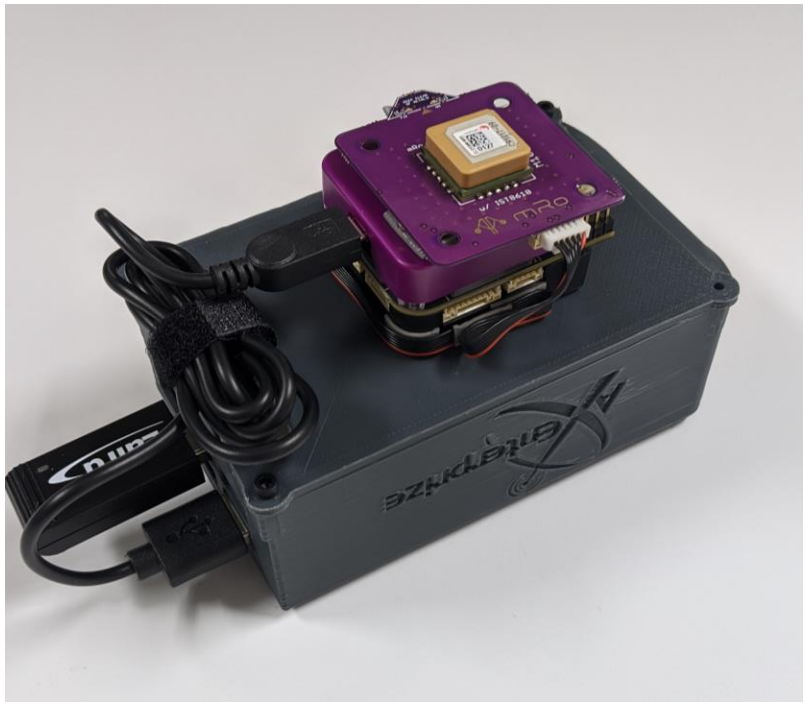
DRIP Authentication Formats

draft-wiethuechter-drip-auth-03

Adam Wiethuechter

DRIP WG – IETF 108; July 30, 2020



We Are Flying DRIP!

New York UAS Test Site (NYUASTS)

- AX Enterprize has been flying and doing testing with Trustworthy Multipurpose Remote ID (TMRID)
  - TMRID: Python3 implementation of DRIP drafts
    - auth-formats-00, identity-claims-00, uas-rid-03
  - Extends AX's Python3 implementation of ASTM F3411-19
- Notable findings:
  - Bluetooth 4 can be detected and decoded up to ~300ft (91m) at 400ft AGL
    - Becomes unreliable around 350ft@400ft AGL away
  - Bluetooth 5 can be detected and decoded to ~1800ft (548m) at 400ft AGL
  - Full Certificate messages are obtained in a wide range from 2.21 seconds to 45 seconds from receipt of first certificate page
    - Still using draft-v00 authentication format, working on updating implementation

From the DRIP Charter

DRIP's goal is to specify how RID can be made trustworthy and available in both Internet and local-only connected scenarios

The DRIP Auth. Solution

- Use the HHIT as the UAS ID
  - See draft-moskowitz-drip-uas for details
- Use the small signature size of EdDSA25519
  - Easily fits in ASTM Authentication Message
    - UA HHIT (16) + Timestamp (4) + Signature (64) = 84 bytes out of 109 bytes
    - 25 bytes left for data to be signed
- Increase Auth. Page limit from 5 to 10
  - We have approached ASTM and they have been receptive to this change
  - Now we have 224 bytes!
- Add Forward Error Correction to help loss of pages in Bluetooth 4.X
- Send short Certificate via Authentication Message making RID trustworthy in local-only scenarios

Authentication Formats

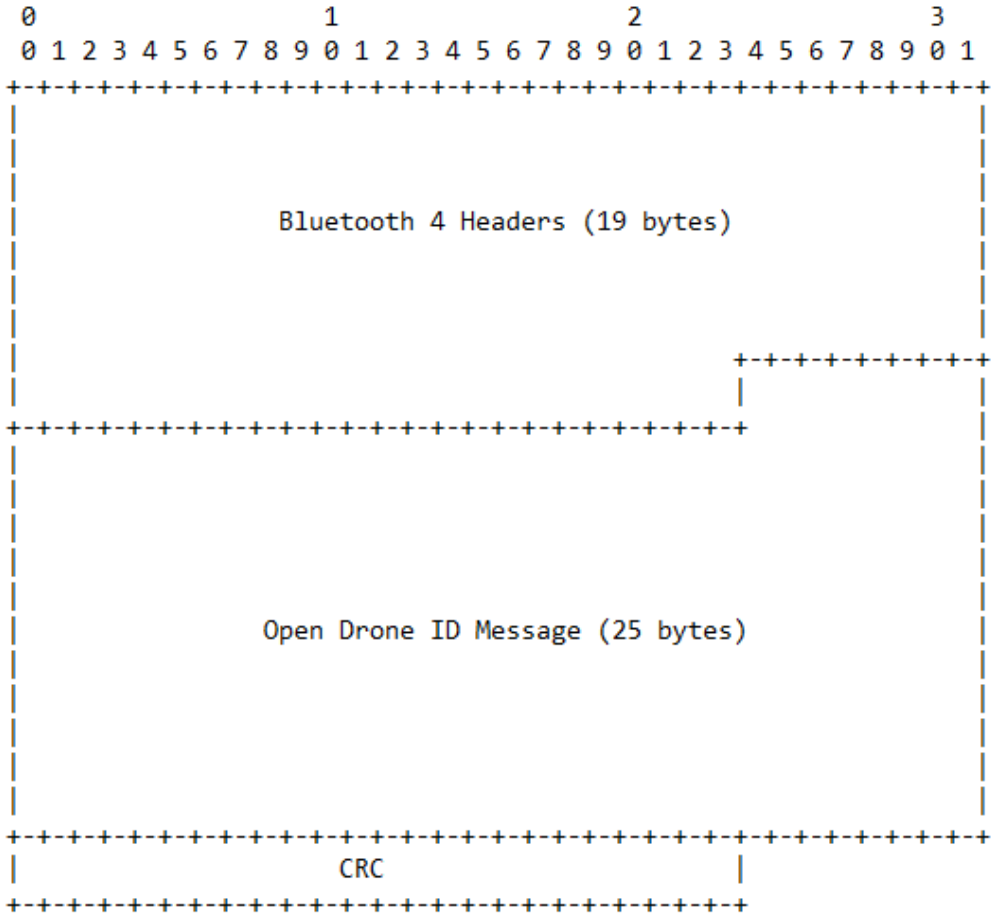
Background and Updates

Background & Problem

- ASTM F3411-19 Broadcast RID
  - Disjointed information delivery
    - Identity information of UA sent in Basic ID
    - Position information of UA sent in Location
      - But no ID in the Location Message
    - Authentication information of UA sent in Auth
    - All of these are sent and received separately (under Bluetooth 4.X)!
  - Fragmented data across Authentication Message pages
- Overall a lack of trust in Broadcast messages
  - Especially in Bluetooth 4.X

Bluetooth Background

- Why so small?
  - Bluetooth 4 legacy frames only give 25 bytes to play with (after Bluetooth headers)
  - 1 byte is for a main header in ASTM format that is always present – now only 24 bytes of data to work with per frame/page



ASTM Authentication

- ASTM F3411-19 “Standard Specification for Remote ID and Tracking”
- Authentication Message
  - 5 pages long with a 109 byte max payload (17 + 23 * 4)
  - Designed to authenticate Message Packs (of up to 5 messages in Bluetooth 5.X frame)

High level draft changes since V00

- Lots of typos
  - Confidence in spelling phonetically != Actually spelling of words correctly
- New format
  - Single ASTM AuthType (0xD selected from Private range; needs allocation into Reserved range from ASTM)
  - Cleaner framing design
    - General and Wrapped – more on this later
  - New DRIP Header
    - Modified shortly after v02 went in
- Addresses DRIP Requirements GEN1, GEN2 and GEN3
  - Certificates address GEN1 and GEN3
    - Provable Ownership and Provable Registration
  - Other DRIP AuthTypes address GEN2
    - Provable Binding

DRIP Framing Structures

General Frame, Wrapper Frame

General Frame

- DRIP Header
  - 1 bit to signal FEC
  - 7 bits for DRIP AuthTypes
- Reed Solomon FEC always fills last page
  - Taken over all pages of Auth. Message
  - FEC is SHOULD on Bluetooth 4, SHOULD NOT on Bluetooth 5
    - See Backup Slides for details
- 223 bytes of data w/o FEC
- 200 bytes of data w/FEC

DRIP Header Details

- Independent FEC flag
  - Previously was tied to auth. type being sent
  - Each DRIP AuthType specifies SHOULD/SHOULD NOT use of FEC
- 128 possible DRIP AuthTypes
  - 9 total currently defined
- 7 bit space broken into 5 areas
  - Half (8) of Wrapped Messages defined
  - One (1) Certificate defined
- Question to WG:
  - Is this the best way to carve up this single byte?

```
DRIP Header (1 byte):
      7     6     5     4     3     2     1     0
   +-----+-----+-----+-----+-----+-----+-----+-----+
   | FEC |              DRIP AuthType              |
   +-----+-----+-----+-----+-----+-----+-----+-----+

   FEC (1 bit):
       Enabled [1] or Disabled [0]. Signals if Page N is
       filled with Reed Solomon FEC.

   DRIP AuthType (7 bits):
       DRIP AuthType                          Values
       -------------                          ------
       0 Wrapped ASTM Message(s)              0
       1 Wrapped ASTM Message(s)              1
       2 Wrapped ASTM Message(s)              2
       3 Wrapped ASTM Message(s)              3
       4 Wrapped ASTM Message(s)              4
       5 Wrapped ASTM Message(s)              5
       8 Byte Manifest                        6
       4 Byte Manifest                        7
       Reserved (Wrapped Messages)            8-15
       Certificate: Registry on Aircraft      16
       Reserved (Certificates)                17-31
       Private Use                            32-63
       Reserved                               64-111
       Experimental Use                       112-127
```

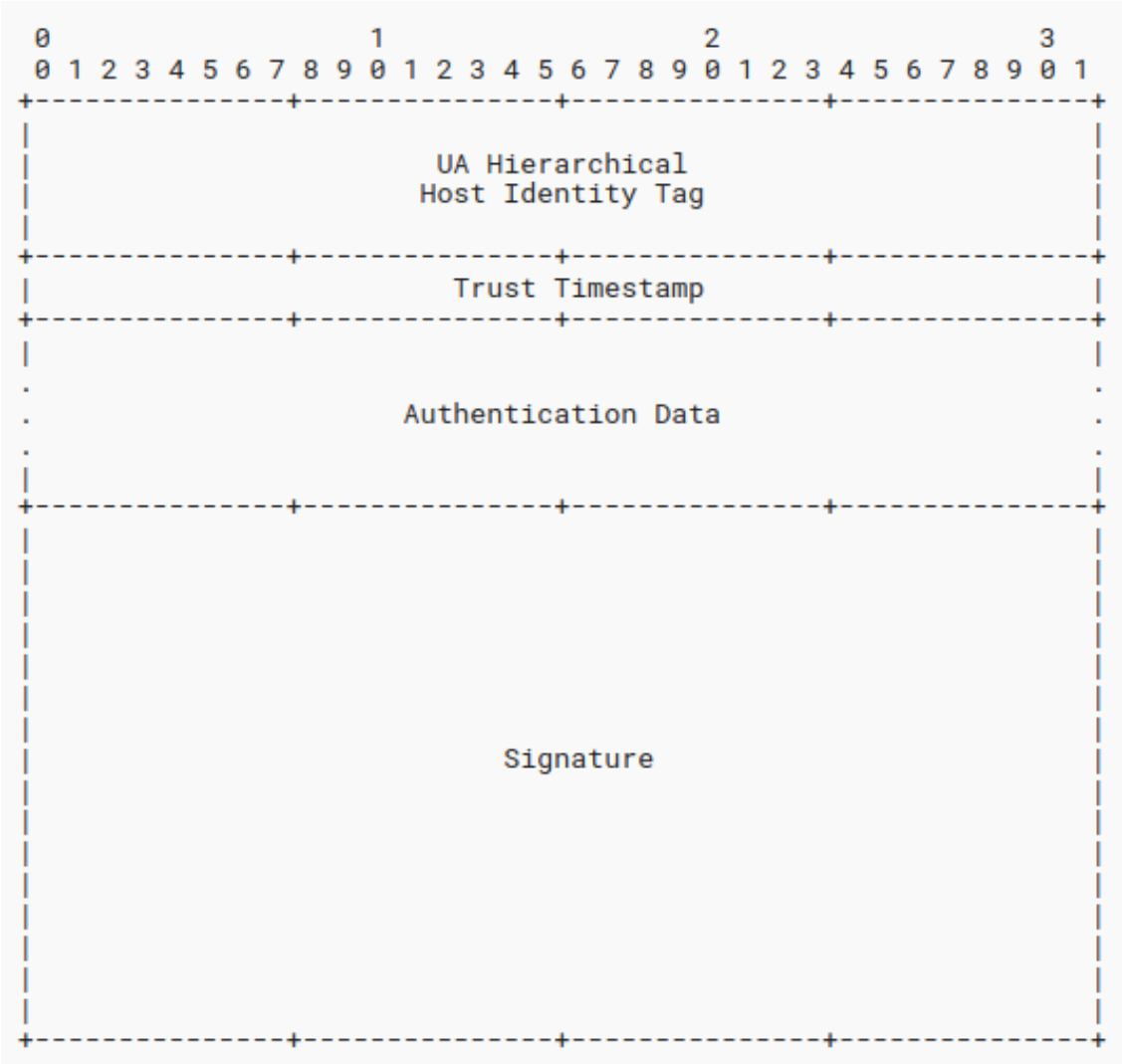
```
000 xxxx (0x00-0x0F): Wrapped Messages (16)
001 xxxx (0x10-0x1F): Certificates (16)
01x xxxx (0x20-0x3F): Private Use (32)
1xx xxxx (0x40-0x6F): Reserved (48)
111 xxxx (0x70-0x7F): Experimental Use (16)
```

Reed Solomon FEC

- Bluetooth (both 4 and 5) have a 3 byte CRC in every frame
  - Full frame is dropped if CRC check fails within Bluetooth stack
  - No signal to upper layers that a frame is being dropped
- To RID applications, we missed a full Authentication page (under Bluetooth 4)
  - Pages are numbered so we know which pages are missing in a set (sets are defined using the AD Counter)
- Reed Solomon can correct 23 bytes of error when we know positions of data lost – which we do!
  - So if we rebuild frames filling in known header bytes (Message Type, ASTM Version, Authentication Type and Page Number) we can correct for 23 bytes which is missing page data
- For Bluetooth 4, FEC gives us an advantage of recovery if any single page is lost in transmission
  - If any more are lost recovery is impossible but if that happens probably more issues going on anyways
- For Bluetooth 5, FEC is useless as it already has FEC at the frame level before CRC check
  - Only with LE Coded PHY, which is what is specified by ASTM
- Also for Bluetooth 5, FEC is useless as per ASTM the Message Pack must be used
  - This uses the 255 byte extended Bluetooth 5 payload to fit multiple ASTM Messages in single frame
  - So if we lose a Bluetooth 5 frame we are already losing anyways as full Authentication Message was together, not physically paged like Bluetooth 4

Wrapper Frame

- Fits inside General Frames DRIP Auth. Data
- Authentication Data
  - 116 bytes with FEC
  - 139 bytes w/o FEC
- Signature computed over all preceding data fields in Wrapper Frame
  - Avoid DRIP Header can change (FEC bit) after signing



[Trust] Timestamp Details

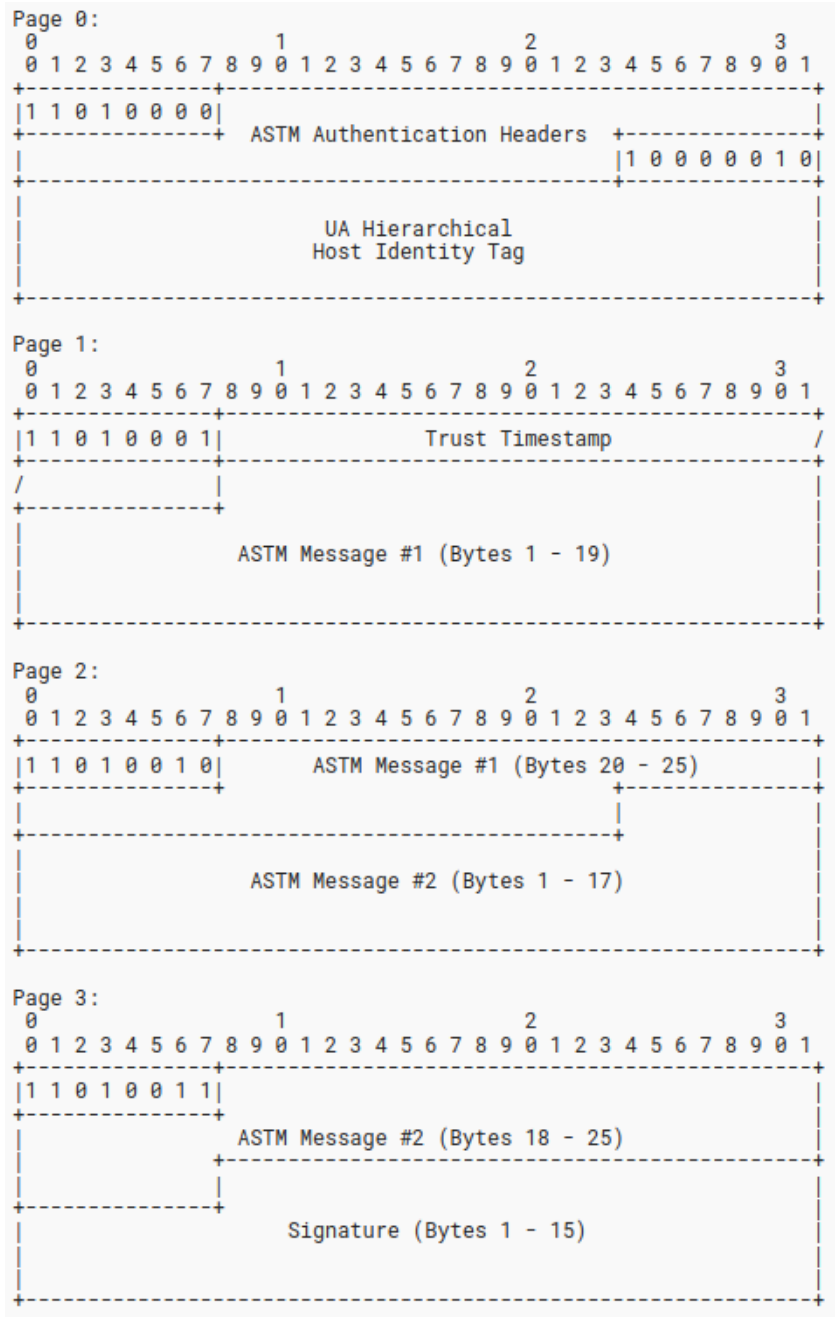
- Different types of timestamp in ecosystem:
  - ASTM Authentication Message style [4 bytes]
    - Offset from 01/01/2019 00:00:00
    - Defined encoding and decoding by ASTM to/from UNIX time
    - Used for DRIP Trust Timestamp in Wrapper Frame
  - UNIX style [4 bytes]
    - Raw UNIX style timestamp
    - Used in DRIP Certificates
  - UTM style (X.509 Validity --> ASN.1)
- Question to WG:
  - What should DRIP adopt for timestamps?

Bluetooth 4.X Auth. Formats

Wrapped ASTM Message(s), Certificate, Manifest(s)

1-5 Wrapped ASTM Message(s)

- DRIP AuthTypes 1-5
  - AuthType signals number of messages being wrapped
- Wrapper Frame Auth. Data filled with ASTM Messages
  - Messages must be in Message Type order
- Special Case: 5 Wrapped Messages
  - Acts as a pseudo-ASTM Message Pack (Type 0xF) over Bluetooth 4
  - FEC MUST be disabled to fit all messages
  - Can fit all ASTM Messages excluding an Auth. Message

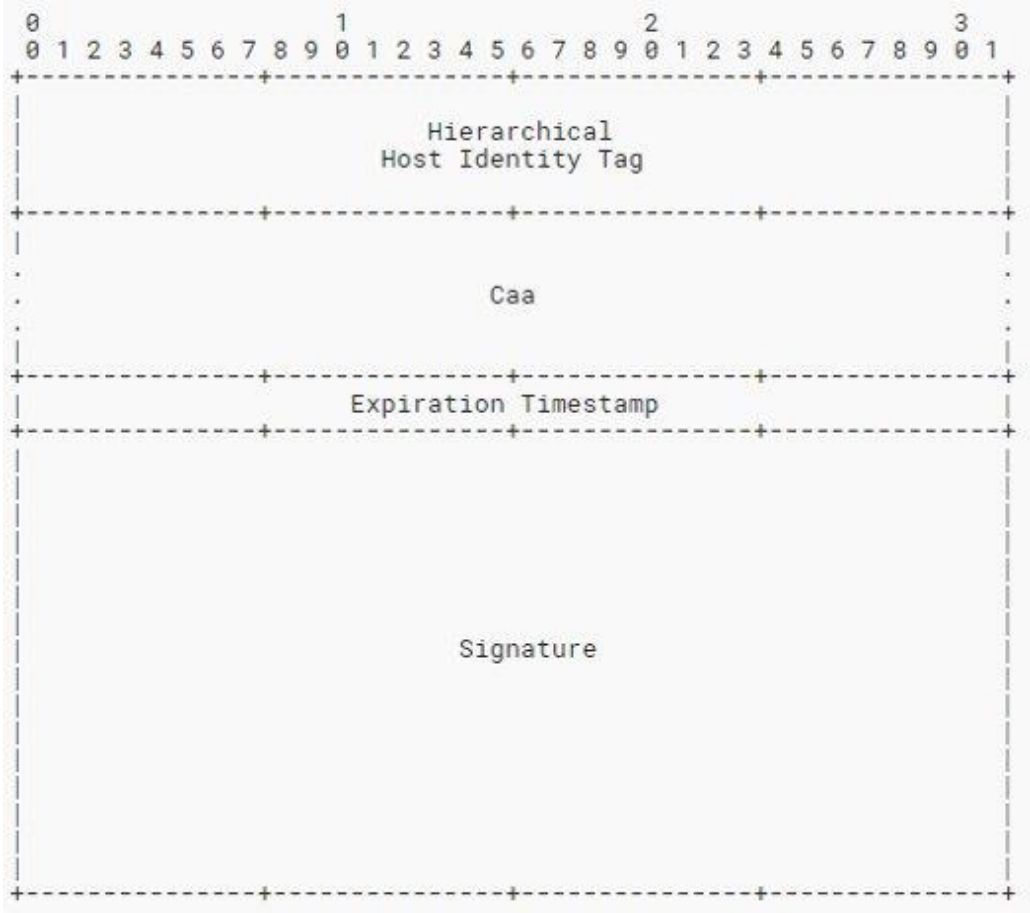

Page 0:
0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
1 1 0 1 0 0 0 0
ASTM Authentication Headers
1 0 0 0 0 0 1 0
UA Hierarchical
Host Identity Tag
Page 1:
0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
1 1 0 1 0 0 0 1
Trust Timestamp
ASTM Message #1 (Bytes 1 - 19)
Page 2:
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
1 1 0 1 0 0 1 0
ASTM Message #1 (Bytes 20 - 25)
ASTM Message #2 (Bytes 1 - 17)
Page 3:
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
1 1 0 1 0 0 1 1
ASTM Message #2 (Bytes 18 - 25)
Signature (Bytes 1 - 15)
Page 4:
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
1 1 0 1 0 1 0 0
Signature (Bytes 16 - 38)
Page 5:
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
1 1 0 1 0 1 0 1
Signature (Bytes 38 - 61)
Page 6:
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
1 1 0 1 0 1 1 0
Signature (Bytes 62 - 64)
Null Padding
Page 7:
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
1 1 0 1 0 1 1 1
Forward Error Correction

Manifests

- DRIP AuthTypes 6, 7
- Wrapper Frame Auth. Data filled with hashes
  - Hashes are of previous non-paged messages sent
- Two special hashes for pseudo-blockchain
  - Links manifests together
  - Hash of previous manifest
  - Hash of current manifest
    - Order of operations?
- Two variants based on hash length; 8 bytes and 4 bytes
  - 27 hashes with 4 bytes, 12 hashes with 8 bytes
  - Uses same hash algorithm as HHIT (in UAS RID this is cSHAKE128)
    - Can use OGA ID of HHIT to signal different hashing methods

Certificate: Registry on Aircraft (Cra)

- DRIP AuthType 16
- General Frame DRIP Auth. Data filled with Cra
  - Exactly 200 bytes in length
  - Binding between entities, asserting trust
  - Contains HI of UA; instant verification of UA
  - Registry HHIT used for lookup on local cached Registry list
    - On Observer device, only ones trusted by User
- See draft-wiethuechter-drip-identity-claims for details



Bluetooth 5.X Auth. Formats

0 Wrapped ASTM Message(s), Certificate

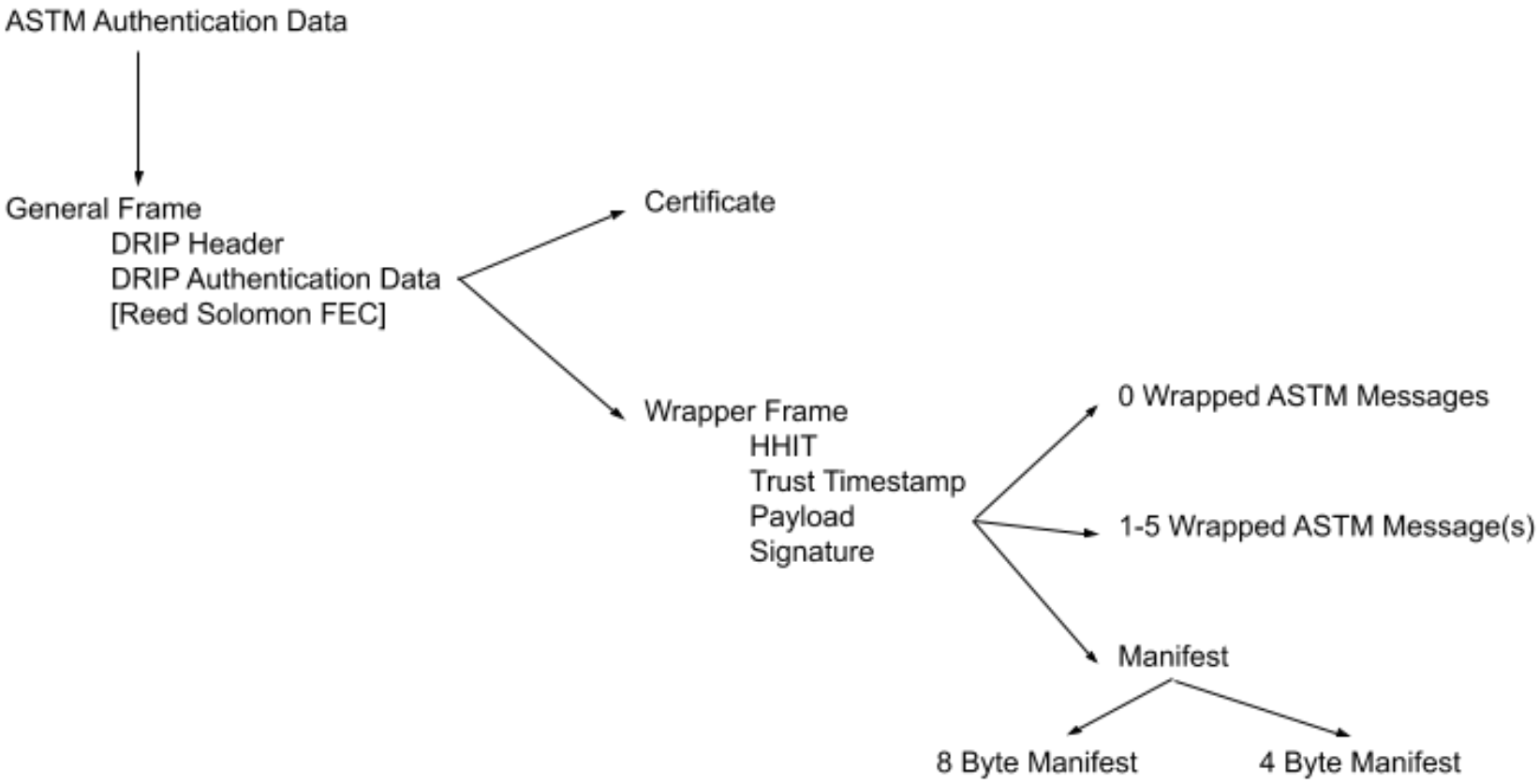
Certificate: Registry on Aircraft (Cra)

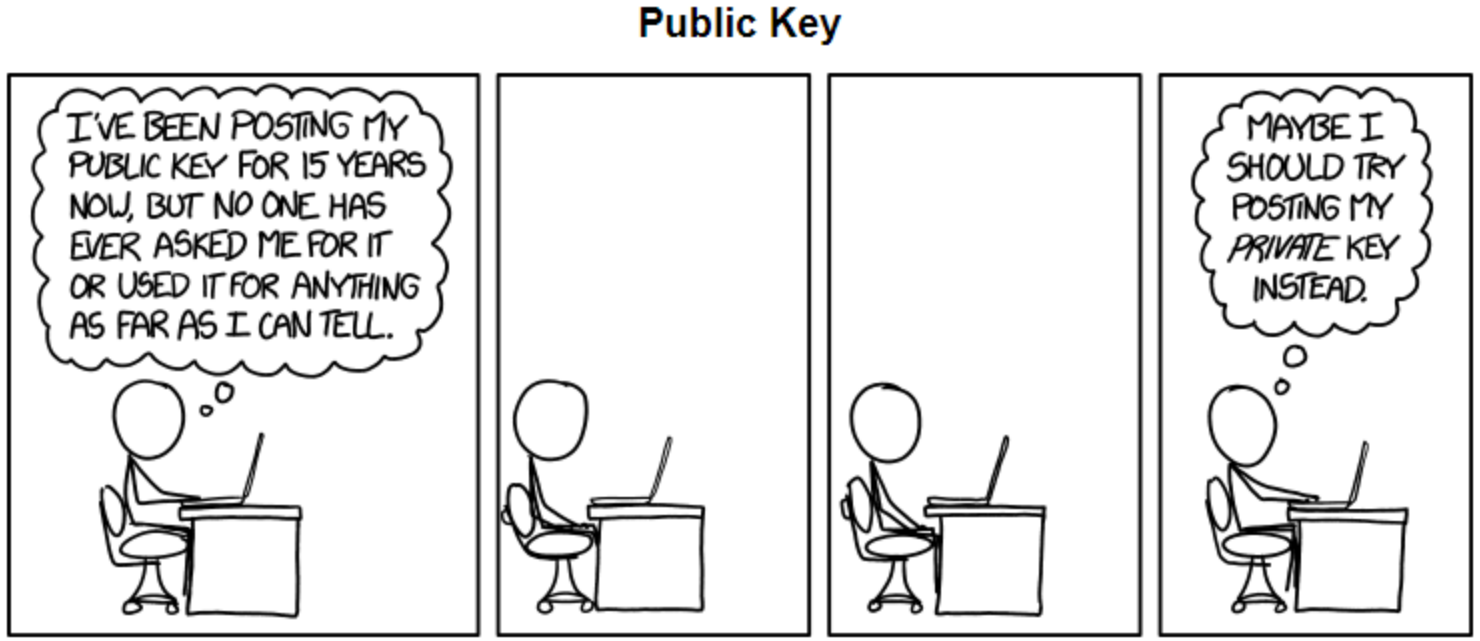
- DRIP AuthType 16
- General Frame DRIP Auth. Data filled with Cra
  - See draft-wiethuechter-drip-identity-claims
- Last 25 bytes of Message Pack can be filled with another ASTM Message
  - Suggested to use Location Message

0 Wrapped ASTM Message(s)

- DRIP AuthType 0
- Special case of Wrapped ASTM Message(s) format
  - Only used for Message Pack under Bluetooth 5.X
- Wrapper Frame Auth. Data virtually filled with ASTM Messages in Message Pack
  - Messages must be in Message Type order
- Discussion for WG
  - Perhaps a better title?

DRIP AuthType Tree





https://xkcd.com/1553/

Discussion

Questions, Comments, Concerns?