td>
</tr>
<tr>
<td>dtIstmCertMapType(3)</td>
<td>...</td>
</tr>
<tr>
<td>dtIstmCertIssuer*</td>
<td>C=US, ST=North Carolina ...</td>
</tr>
<tr>
<td>dtIstmCertSubject*</td>
<td>wes</td>
</tr>
<tr>
<td>dtIstmCertSecurityName(4)</td>
<td>nonVolatile</td>
</tr>
<tr>
<td>dtIstmCertStorageType(5)</td>
<td>createAndGo</td>
</tr>
<tr>
<td>dtIstmCertRowStatus(6)</td>
<td></td>
</tr>
</tbody>
</table>
Outgoing Certificate Selection

- The `dtlstmParamsTable` maps an outgoing `securityName` to a certificate.
- The certificate is referenced by an Issuer and Subject

### dtlstmParamsTable

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>snmpTargetParamsName(1)</code></td>
<td>wes</td>
</tr>
<tr>
<td><code>dtlstmCertIssuer</code></td>
<td>...</td>
</tr>
<tr>
<td><code>dtlstmParamsSubject(1)</code></td>
<td>C=US, ... CN=hardaker...</td>
</tr>
<tr>
<td><code>dtlstmParamsStorageType(2)</code></td>
<td>nonVolatile</td>
</tr>
<tr>
<td><code>dtlstmParamsRowStatus(3)</code></td>
<td>createAndGo</td>
</tr>
</tbody>
</table>
Issues

- A few MIB changes needed
- Awaiting completion of the other documents before WG consideration
- Need people to review it
- DTLS implementations are still few
  - OpenSSL: implemented but poorly documented
  - GnuTLS: not implemented