# PASSporT Extensions

STIR Virtual Interim

June 2017

# draft-peterson-passport-divert-01

- A feature many people have asked about
  - How do we handle **retargeting**?
  - To header field of SIP is signed by PASSporT
    - Original value may be lost with retargeting
- We define a special Identity header track it
  - With its own “ppt” – “**div**” for “divert”
- Different from History-Info and Diversion?
  - Yes, as it is signed by the original destination domain
  - Moreover, it only captures “major” changes
    - Thanks to our canonicalization procedures
- Useful for things like **SIPBRANDY** where integrity protection for retargeting matters

# Inverting the signer

- *A diverting auth service takes an existing PASSporT, moves the "dest" to "div," and populates "dest" with the new target*
- An Identity header with "div" always points to some prior Identity header
  - Though that header may in turn contain a div…
  - Chains back to an original assertion
- Instead of signing for the "orig" value, the auth service for "div" signs the "dest"
  - So relying parties get a direct cryptographic attestation that the original destination domain authorized the new target

# Original vs. Divert Passport

Header:

```
{ "typ":”passport",
  "alg":”ES256“,
    "x5u":"https://www.example.com/cert.pkx" }
```

Original PASSporT

Claims:

```
{ ”orig":{“uri”:”alice@example.com”},
  ”dest":{“uri”:”firsttarget@example.com”}, <- original target
    "iat": 1443208345 }
```

Header:

```
{ "typ":”passport",
  "alg":”ES256“,
  "ppt":”div“,
  "x5u":"https://www.example.com/cert.pkx" }
```

Added when retargeting

Claims:

```
{ ”orig":{“uri”:”alice@example.com”},
  ”dest":{“uri”:”secondtarget@example.com”},  <- new target
  "iat": 1443208345,
  “div”:{“uri”:”firsttarget@example.com”} }  <- original target
```

# Issues

- Do we need a reason?
  - That is, a cause for the retargeting to be recorded
  - Any actual security value for the threat model?
- Has some interesting interactions with out of band
  - Ideally, this should work with out of band, but…
  - We can talk about that later

# Next Steps

- Adopt?
- I keep hearing people need this
- It’s pretty straightforward, this seems relatively baked

# draft-peterson-stir-cnam-02

- Adds a “cna” array to PASSporT
  - Baseline include a “nam” key-value pair containing a display-name
- But the “cna” element is richer than Caller-ID
  - Scope: anything rendered to the called user to help them decide to pick up the phone or not - extensible
  - Could include information about organizations
    - Government, bank, etc.
    - Maybe some fields in Henning’s Caller-Info parameters
  - Location, potentially
    - Likely by reference rather than by value
  - Other rich data associated with the originating persona
    - Social network data, crowdsourced reputation, and so on
    - Creates an IANA registry allowing allocation of more related elements

# First and Third

- Operates in two modes
- Without "**ppt**"
  - This signifies that an originating authentication service provides the caller name
    - Same entity that signs for the originating number
- With "**ppt**"
  - This signifies that a third party provides the assertion
    - *Different* entity than signs for the originating number
      - Signature can come from someone that doesn't own the TN
      - Instead the "iss" field identifies who generated it
    - Different Identity header field as well

# “cna” without “ppt”

Header:

```
{ "typ":”passport",
  "alg":”ES256“,
      "x5u":"https://www.example.com/cert.pkx" }
```

Claims:

```
{ ”orig":{“tn”:"12155551212”},
  ”dest":{“tn”:"12155551213”},
      "iat": 1443208345,
  “cna”:{“nam:”Alice Atlanta”} }
```

# “cna” with “ppt”

Header:

```
{ "typ":”passport",
  "alg":”ES256“,
  "ppt":”cna“,
  "x5u":"https://www.example.org/cert.pkx" }
```

Third Party Signer

Claims:

```
{ ”orig":{“tn”:"12155551212”},
  ”dest":{“tn”:"12155551213”},
     "iat": 1443208345,
  “cna”:{“nam:”Alice Atlanta”} }
```

# Issues

- Richer information can be more personal
  - Privacy issues with carrying a “cna” payload
  - Confidentiality required for these PASSporTs?
- What is the interface for third-person “cna”?
  - Out of band?
  - There are some interactions with OOB here…
- Need to make sure information propagates down to end user devices…

# Next Steps

- Adopt?
- Figure out what other elements we hope to cover

# draft-rescorla-fallback-02

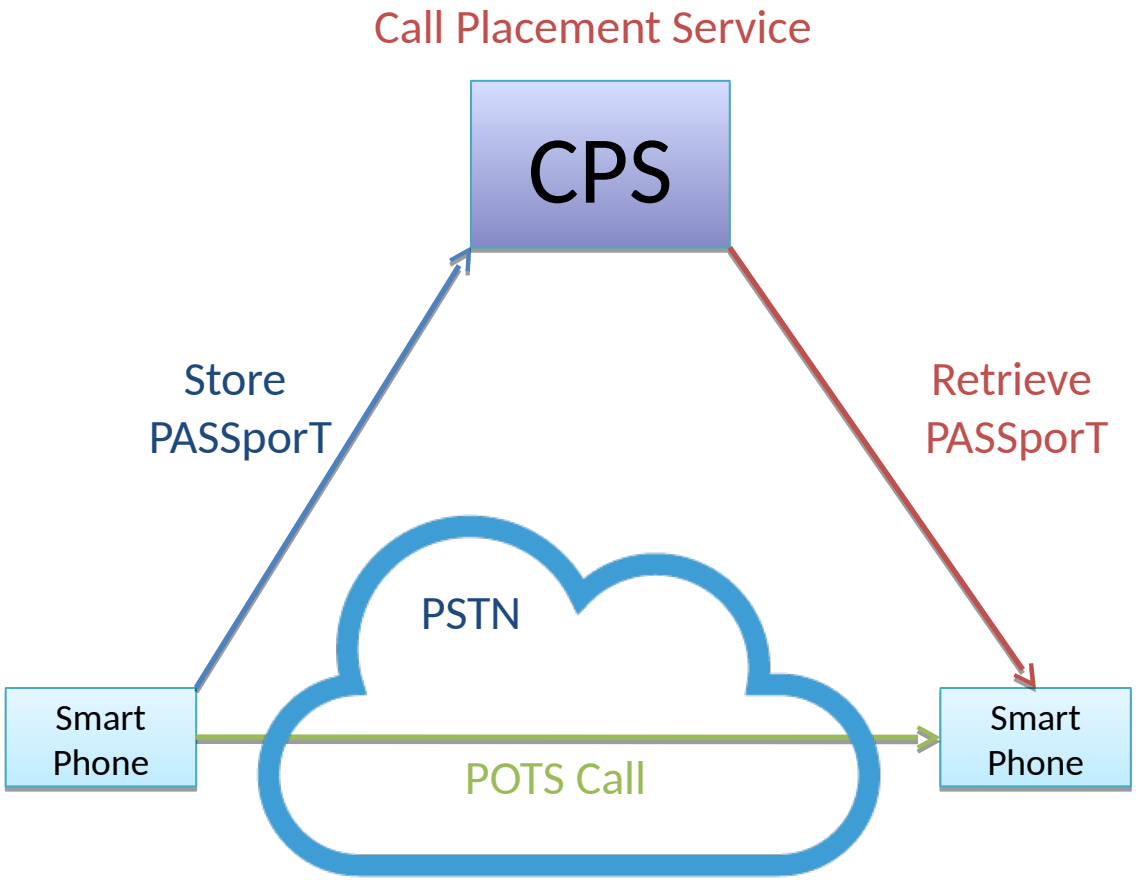
STIR Virtual Interim

June 2017

# Limits of RFC4474bis

- It's in-band – end-to-end IP-IP
  - At best, it addresses the SIP-to-SIP use case
  - Not going to help with SIP-to-PSTN, PSTN-to-PSTN
    - Import for transitional adoption, legacy networks, enterprises, etc.
  - We did in-band first because existing deployments need it
    - Like the IPNNI, now the SHAKEN profile
- Even some IP-IP deployments may not pass Identity e2e
  - Difficult to anticipate what will survive administrative boundaries
    - You can understand "boundaries" pretty broadly
  - And some existing deployments might just block Identity
    - As they block all new headers; especially B2BUAs

# Basic STIR Out of Band



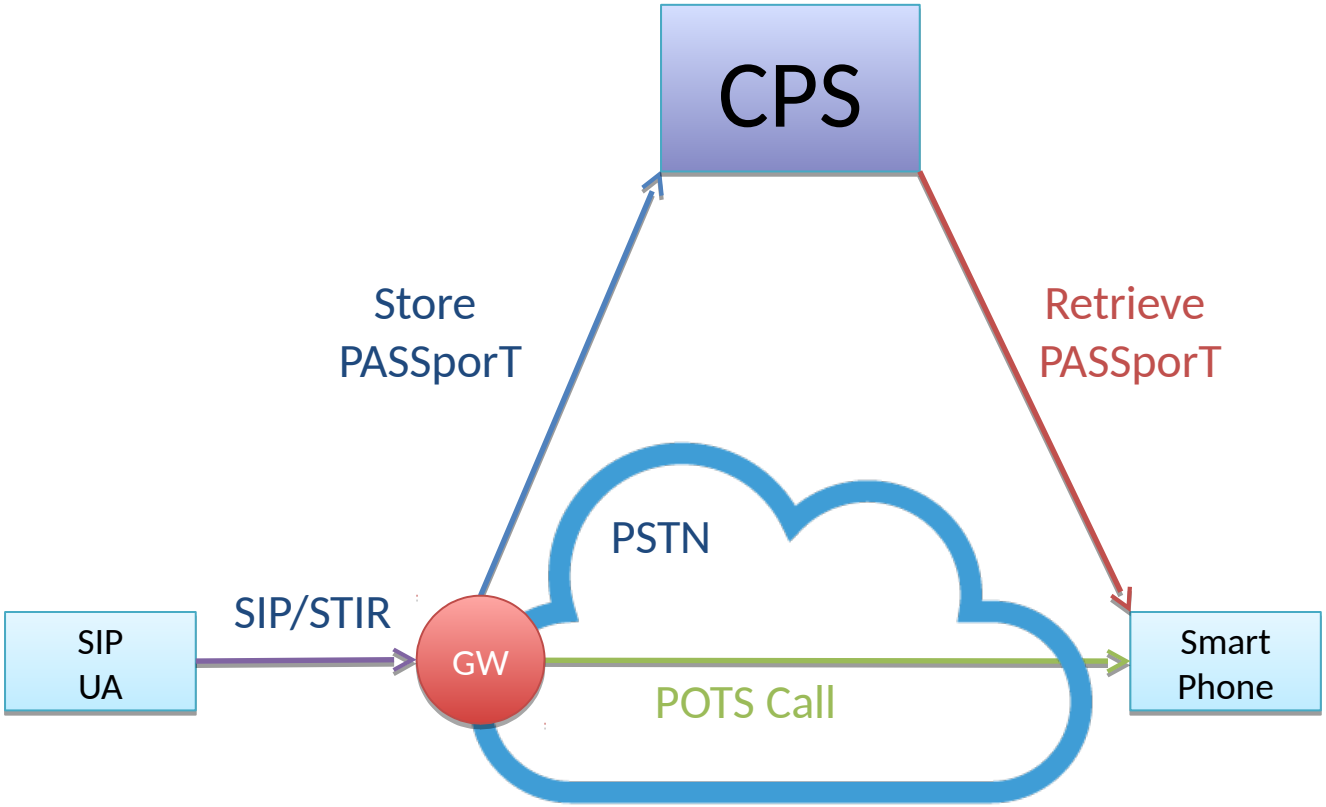

Smart Phones are not just mobile phones, and not just end-user devices

# Obvious Questions

- Okay, how does the originating side know where to find a CPS?
  - And how do we make sure the terminating side comes to exactly the same conclusion?
    - Need a service discovery mechanism
    - A few initial ideas in the draft now – not the focus today
- How do we make sure the right parties store and retrieve PASSporTs from a CPS?
  - Mostly, to manage the risk that someone other than the called party will fetch them?
    - Significant privacy concerns

- These are the things its time to work on

# Who Gets to Store PASSporTs?

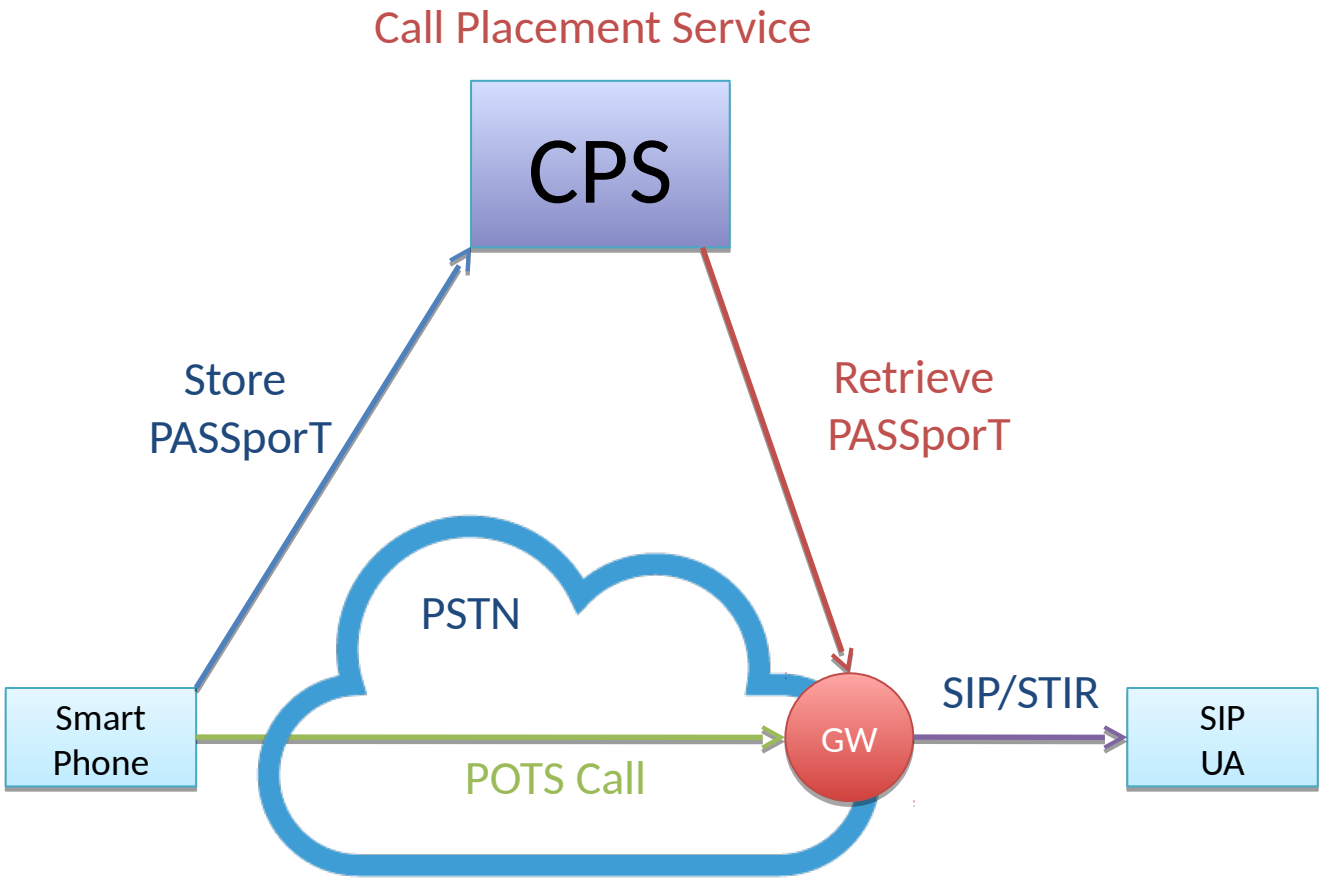



How to authorize a gateway to store it?

# Anyone with a valid PASSporT?

- Assume we have STIR credentials
  - Not necessarily TN credentials, works for SPC too
- PASSporTs are signed, so it almost doesn't matter who stores them
  - Almost – need some kind of DDoS protection from attackers storing millions
- The authority to store is really invested in the PASSporT itself
  - The signature authorizes storage, basically
  - Multiple entities may be authorized to sign for the same "orig" in PASSporT
- Relying parties trust a PASSporT based on its signature, not based on the CPS they got it from
  - At a high level, a CPS can also act as a verification service and only store it if it is valid
  - Maybe don't allow identical PASSporT copies at the CPS to prevent DDoS
- Ultimately, a GW could be authorized to store it
  - Should a GW need any pre-association with the CPS?

# Consider the Following




How to authorize an intermediary to retrieve, if it doesn't have a STIR credential?

# Retrieving What?

- Authorizing retrieval is harder than storage
- What question does the retrieval side ask of the CPS? Three potential semantics:
  - (a) “Give me PASSporTs for the calling number”
  - (b) “Give me PASSporTs for the called number (me?)”
  - (c) “Give me PASSporTs for with both (a) and (b)”
- Those three options have different security implications
  - For case (b), can require a STIR credential
    - (b) however has some complications in call forwarding cases (divert?)
  - How to authorize for case (a)?
    - This is where there are serious privacy risks
    - Effectively, require a STIR credential for the called number, so this ends up with semantics very similar to (c)
      - Right now, that’s the best idea in the draft

# Encrypting PASSporTs

- Encrypting PASSporTs is promising
  - Hides data from a nosy CPS (a likely PERPASS target)
  - Makes retrieval less perilous
    - Need to decrypt PASSporTs to get any value from retrieval
    - Provided of course CPSs always give back an encrypted blob when a retrieval request is made, even when there are no PASSporTs
- But there are costs
  - CPS can no longer validate PASSporTs, so authentication for storage is required
    - Maybe it should be required anyway; belt & suspenders
  - Much harder to manage call forwarding cases
    - Divert requires linking PASSporTs in a way that might be hard to retrieve if things are encrypted blobs
- Will never really deprive the CPS of metadata
  - CPS still needs to know enough about the call that it can field retrieval requests
  - No good story yet about hashing the metadata in a way that the storer and retriever understand, but the CPS can't

# Next Steps

- Already on the charter, targeting WG item adoption
- To Do
  - Need to describe the storage/retrieval protocol
    - Pro tip: it's HTTP
  - Need to specify an OOB authentication and verification service procedure
    - Varies from RFC4474bis because that text is based on comparison to SIP fields
  - Need more on interaction with divert