DRIP Authentication Protocols & Formats for Broadcast RID

draft-ietf-drip-auth-15

Adam Wiethuechter (AX Enterprize, LLC)

ICAO SAM Codes

- Used for Authentication Type 0x5 and our main outer encapsulation in F3411
- Status of ICAO process for allocation is unknown
  - Plan to contact someone in F3411 to get an idea who to talk to and get the process started informally (to at least give heads up)
- We need a IETF liaison to perform this for the WG officially
- Is a circular dependency as we need values allocated to publish draft, they need draft to review for allocation approval

Draft Status

Changes from –05 to -15

- Reordered some sections
- Restructured Section 6 to make it flow better
- Lots of explanation text added

Next Steps

- Technical side document is ready
- Send for review to selection of ASTM F3411 members
- Other reviews (need up to 3)?

DRIP Entity Tag Registration & Lookup

draft-ietf-drip-registries-05

Adam Wiethuechter (AX Enterprize, LLC)

Tim Mesker (InfoNetworks LLC)

Changes from –01 to -05

- Closed items
  - Contributors fix, Andrei's text (Appendix C), RAA/HDA text from –uas-rid, title change
- New items
  - ICAO DNS Structure
  - CERT OID
  - Improved EPP
    - Thanks to Len Bayles from InfoNetworks LLC on the review and modifications
  - X.509 Appendix
    - Stub from Bob, needs more work from him

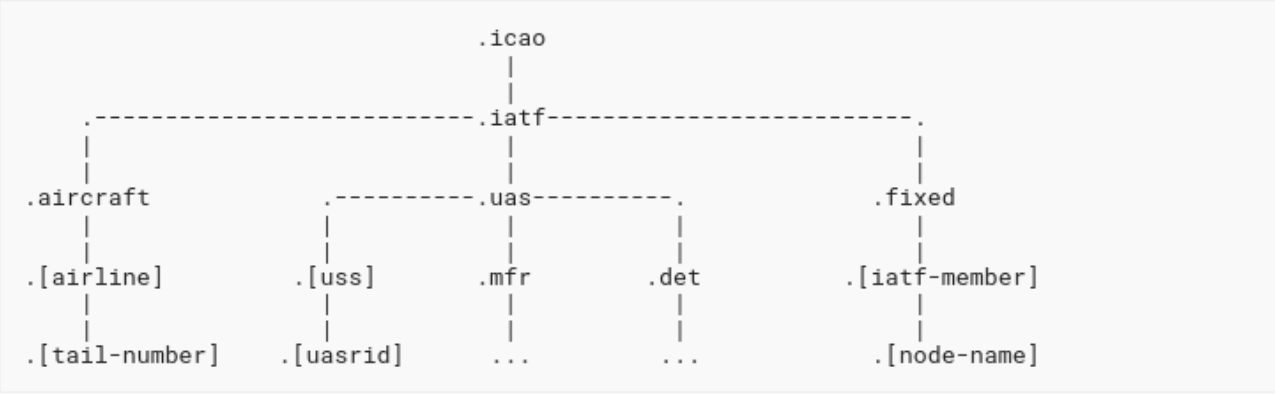
Working Items

CERT OIDs for DRIP Attestation/Certificates

- Using Bob's personal OID (1.3.6.1.4.1.6715)
- Using 2 for DRIP work
- Next value is for format type (based on Attestation/Certificate type)
- Next value is for entity type
  - UA, GCS, HDA, etc.
- Example
  - 1.3.6.1.4.1.6715.6.4.1 => Broadcast Attestation: HDA on UA

ICAO DNS Structure

- All DRIP DNS work should fit under "uas.icao.int"
- "uas.icao.int" would be the "root" of the DRIP DNS structure and follow in subdomains as specified in draft
- This also ties into recent mailing list discssion on use of ".arpa"



Example DET FQDN: `a3ad19520ad0a69e.5.20.10.20010030.det.uas.iatf.icao`

Example MFR FQDN: `Z2T7B8RA85D19LX.8653.mfr.uas.iatf.icao`

RAA Allocation Clarifications

- RAAs are expected to be Civil Aviation Authorities (CAA's)
  - Could be other entities, not in scope to specify
- RAAs uses a value of HDA=0 as part of full hierarchy (HID)
- If CAA wants to do interactions an HDA would do (e.g., register Operators)
  - Has control of the HDA space under them, so just allocate what they need!
  - SHOULD use HDA=1 for this purpose; allocate as such in draft?
- RAA value 1 is allocated for ICAO Registry of Manufacturers (IRM)

Discussion Items

Inter-document dependencies

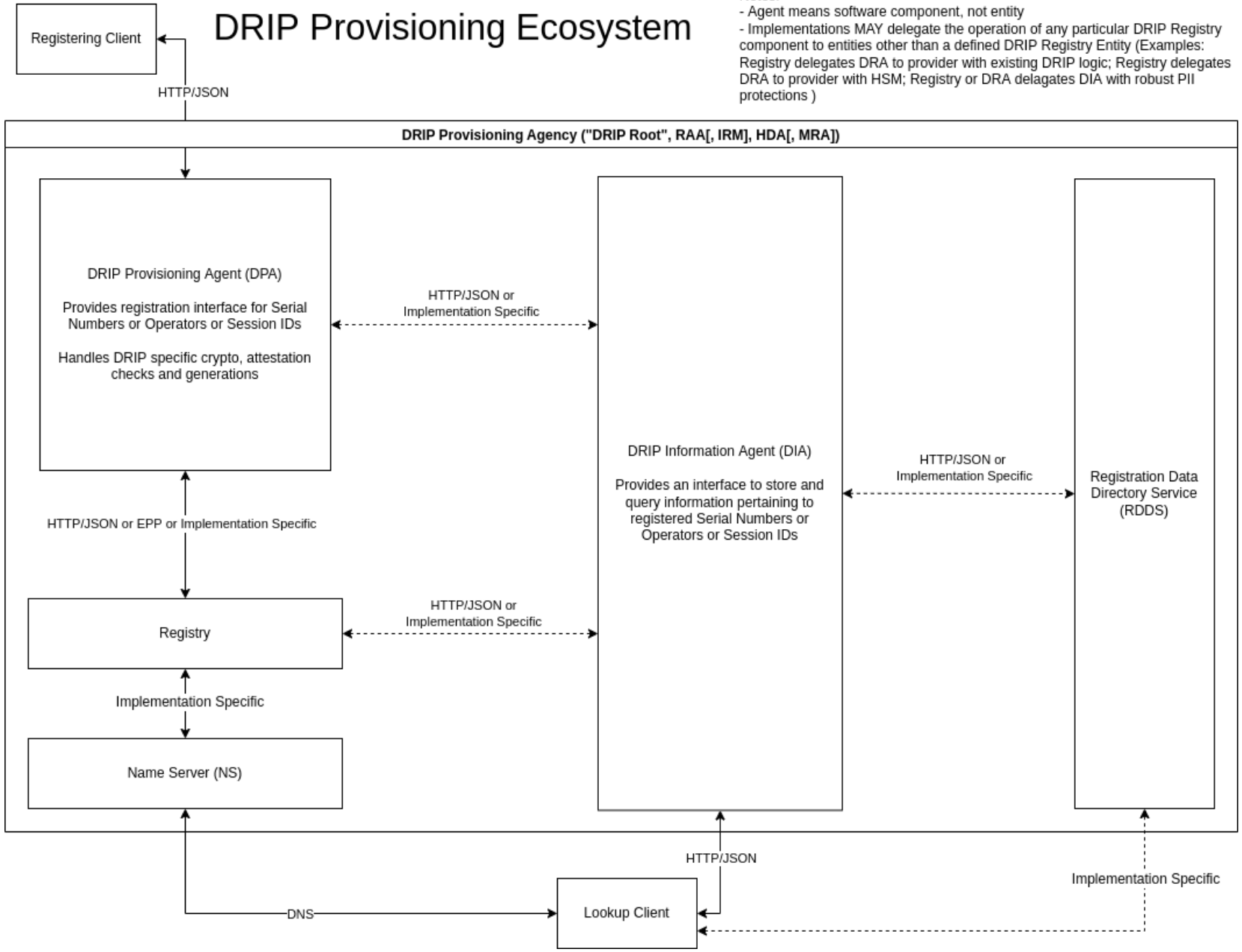
- Very hard to trace and understand DRIP concepts when coming in from registries draft
  - Even with RFC9153 and Architecture citations
  - Took over 20+ hours of Zoom calls (not just emails) to get context and understanding
  - Mix of confusion with DETs/HHIT specification details and how/why various choices made
- Thanks to InfoNetworks for this feedback!
- Personally, will make a more direct effort to avoid this again

Using term registrar

- Being a registrar has a very specific meaning in DNS world
  - Will levy various processes and regulations onto to-be registrar operator
  - This can be very prohibitive for the DRIP use case; impeding adopting
    - We are targeting USS vendors to run the functions as it makes sense in their existing architecture
- In short; specify registrar like functions but don't call it that
- Need different term;
  - DRIP Provisioning Agency? RID-USS?
- Will affect DRIP Architecture that uses the term; should be replaced with selected term and definition added

DRIP Provisioning Ecosystem

Notes:
- Agent means software component, not entity
- Implementations MAY delegate the operation of any particular DRIP Registry component to entities other than a defined DRIP Registry Entity (Examples: Registry delegates DRA to provider with existing DRIP logic; Registry delegates DRA to provider with HSM; Registry or DRA delagates DIA with robust PII protections)



DRIP Provisioning Agent (DPA)

- The main logical entity for user interaction during registering a DET Session ID
- Handles the DRIP based logic such as
  - Cryptographic operations (creating Attestations/Certificates)
  - Validation of DET properties and if valid for registration
  - Generation of the DNS based record content
- RRs made by DPA MUST be put DNS
  - If not co-located, in scope using either an OpenAPI or EPP
  - If co-located out of scope as implementation specific

DRIP Information Agent (DIA)

- Main entity for client interaction looking up "private" data via DET Session ID
- MUST use a differentiated access mechanism and policy for information release
  - DRIP standardizes RDAP as required in this capacity
- Backed by RDDS via some mechanism (out of scope)

Proposed new document(s) structure

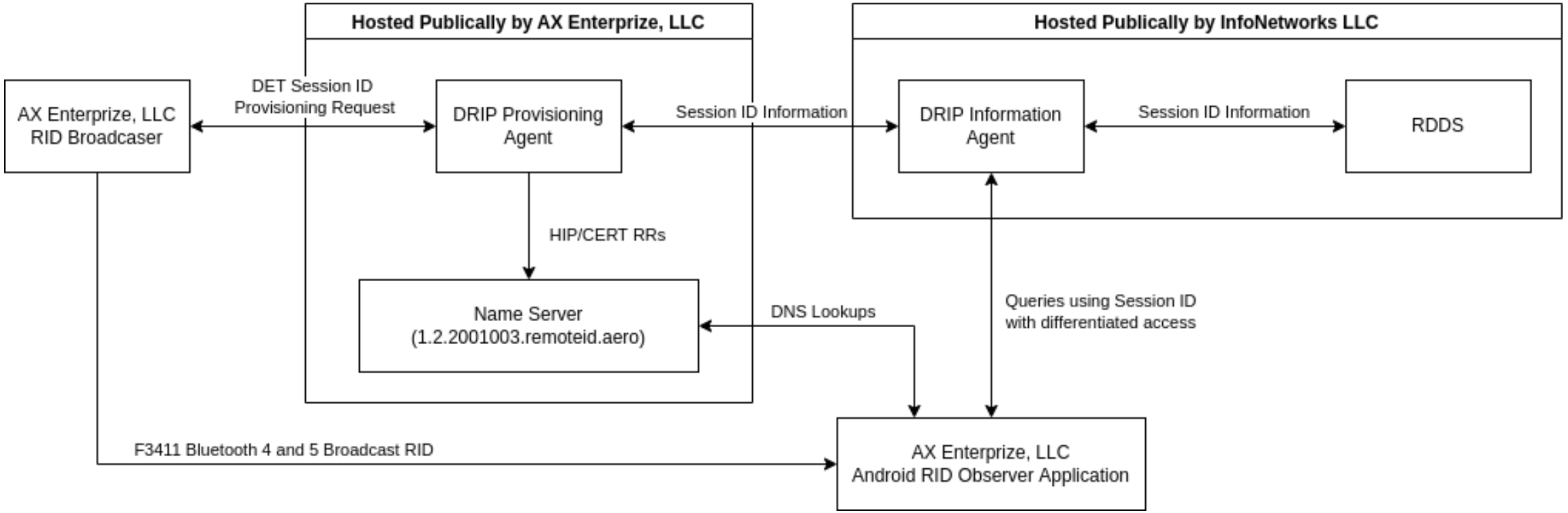
- Current document becomes an Architecture for DET based registries
  - High level architecture and entities (previous slides)
    - Specific interaction examples and reasoning in UAS use case
  - Lists of supported/required RRs and reasoning
  - Definitions of Attestations/Certificates
- New documents for:
  - Interface between Provisioning Clients and DPA (OpenAPI)
  - Interface between DPA and Registry
    - EPP
    - OpenAPI
  - Interface between DIA and DPA/Registry/Lookup Clients (OpenAPI)

Next Steps

- Handle registrar terminology swap
  - Mainly in –arch as priority
- Draft scope changes
  - Do we put registry architecture text in the main architecture document, or do we just change this document to be a registry architecture?
  - Spawn new implementation documents (based on previous question)?
  - Can perform a dry run and send to mailing list a zip of all the drafts to get opinion of a rough cut of the changes
    - Not final titles, but headers with content moved around and roughly explained

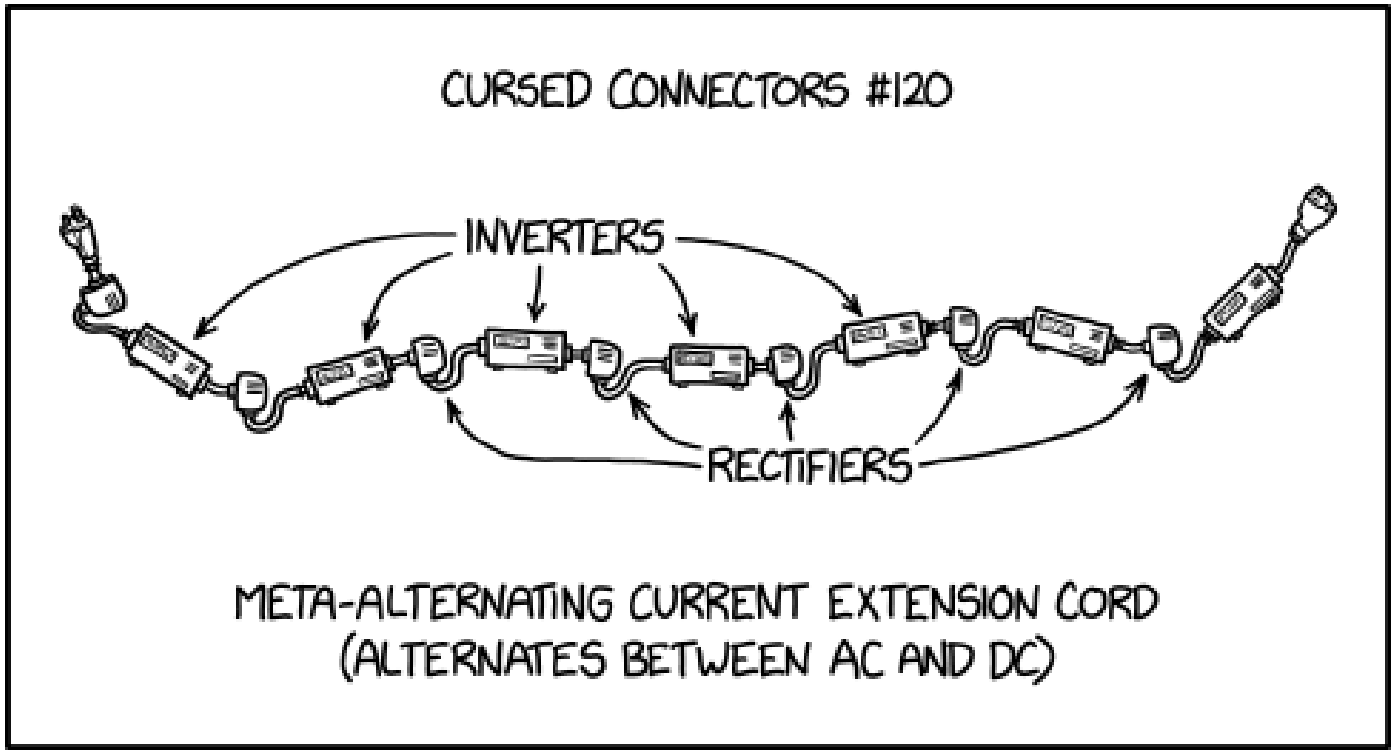
Hackdemo Happy Hour

- Live demos of registration and differentiated lookup



Discussion

Questions, Comments, Concerns?



"It's always bothered me that you can't cancel out an inverter by putting a second inverter after it."
https://xkcd.com/2642