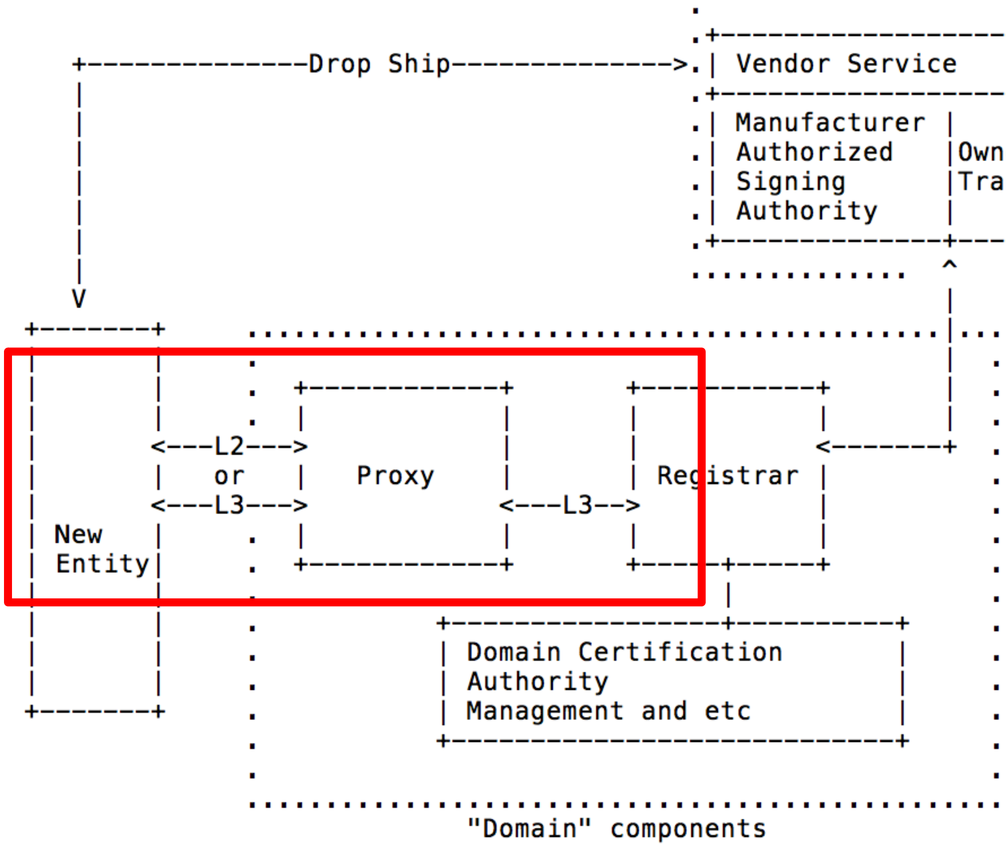
Bootstrapping Key Infrastructures

Max Pritikin

IETF 94, 2 Nov 2015

s2 Architectural Overview



```
                                    .
                                    .+------------------------+
 +--------------Drop Ship--------------->.| Vendor Service         |
 |                                  .+------------------------+
 |                                  .| Manufacturer |         |
 |                                  .| Authorized   |Ownership|
 |                                  .| Signing      |Tracker  |
 |                                  .| Authority    |         |
 |                                  .+--------------+---------+
 |                                  ..............  ^
 V                                                  |
+-------+     ............................................|...
|       |     .                                           |  .
|       |     .  +------------+        +-----------+      |  .
|       |     .  |            |        |           |      |  .
|       |<--L2-->|            |        |           |<-------+  .
|       |   or   |   Proxy    |        | Registrar |         .
|       |<--L3-->|            |<--L3-->|           |         .
| New   |     .  |            |        |           |         .
| Entity|     .  +------------+        +---+-------+         .
|       |     .                             |                 .
|       |     .          +-----------------+----------+      .
|       |     .          | Domain Certification       |      .
|       |     .          | Authority                  |      .
|       |     .          | Management and etc         |      .
+-------+     .          +----------------------------+      .
              .                                              .
              ................................................
                         "Domain" components
```

Figure 1

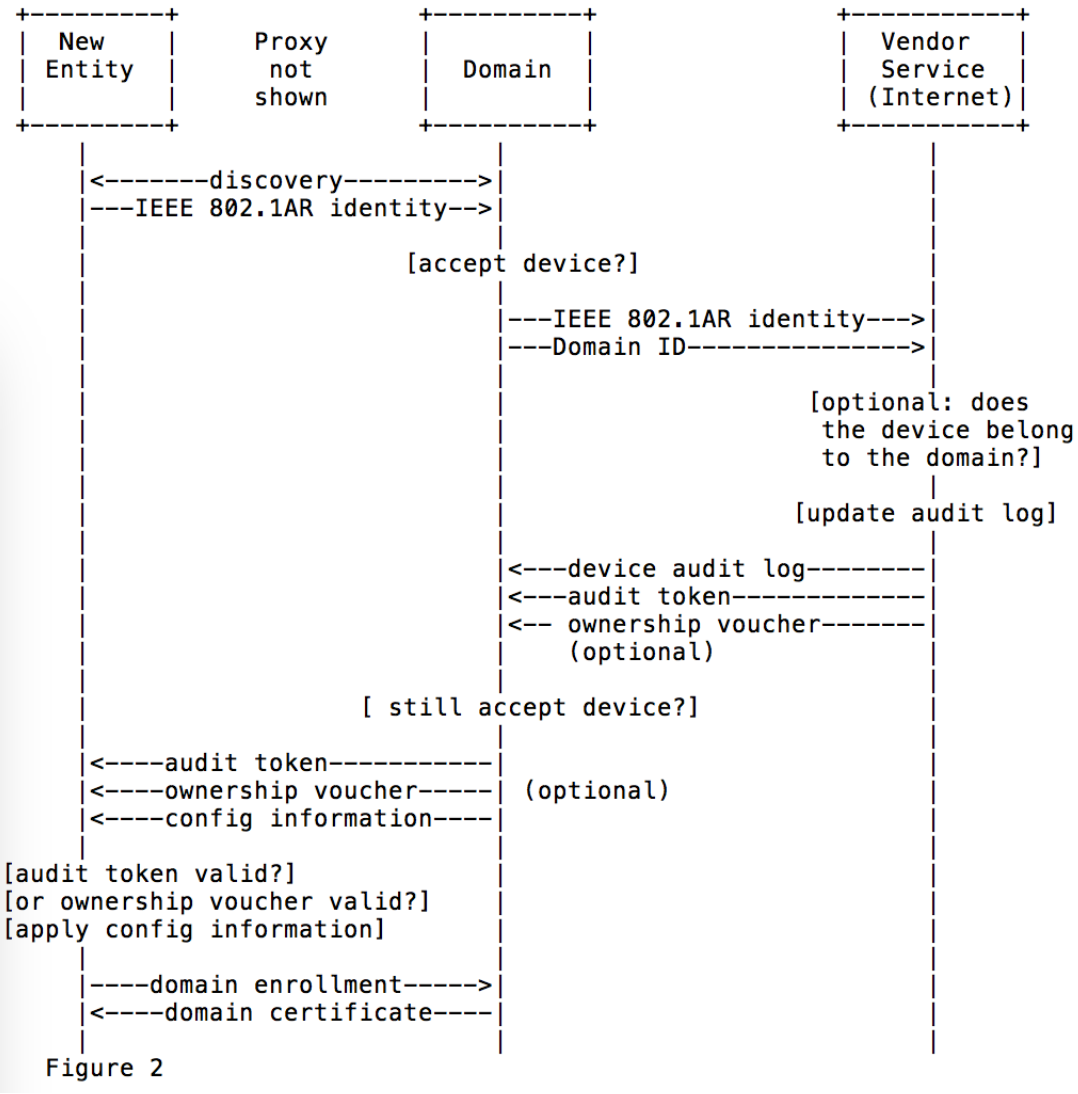
New Entity always communicates with an AN Proxy.

Proxy forwards communication to the Registrar.

Proxy behavior is “dumb” – it only forwards the packets. See s3.2.

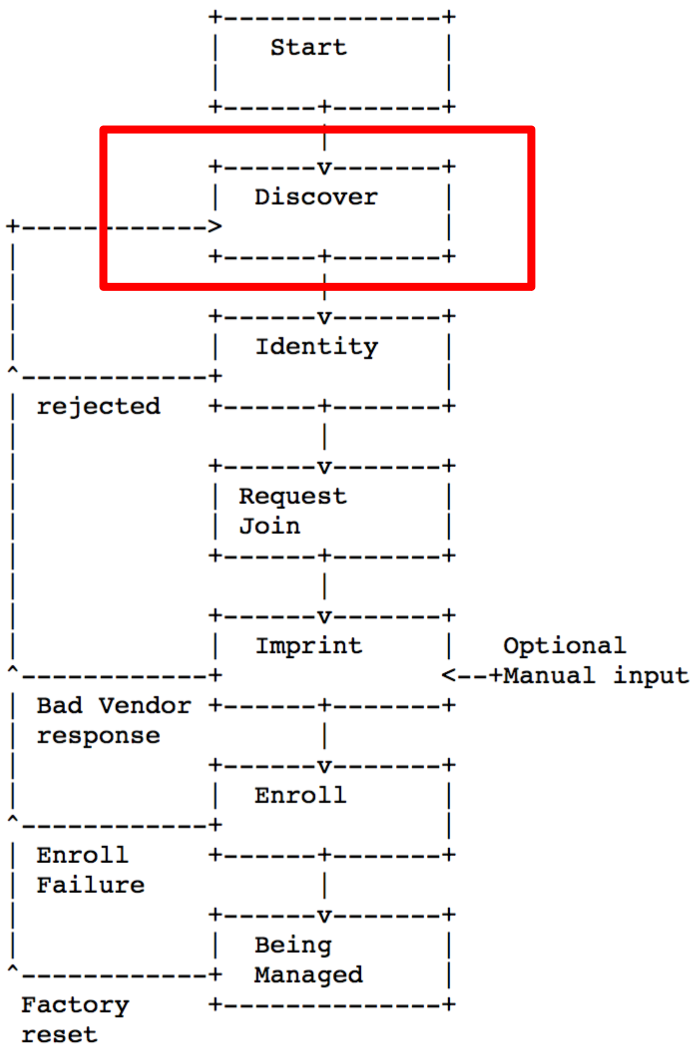
avoiding:
EAP/PANA/1x/provisional

s3 Functional Overview



```
 +----------+               +----------+                +-----------+
 |  New     |    Proxy      |          |                |  Vendor   |
 | Entity   |     not       |  Domain  |                |  Service  |
 |          |    shown      |          |                | (Internet)|
 +----------+               +----------+                +-----------+
     |                           |                            |
     |<-------discovery--------->|                            |
     |---IEEE 802.1AR identity-->|                            |
     |                           |                            |
     |                     [accept device?]                   |
     |                           |                            |
     |                           |---IEEE 802.1AR identity--->|
     |                           |---Domain ID--------------->|
     |                           |                            |
     |                           |                    [optional: does
     |                           |                     the device belong
     |                           |                     to the domain?]
     |                           |                            |
     |                           |                   [update audit log]
     |                           |                            |
     |                           |<---device audit log--------|
     |                           |<---audit token-------------|
     |                           |<-- ownership voucher-------|
     |                           |    (optional)              |
     |                           |                            |
     |                  [ still accept device?]               |
     |                           |                            |
     |<----audit token-----------|                            |
     |<----ownership voucher-----| (optional)                 |
     |<----config information----|                            |
     |                           |                            |
[audit token valid?]             |                            |
[or ownership voucher valid?]    |                            |
[apply config information]       |                            |
     |                           |                            |
     |----domain enrollment----->|                            |
     |<----domain certificate----|                            |
     |                           |                            |
   Figure 2
```

s3.1.1 StateMachine:Discovery



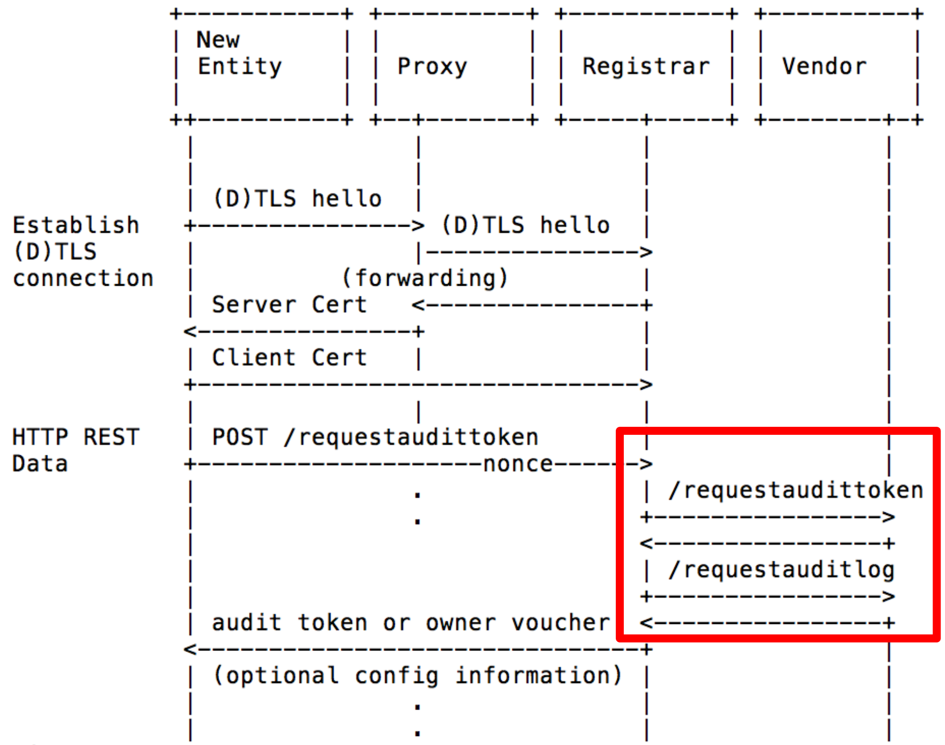
```
           +---------------+
           |   Start       |
           |               |
           +-------+-------+
                   |
           +-------v-------+
           | Discover      |
+------------>             |
|          +-------+-------+
|                  |
|          +-------v-------+
|          | Identity      |
^------------+             |
| rejected +-------+-------+
|                  |
|          +-------v-------+
|          | Request       |
|          | Join          |
|          +-------+-------+
|                  |
|          +-------v-------+
|          | Imprint       |   Optional
^------------+             <--+Manual input
| Bad Vendor +-------+-------+
| response         |
|          +-------v-------+
|          | Enroll        |
^------------+             |
| Enroll   +-------+-------+
| Failure          |
|          +-------v-------+
|          | Being         |
^------------+ Managed     |
  Factory  +---------------+
  reset
```

Figure 3

- MUST: Obtains a local address using either IPv4 or IPv6 methods.

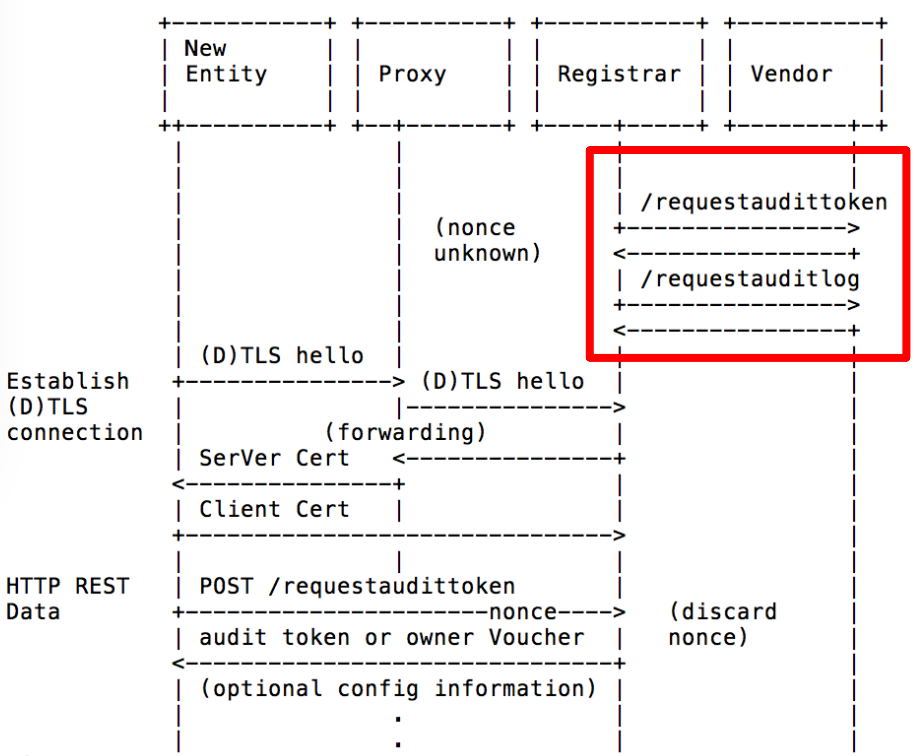
1. MUST: Attempt to establish a (D)TLS to well known AN port of neighbor.
2. MUST: unsecured-GRASP as a link local discovery method.
3. MAY: Performs DNS-SD over Multicast DNS for "_bootstrapks._tcp.local.”
4. MAY: Performs DNS-SD "_bootstrapks._tcp.example.com”.
5. MAY: contacts a well known vendor provided "bootstrapks.vendor-example.com".

s5 Protocol Details



```
            +-----------+ +----------+ +-----------+ +----------+
            | New       | |          | |           | |          |
            | Entity    | | Proxy    | | Registrar | | Vendor   |
            |           | |          | |           | |          |
            ++----------+ +--+-------+ +-----+-----+ +--------+-+
             |               |               |                |
             |               |               |                |
             | (D)TLS hello  |               |                |
Establish    +---------------> (D)TLS hello  |                |
(D)TLS       |               |--------------->                |
connection   |           (forwarding)        |                |
             | Server Cert   <---------------+                |
             <---------------+               |                |
             | Client Cert   |               |                |
             +------------------------------->                |
             |               |               |                |
HTTP REST    | POST /requestaudittoken       |                |
Data         +--------------------nonce------>                |
             |               .               | /requestaudittoken
             |               .               +---------------->
             |                               <----------------+
             |                               | /requestauditlog
             |                               +---------------->
             | audit token or owner voucher  <----------------+
             <-------------------------------+                |
             | (optional config information) |                |
             |               .               |                |
             |               .               |                |
```

Figure 5



```
            +-----------+ +----------+ +-----------+ +----------+
            | New       | |          | |           | |          |
            | Entity    | | Proxy    | | Registrar | | Vendor   |
            |           | |          | |           | |          |
            ++----------+ +--+-------+ +-----+-----+ +--------+-+
             |               |               |                |
             |               |               |                |
             |               |               | /requestaudittoken
             |               | (nonce        +---------------->
             |               | unknown)      <----------------+
             |               |               | /requestauditlog
             |               |               +---------------->
             |               |               <----------------+
             | (D)TLS hello  |               |                |
Establish    +---------------> (D)TLS hello  |                |
(D)TLS       |               |--------------->                |
connection   |           (forwarding)        |                |
             | SerVer Cert   <---------------+                |
             <---------------+               |                |
             | Client Cert   |               |                |
             +------------------------------->                |
             |               |               |                |
HTTP REST    | POST /requestaudittoken       |                |
Data         +--------------------nonce------>  (discard      |
             | audit token or owner Voucher  |  nonce)        |
             <-------------------------------+                |
             | (optional config information) |                |
             |               .               |                |
             |               .               |                |
```

Figure 6

Nonce based to avoid depending on a valid clock.
Supports nonce-less operations but see Security Considerations for discussion

TCP and UDP

- [anima-bootstrapping-keyinfra] just says “(D)TLS” but protocol details are written as if EST was the base protocol. This assumes HTTP which only works over UDP if the MTU is respected.
- [grasp] s3.3.2 “The protocol is capable of running over UDP or TCP, except for link-local multicast discovery messages, which can only run over UDP and MUST NOT be fragmented, and therefore cannot exceed the link MTU size.”
- [RESTCONF] also uses HTTP and requires a reliable transport-layer.

- This is an area for discussion

Transport Protocol

- Current doc builds on EST [RFC7030] as a simple to extend REST interface that results in a certificate enrollment.
- Discussions have included overlap/integration with GRASP.
- CoAP? Talk to ACE?

MASA/NetConf Zerotouch

- Feeling more comfortable with alignment
- Need some text about vendor going out of business or unresponsive or malicious