DANE for IoT

Shumon Huque, Ash Wilson

November 16th 2020

IETF 109; SecDispatch Working Group

Topics

- Background: DANE for client authentication (IOT, SMTP etc)
  - Pointers to drafts, ongoing/prior discussions; prior engagement with existing WGs
  - Pointers to other background material
  - Summarize who would want to advance this work (us, NIST, ICANN, LoRa)
- Current & Planned work:
  - Refresh and revise DANE TLS client authentication drafts
  - Interfaces for IoT applications
  - Planned upcoming work: expand scope to include IOT object security & cert discovery
- Desired outcome
  - Gauge interest, recruit more collaborators
  - Identify IETF venues for this this work

DANE for TLS Client Authentication

- Original drafts developed in mid 2015
  - TLS Client Authentication via DANE TLSA Records:
    - https://tools.ietf.org/html/draft-huque-dane-client-cert
  - TLS Extension to convey DANE Client Identity:
    - https://tools.ietf.org/html/draft-huque-tls-dane-clientid
- Target use cases: IOT & SMTP Transport Security
- Presented at IETF93, July 2015, Prague, DANE Working Group
  - https://datatracker.ietf.org/meeting/93/materials/slides-93-dane-0
  - Subsequent discussion on list
  - DANE working group shutdown without recharter for new work; Authors did not have energy or time to continue working on it (at that time)
  - Every now & then, we are approached by misc parties about reviving this work.

Protocol Summary

- Client has a DNS domain name identity
  - A public/private keypair a certificate binding the public key to the domain name
  - Corresponding DANE TLSA record published in DNS
- (D)TLS server
  - Sends Certificate Request message in handshake; extracts client identity from presented certificate, constructs TLSA record; queries, and validates DANE TLSA response
- New TLS extension for conveying client's identity
  - For signaling support for DANE TLS client authentication (empty extension if signal only)
  - For conveying client DNS identity when used with TLS raw public key auth (RFC 7250)
  - Protect extension with ECH (Encrypted Client Hello) for privacy

Revising & Expanding the work

- Draft Revisions
  - Simplify SAN fields: dNSName & SRVname -> dNSName only
  - Record owner format changes; less proscriptive; more formats
- Object security applications of DANE
  - Neither TLSA or SMIMEA in their currently defined forms are ideally suited to this
  - TLSA is defined for TLS channel authentication
  - SMIMEA is object security for email applications. Its record format is defined in terms of email addresses, not necessarily ideally suited as an IOT identifier, etc.
  - New RRtype? Longer development and adoption lifecycle
  - Desired outcome: expand the scope of TLSA to cover object security
- DANE for Certificate Discovery
  - More on this later

Who wants to advance this work?

- Authors (of course)
- Colleagues at NIST, ICANN, LoRA Alliance and more ..
- DNSSEC and DANE proponents

Detailed Motivation

- Identity:
  - A name
  - A way to prove ownership of the name.
- Value of an identity system:
  - How widely-recognized is the name?
  - How resistant is the proof-of-identity to impersonation?
- IoT challenges:
  - Discovery of public key for message authentication/encryption -> proprietary APIs
  - Subjective entity names -> namespace collisions across CAs
  - Mfr to Enterprise PKI bootstrapping -> costly and time-consuming.
  - Constrained platforms, decoupled architecture
  - certdata.txt: **1.2MB** Espressif ESP32 SOC: **4MB flash**

DANE for IoT

DNS is the most widely-recognized namespace on the Internet

Public-key authentication is extremely resilient to impersonation

DANE binds **DNS** names to **public keys**.

- Eliminates naming collisions across CAs
- SDK for certificate discovery is already in the OS
- Attribution to responsible party via DNS hierarchy
- Current public key is always in DNS: **simplify certificate rotation**
- No need to distribute CA certs to devices: **discovery > distribution**

DANE for IoT: Implementation

- DNS labels
    - **_device** for organization/delegation point
    - a1b2c3._device.example.com
    - Similar to how **_smimecert** label (**RFC 8162**) organizes email identities.
    - Does not carry **RFC 8162**’s complexity for hashing email local part for DNS name.
    - Multiple sub-identities represented by left-hand labels (see **BCP 222**)
    - Underscore challenge: disallowed for publicly trusted certificates.
- Record type
    - TLSA: Allows a variety of representations for certs.
    - No changes in record required for client certificates.
- Not exclusively an IoT protocol
    - This could also simplify B2B, microservices commsec (think **_service** label)

DANE for IoT: Simplification

Network authentication:
    802.1x, EAP-TLS

Transport authentication:
    Mutual TLS authentication
    DNS-SD/mDNS companion

Message authentication:
    Public key discovery

Authz Policy:
    Permitted communication can be described as simply as network ACLs

DANE for IoT: Network Authentication

- **Simplify RADIUS config/management**
  - **Allow list is just a list of permitted entity DNS names**
  - **No CA certificate management**
  - **Less need to re-key to enterprise PKI**

- **Simplify support for raw public keys:**
  - **RFC 7250 - TLS with raw public keys**
  - **https://tools.ietf.org/id/draft-chen-emu-eap-tls-ibs-00.html**

DANE for IoT: Transport Authentication

- **Traverse signing authorities**
  - **DNS names are not bound to CA namespace guarantee**
  - **Any two devices with public keys in DNS may mutually authenticate**

- **Complement existing discovery capabilities**
  - **mDNS indicates services available and entity's DANE id**
  - **DNS/DANE used to authenticate**

- **Simplify configuration**
  - **Permitted Interactions can be represented as simply as a L3ACL**
    - **${CLIENT} may authenticate to ${SERVER}**

DANE for IoT: Object Security

- **Simplify object authentication**
  - **Message bears signer's DNS name**
  - **Signer's DNS name used to retrieve public key**
  - **No need to sync cert store with CA API**

- **Simplify object encryption**
  - **Sender uses recipient's DNS name to retrieve public key**
  - **End-to-end encryption w/ DNS as public key discovery mechanism**

- **Simplify configuration**
  - **Permitted interaction patterns are simplified:**
    - **${SENDER} may use ${MIDDLEWARE} to reach ${RECIPIENT}**

DANE for IoT: Life with DANE

- **Identity suppliers**
  - ○ **Attribution via DNS hierarchy.**
  - ○ **Owner can repudiate identity by deleting record.**
  - ○ **Secure hardware can ship with immediately-usable DANE identity.**

- **Implementers**
  - ○ **Describe interactions within applications using DNS names.**
  - ○ **Reduce time to implement.**

- **Application owners**
  - ○ **Authentication mechanism is not proprietary.**
  - ○ **Pick best-of-breed components, based on standardized protocols.**
  - ○ **Simplify application component lifecycle.**

Current Work

DANE for Client Identity, Dane ClientID extension for TLS

- Immediate protocol benefits:
  - mTLS
  - Oauth2 mTLS (RFC 8705)
  - EAP-TLS: Use mfr-issued PKI for network authentication

Upcoming Work

DANE for certificate discovery

- DANE-lite: use TLSA for certificate discovery, PKIX authenticated cert
- Proposal: add a new Cert Usage mode (4: PKIX-CD)
- Ultimate goal is to use DNSSEC authenticated TLSA everywhere
- But as we know, DNSSEC is today very sparsely deployed, and this presents a significant challenge to DANE adoption
- Narrow use case allows initial adoption of DANE w/o DNSSEC
- Incentive to gradually realize DNSSEC benefits later

Immediate benefits: JOSE/COSE/OSCORE

How should the work continue?

Working group placement:

- TLS might be a good choice, given the mutual auth use case + extension
- UTA?
- DNSOP?
- A new working group?
- [Something else]

Discussion