Workgroup: drip Working Group
Internet-Draft:
draft-wiethuechter-drip-detim-arch-00
Published: 27 September 2022
Intended Status: Standards Track
Expires: 31 March 2023
Authors: A. Wiethuechter   S. Card
         AX Enterprize, LLC   AX Enterprize, LLC
         R. Moskowitz   J. Reid
         HTT Consulting   RTFM llp

DRIP Entity Tag (DET) Identity Management Architecture

Abstract

This document describes the high level architecture for the registration and discovery of DRIP Entity Tags (DETs) using DNS technologies and practices. Discovery of DETs and their artifacts are through the existing DNS structure and methods by using FQDNs. A general overview of the interfaces required between components is described in this document with supporting documents giving technical specifications.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Registries are fundamental to Remote ID (RID). Only very limited operational information can be Broadcast, but extended information is sometimes needed. The most essential element of information sent is the UAS ID itself, the unique key for lookup of extended information in registries.

While it is expected that registry functions will be integrated with USS, who will provide them is not yet determined in most, and is expected to vary between, jurisdictions. However this evolves, the essential registry functions, starting with management of identifiers, are expected to remain the same, so are specified herein.

While most data to be sent via Broadcast or Network RID is public, much of the extended information in registries will be private. Thus AAA for registries is essential, not just to ensure that access is granted only to strongly authenticated, duly authorized parties, but also to support subsequent attribution of any leaks, audit of who accessed information when and for what purpose, etc. As specific AAA requirements will vary by jurisdictional regulation, provider philosophy, customer demand, etc., they are left to specification in policies, which should be human readable to facilitate analysis and discussion, and machine readable to enable automated enforcement, using a language amenable to both, e.g., XACML.

The intent of the negative and positive access control requirements on registries is to ensure that no member of the public would be hindered from accessing public information, while only duly authorized parties would be enabled to access private information. Mitigation of Denial of Service attacks and refusal to allow database mass scraping would be based on those behaviors, not on identity or role of the party submitting the query per se, but querant identity information might be gathered (by security systems protecting DRIP implementations) on such misbehavior.

Registration under DRIP is vital to manage the inevitable collisions in the hash portion of the DET. Forgery of the DET is still possible, but including it as a part of a public registration mitigates this risk. This document creates the DRIP DET registration and discovery ecosystem. This includes all components in the ecosystem (e.g., RAA, HDA, UA, GCS, USS).

1.1. Abstract Process & Reasoning

In DRIP each entity (registry, operator and aircraft) is expected to generate a full DRIP Entity ID [drip-rid] on the local device their key is expected to be used. These are registered with a Public Information Registry within the hierarchy along with whatever data is required by the cognizant CAA and the registry. Any PII is stored in a Private Information Registry protected through industry practice AAA or better. In response, the entity will obtain an endorsement from the registry proving such registration.

Manufacturers that wish to participate in DRIP should not only support DRIP as a Session ID type for their aircraft but also generate a DET then encode it as a Serial Number. This would allow aircraft under CAA mandates to fly only ID Type 1 (Serial Number) could still use DRIP and most of its benefits. Even if DRIP is not supported for Serial Numbers by a Manufacturer it is hoped that they would still run a registry to store their Serial Numbers and allow look ups for generic model information. This look up could be especially helpful in UTM for Situational Awareness when an aircraft flying with a Serial Number is detected and allow for an aircraft profile to be displayed.

Operators are registered with a number of registries or their regional RAA. This acts as a verification check when a user performs other registration operations; such as provisioning an aircraft with a new Session ID. It is an open question if an Operator registers to their CAA (the RAA) or multiple USS's (HDA's). PII of the Operator would vary based on the CAA they are under and the registry.

Finally aircraft that support using a DET would provision per flight to a USS, proposing a DET to the registry to generate a binding between the aircraft (Session ID, Serial Number and Operational Intent), operator and registry. Aircraft then follow [drip-auth] to meet various requirements from [RFC9153] during flight.

2. Terminology

2.1. Required Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2.2. Additional Definitions

See [RFC9153] for common DRIP terms and [drip-arch] Section 2.2 for additional terms used in this document.

HDA:

Hierarchial HIT Domain Authority. The 14 bit field identifying the HIT Domain Authority under a RAA.

HID:

Hierarchy ID. The 28 bit field providing the HIT Hierarchy ID.

PII:

Personally Identifiable Information. Any information a cognizant authority (such as a government agency) or a user requires differentiated access to obtain.

RAA:

Registered Assigning Authority. The 14 bit field identifying the Hierarchical HIT Assigning Authority.

3. DIME Roles

The DRIP Identity Management Entity (DIME) is an entity encompassed various logical components (Section 4) and can be classified to serve a number of different roles (this section). The general hierarchy of these roles are illustrated in Figure 1.

```
                +----------+
                |   Apex   |
                +-o------o-+
                  |      |
******************|******|*****************************
                  |      |
            +-----o-+  +-o-----+
RAAs        |  IRM  |  |  RAA  o------.
            +---o---+  +---o---+      '
                |          |          |
****************|**********|**********|****************
                |          |          |
            +---o---+  +---o---+  +---o---+
HDAs        |  MRA  |  | RIDR  |  |  HDA  |
            +-------+  +-------+  +-------+
```

Figure 1: Registry Hierarchy

3.1. Apex

The apex is special DIME role that holds the value of RAA=0 and HDA=0. It serves as the branch point from the larger DNS system in which HHITs are defined. The Apex generally has prefix portions of the HHIT associated with it (such as 2001:0030/28) which are assigned by IANA from the non-routable special IPv6 address space for ORCHIDs (where HHITs are derived from).

The Apex manages all delegations and allocations of the HHIT's RAA to various parties with NS records to redirect DNS queries to proper sub-branches.

3.2. Registered Assigning Authority (RAA)

RAA's are the upper hierarchy in DRIP (denoted by a 14-bit field (16,384 RAAs) of an HHIT). An RAA is a business or organization that manages a registry of HDAs (Section 3.3). Most are contemplated to be Civil Aviation Authorities (CAA), such as the Federal Aviation Authority (FAA), that then delegate HDAs to manage their National Air Space (NAS). This is does not preclude other entities to operate an RAA if the Apex allows it.

For DRIP and the UAS use case ICAO will handle the registration of RAAs. Once ICAO accepts an RAA, it will assign a number and create a zone delegation under the <prefix>.hhit.arpa. DNS zone for the RAA.

As DETs may be used in many different domains, RAA should be allocated in blocks with consideration on the likely size of a particular usage. Alternatively, different prefixes can be used to separate different domains of use of HHITs.

An RAA must provide a set of services to allocate HDAs to organizations. It must have a public policy on what is necessary to obtain an HDA. It must maintain a DNS zone minimally for discovering HID RVS servers. All RAA's use an HDA value of 0 and have their RAA value delegated to them by the Root.

3.2.1. ICAO Registry of Manufacturers (IRM)

An RAA-level DIME that hands out HDA values to participating Manufacturer's that hold an ICAO Manufacturer Code used in [CTA2063A].

To manage the large ICAO Manufacturer Code space (34 character set; 4 characters; 1,336,336 possible codes) a range of RAA values are set aside for the DRIP use case. These are the RAA values of 2 (0x0002) up to 96 (0x0060). This allows a single HDA for each Manufacturer Code.

All IRM's have two reserved HDA values. 0 (0x0000) for itself in its role as an RAA and 1 (0x0001) if it wishes to offer HDA services.

3.3. Hierarchial HIT Domain Authority (HDA)

An HDA may be an USS, ISP, or any third party that takes on the business to register the actual UAS entities that need DETs. This includes, but is not limited to UA, GCS, and Operators. It should also provide needed UAS services including those required for HIP-enabled devices (e.g. RVS).

The HDA is a 14-bit field (16,384 HDAs per RAA) of a DET assigned by an RAA. An HDA should maintain a set of RVS servers for UAS clients that may use HIP. How this is done and scales to the potentially millions of customers are outside the scope of this document. This service should be discoverable through the DNS zone maintained by the HDA's RAA.

An RAA may assign a block of values to an individual organization. This is completely up to the individual RAA's published policy for delegation. Such policy is out of scope.

3.3.1. Manufacturers Registry of Aircraft (MRA)

An HDA-level DIME run by a manufacturer of UAS systems that participate in Remote ID. Stores UAS Serial Numbers under a specific ICAO Manufacturer Code (assigned to the manufacturer by ICAO).

A DET can be encoded into a Serial Number (see [drip-rid]) and this registry would hold a mapping from the Serial Number to the DET and its artifacts.

3.3.1.1. Remote ID Registries (RIDR)

An HDA-level DIME that holds the binding between a UAS Session ID (for DRIP the DET) and the UA Serial Number. The Serial Number MUST have its access protected to allow only authorized parties to obtain. The Serial Number SHOULD be encrypted in a way only the authorized party can decrypt.

As part of the UTM system they also hold a binding between a UAS ID (Serial Number or Session ID) and an Operational Intent. They may either be a direct logical part of a UAS Service Supplier (USS) or be a UTM wide service to USS's.

3.4. Role Abbreviation in DETs

On receiver devices a DET can be translated to a more human readable form such as: {RAA Abbreviation} {HDA Abbreviation} {Last 4 Characters of DET Hash}. An example of this would be US FAA FE23. To support this DIMEs are RECOMMENDED to have an abbreviation that could be used for this form. These abbreviations SHOULD be a maximum of six characters in length. Spaces SHOULD NOT be used and be replaced with either underscores (_) or dashes (-). For RAAs the abbreviation is RECOMMENDED to be set to the ISO 3166 country code (either Alpha-2 or Alpha-3) for the CAA.

If a DIME does not have an abbreviation or it can not be looked up then the receiver SHOULD use the hexadecimal encoding of the field it is missing.

4. DIME Architecture

```
+--------------------+
| Registering Client |
+------------o-------+
             |
*************|**********************************************************
*            |              DRIP Indentity Management Entity
*            |
*     +------o-------+              +-------------+              +------
*     | DRIP         |              |             |              |
*     | Provisioning o--------------o             |              |
*     | Agent        |              |             |              |
*     +-------o------+              |             |              |
*             |                     |             |              |
*             |                     | DRIP        |              | Regis
*     +-------o--+                  | Information o--------------o Data
*     | Registry o------------------o Agent       |              | Direc
*     +-------o--+                  |             |              | Servi
*             |                     |             |              |
*             |                     |             |              |
*     +-------o-----+               |             |              |
*     | Name Server |               |             |              |
*     +------o------+               +-----o-------+              +------
*            |                            |
*            |                            |
*************|****************************|*****************************
             |                            |
             |                    +-------o-------+
             '--------------------o Lookup Client o---------------------
                                  +---------------+
```

Figure 2: Registry Hierarchy

The DIME, in any of its roles (Section 3), is comprised of a number of logical components that perform specific functions. Any of these components in Section 4 could be delegated to other entities as a service both co-located or remote. For example the Name Server component could be handled by a well established DNS registrar/registry with the DRIP Provisioning Agent (DPA) (Section 4.1) interfacing to them. Another common example may be the DPA, Registry and Name Server are all co-located in one implementation with an interface to a DRIP Information Agent (DIA) offered by another organization.

4.1. DRIP Provisioning Agent (DPA)

The DPA performs the important task of vetting information (such as the DRIP Endorsements) coming from clients wishing to register and then delegate (internally or externally) various items to other components in the DIME.

A standard interface over HTTPS MUST be provided for clients to access with JSON or CBOR encoding of objects being sent to the DPA. This interface specification is out of scope for this document.

There MUST be an interface from the DPA to a Registry (Section 4.2) component which handles the DNS specific requirements of the DIME as defined by the Registry. There MAY also be interface from the DPA to a DRIP Information Agent (Section 4.4) as defined by the DIA.

4.2. Registry

The Registry component handles all the required DNS based requirements of the DIME to function for DRIP. This includes the registration and maintenance of various DNS Resource Records which use the DRIP FQDNs (Section 7.2).

A standardized interface MUST be implemented for interactions with the DPA (Section 4.1). This interface MAY be over HTTPS using JSON/CBOR encoding or MAY use the Extensional Provisioning Protocol (EPP) [RFC5730]. The specifications of either of these interfaces is out of scope for this document.

There MAY be interface from the Registry to a DRIP Information Agent (Section 4.4) as defined by the DIA.

4.3. Name Server (NS)

This may be very important here as we should not preclude a USS from running his own Name Server but they are not DNS experts and will need guidance or at least pointers to it to not mess it up. Such as SOA and NS formats to allow delegation if as RAA.

Most of time is probably outsourced.

The interface of the Name Server to any component (nominally the Registry) in a DIME is out of scope as typically they are implementation specific.

4.4. DRIP Information Agent (DIA)

The DIA is the main component handling requests for information from entities outside of the DIME. Typically this is when an Observer looks up a Session ID from an UA and gets pointed to the DIA via a SVR RR to obtain information not available via DNS.

The information contained in the DIA is generally more oriented around the Operator of a given UAS and is thus classified as Personally Identifiable Information (PII). To protect the privacy of an Operator of the UAS this information is not publicly accessible and is only available behind policy driven differentiated access mechanisms. As an example the Serial Number, under the FAA, is classified as PII and can only be accessed by federal entities (such as the FAA themselves).

For DRIP the Registration Data Access Protocol (RDAP) ([RFC7480], [RFC9082] and [RFC9083]) is the selected protocol to provide policy driven differentiated access for queries of information.

A standard interface over HTTPS MUST be provided for clients to access with JSON/CBOR encoding of objects being sent to the DIA. There MUST also be a standardized interface for the DPA or Registry to add, update or delete information into the DIA. Both of these interfaces are out of scope for this document.

An interface defined by the Registration Data Directory Service (RDDS) (Section 4.5) is also required as specified by the RDDS.

4.5. Registration Data Directory Service (RDDS)

This is the primary information database for the DIA. An interface MUST be provided to the DIA but its specification is out of scope as they are typically implementation specific.

5. Registration/Provisioning Process

The general process for a registering party is as follows:

1. Verify input Endorsement(s) from registering party

2. Check for collision of DET and HI

3. Populate Registry/Name Server with required/optional resource records using the FQDN

4. Populate DIA/RDDS with PII and other info

5. Generate and return required/optional Endorsements/Certificates

In the following subsections an abbreviated form of Section 4 using component abbreviations is used to describe the flow of information. The data elements being transmitted between entities is marked accordingly in each figure for the specific examples.

5.1. Serial Number

Primarily registered to MRA's (Section 3.3.1) by the Manufacturers. Could be also registered to CAA's (using their HDA functionality) as part of Operator registration or to USS's in their capacity as HDAs. In the later two cases no DNS RRs are made to protect the privacy of the registering parties.

When the Serial Number is really an encoded DET the DET FQDN is used to point to HIP and CERT RRs rather than the Serial Number FQDN. Instead a CNAME is made between the Serial Number FQDN and the DET FQDN. The same can still happen if the manufacturer chooses to use their own Serial Number formatting (still within the specification of [CTA2063A]) and create the CNAME back to a DET loaded into the unmanned aircraft.

```
    +-------------------+
    | Unmanned Aircraft |
    +--o---o------------+
       |   ^
   (a) |   | (b)
       |   |
*******|***|*****************************
*      |   |    DIME: MRA               *
*      |   |                            *
*      v   |             +----------+   *
*   +--o---o--+          |          |   *
*   |   DPA   o--------->o          |   *
*   +----o----+   (d)    |          |   *
*        |               |          |   *
*        | (c)           | DIA/RDDS |   *
*        v               |          |   *
*   +----o--------+      |          |   *
*   | Registry/NS |      |          |   *
*   +-------------+      |          |   *
*                        +----------+   *
*                                       *
*****************************************

(a) Serial Number, UA Information, UA Self-Endorsement
(b) Success Code, Endorsement: MRA on UA
(c) HIP RR, CERT RRs
(d) UA Information
```

Figure 3: Example DIME:MRA with Serial Number (DET) Registration

The unmanned aircraft, intending to use DRIP, generates a keypair, DET and Self-Endorsement: UA using the RAA and HDA values specified by the manufacturers DIME (running as an MRA). The DET is converted into a Serial Number (per [drip-rid]) or the manufacturer creates their own Serial Number.

The Serial Number, UA information and the Self-Endorsement: UA are sent to the manufacturers DIME. The DIME validates the Self-Endorsement and checks for DET and HI collisions in the Name Server/DIA. A Broadcast Endorsement: DIME on UA is generated which is provisioned into the aircraft for use when using the Serial Number as its UAS ID. In the Name Server HIP RRs are created using the DET FQDN while a CNAME points the Serial Number FQDN to the DET FQDN.

> Note: Figure 3 is specific for a DET encoded Serial Number. The Endorsements in (a) and (b) as well as RRs in (c) would not be present for non-DET based Serial Numbers.

5.2. Operator

Either by USS or CAA run HDAs. Regulation might require interaction between them. An Operator can request that certain information normally generated and provisioned into DNS be omitted due to privacy concerns.

```
            +----------+
            | Operator |
            +--o---o---+
               |   ^
           (a) |   | (b)
               |   |
        *******|***|*****************************
        *      |   |    DIME: HDA               *
        *      |   |                            *
        *      v   |             +----------+   *
        *   +--o---o--+          |          |   *
        *   |   DPA   o--------->o          |   *
        *   +----o----+   (d)    |          |   *
        *        |               |          |   *
        *        | (c)           | DIA/RDDS |   *
        *        v               |          |   *
        *   +----o--------+      |          |   *
        *   | Registry/NS |      |          |   *
        *   +-------------+      |          |   *
        *                        +----------+   *
        *                                       *
        *****************************************

        (a) Operator Information, Operator Self-Endorsement
        (b) Success Code, Endorsement: HDA on Operator
        (c) HIP RR, CERT RRs
        (d) Operator Information
```

Figure 4: Example DIME:HDA with Operator (DET) Registration

The Operator generates a keypair and DET as specified in [drip-rid] along with a self-signed endorsement (Self-Endorsement: Operator). The RAA and HDA values used in the DET generation for the Operator are found by referencing their selected DIME of choice (in Figure 4 an HDA).

The self-signed endorsement along with other relevant information (such as Operator PII) is sent to the DIME over a secure channel. The specification of this secure channel is out of scope for this document.

The DIME cross checks any personally identifiable information as required. Self-Endorsement: Operator is verified. The DET and HI is searched in the DIME DIA and Name Server to confirm that no collisions occur. A new endorsement is generated (Endorsement: DIME on Operator) and sent securely back to the Operator. Resource Records for the HI and Endorsements are added to the DIME Registry/ Name Server.

With the receipt of Endorsement: DIME on Operator the registration of the Operator is complete.

> Note: (c) in Figure 4 MAY be requested by the Operator to be omitted due to PII concerns.

5.3. Session ID

Session IDs are generally handled by HDAs, specifically RIDRs. In Figure 5 the UAS comprises of an unmanned aircraft and a Ground Control Station (GCS). Both parties are involved in the registration process.

```
    +---------+
    |   UAS   |
    +--o---o--+
       |   ^
   (a) |   | (b)
       |   |
*******|***|*****************************
*      |   |    DIME: RIDR             *
*      |   |                           *
*      v   |            +----------+   *
*   +--o---o--+         |          |   *
*   |   DPA   o--------->o          |   *
*   +----o----+   (d)   |          |   *
*        |              |          |   *
*        | (c)          | DIA/RDDS |   *
*        v              |          |   *
*   +----o--------+     |          |   *
*   | Registry/NS |     |          |   *
*   +-------------+     |          |   *
*                       +----------+   *
*                                      *
*****************************************

(a) Mutual Endorsement: RIDR on GCS, Endorsement: GCS on UA, Session ID
(b) Success Code, Broadcast Endorsement: RIDR on UA, Endorsement: RIDR o
(c) HIP RR, CERT RRs
(d) Session ID Information
```

Figure 5: Example DIME:RIDR with Session ID (DET) Registration

Through mechanisms not specified in this document the Operator should have methods (via the GCS) to instruct the unmanned aircraft onboard systems to generate a keypair, DET and Self-Endorsement: UA. The Self-Endorsement: UA is extracted by the Operator onto the GCS.

The GCS is already pre-provisioned and registered to the DIME with its own keypair, DET, Self-Endorsement: GCS and Endorsement: RIDR on GCS. The GCS creates a new Endorsement: GCS on UA and also creates Mutual Endorsement: RIDR on GCS. These new endorsements along with Session ID Information are sent to the DIME via a secure channel.

The DIME validates all the endorsements and checks for DET and HI collisions in the Name Server/DIA using the proposed UA DET. A Broadcast Endorsement: DIME on UA is generated. An Endorsement: RIDR on UAS is generated using the Endorsement: GCS on UA. HIP and CERT RRs are provisioned into the Registry/Name server. Both endorsements are back to the GCS on a secure channel.

The GCS then injects the Broadcast Endorsement: RIDR on UA securely into the unmanned aircraft. Endorsement: RIDR on GCS is securely stored by the GCS.

> Note: in Figure 5 the Session ID Information is expected to contain the Serial Number along with other PII specific information (such as UTM data) related to the Session ID.

5.3.1. UA Based

There MAY be some unmanned aircraft that have their own Internet connectivity allowing them to register a Session ID themselves without outside help from other devices such as a GCS. When such a system is in use its imperative that the Operator has some method to create the Endorsement: Operator on UA to send to the DIME. The process and methods to perform this are out of scope for this document but MUST be done in a secure fashion.

5.3.2. UAS Based

Most unmanned aircraft will not have their own Internet connectivity but will have a connection to a GCS. Typically a GCS is an application on a user device (such as smartphone) that allow the user to fly their aircraft. For the Session ID registration the DIME MUST be provided with an Endorsement: GCS on UA which implies there is some mechanism extracting and inserting information from the unmanned aircraft to the GCS. These methods MUST be secure but are out of scope for this document.

With this system it is also possible to have the GCS generate the DET based Session ID and insert it securely into the unmanned aircraft after registration is done. This is NOT RECOMMENDED as this invalidates the objective of the asymmetric cryptography in the underlying DET as the private key MAY get in the posession of another entity other than the unmanned aircraft. See Section 11.2 for more details.

5.4. Child DIME

TODO

6. Differentiated Access Process

High level explanation of differentiated access goals and requirements.

7. DRIP in the Domain Name System

The individual DETs may be potentially too numerous (e.g., 60 - 600M) and dynamic (e.g., new DETs every minute for some HDAs) to store in a signed, DNS zone. The HDA SHOULD provide DNS service for its zone and provide the DET detail response.

DNSSEC is strongly recommended (especially for RAA-level zones). Frequency of updates, size of the zone, and registry policy may impact its use.

Per [drip-arch] all public information is stored in the DNS to satisfy REG-1 from [RFC9153]. CERT RRs (Section 7.3.3) contain public Endorsements or X.509 Certificate relevant to a given Session ID. SVR RRs (Section 7.3.5) point an Observer to a service to obtain further information if they have and can prove duly constituted authority.

7.1. Prefix to TLD Mapping

For DRIP, the prefix 2001:0030/28 is slated for DETs being used in UAS. Other prefixes may be allocated by IANA in future for different use cases that do not fit cleanly into an existing prefix.

IANA registry for this?

If so we could remove prefix from FQDN form...Stu would like this to happen

7.2. DRIP Fully Qualified Domain Names

7.2.1. DRIP Entity Tag

7.2.1.1. Forward Lookup

The DET has the following FQDN form:

```
{hash}.{oga_id}.{hda}.{raa}.{prefix}.hhit.arpa.
```

When building a DET FQDN the following two things must be done:

1. The RAA, HDA and OGA ID values MUST be converted from hexadecimal to decimal form.

2. The FQDN must be built using the exploded (all padding present) form of the IPv6 address.

Below is an example:

```
DET: 2001:0030:00a0:0145:a3ad:1952:0ad0:a69e
ID: a3ad:1952:0ad0:a69e
OGA: 5
HDA: 0014 = 20
RAA: 000a = 10
Prefix: 2001003
FQDN: a3ad19520ad0a69e.5.20.10.2001003.hhit.arpa.
```

Note: any of the fields in the FQDN could be CNAME'd to more human readable interpretations. For example the DET FQDN 204.2001003.hhit.arpa. may have a CNAME to uas.faa.gov; if RAA 204 was delegated to the FAA.

7.2.1.2. Reverse Lookup

The DET reverse lookup should be a standard IPv6 reverse address in ip6.arpa..

```
$ORIGIN  5.4.1.0.0.a.0.0.0.3.0.0.1.0.0.2.ip6.arpa.
e.9.6.a.0.d.a.0.2.5.9.1.d.a.3.a    IN   PTR
```

7.2.2. Serial Number

See Section 4.2 of [drip-rid] for how to encode DETs as Serial Numbers.

```
Serial Number: 8653FZ2T7B8RA85D19LX
ICAO Mfr Code: 8653
Length Code: F
ID: FZ2T7B8RA85D19LX
FQDN: Z2T7B8RA85D19LX.8653.mfr.hhit.arpa.
```

Serial Number pose a unique problem. If we explicitly only allow HHITs be under the hhit.arpa. domain structure how do we standardize the lookup of Serial Numbers? Perhaps to look up Serial Numbers one must go to a different tree like mfr.icao.int.? We can have CNAMEs in MRAs for this but they probably need the same TLD (hhit.arpa.) to be found properly and these are clearly not HHITs.

7.3. Supported DNS Records

7.3.1. HIP

All DIMEs will use HIP RR [RFC8005] as the primary public source of DET HIs. The DETs are encoded in an FQDN (Section 7.2.1) and are the lookup key for the RR. DIMEs have their own DET associated with them and their respective name server will hold a HIP RR that is pointed to by their DET FQDN.

MRA (Section 3.3.1) and RIDR (Section 3.3.1.1) DIMEs will also have HIP RRs for their registered parties (aircraft and operators respectfully).

7.3.2. TLSA

This RR, [RFC6698], is mainly used to support DTLS deployments where the DET is used (e.g. Network RID and the wider UTM system). The HI is encoded using the SubjectPublicKeyInfo selector. DANE [RFC6698] is for servers, DANCE [dane-clients] is for clients.

The TLSA RR MAY be used in place of the HIP RR, where to primary need of the DET HI is for DTLS authentication. This DNS server side optimization is for where the overhead of both RR is onerous. Thus all clients that work with the HIP RR SHOULD be able to able to extract the HI from the TLSA RR.

7.3.3. CERT

Endorsements can be placed into DNS in the CERT RRs [RFC4398]. An exception to this is the Attestation Certificate made during Session ID registration. This is as this particular certificate acts similar to a car registration and should be held safe by the operator.

Endorsements will be stored in Certificate Type OID Private (value 254) with a base OID of 1.3.6.1.4.1.6715.2 and further classified by the Endorsement/Certificate Type and then Entities involved.

Endorsement Type	OID Value
Self-Endorsement	1
Endorsement	2
Concise Endorsement	3
Mutual Endorsement	4
Link Endorsement	5
Broadcast Endorsement	6

Table 1

Entity Type	OID Value
Unmanned Aircraft (UA)	1
Ground Control Station (GCS)	2
Operator (OP)	3
HDA	4
RAA	5
Root	6

Table 2

As an example the following OID: 1.3.6.1.4.1.6715.2.6.4.1 would decompose into: the base OID (1.3.6.1.4.1.6715.2), the Endorsement

Type (6: Broadcast Endorsement) and then the parties involved (4: HDA, 1: UA)

Certificate Type X.509 as per PKIX (value 1) MAY be used to store X.509 certificates as discussed in (Appendix B).

Editor Note: This OID is an initial allocation under the IANA Enterprise Number OID. It is expect that a general OID will be allocated at some point.

7.3.4. NS

Used to interconnect entities

7.3.5. SVR

The SVR RR for DRIP always is populated at the "local" registry level. That is an HDA's DNS would hold the SVR RR that points to that HDAs private registry for all data it manages. This data includes data being stored on its children.

The best example of this is RIDR (Section 3.3.1.1) would have a SVR RR that points to a database that contains any extra information of a Session ID it has registered. Another example is the MRA (Section 3.3.1) has a SVR RR pointing to where the metadata of a UA registered in the MRA can be located.

In all cases the server being pointed to MUST be protected using AAA, such as using RDAP.

7.3.6. CNAME

Used for SN -> DET mapping and other cross TLD jumps?

8. Endorsements

Under DRIP Endorsements are defined in a JSON structure that can be encode to CBOR or have their keys removed and be sent as a binary blob. When the latter is used very specific forms are defined with naming conventions to know the data fields and their lengths for parsing.

The first subsection defines the structure of an Endorsement while the remain subsections define specific forms that are commonly used. The binary forms of the subsections can be found in Appendix A.

8.1. Endorsement Structure

```
endorsement_struct = {
    "identity": {
        "hhit": "base16 HHIT/DET",
        "hi_b16": "base16 HI",
        "hi_b64": "base64 HI"
    },
    "evidence": [
        endorsement_struct,
        "base16 data",
        "base64 data"
    ],
    "scope": {
        "vnb": 0,
        "vna": 0
    }
    "signature": {
        "sig_b16": "base16 Signature",
        "sig_b64": "base64 Signature"
    }
}
```

Figure 6: Endorsement JSON Structure

8.1.1. Identity

The identity section is where the main identity information of the signer of the endorsement is found. This can be in many forms such as the the Base16 encoded HHIT or the raw Host Identity (HI) in either Base16 or Base64.

8.1.2. Evidence

The evidence section contain a list of the claims being asserted in the endorsement. The list order after signing can not be tampered with (resulting in different signatures) and is its content is generally well defined in specific endorsements.

The content may be a blob in Base16/Base64 or be another endorsement structure.

8.1.3. Scope

The scope section is more formally "the scope of validity of the endorsement". The scope can come in various forms but MUST always have a "valid not before" (vnb) and "valid not after" (vna) timestamps.

Other forms of the scope could for example be a 4-dimensional volume definition. This could be in raw latitude, longitude, altitude pairs or may be a URI pointing to scope information.

8.1.4. Signature

The signature section contain the signature data for the endorsement. The signature itself MUST be in either Base16 or Base64 strings. Other forms or data elements could also be present in the signature section if specified in a specific endorsement.

8.2. Self-Endorsement (SE-x)

```
self_endorsement = {
    "identity": {
        "hhit": "base16 HHIT/DET"
    },
    "evidence": [
        "base16 host identity"
    ],
    "scope": {
        "vnb": 0,
        "vna": 0
    }
    "signature": {
        "sig_b16": "base16 Signature"
    }
}
```

Figure 7: Self-Endorsement JSON Structure

In a Self-Endorsement the identity is a Base16 HHIT/DET, the evidence is a single element array containing the Base16 HI, and the signature is in Base16.

8.3. Endorsement (E-x.y)

```
endorsement = {
    "identity": {
        "hhit": "base16 HHIT/DET of X",
        "hi_b16": "base16 HI of X"
    },
    "evidence": [
        self_endorsement
    ],
    "scope": {
        "vnb": 0,
        "vna": 0
    }
    "signature": {
        "sig_b16": "base16 Signature of X"
    }
}
```

Figure 8: Endorsement JSON Structure

8.4. Concise Endorsement (CE-x.y)

In constrained environments and when there is the guarantee of being able to lookup the DETs to obtain HIs this endorsement can be used.

```
concise_endorsement = {
    "identity": {
        "hhit": "base16 HHIT/DET of X",
    },
    "evidence": [
        "base16 HHIT/DET of Y"
    ],
    "scope": {
        "vnb": 0,
        "vna": 0
    }
    "signature": {
        "sig_b16": "base16 Signature of X"
    }
}
```

Figure 9: Concise Endorsement JSON Structure

8.5. Mutual Endorsement (ME-x.y)

An endorsement that perform a sign over an existing Endorsement where the signer is the second party of the embedded endorsement. The DET of party Y is used as the identity.

```
mutual_endorsement = {
    "identity": {
        "hhit": "base16 HHIT/DET of Y",
    },
    "evidence": [
        endorsement
    ],
    "scope": {
        "vnb": 0,
        "vna": 0
    }
    "signature": {
        "sig_b16": "base16 Signature of Y"
    }
}
```

Figure 10: Mutual Endorsement JSON Structure

8.6. Link Endorsement (LE-x.y)

An endorsement that perform a sign over an existing Concise Endorsement where the signer is the second party of the embedded endorsement. The DET of party Y is used as the identity.

```
link_endorsement = {
    "identity": {
        "hhit": "base16 HHIT/DET of Y",
    },
    "evidence": [
        concise_endorsement
    ],
    "scope": {
        "vnb": 0,
        "vna": 0
    }
    "signature": {
        "sig_b16": "base16 Signature of Y"
    }
}
```

Figure 11: Link Endorsement JSON Structure

8.7. Broadcast Endorsement (BE-x.y)

```
broadcast_endorsement = {
    "identity": {
        "hhit": "base16 HHIT/DET of X",
    },
    "evidence": [
        "base16 HHIT/DET of Y",
        "base16 HI of Y
    ],
    "scope": {
        "vnb": 0,
        "vna": 0
    }
    "signature": {
        "sig_b16": "base16 Signature of X"
    }
}
```

Figure 12: Broadcast Endorsement JSON Structure

This endorsement is required by DRIP Authentication Formats & Protocols for Broadcast RID ([drip-auth]) to satisfy [RFC9153] GEN-1 and GEN-3 and is sent in its binary form (Appendix A.6).

8.8. Abbreviations & File Naming Conventions

The names of endorsements can become quite long and tedious to write out. As such this section provides a guide to a somewhat standardized way they are written in text.

8.8.1. In Text Abbreviation

In a long form the name of a particular endorsement can be written as follows:

*Self-Endorsement: Unmanned Aircraft

*Endorsement: Operator on Aircraft or Endorsement: Operator, Aircraft

When multiple entities are listed they can be separated by either on or by ,. These long forms can be shortened:

*SE(Unmanned Aircraft) or SE-ua

*E(Operator, Unmanned Aircraft) or E-op.ua

Typical abbreviations for the entity can be used such as Unmanned Aircraft being shorthanded to ua.

8.8.2. File Naming

For file naming of various endorsements a similar format to the short form is used:

*se-{hash of entity}

*e-{hash of entity x}_{hash of entity y}

Some examples of file names:

*se-79d8a404d48f2ef9.cert

*e-120b8f25b198c1e1_79d8a404d48f2ef9.cert

9. X.509 Certificates

9.1. Certificate Policy and Certificate Stores

X.509 certificates are optional for the DRIP entities covered in this document. DRIP endpoint entities (EE) (i.e., UA, GCS, and Operators) may benefit from having X.509 certificates. Most of these certificates will be for their DET and some will be for other UAS identities. To provide for these certificates, some of the other entities covered in this document will also have certificates to create and manage the necessary PKI structure.

Any Certificate Authority (CA) supporting DRIP entities SHOULD adhere to the ICAO's International Aviation Trust Framework (IATF) Certificate Policy [ICAO-IATF-CP-draft]. The CA(s) supporting this CP MUST either be a part of the IATF Bridge PKI or part of the IATF CA Trust List.

EEs may use their X.509 certificates, rather than their rawPublicKey (i.e. HI) in authentication protocols (as not all may support rawPublicKey identities). Some EE HI may not be 'worth' supporting the overhead of X.509. Short lived DETs like those used for a single operation or even for a day's operations may not benefit from X.509. Creating then almost immediately revoking these certificates is a considerable burden on all parts of the system. Even using a short not AfterDate will completely mitigate the burden of managing these certificates. That said, many EEs will benefit to offset the effort. It may also be a regulator requirement to have these certificates.

Typically an HDA either does or does not issue a certificate for all its DETs. An RAA may specifically have some HDAs for DETs that do not want/need certificates and other HDAs for DETs that do need them. These types of HDAs could be managed by a single entity thus providing both environments for its customers.

It is recommended that DRIP X.509 certificates be stored as DNS TLSA Resource Records. This not only generally improves certificate lookups, but also enables use of DANE [RFC6698] for the various servers in the UTM and particularly DRIP registry environment and DANCE [dane-clients] for EEs (e.g. [drip-secure-nrid-c2]). All DRIP certificates MUST be available via RDAP. LDAP/OCSP access for other UTM and ICAO uses SHOULD also be provided.

9.2. Certificate Management

(mostly TBD still)

PKIX standard X.509 issuance practices should be used. The certificate request SHOULD be included in the DET registration request. A successful DET registration then MUST include certificate creation, store, and return to the DET registrant.

Certificate revocation will parallel DET revocation. TLSA RR MUST be deleted from DNS and RDAP, LDAP, and OCSP return revoked responses. CRLs SHOULD be maintained per the CP.

Details of this are left out, as there are a number of approaches and further research and experience will be needed.

9.3. Examples

TBD

9.4. Alternative Certificate Encoding

(CBOR encoded certs here. TBD)

10. IANA Considerations

TODO: requesting hhit.arpa

10.1. IANA DRIP Registry

This document requests a two new subregistries for Endorsement Type and Entity Type under the DRIP registry.

DRIP Endorsement Type: This 8-bit valued subregistry is for Endorsement Types to be used in OID's for CERT Resource Records. Future additions to this subregistry are to be made through Expert Review (Section 4.5 of [RFC8126]). The following values are defined:

Endorsement Type	Value
Self-Endorsement	1
Endorsement	2
Concise Endorsement	3
Mutual Endorsement	4
Link Endorsement	5
Broadcast Endorsement	6

DRIP Entity Type: This 8-bit valued subregistry is for Entity Types to be used in OID's for CERT Resource Records. Future additions to this subregistry are to be made through Expert Review (Section 4.5 of [RFC8126]). The following values are defined:

Entity Type	Value
Unmanned Aircraft (UA)	1
Ground Control Station (GCS)	2
Operator (OP)	3
HDA	4
RAA	5
Root	6

11. Security Considerations

11.1. Key Rollover & Federation

During key rollover the DIME MUST inform all children and parents of the change - using best standard practices of a key rollover. At time of writing this is signing over the new key with the previous key in a secure fashion and it being validated by others before changing any links (in DRIPs case the NS RRs in the parent registry).

A DET has a natural ability for a single DIME to hold different cryptographic identities under the same HID values. This is due to the lower 64-bits of the DET being a hash of the public key and the HID of the DET being generated. As such during key rollover, only the lower 64-bits would change and a check for a collision would be required.

This attribute of the DET to have different identities could also allow for a single registry to be "federated" across them if they share the same HID value. This method of deployment has not been thoroughly studied at this time.

11.2. DET Generation

Under the FAA [NPRM], it is expecting that IDs for UAS are assigned by the UTM and are generally one-time use. The methods for this however are unspecified leaving two options.

Option 1:

The entity generates its own DET, discovering and using the RAA and HDA for the target registry. The method for discovering a registry's RAA and HDA is out of scope here. This allows for the device to generate an DET to send to the registry to be accepted (thus generating the required Self-Endorsement) or denied.

Option 2:

The entity sends to the registry its HI for it to be hashed and result in the DET. The registry would then either accept (returning the DET to the device) or deny this pairing.

Keypairs are expected to be generated on the device hardware it will be used on. Due to hardware limitations and connectivity it is acceptable, though not recommended, under DRIP to generate keypairs for the Aircraft on Operator devices and later securely inject them into the Aircraft. The methods to securely inject and store keypair information in a "secure element" of the Aircraft is out of scope of this document.

12. References

12.1. Normative References

[drip-arch] Card, S. W., Wiethuechter, A., Moskowitz, R., Zhao, S., and A. Gurtov, "Drone Remote Identification Protocol (DRIP) Architecture", Work in Progress, Internet-Draft, draft-ietf-drip-arch-29, 16 August 2022, <https://www.ietf.org/archive/id/draft-ietf-drip-arch-29.txt>.

[drip-rid] Moskowitz, R., Card, S. W., Wiethuechter, A., and A. Gurtov, "UAS Remote ID", Work in Progress, Internet-Draft, draft-ietf-drip-uas-rid-01, 9 September 2020, <https://www.ietf.org/archive/id/draft-ietf-drip-uas-rid-01.txt>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/

RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9153] Card, S., Ed., Wiethuechter, A., Moskowitz, R., and A. Gurtov, "Drone Remote Identification Protocol (DRIP) Requirements and Terminology", RFC 9153, DOI 10.17487/RFC9153, February 2022, <https://www.rfc-editor.org/info/rfc9153>.

12.2. Informative References

[CTA2063A] "ANSI/CTA 2063-A Small Unmanned Aerial Systems Numbers", September 2019, <https://shop.cta.tech/products/small-unmanned-aerial-systems-serial-numbers>.

[dane-clients] Huque, S., Dukhovni, V., and A. Wilson, "TLS Client Authentication via DANE TLSA records", Work in Progress, Internet-Draft, draft-ietf-dance-client-auth-00, 24 March 2022, <https://www.ietf.org/archive/id/draft-ietf-dance-client-auth-00.txt>.

[drip-auth] Wiethuechter, A., Card, S. W., and R. Moskowitz, "DRIP Entity Tag Authentication Formats & Protocols for Broadcast Remote ID", Work in Progress, Internet-Draft, draft-ietf-drip-auth-21, 1 September 2022, <https://www.ietf.org/archive/id/draft-ietf-drip-auth-21.txt>.

[drip-secure-nrid-c2] Moskowitz, R., Card, S. W., Wiethuechter, A., and A. Gurtov, "Secure UAS Network RID and C2 Transport", Work in Progress, Internet-Draft, draft-moskowitz-drip-secure-nrid-c2-11, 23 July 2022, <https://www.ietf.org/archive/id/draft-moskowitz-drip-secure-nrid-c2-11.txt>.

[NPRM] "Notice of Proposed Rule Making on Remote Identification of Unmanned Aircraft Systems", December 2019.

[RFC4398] Josefsson, S., "Storing Certificates in the Domain Name System (DNS)", RFC 4398, DOI 10.17487/RFC4398, March 2006, <https://www.rfc-editor.org/info/rfc4398>.

[RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)", STD 69, RFC 5730, DOI 10.17487/RFC5730, August 2009, <https://www.rfc-editor.org/info/rfc5730>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer

Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <https://www.rfc-editor.org/info/rfc6698>.

[RFC7480] Newton, A., Ellacott, B., and N. Kong, "HTTP Usage in the Registration Data Access Protocol (RDAP)", STD 95, RFC 7480, DOI 10.17487/RFC7480, March 2015, <https://www.rfc-editor.org/info/rfc7480>.

[RFC8005] Laganier, J., "Host Identity Protocol (HIP) Domain Name System (DNS) Extension", RFC 8005, DOI 10.17487/RFC8005, October 2016, <https://www.rfc-editor.org/info/rfc8005>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC9082] Hollenbeck, S. and A. Newton, "Registration Data Access Protocol (RDAP) Query Format", STD 95, RFC 9082, DOI 10.17487/RFC9082, June 2021, <https://www.rfc-editor.org/info/rfc9082>.

[RFC9083] Hollenbeck, S. and A. Newton, "JSON Responses for the Registration Data Access Protocol (RDAP)", STD 95, RFC 9083, DOI 10.17487/RFC9083, June 2021, <https://www.rfc-editor.org/info/rfc9083>.

Appendix A. Binary Endorsements

A.1. Self-Endorsement (SE-x)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---------------+---------------+---------------+---------------+
|                                                               |
|                             DRIP                              |
|                          Entity Tag                           |
|                                                               |
+---------------+---------------+---------------+---------------+
|                                                               |
|                                                               |
|                                                               |
|                          Host Identity                        |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
+---------------+---------------+---------------+---------------+
|                        Valid Not Before                       |
+---------------+---------------+---------------+---------------+
|                        Valid Not After                        |
+---------------+---------------+---------------+---------------+
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                            Signature                          |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
|                                                               |
+---------------+---------------+---------------+---------------+
```

Figure 13: Binary Self-Endorsement (Length: 120-bytes)

A.2. Endorsement (E-x.y)

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                             DRIP                              |
    |                        Entity Tag of X                        |
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                                                               |
    |                                                               |
    |                       Host Identity of X                      |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    .                                                               .
    .                             SE-y                              .
    .                                                               .
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                     Valid Not Before by X                     |
    +---------------+---------------+---------------+---------------+
    |                     Valid Not After by X                      |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                         Signature by X                        |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    +---------------+---------------+---------------+---------------+
```

Figure 14: Binary Endorsement (Length: 240-bytes

A.3. Concise Endorsement (CE-x.y)

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +---------------+---------------+---------------+---------------+
   |                                                               |
   |                             DRIP                              |
   |                        Entity Tag of X                        |
   |                                                               |
   +---------------+---------------+---------------+---------------+
   |                                                               |
   |                             DRIP                              |
   |                        Entity Tag of Y                        |
   |                                                               |
   +---------------+---------------+---------------+---------------+
   |                     Valid Not Before by X                     |
   +---------------+---------------+---------------+---------------+
   |                     Valid Not After by X                      |
   +---------------+---------------+---------------+---------------+
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                         Signature by X                        |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   +---------------+---------------+---------------+---------------+
```

Figure 15: Binary Concise Endorsement (Length: 104-bytes

A.4. Mutual Endorsement (ME-x.y)

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                             DRIP                              |
    |                        Entity Tag of Y                        |
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    .                                                               .
    .                              E-xy                             .
    .                                                               .
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                     Valid Not Before by Y                     |
    +---------------+---------------+---------------+---------------+
    |                     Valid Not After by Y                      |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                         Signature by Y                        |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    +---------------+---------------+---------------+---------------+
```

Figure 16: Binary Mutual Endorsement (Length: 328-bytes

A.5. Link Endorsement (LE-x.y)

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +---------------+---------------+---------------+---------------+
   |                                                               |
   |                             DRIP                              |
   |                        Entity Tag of Y                        |
   |                                                               |
   +---------------+---------------+---------------+---------------+
   |                                                               |
   .                                                               .
   .                             CA-xy                             .
   .                                                               .
   |                                                               |
   +---------------+---------------+---------------+---------------+
   |                     Valid Not Before by Y                     |
   +---------------+---------------+---------------+---------------+
   |                     Valid Not After by Y                      |
   +---------------+---------------+---------------+---------------+
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                         Signature by Y                        |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   |                                                               |
   +---------------+---------------+---------------+---------------+
```

Figure 17: DRIP Link Endorsement (Length: 192-bytes)

A.6. Broadcast Endorsement (BE-x.y)

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                             DRIP                              |
    |                        Entity Tag of X                        |
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                             DRIP                              |
    |                        Entity Tag of Y                        |
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                                                               |
    |                                                               |
    |                       Host Identity of Y                      |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    +---------------+---------------+---------------+---------------+
    |                     Valid Not Before by X                     |
    +---------------+---------------+---------------+---------------+
    |                     Valid Not After by X                      |
    +---------------+---------------+---------------+---------------+
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                         Signature by X                        |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    |                                                               |
    +---------------+---------------+---------------+---------------+
```

Figure 18: DRIP Broadcast Endorsement (Length: 136-bytes)

Authors' Addresses

Adam Wiethuechter

AX Enterprize, LLC
4947 Commercial Drive
Yorkville, NY 13495
United States of America

Email: adam.wiethuechter@axenterprize.com

Stuart Card
AX Enterprize, LLC
4947 Commercial Drive
Yorkville, NY 13495
United States of America

Email: stu.card@axenterprize.com

Robert Moskowitz
HTT Consulting
Oak Park, MI 48237
United States of America

Email: rgm@labs.htt-consult.com

Jim Reid
RTFM llp
St Andrews House
382 Hillington Road, Glasgow Scotland
G51 4BL
United Kingdom

Email: jim@rfc1035.com