

drip Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: 6 May 2021

A. Wiethuechter  
S. Card  
AX Enterprize, LLC  
R. Moskowitz  
HTT Consulting  
2 November 2020

DRIP Identity Claims  
draft-wiethuechter-drip-identity-claims-03

## Abstract

This document describes the Identity Proofs (in the form of Claims, Certificates and Attestations) for use in various Drone Remote ID Protocols (DRIP) and the wider Unmanned Traffic Management (UTM) system.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 6 May 2021.

## Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1. Introduction](#) . . . . . [2](#)
- [1.1. Claims, Assertions, Attestations, and Certificates](#) . . . [2](#)
- [1.1.1. Claims](#) . . . . . [3](#)
- [1.1.2. Assertions](#) . . . . . [3](#)
- [1.1.3. Attestations](#) . . . . . [3](#)
- [1.1.4. Certificates](#) . . . . . [3](#)
- [2. Terminology](#) . . . . . [3](#)
- [2.1. Required Terminology](#) . . . . . [3](#)
- [2.2. Definitions](#) . . . . . [4](#)
- [3. DRIP Proofs](#) . . . . . [4](#)
- [3.1. Certificate: X on X \(Cxx Form\)](#) . . . . . [4](#)
- [3.1.1. Certificate: X on X \(Short Form\)](#) . . . . . [6](#)
- [3.2. Attestation: X on Y \(Axy Form\)](#) . . . . . [6](#)
- [3.2.1. Attestation: X on Y \(Short Form\)](#) . . . . . [8](#)
- [3.2.2. Attestation: X on Y \(Offline Form\)](#) . . . . . [9](#)
- [3.3. Timestamps](#) . . . . . [11](#)
- [3.4. Signatures](#) . . . . . [11](#)
- [4. Provisioning](#) . . . . . [11](#)
- [4.1. HHIT Delegation](#) . . . . . [11](#)
- [4.2. Manufacturer](#) . . . . . [12](#)
- [4.3. Registry](#) . . . . . [12](#)
- [4.4. Operator](#) . . . . . [13](#)
- [4.5. Aircraft](#) . . . . . [14](#)
- [4.5.1. Standard Provisioning](#) . . . . . [14](#)
- [4.5.2. Operator Assisted Provisioning](#) . . . . . [16](#)
- [4.5.3. Initial Provisioning](#) . . . . . [18](#)
- [5. Security Considerations](#) . . . . . [18](#)
- [6. References](#) . . . . . [18](#)
- [6.1. Normative References](#) . . . . . [18](#)
- [6.2. Informative References](#) . . . . . [18](#)
- Authors' Addresses . . . . . [19](#)

[1. Introduction](#)

DRIP Proofs are the backbone of trust in DRIP UAS RID, consisting of a chain of special certificates/attestations that results in a something useful in Broadcast RID. Some of the certificates are stored in and are generated by the Registries (defined in [[hhit-registries](#)]) and allow a user to confirm the trustworthiness of an Unmanned Aircraft (herein referred to as Aircraft) even in the scenario that the Observer is disconnected from the Internet.

## [1.1.](#) Claims, Assertions, Attestations, and Certificates

The authors wish to make a clear distinction on exactly what these terms mean in the context of DRIP.

Wiethuechter, et al.

Expires 6 May 2021

[Page 2]

---

Internet-Draft

identity-claims

November 2020

This is due to the term "certificate" having significant technologic and legal baggage associated with it, specifically around X.509 certificates. These type of certificates and Public Key Infrastructure invokes more legal and public policy considerations than probably any other electronic communication sector. It emerged as a governmental platform for trusted identity management and was pursued in intergovernmental bodies with links into treaty instruments.

As such much discussion has been made around the terms being used.

### [1.1.1.](#) Claims

For DRIP claims are used in the form of a predicate (X is Y, X has property Y, and most importantly X owns Y). The basic form of a claim is an entity using a HHIT as an identifier in the DRIP UAS system.

### [1.1.2.](#) Assertions

Assertions, under DRIP, are defined as being a set of one or more claims. This definition is borrowed from JWT/CWT. An HHIT in of itself is a set of assertions. First that the identifier is unique and is a handle to an asymmetric keypair owned by the entity and that it also is part of the given registry (specified by the HID).

### [1.1.3.](#) Attestations

An attestation is a signed claim. The signee may be the claimant themselves or a third party. Under DRIP this is normally used when a set of entities asserts a relationship between them along with other information.

### [1.1.4.](#) Certificates

Certificates in DRIP have a narrow definition of being signed

exclusively by a third party and are only over identities.

## [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.](#) Definitions

See [[drip-requirements](#)] for common DRIP terms.

HDA: Hierarchical HIT Domain Authority. The 16 bit field identifying the HIT Domain Authority under a RAA.

HID: Hierarchy ID. The 32 bit field providing the HIT Hierarchy ID.

RAA: Registered Assigning Authority. The 16 bit field identifying the Hierarchical HIT Assigning Authority.

## [3.](#) DRIP Proofs

The DRIP Proofs is a set of custom structures to be used in the USS/UTM system. They are created during the provision of an Aircraft and are tied to the UAS ID (expected to be a HHIT, see [[drip-rid](#)] for details).

These structures when chained together can create a root of trust all the way back to the manufacturer itself during the initial production of a given Aircraft. The chain can also be used by authorized entities to trace an Aircraft through all owners and flights in the Aircraft's lifetime (something of interest to ICAO).

The rest of this section will define the formats of proofs in DRIP as forms of certificates and attestations and their common uses.

### [3.1.](#) Certificate: X on X (Cxx Form)

The Cxx Form of DRIP Proofs is a self-signed certificate (by an entity known as 'X') staking an unverified claim on a HHIT/HI pairing until an expiration date/time.

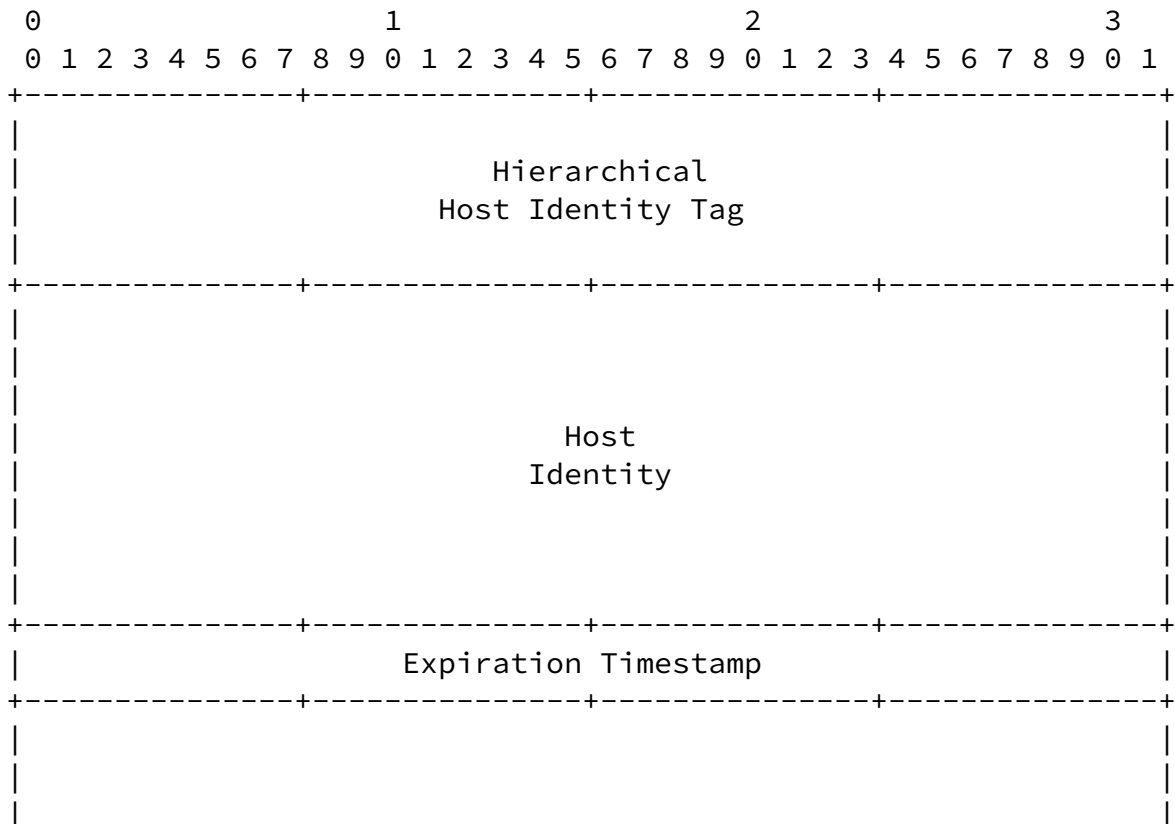




Figure 1: Certificate: X on X

Certificates of the Cxx Form are 116 bytes. The offset of the Expiration Timestamp SHOULD be of significant length (possibly years).

These are 5 (five) Cxx Certificates that can be created in a standard DRIP UAS RID system: Manufacturer on Manufacturer, RAA on RAA, HDA on HDA (Registry on Registry), Operator on Operator, and Aircraft on Aircraft. This is not an exhaustive list as any entity with the DRIP UAS system SHOULD have a Cxx for itself.

### [3.1.1.](#) Certificate: X on X (Short Form)

A smaller version of Certificate: X on X exists where the Host Identity is removed allowing a claim to be made in 84 bytes.

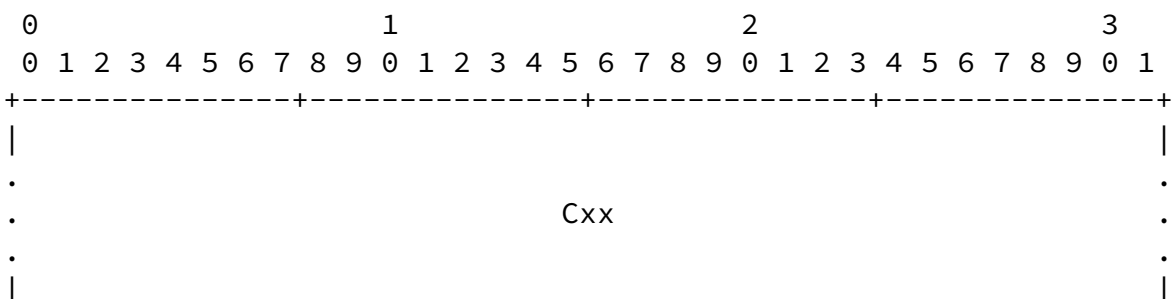




Figure 2: Certificate: X on X (Short Form)

[3.2.](#) Attestation: X on Y (Axy Form)

This DRIP Proof is an attestation where Entity X asserts trust in the binding claimed by Entity Y (in Cyy) and signs this asserting with a timestamp and an expiration of when the binding is no longer asserted by Entity X.



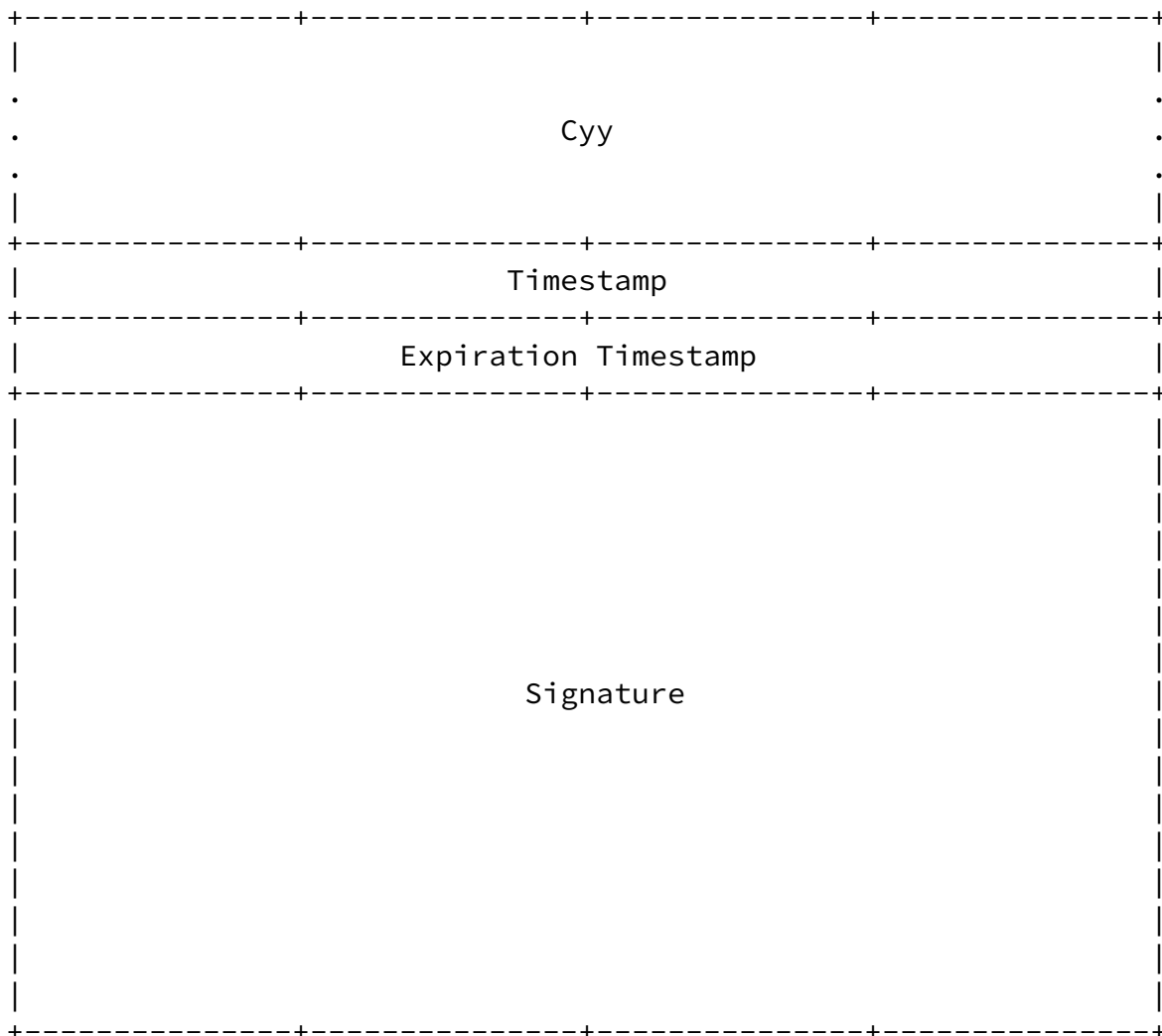


Figure 3: Attestation: X on Y

Axy Form wraps both self-signed certificates of the entities and is signed by Entity X. Two timestamps, one taken at the time of signing and one as an expiration time are used to set boundaries to the assertion. Care should be given to how far into the future the Expiration Timestamp is set, but is left up to system policy.

Most attestations of this form have a length of 304 bytes. Attestation: Registry on Operator on Aircraft is unique in that is 680 bytes long, binding of two Axy forms (in this specific case



Aircraft).

3.2.1. Attestation: X on Y (Short Form)

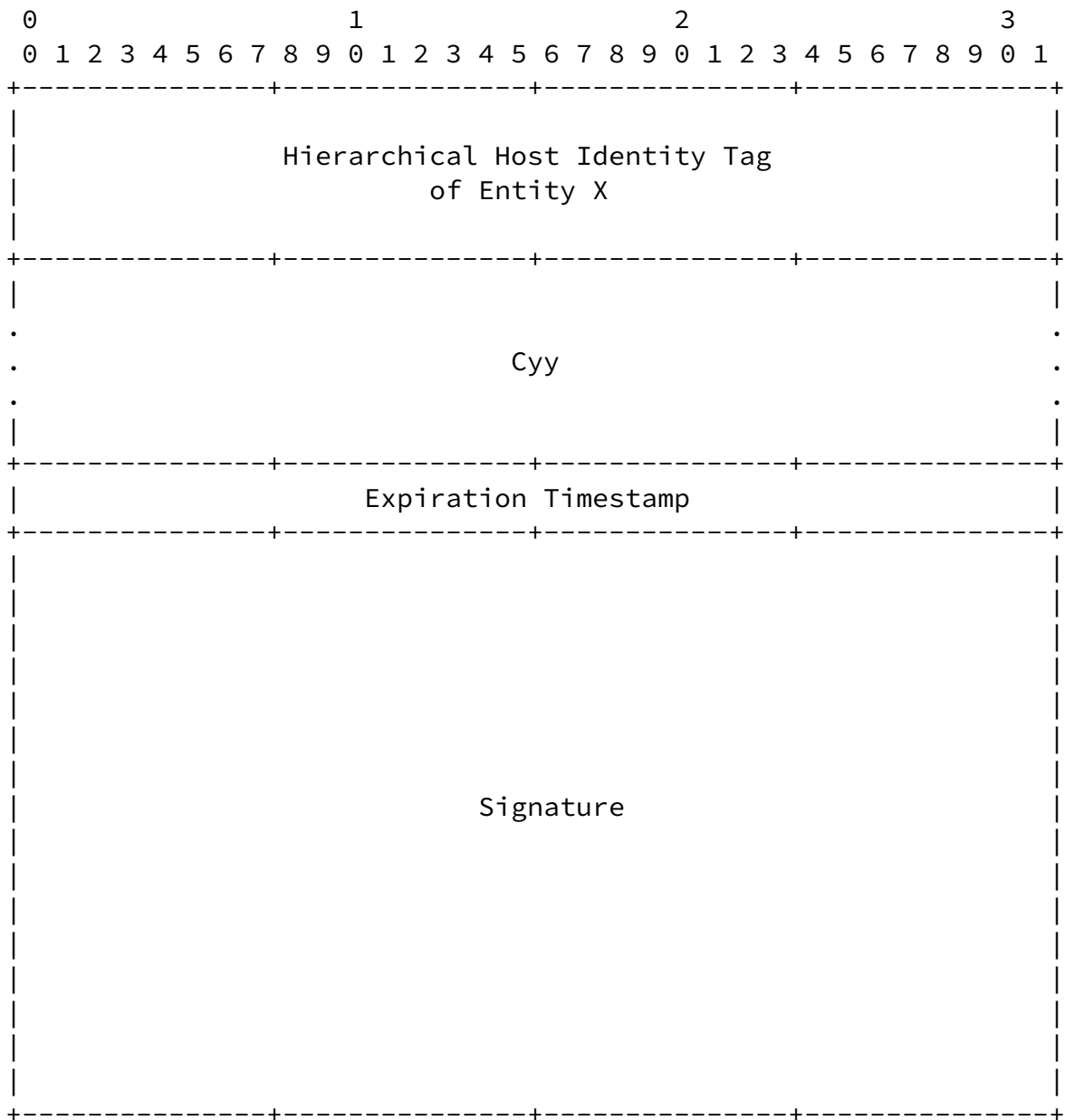


Figure 4: Attestation: X on Y (Short Form)

The short form of the Axy this attestation is 200 bytes long and is designed to fit inside the framing of the ASTM F3411 Authentication Message. The HHIT of Entity X is used in place of the full Cxx (see [Section 5](#) for comments). The timestamp is removed and only an expiration timestamp is present.

During creation the Expiration Timestamp MUST be no later than the Expiration Timestamp found in Cyy.

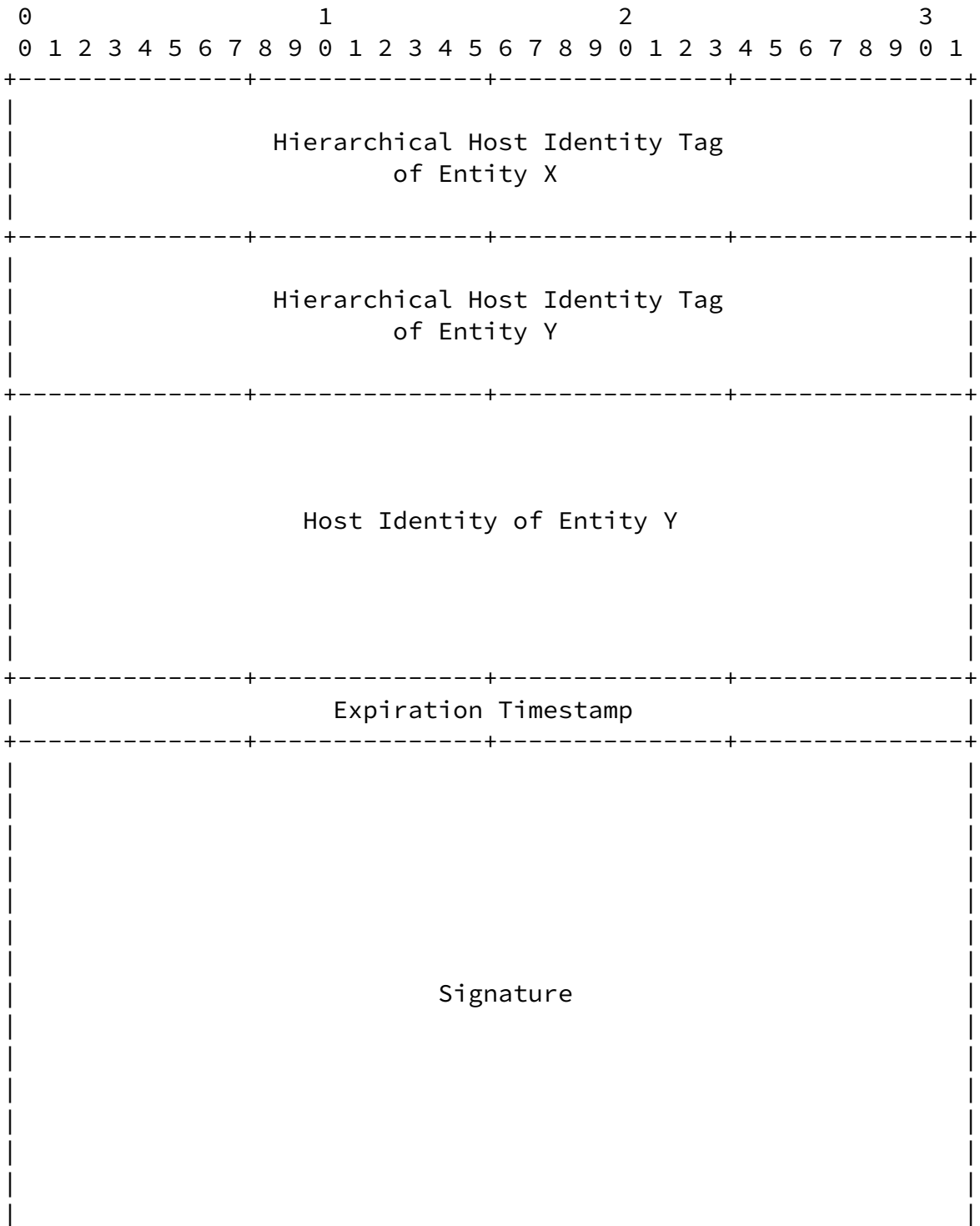
[3.2.2.](#) Attestation: X on Y (Offline Form)

A special attestation that is the basis for a certificate finalized onboard the aircraft during flight. It is used in Broadcast RID to provide the trustworthiness of the Aircraft without the need of the Observer to be connected to the Internet.

Internet-Draft

identity-claims

November 2020



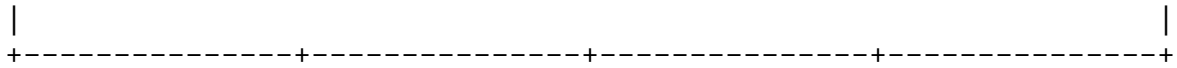


Figure 5: Attestation: X on Y (Offline Form)

The signature is generated using Entity X's keypair.

### [3.3.](#) Timestamps

Timestamps MAY be the standard UNIX time or a protocol specific timestamp, to avoid programming complexities. For example [[F3411-19](#)] uses a 00:00:00 01/01/2019 offset. When a Expiration Timestamp is required a desired offset is added, setting the timestamp into the future. The amount of offset for specific timestamps is left to best practice.

### [3.4.](#) Signatures

Signatures are ALWAYS taken over the preceding fields in the certificate/attestation. For DRIP the EdDSA25519 algorithm from [[RFC8032](#)] is used.

## [4.](#) Provisioning

Under DRIP UAS RID a special provisioning procedure is required to properly generate and distribute the certificates and attestations to all parties in the USS/UTM ecosystem using DRIP RID.

Keypairs are expected to be generated on the device hardware it will be used on. Due to hardware limitations (see [Section 5](#)) and connectivity it is acceptable under DRIP RID to generate keypairs for the Aircraft on Operator devices and later securely inject them into the Aircraft (as defined in [Section 4.5.2](#)). The methods to securely inject and store keypair information in a "secure element" of the Aircraft is out of scope of this document.

### [4.1.](#) HHIT Delegation

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.

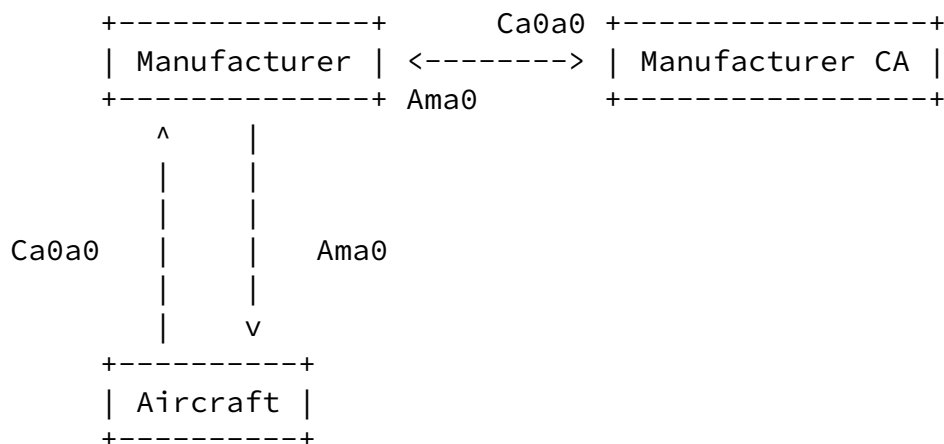
- 1 The entity generates its own HHIT, discovering and using thr RAA and HDA for the target Registry. The method for discovering a Registry's RAA and HDA is out of scope here. This allows for the device to generate an HHIT to send to the Registry to be accepted (thus generating the required Host Identity Claim) or denied.
- 2 The entity sends to the Registry its HI for it to be hashed and result in the HHIT. The Registry would then either accept (returning the HHIT to the device) or deny this pairing.

In either case the Registry must decide on if the HI/HHIT pairing is valid. This in its simplest form is checking the current Registry for a collision on the HHIT.

Upon accepting a HI/HHIT pair the Registry MUST populate the required the DNS serving the HDA with the HIP RR and other relevant RR types (such as TXT and CERT). The Registry MUST also generate the appropriate Host Identity Claim for the given operation.

If the Registry denied the HI/HHIT pair, because there was a HHIT collision or any other reason, the Registry MUST signal back to the device being provisioned that a new HI needs to be generated.

#### [4.2.](#) Manufacturer



During the initial configuration and production at the factory the

Aircraft MUST be configured to have a serial number. ASTM defines this to be an ANSI/CTA-2063A. Under DRIP a HHIT can be encoded as such to be able to convert back and forth between them. This is out of scope for this document.

Under DRIP the Manufacturer SHOULD be using HHITs and have their own keypair and Cxx (Certificate: Manufacturer on Manufacturer). (Ed. Note: some words on aircraft keypair and certs here?).

Certificate: Aircraft 0 on Aircraft 0 (Ca0a0) is extracted by the manufacturer and send to their Certificate Authority (CA) to be verified and added. A resulting certificate (Attestation: Manufacturer on Aircraft 0) SHOULD be a DRIP Attestation in the Axy Form - however this could be a X.509 certificate binding the serial number to the manufacturer.

### [4.3.](#) Registry

TODO

DRIP UAS RID defines two levels of hierarchy maintained by the Registration Assigning Authority (RAA) and HHIT Domain Authority (HDA). The authors anticipate that an RAA is owned and operated by a

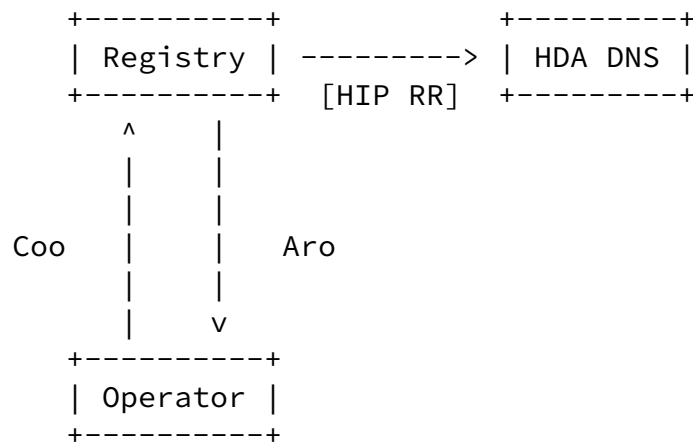
regional CAA (or a delegated party by an CAA in a specific airspace region) with HDAs being contracted out. As such a chain of trust for registries is required to ensure trustworthiness is not compromised. More information on the registries can be found in [[hhit-registries](#)].

Both the RAA and HDA generate their own keypairs and self-signed certificates (Certificate: RAA on RAA and Certificate: HDA on HDA respectively). The HDA sends to the RAA its self-signed certificate to be added into the RAA DNS.

The RAA confirms the certificate received is valid and that no HHIT collisions occur before added a HIP RR to its DNS for the new HDA. An Attestation: RAA on HDA is sent as a confirmation that provisioning was successful.

The HDA is now a valid "Registry" and uses its keypair and Certificate: HDA on HDA with all provisioning requests from downstream.

#### 4.4. Operator



The Operator generates a keypair and HHIT as specified in DRIP UAS RID. A self-signed certificate (Certificate: Operator on Operator) is generated and sent to the desired Registry (HDA). Other relevant information and possibly personally identifiable information needed may also be required to be sent to the Registry (all over a secure channel - the method of which is out of scope for this document).

The Registry cross checks any personally identifiable information as required. Certificate: Operator on Operator is verified (both using the expiration timestamp and signature). The HHIT is searched in the Registries database to confirm that no collision occurs. A new attestation is generated (Attestation: Registry on Operator) and sent securely back to the Operator. Optionally the HHIT/HI pairing can be added to the Registries DNS in to form of a HIP Resource Record (RR).

Other RRs, such as CERT and TXT, may also be used to hold public information.

With the receipt of Attestation: Registry on Operator the provisioning of an Operator is complete.

#### 4.5. Aircraft

##### 4.5.1. Standard Provisioning

Under standard provisioning the Aircraft has its own connectivity to

the Registry, the method which is out of scope for this document.

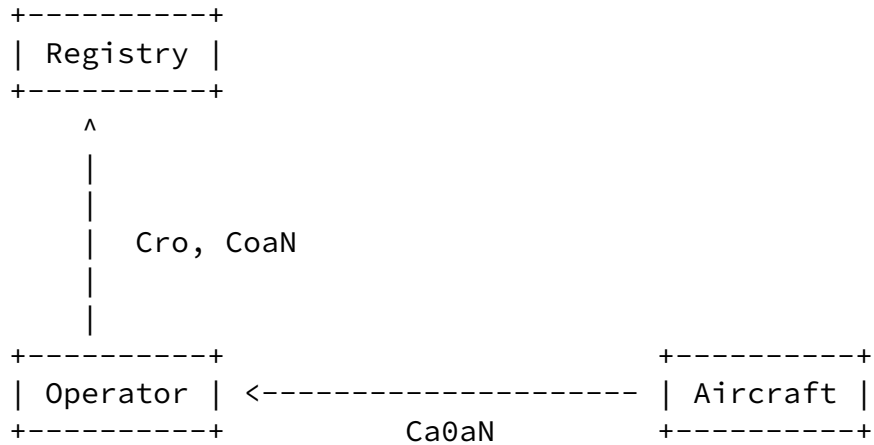


Figure 6: Standard Provision: Step 1

Through mechanisms not specified in this document the Aircraft should have methods to instruct the Aircrafts onboard systems to generate a keypair and certificate. This certificate is chained to the factory provisioned certificate (Certificate: Aircraft 0 on Aircraft 0). This new attestation (Attestation: Aircraft 0 on Aircraft N) is securely extracted by the Operator.

With Attestation: Aircraft 0 on Aircraft N the sub certificate (Certificate: Aircraft N on Aircraft N) is used by the Operator to generate Attestation: Operator on Aircraft N. This along with Attestation: Registry on Operator is sent to the Registry.

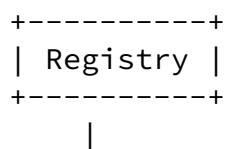






Figure 7: Standard Provision: Step 2

On the Registry, Attestation: Registry on Operator is verified and used as confirmation that the Operator is already registered. Attestation: Operator on Aircraft N also undergoes a validation check and used to generate a token to return to the Operator to continue provisioning.

Upon receipt of this token, the Operator injects it into the Aircraft and its used to form a secure connection to the Registry. The Aircraft then sends Attestation: Manufacturer on Aircraft 0 and Attestation: Aircraft 0 to Aircraft N.

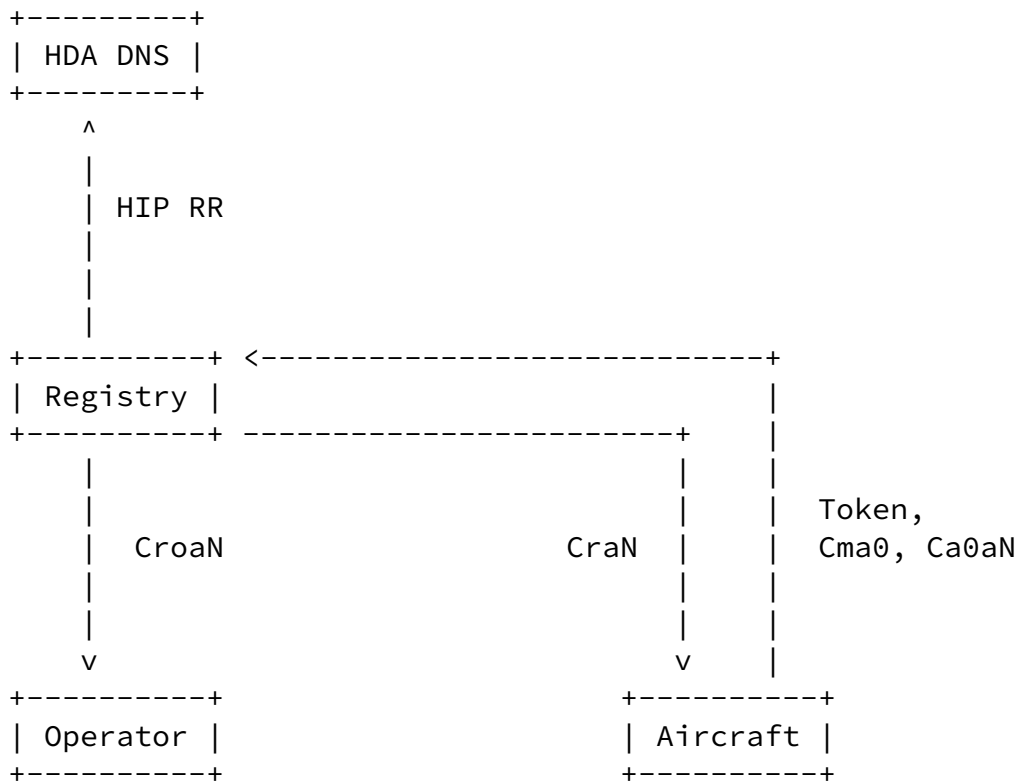


Figure 8: Standard Provision: Step 3

The Registry uses Attestation: Manufacturer on Aircraft 0 (with an external database if supported) to confirm the validity of the Aircraft. Attestation: Aircraft 0 on Aircraft N is correlated with Attestation: Operator on Aircraft N and Attestation: Manufacturer on Aircraft 0 to see the chain of ownership. The new HHIT tied to Aircraft N is then checked for collisions in the HDA. With the information the Registry generates two certificates: Attestation: Registry on Operator on Aircraft N and Attestation: Registry on Aircraft N (Offline Form). A HIP RR (and other RR types as needed) are generated and inserted into the HDA.

Attestation: Registry on Operator on Aircraft N is sent via a secure channel back to the Operator to be stored. Attestation: Registry on Aircraft N (Offline Form) is sent to the Aircraft to be used in Broadcast RID.

4.5.2. Operator Assisted Provisioning

This provisioning scheme is for when the Aircraft is unable to connect to the Registry itself or does not have the hardware required to generate keypairs and certificates.

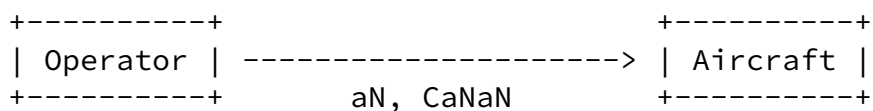
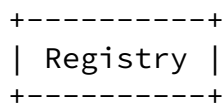


Figure 9: Operator Assisted Provision: Step 1

To start the Operator generates on behalf of the Aircraft a new keypair and Certificate: Aircraft N on Aircraft N. This keypair and certificate are injected into the Aircraft for it to generate Attestation: Aircraft 0 on Aircraft N. After injecting the keypair and certificate, the Operator MUST destroy all copies of the keypair.

Internet-Draft

identity-claims

November 2020

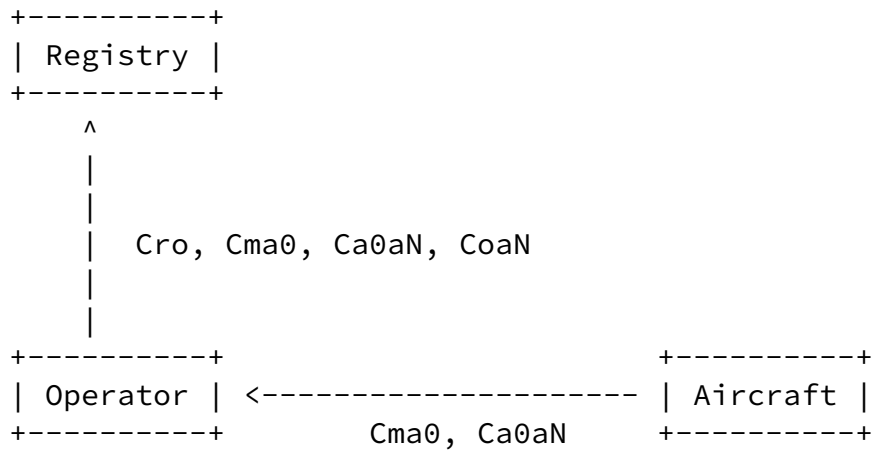


Figure 10: Operator Assisted Provision: Step 2

Attestation: Manufacturer on Aircraft 0 and Attestation: Aircraft 0 on Aircraft N is extracted by the Operator and the following data items are sent to the Registry; Attestation: Registry on Operator, Attestation: Manufacturer on Aircraft 0, Attestation: Aircraft 0 on Aircraft N, Attestation: Operator on Aircraft N.

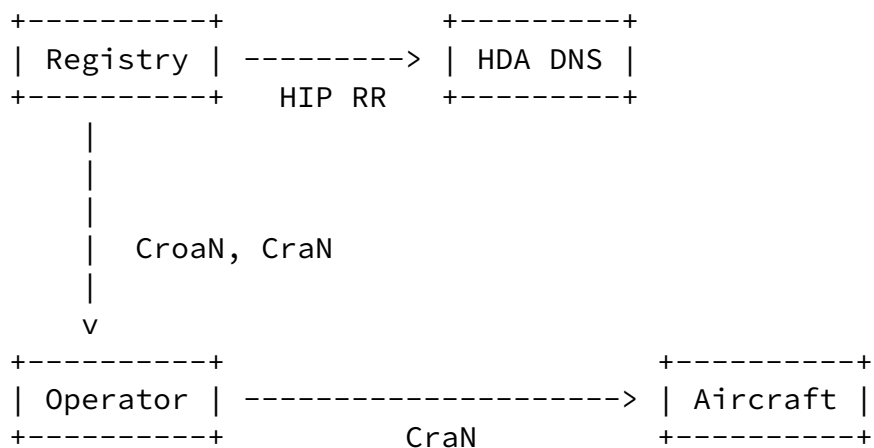


Figure 11: Operator Assisted Provision: Step 3

On the Registry validation checks are done on all attestations as per the previous sections. Once complete then the Registry checks for a HHIT collision, adding to the HDA if clear and generates Attestation: Registry on Operator on Aircraft N and Attestation: Registry on

Aircraft N (Offline Form). Both are sent back to the Operator.

The Operator securely inject Attestation: Registry on Aircraft N (Offline Form) and securely stores Attestation: Registry on Operator on Aircraft N.

### [4.5.3.](#) Initial Provisioning

A special form of provisioning is used when the Aircraft is first sold to an Operator. Instead of generating a new keypair, the built in keypair and certificate done by the Manufacturer is used to provision and register the aircraft to the owner.

For this either Standard or Operator Assisted methods can be used.

## [5.](#) Security Considerations

A major consideration is the optimization done in Attestation: X on Y (Short Form) to get its length down to 200 bytes. The truncation of Certificate: HDA on HDA down to just its HHIT is one that could be used against the system to act as a false Registry. For this to occur an attacker would need to find a hash collision on that Registry HHIT and then manage to spoof all of DNS being used in the system.

The authors believe that the probability of such an attack is low when Registry operators are using best practices in security. If such an attack can occur (especially in the time frame of "one-time use IDs") then there are more serious issues present in the system.

## [6.](#) References

### [6.1.](#) Normative References

[F3411-19] "Standard Specification for Remote ID and Tracking", February 2020.

[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>>.

[RFC8032] Josefsson, S. and I. Liusvaara, "Edwards-Curve Digital Signature Algorithm (EdDSA)", [RFC 8032](#), DOI 10.17487/RFC8032, January 2017, <<https://www.rfc-editor.org/info/rfc8032>>.

[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>>.

## [6.2](#). Informative References

Wiethuechter, et al.

Expires 6 May 2021

[Page 18]

---

Internet-Draft

identity-claims

November 2020

[drip-requirements]

Card, S., Wiethuechter, A., Moskowitz, R., and A. Gurtov, "Drone Remote Identification Protocol (DRIP) Requirements", Work in Progress, Internet-Draft, [draft-ietf-drip-reqs-06](#), 1 November 2020, <<http://www.ietf.org/internet-drafts/draft-ietf-drip-reqs-06.txt>>.

[drip-rid] Moskowitz, R., Card, S., Wiethuechter, A., and A. Gurtov, "UAS Remote ID", Work in Progress, Internet-Draft, [draft-ietf-drip-uas-rid-01](#), 9 September 2020, <<http://www.ietf.org/internet-drafts/draft-ietf-drip-uas-rid-01.txt>>.

[hhit-registries]

Moskowitz, R., Card, S., and A. Wiethuechter, "Hierarchical HIT Registries", Work in Progress, Internet-Draft, [draft-moskowitz-hip-hhit-registries-02](#), 9 March 2020, <<http://www.ietf.org/internet-drafts/draft-moskowitz-hip-hhit-registries-02.txt>>.

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

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

Wiethuechter, et al.

Expires 6 May 2021

[Page 19]

---

Internet-Draft

identity-claims

November 2020

Email: rgm@labs.htt-consult.com

