**ATIS IP-NNI**

**October 12, 2021**

**Contribution**

**Title: “div” transition**

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

**Issue Number:**

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

Abstract

Additional changes proposed for the transition to ubiquitous “div” support.

# Off line discussion

This proposal was discussed in an offline meeting and agreement was reached that:

* In the long term when div support is universal that “dest” validation will be against the r-uri in the case where To & r-uri TNs do not match.
* In the transition to “div” will give verstat=No-TN-Validation when only validation failure is “dest” does not match r-uri TN and To & r-uri TNs do not match.

It was also agreed to recommend (should not shall) that during the transition when adding a “div” PASSporT the To & r-uri TNs should be different – to avoid validation failure by a TSP that doesn’t support “div”

The text proposals in this contribution were drafted after the meeting and not reviewed in the off line discussion.

# Verstat when TN in To & r-uri are different

## Background

ATIS-1000074 specifies If the procedures in ATIS-1000085.v002 are not supported & the To & r-uri are different then the verifier shall skip verification.

Therefore the “div” specification should cover what to do whenever the To & r-uri are different.

One case that is not covered is when there are no “div” PASSporTs and the “shaken” PASSporT “dest” claim matches the r-uri.

One way this could happen is if the OSP implements Option-2 in 5.5.1 of “div”

**Option 2** Update the To header field TN to match the Request-URI TN and perform SHAKEN authentication as specified in ATIS-1000074 [Ref 1].

And if a transit provider changed the To header – e.g. some implementation derive the To header based on the top most diversion header (if present) or r-uri if not.

If do not specify what to do in this case presumably some implementations would treat this as a validation failure.

## Long Term

Before deciding what to do in the interim should consider what to do in the long term – i.e. when “div” PASSporTs are universally supported.

When receiving a “shaken” PASSporT and no “div” PASSporTs, two cases to consider

1. “dest” claim doesn’t match r-uri TN
* Set verstat= TN-Validation-Passed? No since would allow replay (change r-uri TN and leave To header TN and PAI TN unchanged).
* Set verstat=No-TN-Validation? Again would say no even though this is currently what ATIS-10000074 specifies, but the reason for this is the procedures for handling retargeting were not yet worked out.
* Set verstat= TN-Validation-Failed? Yes with universal support for “div” no reason to skip validation in this case.
1. “dest” claim matches r-uri TN
* Set verstat= TN-Validation-Passed? Yes since there is no replay scenario & technically there is a valid chain from the “dest” claim in the “shaken” PASSporT to the TN in the r-uri. (Note if r-uri TN is B, and “shaken” “dest” claim was B, adding a “div” PASSporT with “dest” & “div” claim equal to B would create a valid div chain.)
* Set verstat=No-TN-Validation? No even though this is currently what ATIS-10000074 specifies, but the reason for this is the procedures for handling retargeting were not yet worked out.
* Set verstat= TN-Validation-Failed? No since no replay scenario

## Transition

Now need to decide how to handle this during the transition phase, when the TSP supports “div” but “div” is not universally supported.

When receiving a “shaken” PASSporT and no “div” PASSporTs, two cases to consider

1. “dest” claim doesn’t match r-uri TN
	* Set verstat= TN-Validation-Passed? No since would allow replay (change r-uri TN and leave To header TN and PAI TN unchanged).
	* Set verstat=No-TN-Validation? Yes – since “div” not universally supported should allow for the case where call was retargeted by implementation not supporting “div” and continue applying the ATIS-10000074 treatment..
	* Set verstat= TN-Validation-Failed? No
2. “dest” claim matches r-uri TN
* Set verstat= TN-Validation-Passed? Yes since there is no replay scenario & technically there is a valid chain from the “dest” claim in the “shaken” PASSporT to the TN in the r-uri.
* Set verstat=No-TN-Validation? No even though this is currently what ATIS-10000074 specifies, but the reason for this is the procedures for handling retargeting were not yet worked out.
* Set verstat= TN-Validation-Failed? No since no replay scenario

# Recommendation on setting To header

If a “div” implementation sets the To header TN to be the same as the r-uri TN then a TSP that doesn’t support “div” would attempt verification using only the “shaken” PASSporT and the call would fail verification.

This can be avoided by having the To header & r-uri have different TNs. If this is done then the then a TSP that doesn’t support “div” would skip verification.

# Text Proposal

The intent of the following text proposal is to

1. specify verification procedure when no “div” PASSporT & To & r-uri TN are different
2. In the transition to “div” will give verstat=No-TN-Validation when only validation failure is “dest” does not match r-uri TN and To & r-uri TNs do not match.
3. no change to verification when the To header & r-uri TNs match (i.e. ATIS-1000074 rules apply in this case)
4. No changes to STI-AS procedures – So ATIS-10000074 applies for “shaken” PASSporTs
5. Recommend when adding “div” PASSPorT(s) the To & r-uri TNs should (not must) be different for backward compatibility

Text proposals are against text of IPNNI-2021-00094R000.docx

In section 5.3 propose adding the following new paragraph

In the partial-support transition period, whenever an OSP adds an Identity header with a “div” PASSporT, the OSP should check if the TNs in the To header and Request-URI have the same canonicalized value. If they have the same value, the OSP should change the TN in the To header (e.g. could align it with the “dest” claim in the “shaken” PASSporT). If these TNs are the same the call could fail verification if the TSP does not support “div” PASSporT.

Propose modifying section 5.4 as follows:

**5.4 STI-VS Verification**

**5.4.1 STI-VS Verification with “div’ PASSporTs**

On receiving an INVITE request containing an Identity header with a “shaken” PASSporT, and one or more Identity headers with “div” PASSporTs, an STI-VS shall perform the “div” verification procedures defined in RFC 8946 [Ref 4], with the following restrictions:

* The "div" PASSporT "orig" claim shall be of type “tn”,
* The "div" PASSporT "div" and "dest" claims can each have an identity type of either "tn" or "uri". When the identity type is "uri", the identity value shall identify a service URN in the 'sos' family, as defined in RFC 5031 [Ref 7],

The STI-VS shall verify that the received "div" PASSporTs create an unbroken chain of authority from the INVITE Request-URI TN to the “dest” claim of the "shaken" PASSporT. During construction of the chain, the digits ‘911’ and any services URN in the ‘sos’ family shall be considered equivalent. The STI-VS shall verify each “div” PASSporT as specified in this document. The STI-VS shall verify the “shaken” PASSporT as specified in ATIS-1000074 [Ref 1], with the exception that it shall not use the identity in the To header field to validate the "shaken" PASSporT "dest" claim.

The STI-VS shall verify the freshness of the PASSporT that was last added to the chain, as specified in ATIS-1000074 [Ref 1] (e.g., using the recommended 60 second freshness window). To accommodate call features that legitimately insert a delay before retargeting the INVITE, the STI-VS shall, based on local policy, either skip the freshness check for any earlier PASSporTs, or extend the freshness window of these earlier PASSporTs beyond the normal limit.

If the PASSporT that was last added to the chain fails the "iat" freshness test, then the STI-VS shall remove all received Identity headers. This will avoid the situation where a subsequent retargeting event adds a fresh "div" PASSporT that causes the stale PASSporT to appear fresh to downstream verifiers. This requirement shall be applied to all verification cases; i.e., whether the INVITE request contains a single "shaken" PASSporT, or a "shaken" PASSporT plus one or more "div" PASSporTs.

Note: The STI-VS removes all Identity headers when the last PASSporT fails the freshness test to avoid the problem described by the following scenario:

1. The TSP STI-VS verifies the freshness of the last PASSporT using a 60 second freshness window. The test fails, but to illustrate the problem let’s say the STI-VS does not remove all Identity headers.
2. The TSP then applies terminating features, one of which retargets the INVITE.
3. The STI-AS is invoked to authenticate the retargeting event. As described in clause 5.3, the STI-AS first verifies the freshness of the last PASSporT, and in this case it uses an extended freshness window to accommodate the retargeting feature. Because of the longer freshness window, the last PASSporT now appears fresh. Therefore, the STI-AS adds a new (fresh) "div" PASSporT for the current retargeting event, and sends the INVITE request on to the retarget-to TSP.
4. When the INVITE arrives at the retarget-to TSP, the STI-VS incorrectly generates a passing verification result (verstat=TN-Validation-Passed) because, per the procedures in this clause, the last PASSporT in the authority chain (which was added in step-3) is fresh.

This problem is resolved by having the STI-VS remove all Identity headers when the last PASSporT fails the freshness test in step-1.

**5.4.2 STI-VS Verification without “div” PASSporTs**

On receiving an INVITE request containing an Identity header with a “shaken” PASSporT and no Identity headers with “div” PASSporTs, an STI-VS shall perform verification procedures as follows:

* if the canonicalized value of the Request-URI TN does not match the canonicalized value of the TN in the To header field, the verifier shall perform verification (Note: ATIS-1000074 [Ref 1] specifies skipping verification in this case) and
	+ follow the verification procedures in ATIS-1000074 [Ref 1]  with the following modification
		- “dest” claim shall be validated using the canonicalized value of the Request-URI TN instead of the canonicalized value of the To header TN.
		- Note: Clause 5.7specifies how to set the verstat during the partial-support transition period to full “div” PASSporT support, if verification fails and the only failure is “dest” claim mismatch.
* Otherwise (i.e. if the canonicalized value of the Request-URI TN matches the canonicalized value of the TN in the To header field), the verifier shall perform verification following  the verification procedures in ATIS-1000074 [Ref 1]  (including “using the canonicalized value of the To header TN to validate the “dest” claim)

**Proposal text changes are diff marked against the text in 5.7 of IPNNI-2021-00094R000.docx**

The rules for setting verstat in the partial-support transition period are the same as stated above for the fully-supported case, with the exception that verstat shall be set to No-TN-Validation if the only verification failure is a broken chain of authority between the "shaken" PASSporT "dest" claim and the INVITE Request-URI. For example, consider the case where, during the transition period, a TSP receives an INVITE request that has been retargeted one or more times:

* If the TSP does not support "div" verification, then verstat is set as specified in clause 5.3.1 of ATIS-1000074 [Ref 1] and clause 7.2A.20.3 of TS 29.229 [Ref 10].
* If the TSP does support "div" verification, then the TSP shall set verstat to No-TN-Validation if either of the two following conditions exist:
	+ The INVITE request contains one or more "div" PASSporTs and the only verification failure is a broken chain of authority between the "shaken" PASSporT "dest" claim and the INVITE Request-URI (e.g., when "div" authentication was not performed on all of the INVITE retargeting events), then the TSP shall set verstat to No-TN-Validation, or
	+ The INVITE request contains no "div" PASSporT, and the "shaken" PASSporT "dest" claim identifies a different destination than the INVITE Request-URI but the canonicalized value of the Request-URI TN does not match the canonicalized value of the TN in the To header field..