**ATIS TOPS Council Testbeds Landscape Team**

**June 2017**

**Contribution**

**TITLE: Secure Telephone Identity (STI) Test Plan**

**SOURCE: Testbeds Focus Group, STI subteam**

**ABSTRACT**

This document contains a set of test cases for end-to-end Session Initiation Protocol (SIP) calls with a focus on Secure Telephone Identity (STI). These STI test cases are designed in support of the Signature-based Handling of Asserted information using toKENs (SHAKEN) industry framework. Underlying the SHAKEN framework are a number of Internet Engineering Task Force (IETF) documents. As the foundational SHAKEN framework is expanded, it is anticipated that this document will be updated with additional test cases. This document also incorporates test cases for SHAKEN Governance Model and Certificate Management to demonstrate successful acquisition of certificates.

Testbeds Focus Group

Secure Telephone Identity (STI) Test Plan

# Document Scope

This document contains a test plan for Service Provider (SP) to SP use cases related to Secure Telephone Identity (STI). Specifically, it describes a set of test cases in support of the Signature-based Handling of Asserted information using toKENs (SHAKEN) industry framework [ATIS-1000074]. This framework specifies an end-to-end X.509-based cryptographic authentication and verification of the telephone number (TN) identity (and potentially other information) in an Internet Protocol (IP)-based SP voice network. Underlying the SHAKEN framework are a number of Internet Engineering Task Force (IETF) documents, especially those being managed by the Secure Telephone Identity Revisited (STIR) Working Group. As the currently defined SHAKEN framework is expanded, it is anticipated that this document will be updated with additional test cases. This document also incorporates test cases to demonstrate successful acquisition of certificates as specified in: “SHAKEN: Governance Model and Certificate Management”, PTSC-2017-00093R000, or ATIS-1000080 when approved).

## STI Test Plan Scope

STI is defined to refer to the scope of functions being tested. It comes from STIR, the name of the IETF Working Group focused on RFC 4474bis, Personal Assertion Token (PASSporT) and STI credentials (based on digital certificates).

This test plan focuses on the use of E.164 TNs as SIP identities, and on securing these identities using cryptographic signatures. The test plan focuses on an E.164 TN being asserted as the calling number and how the SIP INVITE message will be signed and validated as part of an end-to-end SIP session.

This version of the test plan document primarily focuses on the SHAKEN framework as defined in ATIS-1000074. Its scope includes the format of STI tokens (including identity claims), the mapping of these tokens to SIP and the Authentication and Verification Services involved in signing and validating telephone calls. More specifically, this version of the test plan will demonstrate that the required Identity header field is created and processed correctly as specified. The intent is to provide assurance that the calling number is a secure telephone identity.

As noted earlier, as the SHAKEN framework is expanded, it is anticipated that this document will be updated with additional test cases. This document also incorporates test cases to demonstrate successful acquisition of certificates as specified in: “SHAKEN: Governance Model and Certificate Management”, PTSC-2017-00093R000. To facilitate initial testing, informal procedures will be used as needed for configuring the components involved in testing. Examples include the public key certificate generation and management, as well as local policy-based decisions based on a positive or negative verification of the signature.

This test plan will focus on the description of call flows involving originating and terminating SPs but not initially involving transit SPs. For reference, it also defines network entities based on the 3GPP IMS architecture. Such a network entity definition is not intended to mandate any particular deployment and/or implementation. It also takes a specific network call flow approach to make test configurations consistent, while recognizing that there isn’t a single network configuration that could be used for STI. These assumptions, taken together, are intended to limit the number of test points and terminology references for the testing described herein.

This version of the test plan assumes that public key certificates (or digital certificates) will be available via HTTPS at a minimum as a reference point for future STI Certificate Repositories (STI-CRs). Note that RFC 4474bis also identifies the use of DNSSEC as an alternate way of retrieving a digital certificate for the verification of signatures. However, DNSSEC is not part of the currently defined SHAKEN framework in ATIS-1000074.

This version of the test plan further assumes a basic and straightforward approach towards credentials or X.509-based digital certificate provisioning. There will be a limited number of Root Certification Authorities (CAs) that sign digital certificates. These, along with an STI Policy Administrator (STI-PA) will authorize an SP, for example, to sign telephone calls. SPs, themselves, can be Root CAs to support initial testing. The associated private keys are expected to be held securely and locally, and digital certificates will be publicly available via HTTPS. The digital certificate for a TN should be retrievable within the appropriate environment via the “info” parameter in the RFC4474bis Identity header field.

## STI Use Case Scope

This test plan supports a set of SP use cases based on the SHAKEN framework, as primarily defined in ATIS-1000074, for demonstrating anti-spoofing. In brief, the scope of these use cases and the expected testing output between originating and terminating SPs include:

* Demonstrating that the appropriate SIP header fields used for digital signing and validation are created and processed correctly across SPs, in order to provide assurance that the calling number is a secure telephone identity;
* Demonstrating the use of STI-CRs in digital certificate validation;
* Testing the interworking between different implementations of STI functions;
* Identifying and collecting issues that could arise when digital signing and validation are used for STI anti-spoofing services;
* Testing different types of telephone calls and associated treatment, including local policy-based decisions and use of the SIP Reason header field (per RFC 3326):
	+ Successfully verifying properly signed calls (200 'OK' response code)
	+ Testing a call with a Date header field value that is older than the local policy for freshness permits (403 'Stale Date' response code)
	+ Testing unsigned calls (428 'Use Identity Header' response code) is optional as not initially recommended for SHAKEN
	+ Testing a URI that cannot be dereferenced (436 'Bad-Identity-Info' response code)
	+ Testing an unsupported credential or improper digital certificate chain (437 'Unsupported credential' response code)
	+ Testing improperly signed calls (438 'Invalid Identity Header' response code)
* Testing of specific systems/subsystems:
	+ Authentication Service and Verification Service
	+ X.509-based digital certificate retrieval and validation, to include testing revoked and outdated certificates used for signing
	+ Specific cases of error generation and response

# STI Overview

This section gives a brief overview of the SHAKEN framework for STI as currently defined in ATIS-1000074. It consists of a System Description, Reference Architecture, Functional Components and Reference Call Flow.

## System Description

STI employs asymmetric key pairs to digitally sign and validate SIP INVITE messages in the particular case when E.164 TNs are used as SIP identities for originating calls. The testing in this version of the document focuses primarily on the format of STI tokens (including identity claims), the mapping of these tokens to SIP and the Authentication and Verification Services involved in signing and validating telephone calls. The systems to be used for this testing are based on the SHAKEN Reference Architecture (reproduced herein as Figure 1).

A number of assumptions are embodied in this version of the test plan and summarized below:

* E.164 TNs are used as SIP identities (i.e., the “orig” and “dest” claims are of type “tn”),
* Each TN used for testing is associated with a private and public key pair,
* Private keys are held securely by the originating SP,
* Public keys are contained in X.509-based digital certificates held by a STI-CR,
* Digital certificates are retrieved via the “info” parameter in the RFC4474bis Identity header field (or cached by the verifying service provider) within the environment under test,
* Digital signing is performed by the originating SP,
* The full form of PASSporT tokens is used and includes all of the baseline claims, as well as the SHAKEN extension claims (i.e., “ppt” PASSporT header parameter with a value of “shaken”), and
* Validation is performed by the terminating SP.

Note that identifying and collecting issues that could arise when digital signing and validation are used for STI anti-spoofing services is an important objective of testing.

## Reference Architecture

This section illustrates the SHAKEN reference architecture for STI testing. The key network entities (using the 3GPP IMS architecture by example) and STI components include:

|  |  |
| --- | --- |
| **Acronym** | **Description** |
| STI-AS | The STI Authentication Service that performs the function of the authentication service defined in RFC 4474bis. It should either itself be highly secured and contain the SKS of private key(s) or have a secure interface to the SKS. |
| STI-VS | The STI Verification Service that performs the function of the verification service defined in RFC 4474bis. It has an HTTPS interface to the STI-CR to retrieve digital certificates for validation. |
| SKS | The Secure Key Store provides secure storage of private key information used directly by the STI-AS. |
| STI-CR | The STI Certificate Repository or Repositories are publically accessible and store X.509-based digital certificates. It supports an HTTPS interface. |
| CSCF | The Call Session Control Function or SIP Registrar and Routing Function. It also has a SIP application server interface. |
| IBCF/TrGW | The Interconnection Border Control Function/Transition Gateway or ingress and egress point for SIP calls between SPs. |
| SIP UA | SIP User Agent, acting as initiating and terminating actors of end-to-end SIP calls. When under direct management control of an SP, the SP can assert the calling party identity in originating SIP INVITE messages initiated by the SIP UA. |
| CVT | Call Validation Treatment after a digital signature is positively or negatively verified (outside the scope of this test plan document). |
| Certificate Provisioning Service | Service used to provision digital certificates (this version of the test plan document assumes and describes a basic and straightforward approach towards digital certificate provisioning). |

Below, Figure 1 (as reproduced from ATIS-1000074) illustrates a reference architecture for STI testing. The focus of testing for this version of the document is primarily on the STI-AS and STI-VS functionality and the relevant SIP signaling and interfaces.



Figure . SHAKEN Reference Architecture

## Functional Components

This section describes core testing requirements for the functional entities directly related to the digital signing and validation functions. The scope is the use of E.164 TNs as calling numbers.

| **Functional Entity** | **Implementation** | **Hardware/****Software Required** | **Implementer** |
| --- | --- | --- | --- |
| STI-AS | Application Server configured via an interface from the CSCF | Application Server software | Originating SP |
| STI-VS | Application Server configured via an interface from the CSCF | Application Server software with HTTPS interface to STI-CR | Terminating SP |
| SKS | Private Key database | At a minimum, a small secure database to store private key(s) if not part of the STI-AS | Originating SP |
| STI-CR  | Certificate Repository Server(s) | At a minimum, server software to store and provide access to X.509-based digital certificates | STI-CR Provider or SP |
| CSCF  | 3GPP IMS SIP Registrar and Routing Server or equivalent | Server software supporting SIP Registrar and Routing function (ability to initiate originating and terminating call triggers and add a P-Asserted-Identity header field asserting that the calling number is a secure telephone identity | Originating and Terminating SP |
| IBCF/TrGW | Session Border Controller (if/as needed) | Session Border Controller or equivalent | Originating and Terminating SP (if/as needed) |
| SIP UA | SIP Client | Command line SIP client software or equivalent that is authenticated by the SP network | Originating and Terminating SP |
| CVT | Outside scope of test plan |  |  |
| Certificate Provisioning Service | Outside scope of this version of test plan | For initial testing, software to acquire a digital certificate from a Root CA. SPs, themselves, can be Root CAs to support initial testing. The associated private key(s) are held in the SKS, and digital certificates stored in a STI-CR | Root CA Provider and/or Originating SP |

## Reference Call Flow

This section describes the use of the Reference Architecture for anti-spoofing testing. Figure 2 (as also contained in ATIS-1000074) shows a flow for a digitally signed and validated call for a SIP session from an originating SP to a terminating SP.

 

Figure 2. Reference Call Flow for STI Testing

A summary of the steps in the call flow for the digitally signing and validating of a SIP INVITE message for STI testing is outlined below. It is assumed that the following have taken place before the call is originated:

* The originating SP (i.e., SP A) provisions its test X.509-based digital certificate(s) in a STI-CR.
* SP A provisions its test private key(s) in the SKS.
1. The originating SIP UA, which first REGISTERs and is authenticated to the CSCF, creates a SIP INVITE with a TN identity.
2. The CSCF of SP A adds a P-Asserted-Identity header field asserting the TN identity of the SIP UA. The CSCF then initiates an originating trigger to the STI-AS for the SIP INVITE message.
3. The STI-AS of SP A first determines the legitimacy of the TN identity being used in the SIP INVITE message. The STI-AS then securely requests its private key from the SKS.
4. The SKS provides the private key in the response, and the STI-AS signs the SIP INVITE message and adds an Identity header field per RFC 4474bis using the TN identity in the P-Asserted-Identity header field.
5. The STI-AS passes the SIP INVITE message back to the SP A CSCF.
6. The originating CSCF, through standard resolution, routes the call to the egress IBCF.
7. The SIP INVITE message is routed to the terminating SP (i.e., SP B) through the standard inter-domain routing configuration.
8. The SP B ingress IBCF receives the SIP INVITE message from SP A.
9. The terminating CSCF initiates a terminating trigger to the STI-VS for the SIP INVITE message (note that the STI-VS must be invoked before terminating call processing).
10. The STI-VS of SP B uses the “info” parameter information in the Identity header field per RFC 4474bis to determine the STI-CR URI and makes an HTTPS request to the STI-CR.
11. The STI-VS checks revocation status of the digital certificate, validates it and then extracts the public key. It constructs the RFC 4474bis format and uses the public key to verify the signature in the Identity header field, which validates the TN identity used when signing the SIP INVITE message on the originating service provider STI-AS.
12. The CVT is an optional function that can be invoked to perform call spam analytics or other mitigation techniques and return a response related to what should be signaled to the terminating SIP UA for a legitimate or illegitimate call. This function is out of the scope for this test plan.
13. Depending on the result of the STI validation, the STI-VS determines that the call is to be completed with any appropriate indicator and the SIP INVITE message is passed back to the terminating CSCF which continues to set up the call to the terminating SIP UA.
14. The terminating SIP UA receives the SIP INVITE message and normal SIP processing of the call continues, returning “200 OK” or optionally setting up media end-to-end.

Testing to demonstrate cases that generate error response codes can be performed based on straightforward modifications to the above steps.

A sequence diagram of events for the reference call flow for anti-spoofing testing is shown in Figure 3.

**STI-VS**

**STI-AS**

UA 1

SP A

SKS

SP B

UA 2

INVITE

Get Private Key

Private Key

Signed INVITE

100 Trying

Get Digital Certificate

Digital Certificate\*

100 Trying

200 OK SDP

ACK

ACK

Signed INVITE

STI-CR

100 Trying

200 OK SDP

200 OK SDP

ACK

More SIP Signaling

\* Certificate contains the
 corresponding Public Key

Figure 3: STI-AS and STI-VS Call Flow Sequence Diagram

## STI Certificate Management Framework Overview

This section describes the certificate management framework for SHAKEN [ATIS-1000074], which establishes an end-to-end architecture that allows an originating Service Provider (SP) to authenticate and assert a telephone identity and provides for the verification of this telephone identity by a terminating service provider. The SHAKEN framework defines a profile, using protocols standardized in the IETF Secure Telephone Identity Revisited (STIR) Working Group (WG). “SHAKEN: Governance Model and Certificate Management”, PTSC-2017-00093R000, provides recommendations and requirements for implementing these IETF specifications, draft-ietf-stir-passport, draft-ietf-stir-rfc4474bis, and draft-ietf-stir-certificates, to support management of Service Provider level certificates within the SHAKEN framework.

The SHAKEN framework uses X.509 certificates, as defined in “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”, IETF [RFC 5280], to verify the digital signatures associated with Session Initiation Protocol (SIP) identifiers. PTSC-2017-00093R000 describes how the certificates are managed and created using the recommended governance model where there is a central STI Policy Administrator (STI-PA) who authorizes Service Providers (SPs) to acquire certificates from trusted Certification Authorities (CAs).

###  System Description

The SHAKEN governance model for certificate management from PTSC-2017-00093R000 is reproduced below in Figure 4.



Figure 4. SHAKEN Governance Model and Certificate Management

The figure identifies the following roles associated with governance and certificate management:

• Secure Telephone Identity Governance Authority (STI-GA)

• Secure Telephone Identity Policy Administrator (STI-PA)

• Secure Telephone Identity Certification Authority (STI-CA)

• Service Provider (SP)

The STI-GA provides the interface to the SHAKEN framework for the enactment of policies established by a National/Regional Regulatory Authority (NRRA). The STI-GA is responsible for:

• Defining the policies and procedures around who can acquire certificates.

• Establishing policies around who can manage the PKI and issue certificates.

The STI-PA role satisfies the requirement “to apply the policies and procedures established for certificate management.” The protocols and message flows between the STI-PA, the SPs, and STI-CAs to support the issuance and management of certificates to support STI are described in PTSC-2017-00093R000.

###  Reference Architecture

The SHAKEN certificate management architecture is illustrated in Figure 5 below.



Figure 5. SHAKEN Certificate Management Architecture

The above SHAKEN certificate management architecture introduces the following additional elements:

* Service Provider Key Management Server (SP-KMS) - The service provider’s server that generates private/public key pair(s) for signing, requests a certificate from the STI-CA, and receives the STI-CA signed public key certificate(s).
* Secure Key Store (SKS) - The store for private keys used by the originating service provider Authentication Service.
* Secure Telephone Identity Certificate Repository (STI-CR) - The HTTPS server that hosts the public key certificates used by the destination service provider’s Verification Service to validate signatures.

Note that the STI-PA functional element also plays a key role in the certificate management architecture and related procedures.

###  Functional Components

For this version of the test plan, the following sections describe the procedures and process steps related to test cases demonstrating the successful acquisition of certificates per PTSC-2017-00093R000. There are three major components: STI-PA, STI-CA, and SPs. They are involved in the following sets of procedures viewed from an SP perspective:

|  |  |
| --- | --- |
| **Test Case** | **Primary Components** |
| STI-PA Registration | SP and STI-PA |
| STI-CA Registration | SP and STI-CA |
| Service Provider Code (SPC) Token Request | SP and STI-PA |
| Certificate Signing Request (CSR) | SP and STI-CA (with STI-CA reliance on STI-PA) |

Some components shown in Figure 5 may be internal to or at the discretion of SPs.

###  Reference Process Steps

For this version of the test plan, the Certificate Management Framework procedures and process steps refer to a set of test cases demonstrating an SP’s successful acquisition of a service provider certificate – identified by a Service Provider Code (SPC) – corresponding to a key pair used to sign and validate calls originated by that SP.

These test cases, including readiness and provisioning steps, are summarized in the following table.

|  |  |  |
| --- | --- | --- |
| **Step Title** | **Description** | **Notes** |
| Step 0. Configuration Readiness | STI-PA and STI-CA are configured | STI-CA is on STI-PA’s list of valid STI-CAs |
| Step 1. STI-PA Registration | SP registers with STI-PA | SP acquires list of valid CAs |
| Step 2. STI-CA Registration | SP registers with STI-CA | SP establishes ACME credentials |
| Step 3. SPC Token Request | SP requests from STI-PA | SP receives signed SPC token |
| Step 4. Certificate Signing Request | SP requests from STI-CA | SP receives signed certificate |
| Step 5. SP Provisioning | SP is ready for SHAKEN calls | SP stores certificate and URL |

The procedures for these steps are described in the following sections.

####  Step 0. Configuration Readiness

* The STI-PA maintains a list of approved/valid STI-CAs.
	+ The list includes each STI-CA’s public key certificate and account registration URL.
* The PA has a PA-admin certificate.
* The STI-PA has access to resources for validating SPs and their SPCs [likely OCNs].

####  Step 1. STI-PA Registration

* The SP registers with the STI-PA using a username, password, and SPC.
* The STI-PA validates the SP and its SPC.
* The STI-PA provides the client\_id/client\_secret to the SP.
* The STI-PA provides access to the list of approved STI-CAs to the SP.
* [The SP acquires the list of approved STI-CAs from the STI-PA.]

####  Step 2. STI-CA Registration

* The SP selects one STI-CA from the list of approved STI-CAs.
* The SP generates a CA-account key pair.
* The SP sends a new registration request to the selected STI-CA.
	+ The request includes the public CA-account key.
	+ The request is signed with the private CA-account key.
* The STI-CA validates the request and responds with a 201 OK.

####  Step 3. SPC Token Request

* The SP requests a SPC token from the STI-PA.
	+ The SP sends a request with username and password.
	+ The request includes the client\_id/client\_secret in the header.
	+ The SP provides its SPC.
	+ The SP provides the fingerprint of its public CA-account key.
* The STI-PA validates the request.
* The STI-PA constructs the signed SPC token.
	+ The token includes the URL of the STI-PA’s PA-admin certificate.
	+ The token includes the SP’s SPC.
	+ The token includes the fingerprint of the SP’s public CA-account key.
	+ The token includes the STI-PA’s signature using its private key.
* The STI-PA sends a 200 OK response to the SP.
	+ The response includes the signed SPC token.

####  Step 4. Certificate Signing Request

* The SP generates a call-signing key pair.
	+ The SP stores its private call-signing key.
* The SP constructs a CSR.
	+ The SP provides its SPC using TN Authorization List extension.
	+ The SP provides its public call-signing key.
* The SP sends its request to the STI-CA.
	+ The SP includes the CSR in the request.
	+ The SP includes a time frame in the request.
	+ The SP signs the request with its private CA-account key.
* The STI-CA processes the certificate signing request.
	+ The STI-CA extracts the SPC from the CSR.
* The STI-CA sends an authorization challenge response to the SP.
	+ The response includes the URL for retrieving authorization details.
* The SP retrieves the authorization challenge details using the supplied link.
	+ The challenge details include the SPC.
	+ The challenge details include the URL for responding to the challenge.
* The SP sends a response to the URL it retrieved for responding to the challenge.
	+ The response includes the (signed) SPC token the SP received from the STI-PA.
	+ The response is signed with the SP’s private CA-account key.
* The STI-CA validates the challenge response it received from the SP.
	+ The STI-CA retrieves the STI-PA’s certificate and extracts the PA-admin public key.
	+ The STI-CA validates the signed SPC token.
	+ The STI-CA creates the certificate.
	+ The STI-CA sets the status to valid.
* The SP polls the STI-CA until it detects that the status has been set to valid.
* The SP sends a request for its certificate to the STI-CA.
* The SP receives a 200 OK response from the STI-CA.
	+ The response includes the call-signing certificate.
	+ The response includes other links and certificates.

####  Step 5. SP Provisioning

* The SP stores the call-signing certificate in its STI-CR.
* The SP associates its URL for the SHAKEN public key certificate with its corresponding SPC private key.

# STI Test Cases

The six test families are:

* End-to-end functional tests
* Detailed tests of protocol/data objects
* STI Authentication Service testing
* STI Verification Service testing
* Interface testing
* Supplemental testing

*Note: A number of the table entries are placeholders for further work.*

***End-to-End Functional Testing***

This family consists of tests demonstrating three of the most expected call scenarios: a valid signed call, an invalid signed call, and an unsigned call. Interactive Voice Response (IVR) messages are suggested as one way to convey the result to the caller.

| **Test #** | **Test Title** | **Test Purpose** | **Comments/Suggestions** |
| --- | --- | --- | --- |
| 3.1 | End-to-End Functional Testing | Demonstrate expected call scenarios. |
| 3.1.1 | Valid Signed Call | Successful end-to-end signed call | Valid certificate associated with calling TN.IVR: valid signed call |
| 3.1.2 | Invalid Signed Call (Response code 436) | 'Bad Identity info' | Can’t retrieve certificate.IVR: invalid, Response code 436 |
| 3.1.3 | Invalid Signed Call (Response code 437) | 'Unsupported Credential' | Can’t validate certificate. IVR: invalid, Response code 437 |
| 3.1.4 | Invalid Signed Call (Response code 438) | 'Invalid Identity Header' | Valid certificate not associated with calling TN.IVR: invalid, Response code 438 |
| 3.1.5 | Unsigned Call (Response code 428) | 'Use Identity Header' | IVR: unsigned call, Response code 428 |
| 3.1.6 | Date Freshness(Response code 403) | “Stale Date” | Date header field value is older than the local policy (e.g., 60 sec) for freshness.IVR: date is older than 60 sec, Response code 403 |

***Detailed Protocol Testing***

This family of tests is intended for detailed testing of the protocols involved. In particular, they include the data objects used by the protocol.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **Test Title** | **Test Purpose** | **Comments/Suggestions** |
| 3.2 | Detailed Protocol Testing | Examine the protocol/data objects used in STIR/SHAKEN |
| 3.2.1 | Certificate Testing | Examine certificate-related errors | Tests errors in validation, contents, and revocation |
| 3.2.1.1 | Certificate Validation and Contents Testing | Certificate validation or content errors | - Invalid dates- Invalid signature |
| 3.2.1.2 | Certificate Revocation Status Testing | Certificate revocation status failures | - CRL/OCSP checking |
| 3.2.1.3 | Trust Anchor Testing | Chain of trust failures | - Invalid chain of trust- Invalid trust anchor |
|  |  |  |  |
| 3.2.2 | Identity Header Testing | Examine header-related errors |  |
| 3.2.2.1 | Valid PASSporT token  | Identity header validation |  |
| 3.2.2.2 | “ppt” parameter | Identity header validation |  |
| 3.2.2.3 | “info” parameter | Identity header validation |  |
|  |  |  |  |
| 3.2.3. | PASSporT object Testing | Examine object-related errors |  |
| 3.2.3.1 | “typ” attribute  | PASSporT token header validation |  |
| 3.2.3.2 | “alg” attribute  | PASSporT token header validation |  |
| 3.2.3.3 | “x5u” attribute | PASSporT token header validation |  |
| 3.2.3.4 | “iat” attribute  | PASSporT token claim validation |  |
| 3.2.3.5 |  “orig” attribute | PASSporT token claim validation |  |
| 3.2.3.6 | “dest” attribute | PASSporT token claim validation |  |
| 3.2.3.7 | “attest” attribute | PASSporT token claim validation |  |
| 3.2.3.8 | “origid” attribute | PASSporT token claim validation |  |

***STI Authentication Service Testing***

This family of tests exercises the Authentication Service subsystem.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **Test Title** | **Test Purpose** | **Comments/Suggestions** |
| 3.3 | STI Authentication Service Subsystem Testing |  |
| 3.3.1 | AS Functional Tests |  |  |
| 3.3.1.1 | If PAI used as Secure Telephone Identity | Test whether PAI is given priority over From | Rules/Criteria:1. “PAI” (if present) else “From”
 |
| 3.3.1.2 | E.164 number in the SIP URI is identified as a TN | URI user-part of the form “+17005551008” in the SIP URI with “user=phone” parameter in it. | For instance,sip:+15552223333@example.com;user=phone |
| 3.3.1.3 | E.164 number in the TEL URI is identified as a TN | URI user-part of the form “+17005551008” in the TEL URI | For instance,tel:+15552223333 |

***STI Verification Service Testing***

This family of tests exercises the Verification Service subsystem.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **Test Title** | **Test Purpose** | **Comments/Suggestions** |
| 3.4 | STI Verification Service Subsystem Testing |  |
| 3.4.1 | VS Function Testing |  |  |
| 3.4.1.1 | Success with valid certificate | Verification to succeed if the certificate is validated to be from trusted source  |  |
| 3.4.1.2 | Failure with expired certificate | Verification to fail if the certificate has expired |  |
| 3.4.1.3 | Failure due to Certification Revocation check | Verification to fail if the certificate is found to have been revoked |  |
| 3.4.1.4 | Date Freshness Test | If the Date parameter freshness test fails, a 403 – ‘Stale Date’ should be returned | The assumption is that the Date value is within 60 seconds from the current time (configurable) |
| 3.4.1.5 | Success on Missing Identity Header | 428 – ‘Use Identity Header’ error should **NOT** be is returned if no Identity header received | ATIS SHAKEN recommends not to return a 428 until wider adoption of SHAKEN. |
| 3.4.1.6 | Failure on Bad Identity Info | 436 – ‘Bad Identity Info’ should be returned if the URL in the ‘info’ parameter couldn’t be dereferenced |  |
| 3.4.1.7 | Failure on Unsupported credential | 437 – ‘Unsupported credential’ when the verifier doesn’t support/trust the certificate chain |  |
| 3.4.1.8 | Failure on Signature Verification Error | 438 – ‘Invalid Identity Header’ returned when the signature verification fails |  |
| 3.4.1.9 | Use of PAI as STI | If present, PAI must be used to derive the orig claim |  |

***Interface Testing***

This family of tests is designed to exercise certain interfaces within a service provider, between a service provider and an external component, or between service providers.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **Test Title** | **Test Purpose** | **Comments/Suggestions** |
| 3.5 | Interface Testing |  |
| 3.5.1 | Private Key Access |  |  |
| 3.5.2 | CSCF-CSCF Interworking |  | (includes NNI) |
| 3.5.3 | Certificate Retrieval | Support call-processing validation and signing requirements | Cover basic certificate, OCSP, and CRL requests. Optionally, test certificate creation/revocation status |
| 3.5.4 | Originating UA Testing |  |  |
| 3.5.5 | Terminating UA Testing |  |  |

***Supplemental Testing***

This family of tests demonstrates the use of additional components or functions that are not necessarily part of the common functional architecture or other five test families above.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **Test Title** | **Test Purpose** | **Comments/Suggestions** |
| 3.6 | Supplemental Testing |  |
| 3.6.1 | Distributed Service Bureau Infrastructure |  |  |
| 3.6.2 | Reference Plane |  |  |

## End-to-End Functional Testing

###  Valid Signed Call

#### Purpose

The purpose of this test is to validate a successful end-to-end signed call.

####  Test Setup

##### Test Pre-conditions

The following test pre-conditions apply:

1. Test environments are reachable, preferably over the public Internet
2. Routing attributes of ingress IBCF/TrGW are shared
3. The test E.164 numbers are shared
4. Public certificate(s) required are issued
5. Share URI (https recommended) to resolve to associated public certificate used as the “info” parameter (or “x5u” claim) in the SIP Identity header
6. Agreement on the accepted codecs (minimum GSM, PCMU, PCMA)
7. Confirm whether both UDP and TCP transports are supported for call signaling. At a minimum, UDP transport should be supported
8. Support for “jumbo” UDP packets is required throughout the test environment to avoid restrictions on UDP packet sizes larger than 1,500 bytes and an outcome of a 403 Forbidden response code
9. Confirm whether both the EC and RSA key algorithms are supported. Use of EC is encouraged since it results in substantially smaller Identity headers
10. For the Verification Service, all suitable trust-related materials are provisioned

##### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header included in the INVITE message includes the full form of the PASSporT token
3. Confirm that the call is established and both the calling party and the called party can communicate with each other

#### Observable Results

##### Message Flows

The associated message flow for this test is Figure 3. This assumes that the optional use of the Reference Plane is not involved.

##### Pass/Fail Criteria

This test will pass if the call setup is successful and the two parties can communicate.

This test will fail if the call could not be setup due to some error.

#### Trace Capture

The SIP trace capture should be enabled to document test and diagnose any errors.

#### Known Issues

None at this time.

###  Invalid Signed Call (Can’t retrieve certificate)

#### Purpose

The purpose of this test is to validate that a SIP 436 response code is returned if referenced certificate can’t be retrieved.

#### Test Setup

Same as for a Valid Signed Call (see Section 3.1.1.2).

##### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2.1).

##### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the INVITE message includes a URI that cannot be dereferenced (i.e., the “info” parameter or “x5u” claim in the SIP Identity header)
3. For testing purposes only, confirm a call failure action and that the destination test environment returns either the response with SIP status code 436 “Bad Identity-Info” or response with SIP status code 200 OK and Reason header field with a value of 436 “Bad Identity-Info”

#### Observable Results

##### Message Flows

The associated message flow for this test will be a modification to Figure 3 and added to this section. This assumes that the optional use of the Reference Plane is not involved.

##### Pass/Fail Criteria

The pass/fail criteria for this test is provided in Section 3.1.2.2.2 above.

#### Trace Capture

The SIP trace capture should be enabled to document test and diagnose any errors.

#### Known Issues

None at this time.

###  Invalid Signed Call (Can’t validate certificate)

#### Purpose

The purpose of this test is to validate that a SIP 437 response code is returned if the referenced certificate can’t be validated.

#### Test Setup

Same as for a Valid Signed Call (see Section 3.1.1.2).

##### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2.1).

##### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the INVITE message includes a URI where its referenced public certificate has any one of the following errors:
	* It is self-signed or untrusted
	* Signed by untrusted or unknown CA
	* Expired
	* Revoked
3. For testing purposes only, confirm a call failure action and that the destination test environment returns the SIP response code 437 “Unsupported Certificate”

#### Observable Results

##### Message Flows

The associated message flow for this test will be a modification to Figure 3 and will be added to this section. This assumes that the optional use of the Reference Plane is not involved.

##### Pass/Fail Criteria

The pass/fail criteria for this test is provided in Section 3.1.3.2.2 above.

#### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

#### Known Issues

None at this time.

###  Invalid Signed Call (Valid certificate but for wrong TN)

#### Purpose

The purpose of this test is to validate that a SIP 438 response code is returned if there is a valid Identity header but certificate is for different TN.

#### Test Setup

Same as for a Valid Signed Call (see Section 3.1.1.2).

##### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2.1).

##### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the INVITE message includes a validly constructed Identity header but the signature value is invalid for destination TN, which in turn leads to a signature verification failure
3. For testing purposes only, confirm a call failure action and that the destination test environment returns either the response with SIP status code 438 “Invalid Identity Header” or response with SIP status code 200 OK and Reason header field with a value of 438 “Invalid Identity Header”

#### Observable Results

##### Message Flows

The associated message flow for this test will be a modification to Figure 3 and added to this section. This assumes that the optional use of the Reference Plane is not involved.

##### Pass/Fail Criteria

The pass/fail criteria for this test is provided in Section 3.1.4.2.2 above.

#### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

#### Known Issues

None at this time.

###  Unsigned Call (no Identity header)

#### Purpose

The purpose of this test is to validate that the appropriate response code is returned if there is no Identity header in the SIP INVITE message when one is locally expected.

#### Test Setup

Same as for a Valid Signed Call (see Section 3.1.1.2).

##### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2.1).

##### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant without an Identity header
2. For testing purposes only, confirm a call failure action and that the destination test environment returns either the response with SIP status code 428 “Use Identity Header” or response with SIP status code 200 OK and Reason header field with a value of 428 “Use Identity Header”

#### Observable Results

##### Message Flows

The associated message flow for this test will be a modification to Figure 3 and added to this section. This assumes that the optional use of the Reference Plane is not involved.

##### Pass/Fail Criteria

The pass/fail criteria for this test is provided in Section 3.1.5.2.2 above.

#### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

#### Known Issues

None at this time.

### Date Freshness Test

####  Purpose

Verify that if the date freshness test fails, a SIP error code of 403 is returned

####  Test Setup

##### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

##### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Test to make sure that the ‘iat’ value falls within 60 seconds (configurable) from the current time
5. Verify that a 403 SIP error code is returned if the ‘iat’ value is older than 60 seconds from the current time

####  Observable Results

##### Message Flows

##### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

####  Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

####  Known Issues

None at this time.

## Detailed Protocol Testing

### Certificate Testing

### Identity Header Testing

#### Valid PASSporT Token

##### Purpose

Verify that a valid PASSporT token exists in the Identity header field

##### Test Setup

Same as for a Valid Signed Call (see Section 3.1.1.2).

###### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2.1).

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header included in the INVITE message includes the full form of the PASSporT token

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

Pass criteria is to observe a valid PASSporT token in the Identity header field.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “ppt” Parameter

##### Purpose

Test to see if the SHAKEN “ppt” parameter exists in the Identity header.

##### Test Setup

###### Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2).

###### Procedure

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

The test passes if the “ppt=shaken” exists in the Identity header field.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “info” Parameter

##### Purpose

Verify that “info” parameter exists in the Identity Header field and its value is a valid URI

##### Test Setup

###### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2).

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header included in the INVITE message includes the ‘info’ parameter field

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

1. ‘info’ parameter is present
2. The value of the ‘info’ parameter is a valid URI
3. The value of the ‘info’ parameter matches the ‘x5u’ attribute value in the PASSporT token

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

### PASSporT Object Testing

#### “typ” attribute

##### Purpose

Verify that the “typ” attribute is set in the header section of the PASSporT object and it is set to the value “passport”

##### Test Setup

###### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the header section of the PASSporT object includes the “typ” attribute and that its value is set to “passport”

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if the value of the “typ” attribute is set to “passport”.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “alg” attribute

##### Purpose

Verify that the “alg” attribute is set in the header section of the PASSporT object and that its value is set to “ES256”

##### Test Setup

###### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the header section of the PASSporT object includes the “alg” attribute and that its value is set to “ES256” or another acceptable value.

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes,

1. If the value of the “alg” attribute is set to “ES256”.
2. If “alg” is set to a valid value other than “ES256” (such as, “RS256”), an “alg” parameter MUST also be set in the Identity header field to match the “alg” attribute in the PASSporT object.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “x5u” attribute

##### Purpose

Verify that the “x5u” attribute exists in the header section of the PASSporT object and that it is set to a valid URI

##### Test Setup

###### Test Pre-conditions

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the header section of the PASSporT object includes the “x5u” attribute and that its value is set to a valid URI.

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes,

1. If the value of the “x5u” attribute is set to a valid URI.
2. If the value of the “x5u” matches that of the “info” parameter of the Identity header field
3. Also, when the URI is resolved, it should point to a valid X.509 certificate

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “iat” Attribute

##### Purpose

Verify that “iat” attribute exists in the claim section of the PASSporT object

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the claim section of the PASSporT object includes the “iat” attribute and that its value is set to the current Unix epoch time.
4. Also, verify that the value type of “iat” is Numeric

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes,

1. If the value of the “iat” attribute is set to a valid Unix epoch time.
2. If the value type of “iat” is Numeric

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “orig” Attribute

##### Purpose

Verify that “orig” attribute exists in the claim section of the PASSporT object and that it contains a TN.

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the INVITE message includes the Identity header
3. Verify that the Identity header field contains the full form of the PASSporT token
4. Verify that the claim section of the PASSporT object includes the “orig” attribute and that it contains the “tn” attribute.
5. Verify that the value of the “tn” should agree with section 8 of RFC 4474bis.

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “dest” Attribute

##### Purpose

Verify that “dest” attribute exists in the claim section of the PASSporT object and that it contains a TN

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the INVITE message includes the Identity header
3. Verify that the Identity header field contains the full form of the PASSporT token
4. Verify that the claim section of the PASSporT object includes the “dest” attribute and that it contains at least one “tn” attribute.
5. Verify that the value of the “tn” agrees with section 8 of RFC 4474bis.

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

####  “attest” Attribute

##### Purpose

Verify that “attest” attribute exists in the claim section of the PASSporT object

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the Identity header includes the full form of the PASSporT token
4. Verify that the claim section of the PASSporT object includes the “attest” attribute and that its value is set to one of the following values: “A”, “B”, or “C”

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3 and 4 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### “origid” Attribute

##### Purpose

Verify that “origid” attribute exists in the claim section of the PASSporT object

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the Identity header field includes the full form of the PASSporT token
4. Verify that the claim section of the PASSporT object includes the “origid” attribute and that its value is set to a valid UUID

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3 and 4 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

## STI Authentication Service Testing

### AS Functional Testing

#### PAI as Secure Telephone Identity

##### Purpose

Verify that PAI header with a TN user part is given precedence over the From header

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the INVITE message also contains the PAI header
4. Verify that the user part of the URI in the PAI is a TN
5. Verify that the “tn” attribute within the “orig” attribute in the claim section of the Identity header reflects the TN in the PAI header

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### TN in the SIP URI

##### Purpose

Verify that E.164 TN in the From header SIP URI is used as STI when PAI is absent

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is included in the INVITE message
3. Verify that the “tn” attribute within the “orig” attribute in the claim section of the Identity header reflects the TN from the SIP URI of the From header

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Step 3 of the Procedure is verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### TN in the TEL URI

##### Purpose

Verify that E.164 TN in the From header TEL URI is used as STI if PAI is absent

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is included in the INVITE message
3. Verify that the “tn” attribute within the “orig” attribute in the claim section of the Identity header reflects the TN from the TEL URI in the From header

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if the Step 3 of the Procedure is verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

## STI Verification Service Testing

### VS Functional Testing

#### Success with Valid Certificate

##### Purpose

Verify that PASSporT token validation succeeds with a valid certificate that is not expired and that is not in a CRL

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that “x5u” and “info” parameters refer to the same certificate
5. Verify that the certificate expiry date is in the future
6. Verify that the certificate CRL check passes
7. Verify that the PASSporT token signature is digitally verified

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3 through 7 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Failure with Expired Certificate

##### Purpose

Verify that PASSporT token validation results in a failure due to expired certificate

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that “x5u” and “info” parameters refer to the same certificate
5. Verify that the certificate expiry date is in the past, which results in validation failure

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Failure due to Certificate Revocation List Check

##### Purpose

Verify that PASSporT token validation fails due to the certificate being revoked

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that the certificate revocation list check is performed and that the result shows that the certificate has been revoked
5. Verify that an Identity header failure error has been returned

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Date Freshness Test

##### Purpose

Verify that if the date freshness test fails, a SIP error code of 403 is returned

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that the ‘iat’ value falls within 60 seconds (configurable) from the current time
5. Verify that a 403 SIP error code is returned if the ‘iat’ value is older than 60 seconds from the current time

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Success on Missing Identity header

##### Purpose

Verify that a 428 ‘Use Identity header’ error is not returned if the Identity header is not present in the INVITE message

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is not present in the INVITE message
3. Per SHAKEN, verify that the call is established successfully and that SIP status code 428 is not raised

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Step 3 of the Procedure is verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Success with Valid Certificate

##### Purpose

Verify that PASSporT token validation succeeds with a valid certificate that is not expired and that is not on a CRL

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that “x5u” and “info” parameters refer to the same certificate
5. Verify that the certificate expiry date is in the future
6. Verify that the CRL check is completed and passes
7. Verify that the PASSporT token signature is digitally verified

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3 through 7 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Failure on Certificate Fetch Error

##### Purpose

SIP error 437 ‘Unsupported Credential’ should be returned if there is a failure in retrieving the certificate as referenced by the ‘info’ parameter

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that “x5u” and “info” parameters refer to the same URI
5. Verify that SIP error 437 is returned when the “info” URL doesn’t yield a certificate

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Failure on Signature Verification Error

##### Purpose

SIP error 438 ‘Invalid Identity Header’ should be returned if signature verification fails

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that SIP error 438 is returned in case of signature verification failure

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3 and 4 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

#### Use of PAI as STI

##### Purpose

Verify that if PAI is present, it is given precedence as STI over the From header

##### Test Setup

###### Test Pre-condition

Same as for a Valid Signed Call (see Section 3.1.1.2)

###### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header is present in the INVITE message
3. Verify that the full form of PASSporT token exists in the Identity header field
4. Verify that PAI header exists and that its user part qualifies as a TN
5. Verify that PAI header is used as STI for signature verification and the call is established successfully

##### Observable Results

###### Message Flows

###### Pass/Fail Criteria

This test passes if Steps 3, 4 and 5 of the Procedure are all verified.

##### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

##### Known Issues

None at this time.

## Interface Testing

###  Private Key Access

Private Key access is performed by the STI Authentication Service (STI-AS) and thus will utilize the preferred key storage mechanism of the AS platform, which may be via the operating system of the call-processing component or a Hardware-Signing-Module if available on the platform.

Details will be worked out as platforms used for testing via the testbed are announced.

###  CSCF-CSCF Interworking

(To be supplied)

###  Certificate Retrieval

#### Purpose

The purpose of these tests is to validate the interface to a Certificate Authority (or TN-CR). For these tests, the bearer protocol is http / https. This interface supports certificate retrieval and additional validation of certificates through Open Certificate Status Protocol (OCSP) and Certificate Revocation Lists (CRL). Optionally, an authoritative TN Reference Plane can be used to mitigate against “fake” CAs not being authoritative for TN certificates.

#### Test Setup

1. The CA is reachable through http / https, preferably on the public Internet
2. A test computer runs scripts to handle certificate-retrieval, OCSP-responses, CRLs, and (optionally) Reference Plane queries. This can be a server with terminal access or any appropriate system with network access (e.g., a Unix machine with openssh and a web-server installed).
3. Test scripts are used to fetch and verify CA and Reference-Plane objects.
4. Root certificates of Test CA(s) in use will be installed and retrievable.

##### Test Pre-conditions

The test machine is connected to the network and can reach the CA (and optionally Reference Plane) used in the tests.

Certificates used for testing include a set of domain-level certificates (e.g., company.com), a set of telephone number (TN) certificates, TN-CR intermediate certificates (if any), and CA-Root certificates. Certain certificates are constructed to be valid, invalid, or have valid or invalid signing certificates from CAs.

For TN certificates, below is a list of respective certificates with attributes for various tests. The TN-CR will be provisioned with a minimum set of test numbers, possibly of the form +1-NXX-555-01XX, and associated certificates with various properties. Private keys may be distributed between test participants.

1. TN-01: Phone number with valid certificate and chain
2. TN-02: Phone number with expired certificate
3. TN-03: Phone number with expired signing certificate
4. TN-04: Phone number with expired root certificate
5. TN-05: Phone number with revoked certificate
6. TN-06: Phone number with revoked signing certificate
7. TN-07: Phone number with revoked root certificate
8. TN-08: Phone number without entry in the TN-CR
9. TN-09: Phone number without entry in the Reference Plane (“fake” CA)

[Description of associated Test URIs to be provided.]

Note that these Test TNs and Test URIs may be used in setting up configurations for certain End-to End-Functional Tests as described in Section 3.1. The following table contains some ideas for using selected TNs and URIs to generate certain results.

| **Test #** | **Test Title** | **Test Purpose** | **Comments/Suggestions** |
| --- | --- | --- | --- |
| 3.1.1 | Valid Signed Call | Successful end-to-end signed call | TN-01 and URI-01 (link to valid certificate) |
| 3.1.2 | Invalid Signed Call (Response Code 436) | 'Bad Identity info' | TN-01 and URI-08 (link has no associated certificate) |
| 3.1.3 | Invalid Signed Call (Response Code 437) | 'Unsupported Credential' | TN-02 and URI-02 (link to expired certificate) |
| 3.1.4 | Invalid Signed Call (Response Code 438) | 'Invalid Identity Header' | Any valid TN other than TN-01 and URI-01 (“wrong” public key) |
| 3.1.5 | Unsigned Call (Response Code 428) | 'Use Identity Header' | TN-08 (no URI) |

##### Procedure *(note: these remaining sub-sections through 3.5.3.5 will be updated)*

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant
2. Inspect the SIP trace to confirm that the Identity header included in the INVITE message includes the full form of the PASSporT token
3. Confirm that the call is established and both the calling party and the called party can communicate with each other

##### Procedure

Test participants will repeat the following steps:

1. Initiate the call to the destination TN of the test participant without an Identity header
2. For testing purposes only, confirm a call failure action and that the destination test environment returns the SIP response code 428 “Use Identity Header”

#### Observable Results

##### Message Flows

The associated message flow for this test will be a modification to Figure 3 and added to this section. This assumes that the optional use of the Reference Plane is not involved.

##### Pass/Fail Criteria

The pass/fail criteria for this test is provided in Section 3.5.3.2.2 above.

#### Trace Capture

The SIP trace capture should be enabled to document this test and diagnose any errors.

#### Known Issues

None at this time.

###  Originating UA Testing

(To be supplied)

###  Terminating UA Testing

(To be supplied)

## Supplemental Testing

## (To be supplied)SHAKEN Certificate Acquisition

(To be supplied)

###  STI-PA Registration

(To be supplied)

###  STI-CA Registration

(To be supplied)

###  SPC Token Request

(To be supplied)

###  Certificate Request

(To be supplied)

# References

Text

# Abbreviations / Acronyms

Text