September 2020

MECHANISM FOR DUAL LOOKUP IN SECURE TELEPHONY IDENTITY REVISITED (STIR)

Kaustubh Inamdar

Gonzalo Salgueiro

MECHANISM FOR DUAL LOOKUP IN SECURE TELEPHONY IDENTITY REVISITED (STIR)

AUTHORS:
Kaustubh Inamdar
Gonzalo Salgueiro

ABSTRACT

Secure Telephony Identity Revisited (STIR) is a framework that enables the cryptographic assertion and verification of an identity of a caller. Through an out-of-band (OOB) mechanism, an entity that cryptographically asserts caller identity places a Personal Assertion Token (PASSporT) in a Call Placement Service (CPS). To verify the identity of the caller, a verification service running at a callee contacts the same CPS to obtain, decrypt, and verify the PASSporT identity assertion. In large scale deployments, such as a contact center, there is likely to be a single main line number for all incoming calls and that called number would be expected to service several hundred calls per minute resulting in a significant load on a verification service. Additionally, the verification service would be expected to sift through several hundred PASSporT entries on a CPS to obtain, decrypt and validate the correct PASSporT for a given call. Techniques are presented herein to address these challenges by providing to a verification service the explicit location of the PASSporT for a given call in the CPS.

DETAILED DESCRIPTION

Secure Telephony Identity Revisited (STIR) is a framework that enables the cryptographic assertion and verification of the identity of a caller. While the STIR framework allows the caller/authentication server to cryptographically assert a caller identity of the Session Initiation Protocol (SIP) Identity header field in a SIP INVITE request, existing deployment realities make it difficult if not impossible for the Identity header field value to be preserved end-to-end from caller to callee. This is in large part due to the significant deployment trail of legacy equipment in carrier networks.

Taking these deployment realities into consideration, STIR proposes an OOB mechanism (see, https://datatracker.ietf.org/doc/draft-ietf-stir-oob/) wherein the entity that cryptographically asserts caller identity (e.g., an authentication service) places a full PASSporT (see, Request For Comments (RFC) 8255) in a rendezvous service known as a Call Placement Service (CPS). To verify the identity of the caller, the verification service running at the callee contacts the same CPS to obtain, decrypt, and verify the PASSporT identity assertion.

When a caller originates a call, it is required for the caller/authentication service to create a full PASSPorT and place the PASSporT in the CPS. However, before doing so, it is required for the authentication service to obtain two pieces of information:

1. The location of the CPS; and
2. A public key for the called number.

The STIR OOB use-case draft (referred to above) discusses how both of these pieces of information may be obtained by the authentication service. Additionally, the draft https://datatracker.ietf.org/doc/draft-peterson-stir-servprovider-oob discusses how a CPS hosted in a service provider (SP) domain may be discovered/authenticated by a caller domain.

The current STIR OOB architecture has structural limitations. For example, the current STIR OOB specification requires the CPS to have a PASSporT (encrypted) that is indexed under the called number. In large scale deployments, such as for a contact center, there is likely to be a single main line number such that all incoming calls to the contact center use the main line number as the called number. The called number would be expected to service several hundred calls per minute resulting in a significant load on the verification service. Additionally, the verification service would be expected to sift through several hundred PASSporT entries on a CPS to obtain, decrypt and validate the correct PASSporT for a given call.

After placing a PASSporT in the CPS, the authentication service/caller receives confirmation of the same and is then allowed to initiate a call. Once the call arrives at the callee, the verification service either contacts the CPS to obtain PASSporTs signed under the called number's public key or receives a notification from the CPS about the presence of PASSporTs for a certain called number.

As noted above, in large scale deployments, such as for a contact center, there likely would be a single main line number that services several hundred calls per minute. In such deployments, there likely would be a significant load on the verification service (either standalone or distributed). This is, in large part, for two reasons:

1. The verification service is required to undertake a not insignificant number of operations before it can cryptographically determine the validity of the asserted identity. Those operations include, for example, obtaining all of the PASSporTs placed in the CPS for the main line number, decrypting the PASSporTs, determining if there is a match between the presented calling number (via signaling) and the origination ("orig") claim of a PASSporT, verifying the signature, verifying the Issued At ("iat") claim, etc.
2. STIR OOB architecturally places constraints such that it is not possible for the CPS to discern the caller who created and stored the PASSporT in the CPS. As a consequence, when a request is made from the verification service (to the CPS) or a push is made from the CPS to the verification service, the verification service is presented with a block of all PASSporTs for the main line called number, resulting in what could be a very large number of such PASSporTs.

Consequently, a framework that reduces the number of PASSporTs presented to the verification service to either one or a small set, while preserving the security principles of the STIR OOB mechanism, would be extremely useful. Techniques are presented herein for such a framework, wherein the verification service performs a "dual lookup" to determine the location of, obtain, and verify a single or a small set of PASSporTS for a given calling number.

It is important to note that the techniques presented herein assume the presence of a CPS that is local to the calling number. This should be possible as the calling number for one call could be the called number for another call.

The techniques presented herein may be described with reference to Figure 1, below.

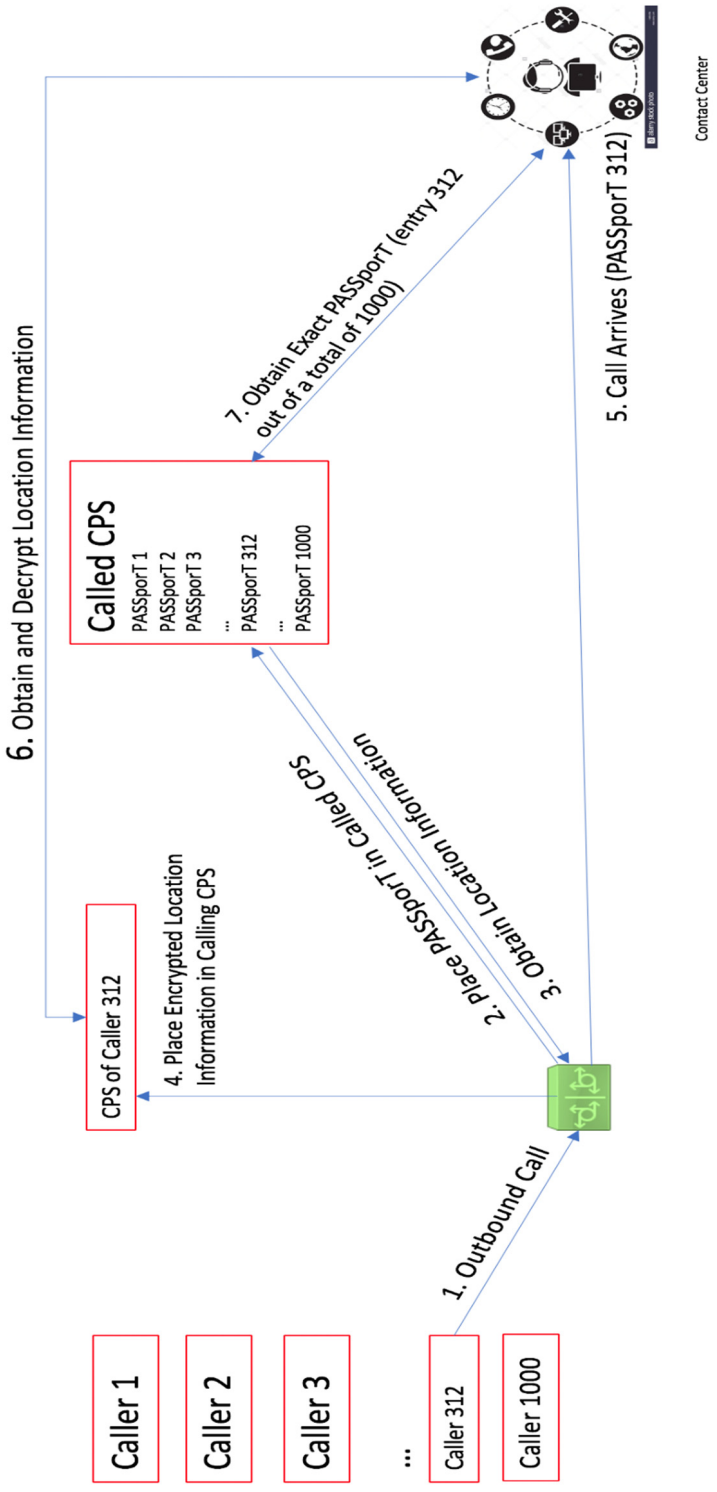

Figure 1: Dual Lookup Framework

As illustrated in the above figure, at Step 1 the authentication service at the caller creates a PASSporT. As per the requirement of the STIR OOB architecture, the authentication service looks up the CPS and public key for the called number. Once the public key is obtained, the PASSporT is signed and placed in the CPS under the called number directory - for example by issuing a Hypertext Transfer Protocol (HTTP) POST request method on the called number resource. An example of this operation is provided below:

```
POST /cps/123456/ppts HTTP/1.1. (123456 is the called mainline number)
Host: cps.example.com
Content-Type: application/passport
rlWuoTpvBvWSHmV1AvVfVaE5pPV6VaOup3Ajo3W0VvjvrQI1VwbvnUE0pUZ6Yl9w
MKW0YzI4LJ1joTHho3WaY3Oup3Ajo3W0YzAypvW9rlWxMKA0Vwc7VaIlnFV6JlWm
nKN6LJkcL2INMKuuoKOfMF5wo20vKK0fVzyuqPV6VwR0AQZlZQtmAQHvYPWipzyaV
wc7VaEhVwbvZGVkAGH1AGRlZGVvsK0ed3cwG1ubEjnxRTwUPaJFjHafuq0-mW6S1
IBtSJFwUOe8Dwcwyx-pcSLcSLfbwAPcGmB3DsCBypxTnF6uRpx7j
```

After an entry is created on the CPS it responds with a 201 response status code, such that the Location header field value provides the explicit location of the PASSporT on the called number CPS. For example:

```
/cps/123456/ppts/ppt7 /*ppt7 is the PASSporT placed in the called party CPS*/
```

Referring back to Figure 1 at Step 2, before the call is placed on the wire, for example via a SIP INVITE request, the authentication service encrypts the location information obtained in the previous step (via the Location header field value of the 201 response) with the public key of the called number and places the encrypted location information in the CPS that is local to the calling number.

In Figure 1 at Step 3, assuming the calling number is 171717, the authentication service issues a HTTP POST request method to the local CPS. For example:

```
POST /localcps/171717/pptlocation HTTP/1.1. /*171717 is the calling number*/
Host: cpslocal.example.com
Content-Type: application/location
```

```
{Encrypted Location Information}
```

There is likely going to be only a single entry under the calling number at the local CPS as a given calling number would be engaged in a single call at any given point in time.

In Figure 1 at Step 4, before a SIP INVITE request is placed on the wire, the authentication service of the calling number has (1) Created a PASSporT and placed it in the CPS (encrypted with the public key of the called number) of the called number and (2) obtained and encrypted the location at which the PASSporT is placed in the called number CPS. The encrypted location information is placed in a CPS local to the authentication service.

In Figure 1 at Step 5, when the call arrives at the destination the verification service first does a lookup for the CPS of the calling number (e.g., the procedure outlined in Section 10 of https://datatracker.ietf.org/doc/draft-ietf-stir-oob/ may be used for this purpose). Once obtained, it then issues a HTTP GET request method for all the entries under the calling number. For example:

```
GET /localcps/171717/pptlocation HTTP/1.1
Host: cpslocal.example.com
```

The CPS returns a list of all encrypted location entries currently in the collection. In most cases this is likely going to be just a single entry. The verification service then issues a new GET request method for the specific encrypted location entry. The encrypted location entry is returned to the verification service. Since the entry is encrypted using the public key of the called number, decryption is possible only with the corresponding private key - which is known to the verification service.

In Figure 1 at Step 6, following decryption the verification service obtains the location of the PASSporT placed in the called number CPS. At this time, the verification service is able to explicitly request a single PASSporT based on the decrypted location information instead of obtaining a block of PASSporTs that may or may not be helpful in cryptographically verifying the asserted identity.

There may be instances in which a given calling number is shared by several entities at a given time. In such instances, it is possible for a local CPS to have more than one

encrypted location entry - particularly if there is more than one simultaneous call with the same calling number. However, even in such instances, the number of entries for the verification service to sift through would be significantly smaller than what it would be expected to sift through without the DUAL lookup mechanism.

It is important to note that the techniques that were presented above do not introduce any new denial-of-service (DoS) vectors that are not already present in the STIR OOB architecture.

In summary, techniques have been presented for a mechanism in a STIR framework through which the load on the verification service for a called number is significantly reduced by providing the verification service with the explicit location of the PASSporT for a given call in the CPS.