ISMS – SNMP over DTLS

draft-ietf-isms-dtls-tm

Wes Hardaker
ietf@hardakers.net

Overview

- Current Status
- Open Issues
- Requirements

Status

Current Status

- Draft -01 published in October
  - (-00 was new since last meeting too)

- WG Last Call
  - Started: Oct 29th
  - Ends: Nov 14th

- Please read the draft and send in comments!
  - Thanks to those that already have done so!

Major Changes Since -(-1)

- MIB Changes
  - Single Fingerprint TC (was 2)
  - SubjectAltName type selection (includes “any”)
  - Added Notifications
    - Server certificate not valid, server authentication failure
- Wording Cleanups
  - Moved TLS/X.509 introduction text to appendices
  - Synchronized further with ISMS' SSH RFC
  - Text changes from readers

Current TLS Vulnerability

- Recent new attack on TLS
  - Uses renegotiation to trick the client and server
  - New man-in-the-middle attack
- Effect on SNMP:
  - Allows attacker to insert arbitrary PDUs into stream
  - Can't see responses though
  - Useful to fake SETs or notifications
- The TLS WG will take care of this

Open Issues

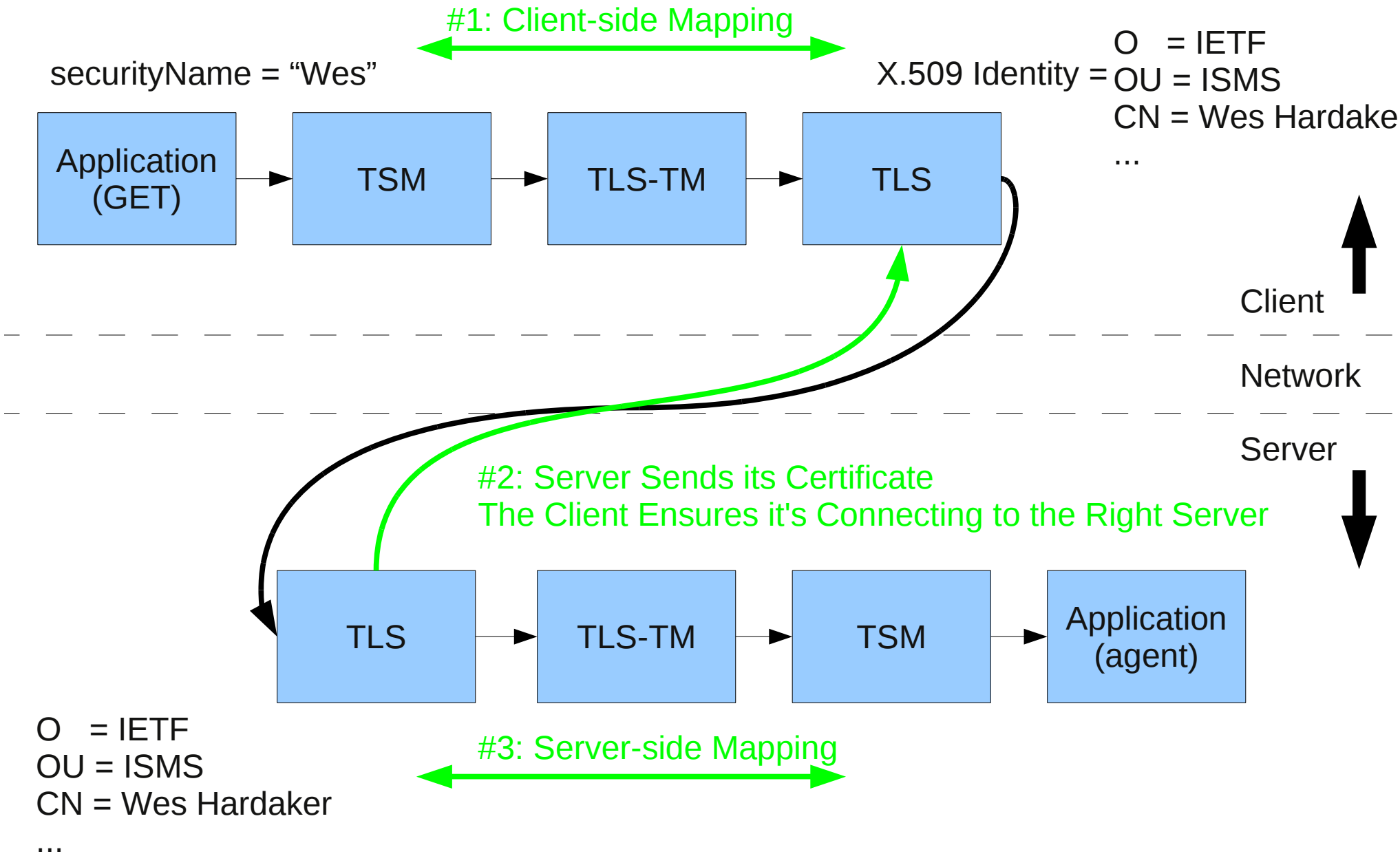
Incoming Connection Refresher

- Client opens (D)TLS Connection
- Client presents X.509 certificate
  - Contains a “subjectAltName” extension
- Server derives the snmpSecurityName from it

- Multiple subjectAltName types:
  - rfc822Name, dNSName, ipAddress, otherName
  - MIB has option for “any” (take the first found)
  - securityName derived from first value of correct type

X.509 Identity / securityName

3 issues:



Issue #1: OtherName Mapping

- “OtherName” choice added after last meeting
- Issue with “otherName”:
  - X.509 SubjectName type “OtherName”
  - An arbitrary field to convert to secName
    - ASN.1: SEQUENCE { OBJECT IDENTIFIER, EXPLICIT }
- Choices for mapping to a securityName

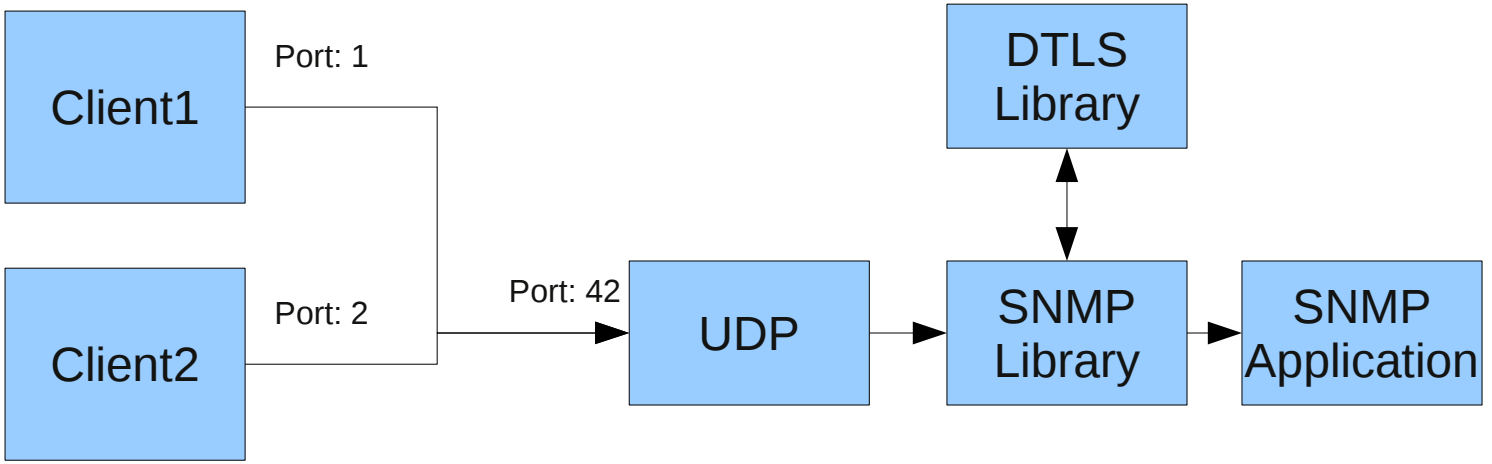
  1) Mapping is implementation dependent.
     - Current Draft
  2) OID selector and direct mapping?
  3) Proposal: Don't do OtherName mapping at all

#2: X.509 Certificate Path Validation

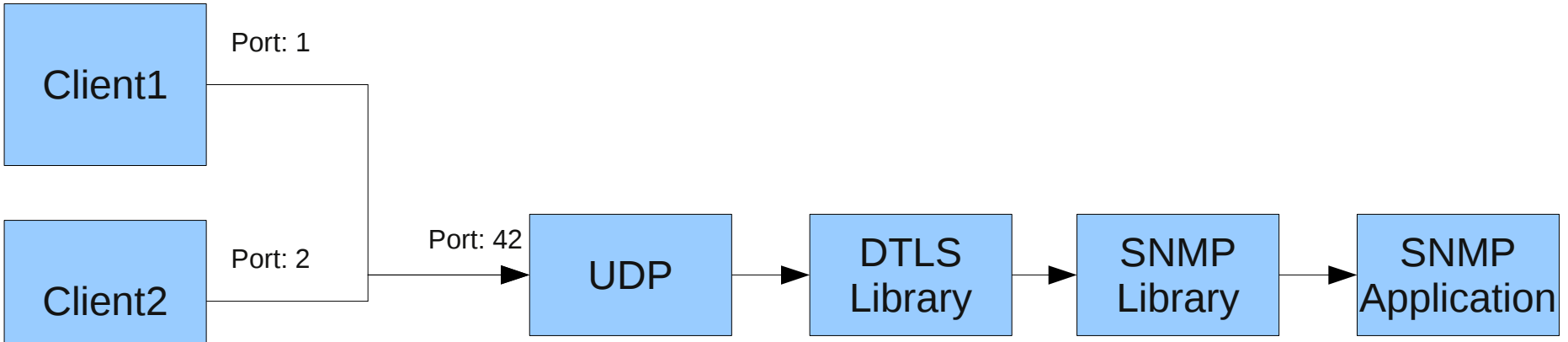
- Two choices when doing X.509 Certificate Validation:
  - Direct FingerPrint specifications
  - Full Path Validation to a trust anchor
- WG Decided to:
  - Provide fingerprint mappings
  - Configuration/definition of full Trust Anchor validation and Configuration is out of scope.

- Proposal: I'll try to make this more clear

DTLS Demultiplexing



Software Architecture #1



Software Architecture #2

#3: Keep UDP Session Handling

- Section 5.5.1+ describes demultiplexing UDP
  - Mandates Unique src/dest addr/port combinations
  - Was written because architecture #1 appears common today
    - May not be in the future?
  - Specifies that the demultiplexing EoP are optional
    - IE: it's “Implementation Guidance”

- Proposal: Leave it.

#4: securityName case sensitivity

- When mapping to a security name we should specify case sensativity
- Proposal:
  - IPv6: Lower Case
    - 2002855d18a500050222fafffeff174c
  - dNSName: Lower Case
    - isms.example.com
  - rfc822name: Lower Case
    - wes@example.com
    - (Pasi proposed just the domain name portion, but 5280 says everything)

#5: Port > || < 1024

- Pasi requested we use a port > 1024
- I'm fine with this

- Proposal: Request > 1024 from IANA

#6: 3 TransportDomains/Addresses

- Pasi wondered why:
  - We have 3 Transport Domains
  - We have 3 Transport Addresses with identical text
  - We can't reuse 1 transport address multiple times for the same TransportDomain identifier
- Answer, unfortunately:
  - “Furthermore, MIB authors SHOULD define a separate TransportAddressType or TransportDomain object for each TransportAddress object.”

    -- TransportAddress TC
  - IE: That's the way it's always been done in SMIv2
- Proposal: keep as is

#7: FingerPrint Crypto Value

- The current TC text says the Fingerprint shouldn't be used as a comparison alternative
  - IE: you must compare the full presented certificate against the fully stored certificate; not just hashes
- Originally allowed for “cheap” (insecure) fingerprints
  - But now we're using only secure hashes
- Proposal: drop the last sentence limiting Fingerprint Usage.
  - IE, allow implementations to just compare hash values

#8: Drop (D)TLS ASIs?

- Draft contains:
  - tlsRead
  - tlsWrite
- I think this derives from early SSH drafts

- Proposal: Not really needed, so drop it.

#9: failure counter in notification

- TlstmServerAuthFailure notification
  - Include TlstmSessionInvalidServerCertificates?

- Proposal: Sure

#10 CreateAndGo vs Active

- Examples currently assume new row creation
  - E.G. sets to createAndGo for creating a row
  - Apparently 3414 uses active instead

- Proposal: umm.....

#11: Dead-Peer Detection

- Pasi wondered if we should say something about when one side drops a DTLS connection if the client should try and detect this?
  - But notes that DTLSTM shouldn't know about PDUs
- Draft currently says (section 8):

  A "broken" session (one side up and one side down) can result if one side of a session is brought down abruptly (i.e., reboot, power outage, etc.). Whenever possible, implementations SHOULD provide graceful session termination through the use of disconnect messages. Implementations SHOULD also have a system in place for dealing with "broken" sessions.

#12: Fate Sharing

- Currently:
  - Can create TLSTM-MIB entries in advance of TARGET-MIB entries being created
  - When TARGET-MIB entries are deleted, corresponding TLSTM-MIB entries are deleted
- Juergen finds this inconsistent.
  - Second bullet decided in previous WG

- Proposal: leave as is

Questions?