**ATIS IP-NNI**

**October 26, 2021**

**Contribution**

**Title: PASSporT signature validation when “div” PASSporT present**

**Source**\***: Charter Communications**

**Issue Number:**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Abstract

RFC 8224 PASSporT signature validation rules need modification when “div” PASSporT present to allow for the case where the To header TN changes during retargeting.

# RFC 8224 PASSporT signature validation

RFC 8224 section 6.2 specifies validating the signature in the Identity header by creating a PASSporT with the dest & orig claims derived from the content of the INVITE. If a “div” PASSporT is present this would need to change since the To header may have changed.

# Alternatives for validating PASSporT signatures

## Use Full form of the PASSporT in the Identity header

**Option 1:**

For each Identity header in the INVITE use the Full form of the PASSporT when verifying the signature versus creating a new PASSporT.

Then verify the claims (per “div” rules) of all the PASSporTs with a valid signature – e.g. verify have valid “div’ chain from “shaken” “dest” claim to the TN in the r-uri and all the “orig” claims match the TN in the PA.I

## Create new PASSporTs from “div” chain

**Option 2:**

Possibly more in the spirit of RFC 8224 would be to create a new PASSporT when validating the signature. No issue in what to use for “orig” claim since this doesn’t change.

For “dest” claim the following is one way to do this

Order the PASSporTs to form a valid “div” chain. P-1 is the 1st PASSporT in the chain (i.e. the “shaken” PASSporT) and P-N is the Nth PASSporT.

When verify the signature of PASSporT P-N follow procedures in RFC 8824 with the exception that the

* “dest” claim for P-N will be either
	+ the “div” claim of P-N+1
	+ or the TN in the r-uri if P-N is the last PASSporT

An issue with this approach is what to do if have a broken “dest” chain. In this case could simply treat it is as if all PASSporTs failed validation.

## Allow both

If a set of Identity headers passes either of the 2 Options then it would pass the other.

 Only potential advantage of Option 1 is that can verify signature of PASSporTs independent of the content of the other PASSporTs so could have some PASSporTs with valid signatures and some without or all valid but have a broken chain. This is only an advantage if an implementation made use of this information – e.g. if in the failure case the “shaken” PASSporT passed signature validation an implementation could decide to treat this as No-TN-Validation and if it failed treat it as TN-Validation-Failed.

So could specify both options as allowed and which to use is based on local policy.