

ETSI TS 103 834-2 V17.0.0 (2022-12)



**Smart Secure Platform (SSP);  
Part 2: Test Specification, SSP Test Tool Interface  
(Release 17)**

---

Reference

DTS/SET-00103834-2vh00

---

Keywords

SCL, SSP, test tool interface, testing

**ETSI**

650 Route des Lucioles  
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° w061004871

---

**Important notice**

The present document can be downloaded from:  
<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at [www.etsi.org/deliver](http://www.etsi.org/deliver).

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at <https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:  
<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

If you find a security vulnerability in the present document, please report it through our Coordinated Vulnerability Disclosure Program:  
<https://www.etsi.org/standards/coordinated-vulnerability-disclosure>

---

**Notice of disclaimer & limitation of liability**

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

---

**Copyright Notification**

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.  
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2022.  
All rights reserved.

# Contents

Intellectual Property Rights .....	5
Foreword.....	5
Modal verbs terminology.....	6
1 Scope .....	7
2 References .....	7
2.1 Normative references .....	7
2.2 Informative references.....	8
3 Definition of terms, symbols, abbreviations and formats.....	8
3.1 Terms.....	8
3.2 Symbols.....	8
3.3 Abbreviations .....	8
3.4 Formats.....	9
3.4.1 Format of the requirements tables.....	9
3.4.2 Numbers and Strings.....	9
3.4.3 Format of test description clauses.....	9
3.4.4 Dynamic content validation in ASN.1 structure .....	12
4 Requirements for the TTI.....	12
4.0 Introduction .....	12
4.1 Requirements from ETSI TS 103 666-1 .....	13
4.1.1 Accessor authentication .....	13
4.1.2 Protocol layers .....	13
4.2 Requirements from ETSI TS 103 666-2.....	14
4.2.1 Runtime model.....	14
4.3 Requirements from ETSI TS 103 999-1 .....	14
4.3.1 Test Tool Interface requirements .....	14
4.4 Requirements from GlobalPlatform - Virtual Primary Platform .....	14
4.4.1 FFS.....	14
4.5 Requirements from ETSI TS 103 834-1 .....	15
4.5.1 TTI MQTT requirements .....	15
4.5.2 TTI Accessor Authentication Service requirements .....	15
4.5.3 TTI Control Service requirements .....	15
4.5.4 TTI Data Service requirements.....	16
4.5.5 RDE Data Service requirements .....	16
4.6 ASN.1 syntax .....	16
4.6.1 Introduction.....	16
4.6.2 Start of ASN.1 .....	16
5 Test descriptions for TTI implementations .....	17
5.1 Test descriptions on ETSI TS 103 666-1 requirements.....	17
5.1.1 Accessor authentication .....	17
5.1.2 Protocol layers .....	17
5.2 Test descriptions on ETSI TS 103 666-2 requirements.....	17
5.3 Test descriptions on ETSI TS 103 999-1 requirements.....	17
5.4 Test descriptions on GP Virtual Primary Platform requirements .....	17
5.5 Test descriptions on ETSI TS 103 834-1 requirements.....	18
5.5.1 TTI underlayers .....	18
5.5.1.1 Configurations.....	18
5.5.1.1.1 CTMQ_001 .....	18
5.5.1.2 Procedures.....	18
5.5.1.3 Test Descriptions.....	19
5.5.1.3.1 TMQ_001 - TTI_UL client/server connection .....	19
5.5.2 TTI Accessor Authentication Service .....	19
5.5.2.1 Configurations.....	19
5.5.2.2 Procedures.....	19
5.5.2.3 Test Descriptions.....	19

5.5.3	TTI Control Service .....	20
5.5.3.1	Configurations.....	20
5.5.3.1.1	CTCS_001 .....	20
5.5.3.1.2	CTCS_002 .....	20
5.5.3.1.3	CTCS_003 .....	21
5.5.3.1.4	ASN.1 Configuration.....	21
5.5.3.2	Procedures.....	23
5.5.3.2.1	PTCS_001 - Open a pipe session with the Identity gate.....	23
5.5.3.2.2	PTCS_002 - Open a pipe session with the Accessor Authentication Service.....	24
5.5.3.2.3	PTCS_003 - Authentication of the root accessor.....	25
5.5.3.2.4	PTCS_004 - Access to the Authentication Service from the root accessor .....	26
5.5.3.2.5	PTCS_005 - Open a pipe session with the Accessor Authentication Service (ROOT) .....	27
5.5.3.2.6	PTCS_006 - Creation of an accessor TTI Control Service Accessor (TTI1) .....	28
5.5.3.2.7	PTCS_007 - Open a pipe session with the Accessor Authentication Service for the TTI Control Service Accessor (TTI1).....	29
5.5.3.2.8	PTCS_008 - Authentication of the accessor (TTI1) .....	29
5.5.3.2.9	PTCS_009 - Access to TTI Control Service with secure pipe (TTI1).....	30
5.5.3.2.10	PTCS_010 - Open a secure pipe session with the TTI Control Service for the TTI1 accessor .....	30
5.5.3.2.11	PTCS_011 - Access to TTI Control Service with secure pipe (ROOT) .....	31
5.5.3.2.12	PTCS_012 - Open a secure pipe session with the TTI Control Service for the root accessor .....	31
5.5.3.3	Test Descriptions.....	32
5.5.3.3.1	TTI-OP-GET-CAPABILITIES .....	32
5.5.3.3.2	TTI-ADMIN-IMPERSONATE.....	33
5.5.3.3.3	TTI-ADMIN-UPDATE-ACL.....	34
5.5.4	TTI Data Service.....	37
5.5.4.1	Configurations.....	37
5.5.4.1.1	CTDS_001 .....	37
5.5.4.2	Procedures.....	38
5.5.4.2.1	PTDS_001 - Access to TTI RDE Service for TTI1 with secure pipe .....	38
5.5.4.2.2	PTDS_002 - Open a secure pipe session with the TTI Data Service for the TTI1 accessor.....	38
5.5.4.3	Test Descriptions.....	39
5.5.4.3.1	TDS_001 - Tunnelling of SCL packets .....	39
5.5.4.3.2	TDS_002 - Tunnelling of SCL packets .....	39
5.5.5	RDE Data Service.....	40
5.5.5.1	Configurations.....	40
5.5.5.1.1	CRDE_001 .....	40
5.5.5.2	Procedures.....	40
5.5.5.2.1	PRDE_001 - Creation of an RDE Service gate .....	40
5.5.5.2.2	PRDE_0002 - Open RDE data service session.....	41
5.5.5.3	Test Descriptions.....	41
5.5.5.3.1	RDE_001 - Extract Time Stamp.....	41
<b>Annex A (informative): References on ETSI forge.....</b>		<b>42</b>
A.1	ETSI forge repository for the TTI test specification .....	42
A.2	License information.....	42
A.3	ASN.1 coding.....	42
<b>Annex B (informative): ASN.1 definition.....</b>		<b>43</b>
B.1	End of ASN.1 .....	43
<b>Annex C (informative): Core specification version information.....</b>		<b>44</b>
<b>Annex D (informative): Change History .....</b>		<b>45</b>
History .....		46

---

# Intellectual Property Rights

## Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

## Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

**DECT™**, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

---

# Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Secure Element Technologies (SET).

The present document is part 2 of a multi-part deliverable covering the Test Tool Interface (TTI) for the Smart Secure Platform (SSP). Full details of the entire series can be found in part 1 [1].

The contents of the present document are subject to continuing work within TC SET and may change following formal TC SET approval. If TC SET modifies the contents of the present document, it will then be republished by ETSI with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 0 early working draft;
  - 1 presented to TC SET for information;
  - 2 presented to TC SET for approval;
  - 3 or greater indicates TC SET approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

---

## Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

---

# 1 Scope

The present document is part of a series of documents specifying the Test Tool Interface (TTI) for the Smart Secure Platform. The present document is the test specification for the Test Tool Interface (TTI) shown in the test environment of ETSI TS 103 999-1 [3], testing the implementations in accordance to ETSI TS 103 834-1 [7].

It contains:

- A list of requirements dedicated to the TTI derived from ETSI TS 103 999-1 [3];
- TTI related requirements from ETSI TS 103 834-1 [7];
- TTI related requirements from ETSI TS 103 666-1 [1] and ETSI TS 103 666-2 [2];
- the principle and requirements for testing the SSP Primary Platform (also known as VPP);
- a selection of test cases to verify the implementation of the TTI independently from the respective manufacturers.

---

## 2 References

### 2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

- In the case of a reference to a TC SET document, a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

Referenced documents which are not found to be publicly available in the expected location might be found at <https://docbox.etsi.org/Reference/>.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

- [1] ETSI TS 103 666-1: "Smart Secure Platform (SSP); Part 1: General characteristics".
- [2] ETSI TS 103 666-2: "Smart Secure Platform (SSP); Part 2: Integrated SSP (iSSP) characteristics".
- [3] ETSI TS 103 999-1: "Smart Secure Platform (SSP); Part 1: Test Specification, general characteristics".
- [4] ETSI TS 103 999-2: "Smart Secure Platform (SSP); Part 2: Integrated SSP (iSSP) characteristics Test Specification".
- [5] GlobalPlatform: "Virtual Primary Platform - Network Protocol" Version 2.0.

NOTE: Available at <https://globalplatform.org/specs-library/globalplatform-technology-virtual-primary-platform/>.

- [6] ETSI TS 102 622: "Smart Cards; UICC - Contactless Front-end (CLF) Interface; Host Controller Interface (HCI)".
- [7] ETSI TS 103 834-1: "Smart Secure Platform (SSP); Part 1: Technical Specification, SSP Test Tool Interface".

## 2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

- In the case of a reference to a TC SET document, a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] Recommendation ITU-T X.680: "Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation".
- [i.2] Recommendation ITU-T X.690: "Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)".

---

## 3 Definition of terms, symbols, abbreviations and formats

### 3.1 Terms

For the purposes of the present document, the terms given in ETSI TS 103 666-1 [1] and ETSI TS 103 999-1 [3] apply.

### 3.2 Symbols

For the purposes of the present document, the symbols given in ETSI TS 103 666-1 [1] and ETSI TS 103 999-1 [3] apply.

### 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI TS 103 666-1 [1], ETSI TS 103 999-1 [3] and the following apply:

AA	Accessor Authentication
FFS	For Further Study
RDE	Router Data Extractor
TT	Test Tool
TTI	Tests Tool Interface



## 3.4 Formats

### 3.4.1 Format of the requirements tables

The columns in the requirement tables in clause 5 have the following meaning.

Column	Meaning
Req.ID	This column shows the ordinal term assigned to a requirement identified in the referenced specification. The following syntax has been used to define the unique R(equirement) terms: R<n><XX><YY>_<ZZ> n: Identification letter for the referenced specification Q: ETSI TS 103 666-1 [1] R: ETSI TS 103 666-2 [2] S: ETSI TS 103 999-1 [3] T: GlobalPlatform, Virtual Primary Platform [5] U: ETSI TS 103 834-1 [7] XX: Main clause of the core specification in which the requirement is listed. YY: Subclause of the main clause in the core specification in which the requirement is listed. ZZZ: Continuously increasing number starting with '001'.
Clause	The "Clause" column helps to identify the location of a requirement by listing the clause hierarchy down to the subclause the requirement is located in.
Release	An optional column that is used if the listed requirement is valid for a specific release or a specific range of releases only, up to a specific release, or from a specific release onwards.
Description	In this column the requirement text is shown. Where the text can either be a copy of the original requirement, or a text analogous to the requirement text (e.g. if the requirement text is descriptive and can be shortened or truncated).

### 3.4.2 Numbers and Strings

The conventions used for decimal numbers, binary numbers and strings are listed in table 3.1.

**Table 3.1: Convention of Numbering and Strings**

Convention	Description
nnnnn	A decimal number, e.g. PIN value or phone number
'b'	A single digit binary number
'bbbbbbb'	An 8-bit binary number
'hh'	A single octet hexadecimal number
'hh hh...hh'	A multi-octet hexadecimal number or string
"SSSS"	A character string
NOTE:	If an 'X' is present in a binary or hexadecimal number, then the digit might have any allowed value. This 'X' value does not need to be interpreted within the particular coding shown.

### 3.4.3 Format of test description clauses

In general clauses with test descriptions use the following basic format:

#### **X.Y. Group of test descriptions for a particular topic**

##### **X.Y.1 Configurations**

This header is to be used in every clause that includes configuration descriptions. It may be followed by a sentence explaining that there are no specific configurations required for this particular topic or:

##### **X.Y.1.1 C<aaa>\_<nnn> <optional>**

Where each sub-header of a required configuration is built from a leading 'C' followed by <aaa>, a minimum three-digit abbreviation for the configuration description group, an underscore and <nnn>, a minimum three-digit number to identify the configuration. This sub-header may include explanatory text following the identification.

Whenever a configuration exists it is presented in a table of the following format:

Configuration ID	C<aaa>_<nnn>
<b>Configuration description</b>	<p><b>Example:</b></p>

Configuration description shall show a drawing representing the entities involved and the connections available between instances. It shall not include explanatory text.

**X.Y.2 Procedures**

This header is to be used in every clause that includes procedure descriptions. It may be followed by a sentence explaining that there are no specific procedures required for this particular topic or:

**X.Y.2.1 P<aaa>\_<nnn> <optional>**

Where each sub-header of a required procedure is built from a leading 'P' followed by <aaa>, a minimum three-digit abbreviation for the procedure description group, an underscore and <nnn>, a minimum three-digit number to identify the procedure. This sub-header may include explanatory text following the identification.

Whenever a procedure exists it is presented in a table of the following format:

<b>Procedure ID</b>	P<aaa>_<nnn>
<b>Procedure objectives</b>	Description of the procedure objectives.
<b>Configuration reference</b>	C<aaa>_<nnn> See note 1
<b>Initial conditions</b>	
Text and/or list of procedure IDs identifying the initial conditions that need to be fulfilled before the procedure sequence defined in this table can be executed. See note 2.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Description of procedure step #1
...	...
n	Description of procedure step #n
NOTE 1: Reference to the appropriate configuration.	
NOTE 2: Procedure IDs can be referenced if the integration of existing procedure sequences can avoid required procedure steps duplication to achieve the initial conditions. Referenced procedures are intended to be executed in given order.	

Procedures are sequences that shall be executed to prepare specific initial conditions for a test. As such they shall not include verifications of any requirements.

**X.Y.3 Test descriptions**

This header is to be used for every clause that includes test descriptions. It may be followed by:

**X.Y.3.1 <aaa>\_<nnn> <optional>**

Where each sub-header of a test description is built from <aaa>, a minimum three-digit abbreviation for the test description group, an underscore and <nnn>, a minimum three-digit number to identify the test description. This sub-header may include explanatory text following the identification.

Whenever a test description exists it is presented in a table of the following format:

<b>Test ID</b>	<aaa>_<nnn>	
<b>Test objectives</b>	Description of the test objectives. See note 1	
<b>Configuration reference</b>	C<aaa>_<nnn> See note 2	
<b>Initial conditions</b>		
Text and/or list of procedure IDs identifying the initial conditions that need to be fulfilled before the test sequence defined in this table can be executed. See note 3.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	Description of test step #1	
...	...	RQ<XX><YY>_<ZZZ>
n	Description of test step #n	
NOTE 1: The descriptions should reflect the objectives of the requirements verified.		
NOTE 2: Reference to the appropriate configuration		
NOTE 3: If possible the initial conditions for the test sequence shall be defined by existing procedures. Referenced procedures are intended to be executed in given order.		

Requirement IDs listed in the Req.ID are references to the requirements listed in clause 5 of the present document. A requirement listed in the test sequence is handled as verified if the response related to the listed requirement has the expected contents. Req.IDs shall always be assigned to a response step.

If there are no test descriptions defined for a group of tests, but related requirements are available, an appropriate clause shall inform about the status of the requirements. E.g.

**X.Y.3.Z Requirements not testable, implicitly verified or verified elsewhere**

The header of this clause shall be adjusted depending on which condition applies for the identified requirements.

**Example text for requirements referenced from a different standardization body:**

The following requirements identified in <XYZ> are not tested in accordance to the present document, as they are referencing requirements from a different standardization body (<NAME>): <XX><YY>\_<ZZZ>, ...

**Example text for requirements implicitly tested:**

The following requirements identified in <XYZ> are generated from descriptive text. An explicit verification is not possible but with correct execution of the related function the requirements can be handled as implicitly verified: <XX><YY>\_<ZZZ>, ...

**Example text for requirements not tested:**

The following requirements identified in <XYZ> are either generated from descriptive text or not testable in the defined test environment. A verification of the listed requirements is not possible: <XX><YY>\_<ZZZ>, ...

The clause with explanatory text for the untested or implicitly tested requirements shall always be the last clause in the Test description clause. Nevertheless, it can be provided as the first clause if no executable test sequences are defined.

The hierarchy given in this example structure is not fixed. If building sub-groups is useful this may be done on any level of the test description hierarchy. Furthermore, it is not required to generate sub-groups for all the three main sections (Configurations, Procedures, Test descriptions) if adding a sub-group is useful in any of these sections.

E.g. common configurations on hierarchy level 3, common procedures on hierarchy level 3 but subgroups for the test descriptions with a new group header on level 4 and the test descriptions on level 5.

### 3.4.4 Dynamic content validation in ASN.1 structure

In certain test cases a dynamic content returned by the DUT (e.g. value within ASN.1 structure, signature, integer, ...) is processed according to the following textX grammar:

```
operations ::= '<' operation ( logical_operator operation)* '>'
operation ::= operation_Identifier ' (' variable_Identifier (',' parameter)* ') '
operation_Identifier ::= 'STORE' | 'REPLACE' | 'COMPARE' | 'ISFIELDNOTEXIST'
logical_operator ::= 'AND' | 'OR' | 'XOR'
variable_Identifier ::= ([A-Z]|[a-z])+[0-9]*
```

where:

- Operation\_Identifier: is the identifier identifying the operation to perform on a dynamic content of aFieldName as:
  - STORE: store the dynamic content of aFieldName into a test tool variable identified by a variable identifier.
  - REPLACE: retrieve a variable identified by Variable\_Identifier and replace the content of aFieldName by the content of the variable.
  - COMPARE: compare the content of aFieldName with the content of a variable and return True or False to the test tool. This operator requires one or more additional parameters. The parameters may be combined for ORing them. The parameters are as follow:
    - GT: the content of the aFieldName shall be strictly greater than the content of a variable.
    - LS: the content of the aFieldName shall be strictly less than the content of a variable.
    - EQ: the content of the aFieldName shall be equal to the content of a variable.
    - DIF: the content of the aFieldName shall be different from the content of a variable.
  - ISFIELDNOTEXIST: return true, if aFieldName field does not exist.
- Variable\_Identifier: variable identifier managed by the test tool. The variable identifier shall consist only of a set of alphanumeric characters.

The operations are inserted within a comment associated to a field as follow:

```
aFieldName ... /* operations */
```

EXAMPLE:

```
aParameter {
  aVersion '0000'H /*<COMPARE(TCSVERSION,GT,EQ)>*/where
aTCSVERSION INTEGER ::= 10 /*<STORE(TCSVERSION)>
```

## 4 Requirements for the TTI

### 4.0 Introduction

As the TTI shall be usable in test environments as defined in ETSI TS 103 999-1 [3], requirements from various specifications shall be considered. Requirements listed in the following clauses are derived from:

- ETSI TS 103 666-1 [1],

- ETSI TS103 666-2 [2]
- ETSI TS 103 999-1 [3],
- GlobalPlatform, Technology, Virtual Primary Platform [5].
- ETSI TS 103 834-1 [7].

NOTE: To avoid ambiguities in requirements derived from the GlobalPlatform - Virtual Primary Platform [5] specification, the requirement tables use normative text rather than the reference as in the underlying ETSI specification(s).

## 4.1 Requirements from ETSI TS 103 666-1

### 4.1.1 Accessor authentication

Reference: ETSI TS 103 666-1 [1], clause 6.13.

RQ number	Clause	Description
<b>6.13.1 Overview</b>		
RQ0613_001	6.13.1	For the accessor authentication service in the TTI Host in general the accessor authentication service requirements listed in ETSI TS 103 999-1 [3], clause 5.2.13 apply.

### 4.1.2 Protocol layers

Reference: ETSI TS 103 666-1 [1], clause 8.3.

RQ number	Clause	Description
<b>8.3.1 Overview</b>		
RQ0803_001	8.3.1	The MTU shall be 20 bytes or greater.
RQ0803_002	8.3.1	For proper operation, the protocol stack underlying the SCL shall provide a means for managing the underlying flow control.
RQ0803_003	8.3.1	There shall be an optional means for controlling (e.g. activating, deactivating) the underlying protocols.
RQ0803_004	8.3.1	There shall be an optional means for getting the notifications from an underlying protocol (e.g. activation/deactivation of the interface by the terminal).
<b>8.3.2 Network layer</b>		
RQ0803_005	8.3.2	The TTI shall support the exchange of VPN packets as defined in GlobalPlatform, Virtual Primary Platform - Network Protocol [5], clause 4.3.1.
RQ0803_006	8.3.2	UUIDs as defined in GlobalPlatform, Virtual Primary Platform - Network Protocol [5], clause 3.3 shall be used as identifiers for Hosts, Gates and Host Domains to fulfil the requirements to a VPN packet as requested in clause 4.3.1 of [5].
<b>8.3.3 Transport layer</b>		
RQ0803_009	8.3.3	The TTI shall support the routing of HCP packets as defined in GlobalPlatform, Virtual Primary Platform - Network Protocol [5], clause 4.4.
RQ0803_010	8.3.3	For routing of HCP packets using static pipes within the destination host, pipes with pipe identifiers (P <sub>ID</sub> ) as defined in table 3-5 of GlobalPlatform, Virtual Primary Platform - Network Protocol [5] shall be used.
RQ0803_011	8.3.3	For routing of HCP packets using dynamic pipes within the destination host, pipes with pipe identifiers (P <sub>ID</sub> ) as defined in clause 5.5.3.2 of GlobalPlatform, Virtual Primary Platform - Network Protocol [5] shall be used.
RQ0803_012	8.3.3	Fragmentation and reassembly of messages larger than the MTU shall be possible as defined in ETSI TS 102 622 [6], clause 5.3.
RQ0803_013	8.3.3	Dedicated Protocol Accommodations as defined in clause 4.6 of GlobalPlatform, Virtual Primary Platform - Network Protocol [5] shall be supported.
<b>8.3.4 Session layer</b>		
RQ0803_014	8.3.4	The TTI shall support a session layer using pipe sessions as defined in GlobalPlatform, Virtual Primary Platform - Network Protocol [5], clause 3.10.
RQ0803_015	8.3.4	The TTI shall be capable to execute a successful pipe binding operation to open a pipe session as defined in GlobalPlatform, Virtual Primary Platform - Network Protocol [5], clause 5.5.3.2.
RQ0803_016	8.3.4	The TTI shall be capable to explicitly close a pipe session as defined in GlobalPlatform, Virtual Primary Platform - Network Protocol [5], clauses 5.5.3.3 and 3.10.3.

## 4.2 Requirements from ETSI TS 103 666-2

### 4.2.1 Runtime model

Reference: ETSI TS 103 666-2 [2], clause 6.6.

RQ number	Clause	Description
RR0606_001	6.6	The TTI shall be extensible for monitoring signals and exceptions from GlobalPlatform, Virtual Primary Platform - Network Protocol [5], clause 5.6 (see note).
NOTE: The requirement is adjusted to TTI specific requirements.		

NOTE: As currently no test specification for the GP VPP is available, no TTI specific requirements, not yet covered by testing, can be identified. This is subject to change as soon as related GlobalPlatform documents are made available.

## 4.3 Requirements from ETSI TS 103 999-1

### 4.3.1 Test Tool Interface requirements

Reference: ETSI TS 103 999-1 [3], clause 4.2.3.

RQ number	Clause	Description
RS0402_001	4.2.3	The test tool interface plugged into the router shall copy all SCL packets routed by the router into it, excluding the SCL packets from the SCL analyser.
RS0402_002	4.2.3	The test tool interface plugged into the router shall timestamp the copy of the SCL packet.
RS0402_003	4.2.3	It shall be possible to disable the identification of the host issuing an SCL packet to impersonate it.
RS0402_004	4.2.3	It shall be possible to impersonate a host domain by a directive to the router.
RS0402_005	4.2.3	It shall be possible to collect events related to the SPB management (e.g. termination, exceptions).
RS0402_006	4.2.3	TTI shall be able to grant or prohibit the access to hosts and services within hosts of the SUT from the TT host according to the services under tests.

## 4.4 Requirements from GlobalPlatform - Virtual Primary Platform

### 4.4.1 FFS

Reference: GlobalPlatform Virtual Primary Platform [5], clause FFS.

RQ number	Clause	Description
RTxxxx_001	FFS	Requirement description from GlobalPlatform

NOTE: Requirements from GlobalPlatform, relevant for the ETSI TTI are expected to be listed in ETSI TS 103 666-2 [2]. Whenever possible requirements from ETSI TS 103 666-2 [2] will be used to avoid trademark and copyright issues.

## 4.5 Requirements from ETSI TS 103 834-1

### 4.5.1 TTI MQTT requirements

Reference: ETSI TS 103 834-1 [7], clause 5.2.

RQ number	Clause	Description
<b>5.2.3 TTI underlayer server</b>		
RU0502_001	5.2.3	The terminal shall be able to connect a MQTT client to an agreed MQTT broker.
RU0502_002	5.2.3	The MQTT connection shall use TLS with a mutual authentication of the broker and the client.
RU0502_003	5.2.4	The MQTT client of the TT shall be able to transfer the TTI_UL URL to the MQTT client of the terminal.
RU0502_004	5.2.5	The MQTT clients shall be able to provision the broker cross certificates.

### 4.5.2 TTI Accessor Authentication Service requirements

Reference: ETSI TS 103 834-1 [7], clause 6.1.

RQ number	Clause	Description
<b>6.1.1 Requirements</b>		
RU0601_001	6.1.1	The TTI Accessor Authentication Service shall operate similar to the Accessor Authentication Service in the SSP, defined in ETSI TS 103 666-1 [1], clause 6.13. Where the TT accessor becomes a member of the accessors allowed to use of the service within a TTI Host outside the SSP host domain.
<b>6.1.2 Test Tool (TT) Accessor Authentication</b>		
RU0601_002	6.1.2	The TT Accessor Authentication supported by the Accessor Authentication Service within the TTI Host shall support the protocol as defined in ETSI TS 103 666-1 [1], clause 9.4.
<b>6.1.3 Access to the TTI Host Services</b>		
RU0601_003	6.1.3	The access of the TTI Host Services as defined in clause 5.4 of the present document are granted by using the AAS-OP-ACCESS-SERVICE-Service-Command as defined in ETSI TS 103 666-1 [1], clause 6.13.5.6.

### 4.5.3 TTI Control Service requirements

Reference: ETSI TS 103 834-1 [7], clause 6.2.

RQ number	Clause	Description
<b>6.2.1 Overview</b>		
RU0602_001	6.2.1	The TTI Control Service supports retrieving the capabilities of the TTI Control Service (i.e. aTTI-OP-GET-CAPABILITIES-Service-Command).
RU0602_002	6.2.1	If the capabilities can be granted, the TTI Control Service shall send a response including an eTTI-OK.
RU0602_003	6.2.1	The TTI Control Service supports updating the ds of a TTI Data Service related to the Router Data Extractor in charge to record the SCL packet traffic.
RU0602_004	6.2.1	With the command aTTI-ADMIN-IMPERSONNATE-Service-Command, a TTI Control Service Application shall be able to impersonate a host.
RU0602_005	6.2.1	If the TTI Control Service Application can impersonate a host, the TTI Control Service shall send a response including an eTTI-OK.
RU0602_006	6.2.1	With the command aTTI-ADMIN-UPDATE-ACL-Service-Command, a TTI Control Service Application shall be able to update the ACL of the TTI control service.
RU0602_007	6.2.1	If the TTI Control Service Application can update the ACL of the TTI control service, the TTI Control Service shall send a response including an eTTI-OK.

## 4.5.4 TTI Data Service requirements

Reference: ETSI TS 103 834-1 [7], clause 6.3.

RQ number	Clause	Description
<b>6.3.1 Overview</b>		
RU0603_001	6.3.1	This service is in charge to tunnel a SCL packets traffic from/to a TT Host

## 4.5.5 RDE Data Service requirements

Reference: ETSI TS 103 834-1 [7], clause 6.4.

RQ number	Clause	Description
<b>6.4.1 Overview</b>		
RU0604_001	6.4.1	Filter the copied SCL packet flowing within the SCL network associate to the Router 2 (SUT) according to the rights associated to the TT accessor
RU0604_002	6.4.1	Timestamp the taped SCL packets and encapsulate them into a RDE_EVT event

## 4.6 ASN.1 syntax

### 4.6.1 Introduction

The description of some data objects in the present document is based on ASN.1 specified in Recommendation ITU-T X.680 [i.1] and encoded in TLV structures using Distinguished Encoding Rule (DER) encoding as specified in Recommendation ITU-T X.690 [i.2]. This provides a flexible description of those data objects. The complete ASN.1 code is divided into a number of ASN.1 sections in the specifications. In order to facilitate the extraction of the complete ASN.1 code from the specification, each ASN.1 section begins with a text paragraph consisting entirely of an ASN.1 start tag, which consists of a double hyphen followed by a single space and the text string "ASN1START" (in all upper-case letters). Each ASN.1 section ends with a text paragraph consisting entirely of an ASN.1 stop tag, which consists of a double hyphen followed by a single space and the text "ASN1STOP" (in all upper-case letters).

The complete ASN.1 code may be extracted by copying all the text paragraphs between an ASN.1 start tag and the following ASN.1 stop tag in the order they appear, throughout the present document.

### 4.6.2 Start of ASN.1

```
-- ASN1START

TTITestsDefinitions { itu-t (0) identified-organization (4) etsi (0) smart-secure-platform (3834)
part1 (2) }
DEFINITIONS
AUTOMATIC TAGS
EXTENSIBILITY IMPLIED ::=
BEGIN

EXPORTS ALL;
/* Imports */
IMPORTS
    AccessMode,
    UUID,
    SessionID,
    AccessorRights,
    AccessControl,
    AccessorConditionsPIN,
    AccessorConditions,
    AAS-SERVICE-GATE-Commands,
    AAS-SERVICE-GATE-Responses,
    TTI-CONTROL-SERVICE-GATE-Commands,
    TTI-CONTROL-SERVICE-GATE-Responses,
    Certificate,
    AuthenticationToken,
    Version
FROM SSPDefinitions
    ECDSA-Sig-Value,
```



```
id-ecPublicKey
FROM PKIX1Algorithms88;
-- ASN1STOP
```

NOTE: The ASN.1 code is ended in annex B.

---

## 5 Test descriptions for TTI implementations

### 5.1 Test descriptions on ETSI TS 103 666-1 requirements

#### 5.1.1 Accessor authentication

The Accessor authentication is tested according to clause 5.5.2 of the present document.

#### 5.1.2 Protocol layers

Protocol layer requirements listed in clause 4.1.2 of the present document do not require TTI specific test descriptions. It is suggested to run appropriate tests from ETSI TS 103 999-1 [3] with running the TTI as defined in the present document.

### 5.2 Test descriptions on ETSI TS 103 666-2 requirements

The test descriptions on the ETSI TS 103 666-2 [2] are covered by the clause 5.4.

### 5.3 Test descriptions on ETSI TS 103 999-1 requirements

The requirements are covered by the test descriptions defined in the clause 5.5.

### 5.4 Test descriptions on GP Virtual Primary Platform requirements

FFS (see note).

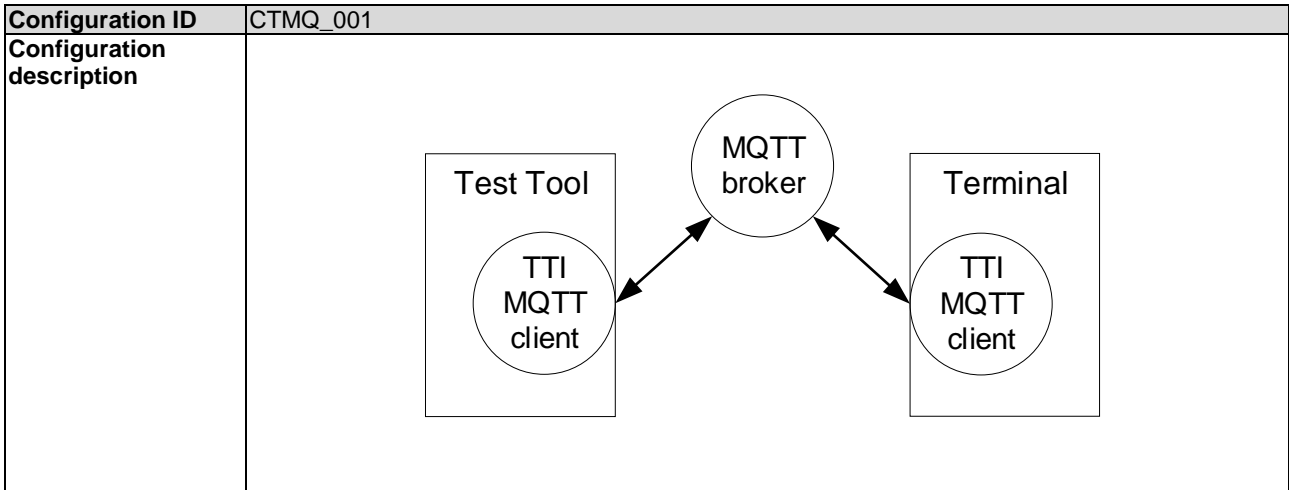
NOTE: Test descriptions from GlobalPlatform, relevant for the ETSI TTI are expected to be tested in accordance to GlobalPlatform specifications. As soon as GlobalPlatform test descriptions are available appropriate references should be added here.

## 5.5 Test descriptions on ETSI TS 103 834-1 requirements

### 5.5.1 TTI underlayers

#### 5.5.1.1 Configurations

##### 5.5.1.1.1 CTMQ\_001



#### 5.5.1.2 Procedures

##### 5.5.1.2.1 PTMQ\_001 - Retrieving of the MQTT credentials

<b>Procedure ID</b>	PTMQ_001
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the Identity gate of the TTI host. From the GATE_LIST registry, the UUID of the root accessor shall be listed. If the procedure is successful, then a pipe session is open between the Identity Application in the other host and the Identity Service in the TTI host.
<b>Configuration reference</b>	CTMQ_001
<b>Initial conditions</b>	
None	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Both MQTT clients (TT and terminal) collect their certificate and other credentials according to the definition in clause 5.6 of ETSI TS 103 834-1 [7].

### 5.5.1.3 Test Descriptions

#### 5.5.1.3.1 TMQ\_001 - TTI\_UL client/server connection

<b>Test ID</b>	TMQ_001	
<b>Test objectives</b>	The MQTT client of the TT shall be able to transfer the URL of its TTI_UL server to the MQTT client of the terminal.	
<b>Configuration reference</b>	CTMQ_001	
<b>Initial conditions</b>		
The procedure PTMQ_001 shall be successfully executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	The MQTT client of the TT executes the procedure defined in annex E of ETSI TS 103 834-1 [7].	RU0502_001 RU0502_002 RU0502_003
2	The TTI_UL client in the terminal connects the TTI_UL server in the TT.	RU0502_004 RQ0803_001 RQ0803_002 RQ0803_003 RQ0803_004

## 5.5.2 TTI Accessor Authentication Service

### 5.5.2.1 Configurations

The configurations defined in ETSI TS 103 999-1 [3], clause 6.13.1 apply with the following exceptions:

- 1) SSP Host Domain is replaced by TTI Host Domain.
- 2) SSP Host is replaced by TTI Host.

### 5.5.2.2 Procedures

The procedures defined in ETSI TS 103 999-1 [3], clause 6.13.2 apply with the following exceptions:

- 1) SSP is replaced by TTI.
- 2) SSP Host is replaced by TTI Host.

### 5.5.2.3 Test Descriptions

The test descriptions defined in ETSI TS 103 999-1 [3], clause 6.13.3 apply with the following exceptions:

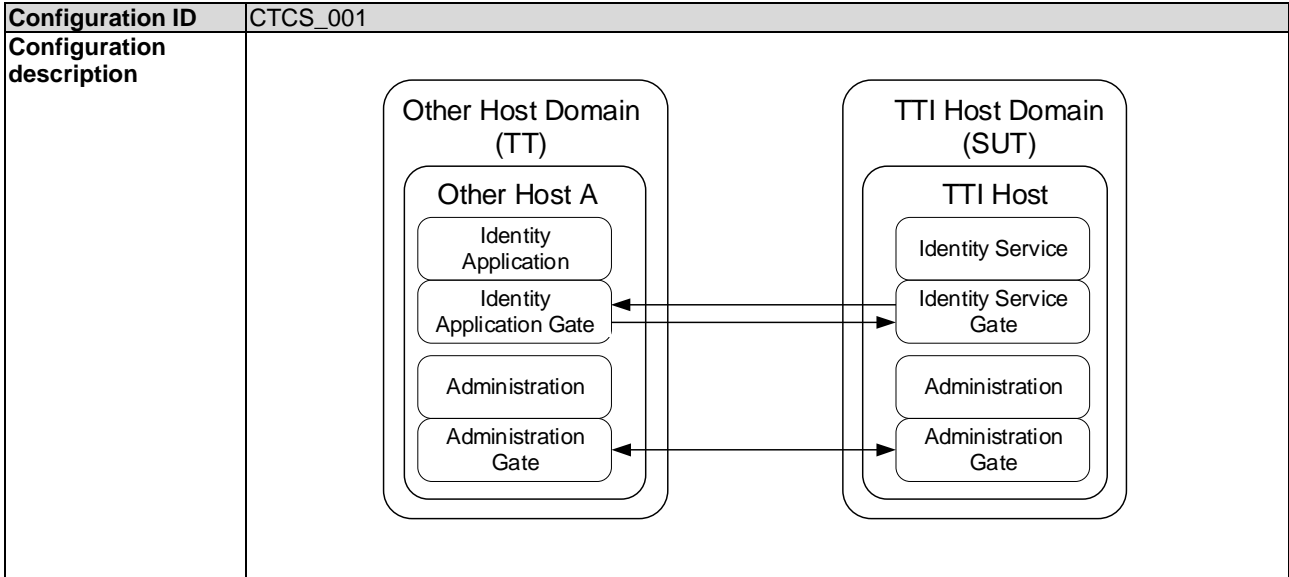
- 1) SSP is replaced by TTI.
- 2) SSP Host is replaced by TTI Host.

The test descriptions support the requirement RQ0613\_001, RU0601\_001, RU0601\_002, RU0601\_003.

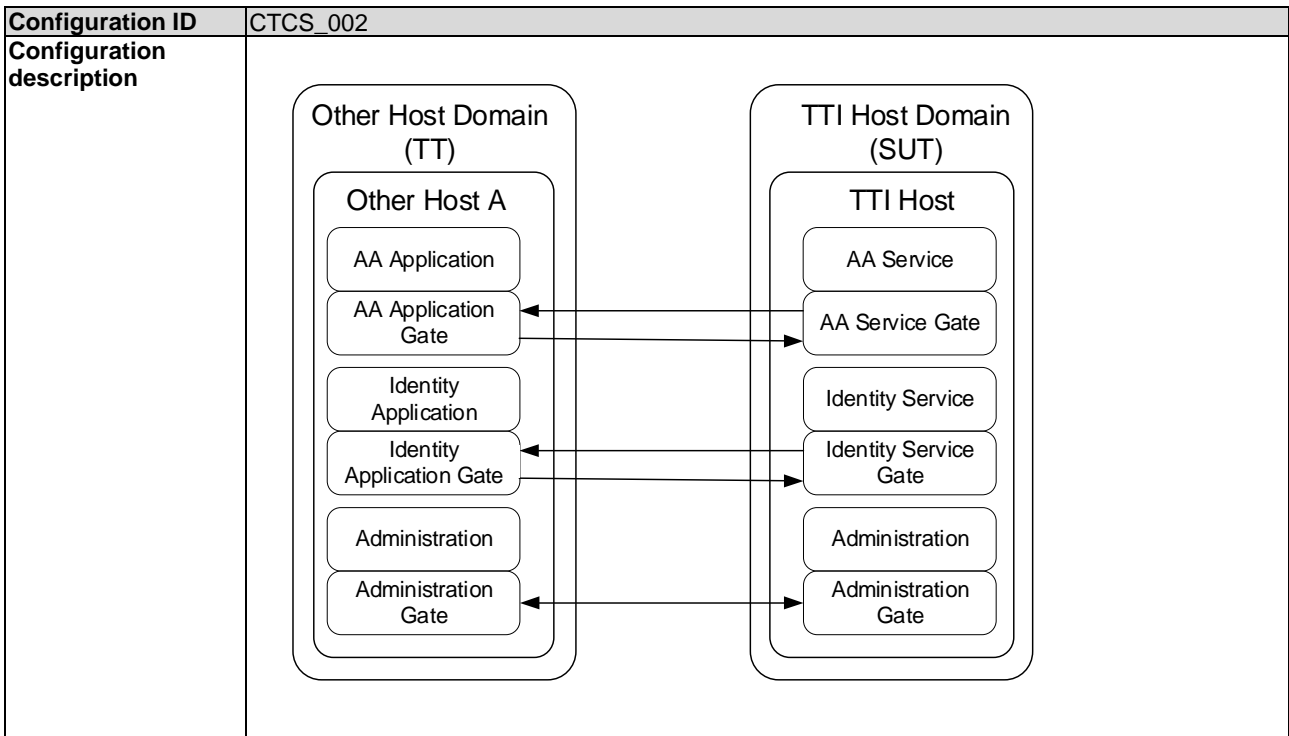
### 5.5.3 TTI Control Service

#### 5.5.3.1 Configurations

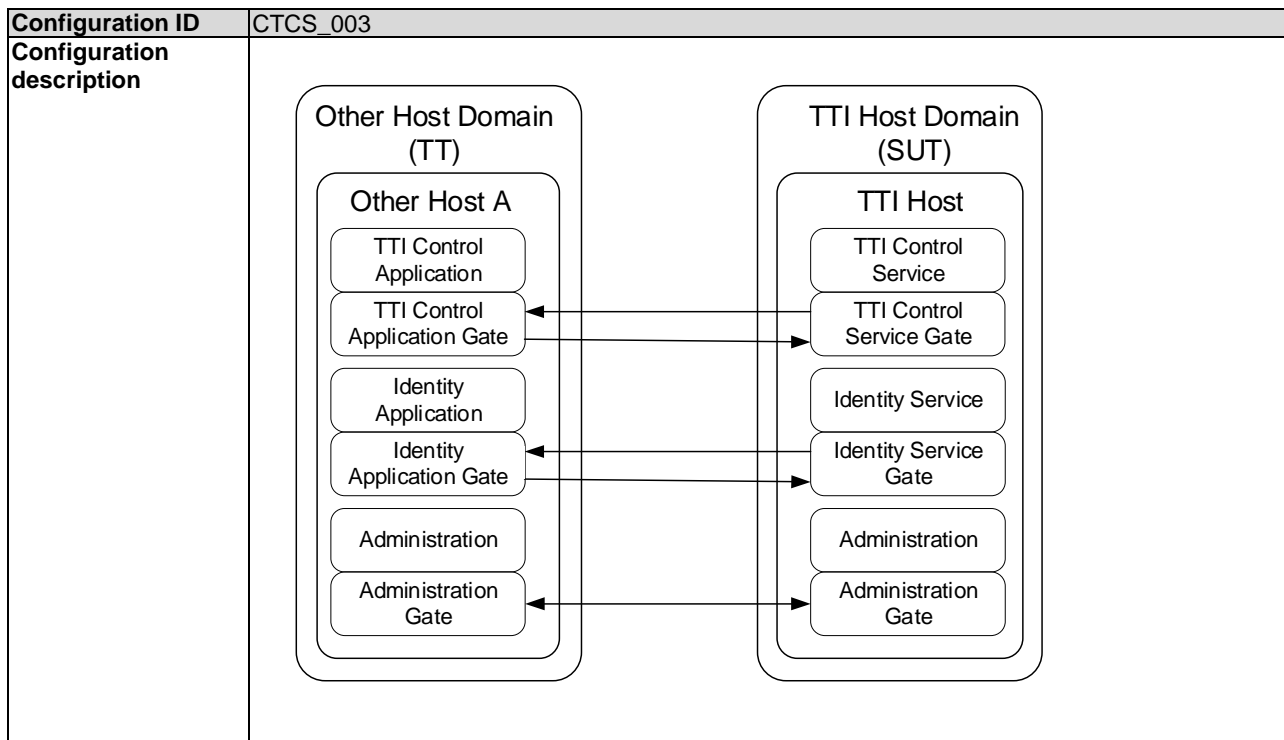
##### 5.5.3.1.1 CTCS\_001



##### 5.5.3.1.2 CTCS\_002



5.5.3.1.3 CTCS\_003



5.5.3.1.4 ASN.1 Configuration

The following configuration is used for the procedures and the test descriptions.

```

-- ASN1START

eTTI-ID-CS          UUID::='09560B78BED958B9A5FF6CAA8384D556'H -- ETSI TTI Control Service identifier

-- urn:etsi.org:asn.1:accessor:tti
eTTI-ACC           UUID::='0F26EB7CEF785F8E84FCB64C9284638E'H
eTTI-ACC-ROOT     UUID::='DD61116FF0DD57F48A4F52EE70276F24'H
eAS-ID-ASS-GateID_1 UUID::='AAAAAAAABBBBCCCCDDDDDEEEEEEEEEEEEEEE'H
eAS-ID-ASS-GateID_2 UUID::='AAAAAAAABBBBCCCCDDDDDEEEEEEEEEEEEEEE'H
eAS-Challenge     UUID::='BA64E9EE888952F4891DA79401758FF4'H
-- urn:etsi.org:test:firmware-01
eTTI-FFI         UUID::='FC4B3800094D523A9A5AD2E9FDFBCEEC'H

--eAASAccessRight-RequiresSecurePipe AccessorRights ::=      { eRight-Bit1 }
--eAASAccessRight-Create AccessorRights ::=                    { eRight-Bit2 }
--eAASAccessRight-Delete AccessorRights ::=                    { eRight-Bit3 }
--eAASAccessRight-Update AccessorRights ::=                    { eRight-Bit4 }
--eAASAccessRight-UpdateACL AccessorRights ::=                 { eRight-Bit5 }
--eAASAccessRight-UpdateGroup AccessorRights ::=               { eRight-Bit6 }
--eAASAccessRight-UpdateCredentialPolicy AccessorRights ::=   { eRight-Bit7 }
--eAASAccessRight-UpdateCredentialStatus AccessorRights ::=   { eRight-Bit8 }

-- The root accessor has all accessor rights

eTTI-ACL-ROOT     AccessorRights ::= {
--eAASAccessRight-RequiresSecurePipe--      eRight-Bit1,
--eAASAccessRight-Create AccessorRights--    eRight-Bit2,
--eAASAccessRight-Delete--                  eRight-Bit3,
--eAASAccessRight-Update AccessorRights--    eRight-Bit4,
--eAASAccessRight-UpdateACL--                eRight-Bit5,
--eAASAccessRight-UpdateGroup--              eRight-Bit6,
--eAASAccessRight-UpdateCredentialPolicy--    eRight-Bit7,
--eAASAccessRight-UpdateCredentialStatus--    eRight-Bit8
}

eTTI-ACL          AccessorRights ::= {
--eAASAccessRight-RequiresSecurePipe--      eRight-Bit1,
--eAASAccessRight-Create AccessorRights--    eRight-Bit2,

```

```

--eAASAccessRight-Delete-- eRight-Bit3,
--eAASAccessRight-Update AccessorRights-- eRight-Bit4,
--eAASAccessRight-UpdateACL-- eRight-Bit5,
--eAASAccessRight-UpdateGroup-- eRight-Bit6,
--eAASAccessRight-UpdateCredentialPolicy-- eRight-Bit7,
--eAASAccessRight-UpdateCredentialStatus-- eRight-Bit8
}
eTTI-CS-ACL AccessorRights ::= {
--eTTIAccessRight-APDUGateAccessAllowed -- eRight-Bit5
}

-- ASN1STOP

```

The Authentication token and certificates are given as examples. Tools are available in the ETSI forge repository in [\[SCP x509v3\]](#) to generate the needed certificates for creating the certification path.

```

-- ASN1START

eAS-ATK-01 AuthenticationToken ::= {
  tbsToken {
    version v1,
    subjectPublicKeyInfo {
      algorithm {
        algorithm { 0 0 }
      },
      subjectPublicKey '0'B
    },
    aATK-Content {
      aChallenge '00000000000000000000000000000000'H,
      aKey-Size e128,
      aStreamCipherIdentifier aAES-CGM-StreamCipherIdentifier
    }
  },
  signatureAlgorithm {
    algorithm { 0 0 }
  },
  signature {
    r 0,
    s 0
  }
}

eAS-CERT-01 Certificate ::= {
  tbsCertificate {
    version v3,
    serialNumber 1,
    signature {
      algorithm { 0 0 },
      parameters OCTET STRING : '00'H
    },
    issuer rdnSequence : {
      {
        {
          type { 0 0 },
          value OCTET STRING : '00'H
        }
      }
    },
    validity {
      notBefore utcTime : "000101000000Z",
      notAfter utcTime : "000101000000Z"
    },
    subject rdnSequence : {
      {
        {
          type { 0 0 },
          value OCTET STRING : '00'H
        }
      }
    },
    subjectPublicKeyInfo {
      algorithm {
        algorithm id-ecPublicKey
      },
      subjectPublicKey '0'B
    },
    issuerUniqueID '0'B,
    subjectUniqueID '0'B,
    extensions {

```

```

    {
      extnID { 0 0 },
      critical FALSE,
      extnValue '00'H
    }
  },
  signatureAlgorithm {
    algorithm { 0 0 },
    parameters OCTET STRING : '00'H
  },
  signature '0'B
}
-- ASN1STOP

```

## 5.5.3.2 Procedures

### 5.5.3.2.1 PTCS\_001 - Open a pipe session with the Identity gate

<b>Procedure ID</b>	PTCS_001
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the Identity gate of the TTI host. From the GATE_LIST registry, the UUID of the root accessor shall be listed. If the procedure is successful, then a pipe session is open between the identity application in the other host and the Identity Service in the TTI host.
<b>Configuration reference</b>	CTCS_001
<b>Initial conditions</b>	
Root accessor (UUID: DD61116F-F0DD-57F4-8A4F-52EE70276F24) is existing. The root accessor is available in TTI prepared for procedures purpose. The TT acting as an accessor shall be able to be authenticated by using an authentication token authenticated by a certification path.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate sends EVT_ADM_BIND to Administration gate in the TTI with: <ul style="list-style-type: none"> <li>PIPE<sub>XY</sub>: a dynamically assigned pipe identifier for the Identity Service gate,</li> <li>GATE<sub>IDENTITY</sub>: the UUID gate identifier of the Identity gate (416B66AC-A134-5082-8160-FA1BA497F917).</li> </ul>
2	Administration gate sends EVT_ADM_BIND to Administration gate in the other host with: <ul style="list-style-type: none"> <li>PIPE<sub>YX</sub>: a dynamically assigned pipe identifier for the Identity Application gate,</li> <li>GATE<sub>IDENTITY</sub>: the UUID gate identifier of the Identity gate (416B66AC-A134-5082-8160-FA1BA497F917).</li> </ul>
3	Identity Application gate sends ANY_GET_PARAMETER command (pipe PIPE <sub>XY</sub> ) to the Identity Service gate in the TTI host with the register '04'H.
4	Identity Service gate sends ANY_GET_PARAMETER response (pipe PIPE <sub>YX</sub> ) to the Identity Application gate in the other host. The service identifier 'DD61116F-F0DD-57F4-8A4F-52EE70276F24' shall be present. The procedure is successful if the previous requirement is satisfied.
5	Administration gate sends EVT_ADM_UNBIND event to the Administration gate in the TTI host with: <ul style="list-style-type: none"> <li>PIPE<sub>XY</sub>: a dynamically assigned pipe identifier for the Identity Service gate.</li> </ul> The pipe session between the Identity Application gate and the Identity Service gate is closed. This step is required to clean up the context of the tests but it is not essential for the procedure objective.

## 5.5.3.2.2 PTCS\_002 - Open a pipe session with the Accessor Authentication Service

<b>Procedure ID</b>	PTCS_002
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the Authentication Service gate of the TTI host. If the procedure is successful then a pipe session is open between the Accessor Authentication Application in the other host and the Accessor Authentication Service in the TTI host.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
Root accessor (UUID: DD61116F-F0DD-57F4-8A4F-52EE70276F24) is existing. This UUID is also the identity of the Root accessor. This root accessor is dedicated for the TT and assigned to the test providers using the ETSI TTI tests. The procedure PTCS_001 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate sends EVT_ADM_BIND to Administration gate in the TTI with: <ul style="list-style-type: none"> <li>• PIPE<sub>BA</sub>: a dynamically assigned pipe identifier for the Accessor Authentication Service gate.</li> <li>• GATE<sub>ROOT</sub>: the UUID gate identifier of the root Accessor Authentication Service gate (DD61116F-F0DD-57F4-8A4F-52EE70276F24).</li> </ul>
2	Administration gate sends EVT_ADM_BIND to Administration gate in the other host with a binding parameter equal to: <ul style="list-style-type: none"> <li>• PIPE<sub>AB</sub>: a dynamically assigned pipe identifier for the Accessor Authentication Application gate.</li> <li>• GATE<sub>ROOT</sub>: the UUID gate identifier of the root Accessor Authentication Application gate (DD61116F-F0DD-57F4-8A4F-52EE70276F24).</li> </ul> GATEROOT shall be present in one of the binding parameters (see VNP [5]. If present then the procedure is successful.



## 5.5.3.2.3 PTCS\_003 - Authentication of the root accessor

<b>Procedure ID</b>	PTCS_003
<b>Procedure objectives</b>	The root accessor shall be able to be authenticated with the Accessor Authentication Service by using: <ul style="list-style-type: none"> <li>• The aAAS-OP-GET-CHALLENGE-Service-Command command.</li> <li>• The aAAS-OP-AUTHENTICATE-ACCESSOR-Service-Command command.</li> </ul> The authentication mean is based on the authentication tokens.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_002 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	AAA gate sends an AAS-SERVICE-GATE-Commands command to AAS gate with: -- ASN1START <pre>aPTCS-003-command-01 AAS-SERVICE-GATE-Commands ::= aAAS-OP-GET-CHALLENGE-Service-Command : { }</pre> -- ASN1STOP
2	AAS gate sends AAS-SERVICE-GATE-Responses response to AAA gate with: -- ASN1START <pre>aPTCS-003-response-01 AAS-SERVICE-GATE-Responses ::= aAAS-OP-GET-CHALLENGE-Service-Response : {   aAAS-Service-Response eAAS-OK,   aParameter {     aChallenge eAS-Challenge,     aCertificates {eAS-CERT-01}   } }</pre> -- ASN1STOP aCertificate is a set of certificates. aChallenge is a random number (128 bit) generated by the AAS. The value expressed in the procedure is given as example.
3	AAA gate sends AAS-SERVICE-GATE-Commands command to AAS gate with: -- ASN1START <pre>aPTCS-003-command-02 AAS-SERVICE-GATE-Commands ::= aAAS-OP-AUTHENTICATE-ACCESSOR-Service- Command : {   aCredential aAccessorTokenCredential : {     aToken eAS-ATK-01, aTokenCertificationPath {eAS-CERT-01}   } }</pre> -- ASN1STOP The authentication token shall contain the challenge as recovered at the step 2. The authentication token shall be verified by using the certification path.
4	AAS gate sends AAS-SERVICE-GATE-Responses response to AAA gate with: -- ASN1START <pre>aPTCS-003-response-02 AAS-SERVICE-GATE-Commands ::= aAAS-OP-AUTHENTICATE-ACCESSOR-Service- Command : {   aCredential aAccessorTokenCredential : {     aToken eAS-ATK-01, aTokenCertificationPath {eAS-CERT-01}   } }</pre> -- ASN1STOP The authentication token shall contain the challenge as recovered at the step 2. The authentication token shall be verified by using the certification path. The procedure is successful if the same challenge is in all authentication tokens and all of them have been verified by their certification path.

## 5.5.3.2.4 PTCS\_004 - Access to the Authentication Service from the root accessor

<b>Procedure ID</b>	PTCS_004
<b>Procedure objectives</b>	The authenticated root accessor shall be able to access the Accessor Authentication Service by using: <ul style="list-style-type: none"> <li>The aAAS-OP-ACCESS-SERVICE-Service-Command command.</li> </ul> If the procedure is successful then the accessor can open a secure pipe session to the Accessor Authentication Service.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_003 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	AAA gate sends an AAS-OP-ACCESS-SERVICE-Service-Command command to AAS gate with: -- ASN1START <pre>aPTCS-004-command-01 AAS-SERVICE-GATE-Commands ::= aAAS-OP-ACCESS-SERVICE-Service-Command : {   aServiceIdentifier 'DD61116FF0DD57F48A4F52EE70276F24 'H,   aUseSecurePipe TRUE } -- ASN1STOP</pre>
2	AAS gate sends an AAS-OP-ACCESS-SERVICE-Service-Response response to AAA gate with: -- ASN1START <pre>aPTCS-004-response-01 AAS-SERVICE-GATE-Responses ::= aAAS-OP-ACCESS-SERVICE-Service-Response : {   aAAS-Service-Response eAAS-OK,   aParameter {     aGateIdentifier eAS-ID-ASS-GateID_1 /* &lt;STORE(eAS-ID-ASS-GateID_1)&gt; */   } } -- ASN1STOP</pre> The AAS returns the gate identifier on which the authenticated root accessor can access the Accessor Authentication Service by using a secure pipe. The procedure is successful if the AAS returns eAAS-OK.

### 5.5.3.2.5 PTCS\_005 - Open a pipe session with the Accessor Authentication Service (ROOT)

<b>Procedure ID</b>	PTCS_005
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the Authentication Service gate of the TTI host. If the procedure is successful then a pipe session is open between the Accessor Authentication Application in the other host and the Accessor Authentication Service in the TTI host.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
Root accessor (UUID: DD61116F-F0DD-57F4-8A4F-52EE70276F24) is existing. This UUID is also the identity of the Root accessor. This root accessor is dedicated for the TT and assigned to the test providers using the ETSI TTI tests. The accessor has obtained the gate identifier on the Accessor Authentication Service for the root accessor by using a secure pipe session. The procedure PTCS_004 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate sends EVT_ADM_BIND to Administration gate in the TTI with: <ul style="list-style-type: none"> <li>• PIPE<sub>CD</sub>: a dynamically assigned pipe identifier for the Accessor Authentication Service Gate,</li> <li>• GATE<sub>ROOTBIS</sub>: The dynamically assigned UUID gate identifier returned by AAS in PTCS_004 (eAS-ID-ASS-GateID_1).</li> </ul>
2	Administration gate sends EVT_ADM_BIND to Administration gate in the other host with a binding parameter equal to: <ul style="list-style-type: none"> <li>• PIPE<sub>DC</sub>: a dynamically assigned pipe identifier for the Accessor Authentication Application Gate,</li> <li>• GATE<sub>ROOTBIS</sub>: The dynamically assigned UUID gate identifier returned by AAS in PTCS_004 (eAS-ID-ASS-GateID_1).</li> </ul> GATE <sub>ROOTBIS</sub> shall be present in one of the binding parameters (see VNP [5]). If present then the procedure is successful. A secure pipe session is opened between the AAA acting for the root accessor and AAS as the authentication service.

## 5.5.3.2.6 PTCS\_006 - Creation of an accessor TTI Control Service Accessor (TTI1)

<b>Procedure ID</b>	PTCS_006
<b>Procedure objectives</b>	The Accessor Authentication Application shall be able to create an TTI1 accessor from the Accessor Authentication Service using an aAAS-ADMIN-CREATE-ACCESSOR-Service-Command. TTI Control Service Accessor 1 (TTI1): Accessor identity: eTTI-ACC The TTI1 accessor authentication mean shall be based on the pin code.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_005 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	<p>AAA gate sends AAS-SERVICE-GATE-Commands to AAS gate with:</p> <pre>-- ASN1START  aPTCS-001-command-01 AAS-SERVICE-GATE-Commands ::= aAAS-ADMIN-CREATE-ACCESSOR-Service-Command : {   aAccessor aAccessorUser : {     aAccessorIdentity eTTI-ACC,     aAccessorConditions {       aAccessConditionsPIN ePinNumeric     },     aACL {       {         aAccessorIdentity eTTI-ACC-ROOT,         aAccessorRights eTTI-ACL-ROOT       },       {         aAccessorIdentity eTTI-ACC,         aAccessorRights eTTI-ACL       }     }   },   aCredential { aPinNumericCredential "1234" },   aCredentialsPolicy {     aPinNumericPolicy {       aIsDisableForbidden FALSE, aMinSize 4, aMaxSize 255, aMaxAttempts 3     }   },   aCredentialsStatus {     aPinNumericStatus {       aCommonStatus {         aIsDisabled FALSE       }     }   } }  -- ASN1STOP</pre> <p>The root accessor has all rights on the procedure accessor. The procedure accessor shall be authenticated by using the pin code.</p>
2	<p>AAS gate sends aAAS-ADMIN-CREATE-ACCESSOR-Service-Response to AAA gate with:</p> <pre>-- ASN1START  aPTCS-001-response-01 AAS-SERVICE-GATE-Responses ::= aAAS-ADMIN-CREATE-ACCESSOR-Service-Response : {   aAAS-Service-Response eAAS-OK }  -- ASN1STOP</pre> <p>The procedure is successful if the aAAS-Service-Response is eAAS-OK.</p>

### 5.5.3.2.7 PTCS\_007 - Open a pipe session with the Accessor Authentication Service for the TTI Control Service Accessor (TTI1)

<b>Procedure ID</b>	PTCS_007
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the Authentication Service gate of the TTI host. If the procedure is successful then a pipe session is open between the Accessor Authentication Application in the other host and the Accessor Authentication Service in the TTI host.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_001 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate sends EVT_ADM_BIND to Administration gate in the TTI with: <ul style="list-style-type: none"> <li>• PIPE<sub>BA</sub>: a dynamically assigned pipe identifier for the Accessor Authentication Service gate.</li> <li>• GATE<sub>TEST</sub>: the UUID gate identifier of the TTI1 accessor AA Service gate (0f26eb7c-ef78-5f8e-84fc-b64c9284638e).</li> </ul>
2	Administration gate sends EVT_ADM_BIND to Administration gate in the other host with a binding parameter equal to: <ul style="list-style-type: none"> <li>• PIPE<sub>AB</sub>: a dynamically assigned pipe identifier for the Identity Application gate.</li> <li>• GATE<sub>TEST</sub>: the UUID gate identifier of the TTI1 accessor AA Application gate (0f26eb7c-ef78-5f8e-84fc-b64c9284638e).</li> </ul> GATE <sub>TEST</sub> shall be present in one of the binding parameters (see VNP [5]). If present then the procedure is successful.

### 5.5.3.2.8 PTCS\_008 - Authentication of the accessor (TTI1)

<b>Procedure ID</b>	PTCS_008
<b>Procedure objectives</b>	The Accessor Authentication Application shall be able to authenticate TTI1 accessor from the Accessor Authentication Service using an aAAS-OP-AUTHENTICATE-ACCESSOR-Service-Command.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_0062 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	AAA gate sends AAS-SERVICE-GATE-Commands to AAS gate with: <pre>-- ASN1START aTCS-003-command-01 AAS-SERVICE-GATE-Commands ::= aAAS-OP-AUTHENTICATE-ACCESSOR-Service-Command : {   aCredential aPinNumericCredential : "1234" } -- ASN1STOP</pre>
2	AAS gate sends AAS-SERVICE-GATE-Responses to AAA gate with: <pre>-- ASN1START aTCS-003-response-01 AAS-SERVICE-GATE-Responses ::= aAAS-OP-AUTHENTICATE-ACCESSOR-Service-Response : {   aAAS-Service-Response eAAS-OK } -- ASN1STOP</pre> The procedure is successful if the aAAS-Service-Response is eAAS-OK.

## 5.5.3.2.9 PTCS\_009 - Access to TTI Control Service with secure pipe (TTI1)

<b>Procedure ID</b>	PTCS_009
<b>Procedure objectives</b>	The Accessor Authentication Application on behalf of TTI Control Service 1 (TTI1) accessor shall be able to access the TTI Control Service from the Accessor Authentication Service using an aAAS-OP-ACCESS-SERVICE-Service-Command. The TTI Control Service Identifier is: 09560b78-bed9-58b9-a5ff-6caa8384d556.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_003 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	AAA gate sends an AAS-OP-ACCESS-SERVICE-Service-Command to AAS gate with: -- ASN1START  aTCS-004-command-01 AAS-SERVICE-GATE-Commands ::= aAAS-OP-ACCESS-SERVICE-Service-Command : { aServiceIdentifier eTTI-ID-CS, aUseSecurePipe TRUE }  -- ASN1STOP
2	AAS gate sends an AAS-OP-ACCESS-SERVICE-Service-Response to AAA gate with: -- ASN1START  aTCS-004-response-01 AAS-SERVICE-GATE-Responses ::= aAAS-OP-ACCESS-SERVICE-Service-Response : { aAAS-Service-Response eAAS-OK, aParameter { aGateIdentifier eAS-ID-ASS-GateID_2 /* <STORE(eAS-ID-ASS-GateID_2)> */ } }  -- ASN1STOP The AAS returns the gate identifier on which the authenticated TTI1 accessor can access the TTI Control Service by using a secure pipe. The procedure is successful if the AAS returns eAAS-OK.

## 5.5.3.2.10 PTCS\_010 - Open a secure pipe session with the TTI Control Service for the TTI1 accessor

<b>Procedure ID</b>	PTCS_010
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the TTI Control Service gate of the TTI host on the behalf of the TTI1 accessor. If the procedure is successful then a pipe session is open between the TTI Control Application in the other host and the TTI Control Service in the TTI host.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_010 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate sends EVT_ADM_BIND to Administration gate in the TTI with: <ul style="list-style-type: none"> <li>PIPE<sub>BA</sub>: a dynamically assigned pipe identifier for the TTI Control Service Gate.</li> <li>GATE<sub>TEST</sub>: the dynamically assigned UUID gate identifier returned by AAS in PTCS_010 (eAS-ID-ASS-GateID_2).</li> </ul>
2	Administration gate sends EVT_ADM_BIND to Administration gate in the other host with a binding parameter equal to: <ul style="list-style-type: none"> <li>PIPE<sub>AB</sub>: a dynamically assigned pipe identifier for the TTI Control Application gate.</li> <li>GATE<sub>TEST</sub>: the dynamically assigned UUID gate identifier returned by AAS PTCS_010 (eAS-ID-ASS-GateID_2).</li> </ul> GATE <sub>TEST</sub> shall be present in one of the binding parameters (see VNP [5]). If present then the procedure is successful. A secure pipe session is opened between the TTI1 accessor and the TTI Control Service gate.

5.5.3.2.11 PTCS\_011 - Access to TTI Control Service with secure pipe (ROOT)

<b>Procedure ID</b>	PTCS_011
<b>Procedure objectives</b>	The Accessor Authentication Application on the behalf of ROOT accessor shall be able to access the TTI Control Service from the Accessor Authentication Service using an aAAS-OP-ACCESS-SERVICE-Service-Command. The TTI Control Service identifier is 09560b78-bed9-58b9-a5ff-6caa8384d556.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_003 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	AAA gate sends an AAS-OP-ACCESS-SERVICE-Service-Command to AAS gate with: -- ASN1START  aTCS-011-command-01 AAS-SERVICE-GATE-Commands ::= aAAS-OP-ACCESS-SERVICE-Service-Command : { aServiceIdentifier eTTI-ID-CS, aUseSecurePipe TRUE }  -- ASN1STOP
2	AAS gate sends an AAS-OP-ACCESS-SERVICE-Service-Response to AAA gate with: -- ASN1START  aTCS-011-response-01 AAS-SERVICE-GATE-Responses ::= aAAS-OP-ACCESS-SERVICE-Service-Response : { aAAS-Service-Response eAAS-OK, aParameter { aGateIdentifier eAS-ID-ASS-GateID_3 /* <STORE(eAS-ID-ASS-GateID_3)> */ } }  -- ASN1STOP The AAS returns the gate identifier on which the root accessor can access the TTI Control Service by using a secure pipe. The procedure is successful if the AAS returns eAAS-OK.

5.5.3.2.12 PTCS\_012 - Open a secure pipe session with the TTI Control Service for the root accessor

<b>Procedure ID</b>	PTCS_012
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the TTI Control Service gate of the TTI host on the behalf of the root accessor. If the procedure is successful then a pipe session is open between the TTI Control Application in the other host and the TTI Control Service in the TTI host.
<b>Configuration reference</b>	CTCS_002
<b>Initial conditions</b>	
The procedure PTCS_011 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate sends EVT_ADM_BIND to Administration gate in the TTI with: <ul style="list-style-type: none"> <li>• PIPE<sub>BA</sub>: a dynamically assigned pipe identifier for the TTI Control Service gate.</li> <li>• GATE<sub>TEST</sub>: the dynamically assigned UUID gate identifier returned by AAS in PTCS_011 (eAS-ID-ASS-GateID_2).</li> </ul>
2	Administration gate sends EVT_ADM_BIND to Administration gate in the other host with a binding parameter equal to: <ul style="list-style-type: none"> <li>• PIPE<sub>AB</sub>: a dynamically assigned pipe identifier for the TTI Control Application gate.</li> <li>• GATE<sub>TEST</sub>: the dynamically assigned UUID gate identifier returned by AAS PTCS_011 (eAS-ID-ASS-GateID_2).</li> </ul> GATE <sub>TEST</sub> shall be present in one of the binding parameters (see VNP [5]). If present then the procedure is successful. A secure pipe session is opened between the root accessor and the TTI Control Service gate.

## 5.5.3.3 Test Descriptions

## 5.5.3.3.1 TTI-OP-GET-CAPABILITIES

## 5.5.3.3.1.1 TCS\_001 - Get Capabilities

<b>Test ID</b>	TCS_001	
<b>Test objectives</b>	The TTI Control Application shall be able to retrieve the capabilities of the TTI Control Service using TTI-OP-GET-CAPABILITIES-Service-Command.	
<b>Configuration reference</b>	CTCS_003	
<b>Initial conditions</b>		
The procedure PTCS_010 is successfully executed. -- ASN1START  aTCSVERSION INTEGER ::= 10 /*<STORE(TCSVERSION)> it contains the value of the major and minor release version supported by the TTI control Service gate */  --ASN1STOP		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	TTI Control Application gate sends aTCS-001-command-01 TCS-to TTI Control Service gate with: -- ASN1START  aTCS-001-command-01 TTI-SERVICE-GATE-Commands ::= aTTI-OP-GET-CAPABILITIES-Service-Command : { }  -- ASN1STOP	
2	TTI Control Service gate sends aTCS-001-response-01 to TTI Control Application gate with: -- ASN1START  aTCS-001-response-01 TTI-SERVICE-GATE-Responses ::= aTTI-OP-GET-CAPABILITIES-Service-Response : { aTTI-Service-Response eTTI-OK, aParameter { aVersion '0000'H /*<COMPARE(TCSVERSION,GT,EQ)>*/ } }  -- ASN1STOP	RU0602_001 RU0602_002



5.5.3.3.2 TTI-ADMIN-IMPERSONATE

5.5.3.3.2.1 TCS\_002 - Successful Host impersonation

<b>Test ID</b>	TCS_002	
<b>Test objectives</b>	The TTI CONTROL application gate shall be able to impersonate a host.	
<b>Configuration reference</b>	CTCS_003	
<b>Initial conditions</b>		
The procedure PTCS_010 shall be successfully executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	TTI Control Application gate sends aTCS-002-command-01 to TTI Control Service gate with: -- ASN1START  aTCS-002-command-01 TTI-SERVICE-GATE-Commands ::= aTTI-ADMIN-IMPERSONATE-Service-Command : { { aFirmwareFamilyID eTTI-FFI, -- Identifier of firmware family of the host to impersonate (REE host domain identifier) aHostDomainID 'E7A14FE4378D51AC85C805F6504A7C91'H -- Host domain identifier of the host to impersonate } } -- ASN1STOP	RU0602_004
2	TTI CONTROL Service gate sends eTTI-OK response to the TTI CONTROL Application gate. -- ASN1START  aTCS-002-response-01 TTI-SERVICE-GATE-Responses ::= aTTI-ADMIN-IMPERSONATE-Service-Response : { aTTI-Service-Response eTTI-OK } -- ASN1STOP	RU0602_005

5.5.3.3.2.2 TCS\_003 - Wrong parameter- Host impersonation

<b>Test ID</b>	TCS_003	
<b>Test objectives</b>	The TTI CONTROL Service gate shall reject the host impersonation when wrong parameters are applied.	
<b>Configuration reference</b>	CTCS_003	
<b>Initial conditions</b>		
The procedure PTCS_010 shall be successfully executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	TTI Control Application gate sends aTCS-003-command-01 to TTI Control Service gate with: -- ASN1START  aTCS-003-command-01 TTI-SERVICE-GATE-Commands ::= aTTI-ADMIN-IMPERSONATE-Service-Command : { } } -- ASN1STOP	
2	TTI CONTROL Service gate sends eTTI-NOK response to the TTI CONTROL Application gate. -- ASN1START  aTCS-003-response-01 TTI-SERVICE-GATE-Responses ::= aTTI-ADMIN-IMPERSONATE-Service-Response : { aTTI-Service-Response eTTI-NOK } -- ASN1STOP	RU0602_006 RU0602_007

## 5.5.3.3.3 TTI-ADMIN-UPDATE-ACL

## 5.5.3.3.3.1 TCS\_004 - Successful ACL update

<b>Test ID</b>	TCS_004	
<b>Test objectives</b>	The TTI Control Application gate shall be able to update the ACL related to the TTI interface resource.	
<b>Configuration reference</b>	CTCS_003	
<b>Initial conditions</b>		
The procedure PTCS_012 shall be successfully executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	<p>TTI Control Application gate sends aTCS-004-command-01 to TTI Control Service gate with:</p> <pre>-- ASN1START  aTCS-004-command-01 TTI-SERVICE-GATE-Commands ::= aTTI-ADMIN-UPDATE-ACL-Service-Command : {   aACL {     {       aAccessorIdentity eTTI-ACC,       aAccessorRights eTTI-CS-ACL     }   } } -- ASN1STOP</pre>	<p>RU0602_003 RU0602_006</p>
2	<p>TTI CONTROL Service gate sends eTTI-OK response to the TTI CONTROL Application gate. TTI control Service gate sends aTCS-004-response-01 to TTI control Application gate with:</p> <pre>-- ASN1START  aTCS-004-response-01 TTI-SERVICE-GATE-Responses ::= aTTI-ADMIN-UPDATE-ACL-Service-Response :{   aTTI-Service-Response eTTI-OK } -- ASN1STOP</pre>	<p>RU0602_007</p>

## 5.5.3.3.3.2 TCS\_005 - Wrong parameter- ACL update

<b>Test ID</b>	TCS_005	
<b>Test objectives</b>	The TTI Control Service gate shall reject the ACL update when the accessor did not grant the rights to	
<b>Configuration reference</b>	CTCS_003	
<b>Initial conditions</b>		
The procedure PTCS_010 shall be successfully executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	<p>TTI Control Application gate sends aTCS-005-command-01 to TTI Control Service gate with:</p> <pre>-- ASN1START  aTCS-005-command-01 TTI-SERVICE-GATE-Commands ::= aTTI-ADMIN-UPDATE- ACL-Service-Command : {   aACL {     {       aAccessorIdentity eTTI-ACC,       aAccessorRights eTTI-CS-ACL     }   } } -- ASN1STOP</pre>	RU0602_006
2	<p>TTI CONTROL Service gate sends eTTI-OK response to the TTI CONTROL Application gate. TTI control Service gate sends aTCS-005-response-01 to TTI control Application gate with:</p> <pre>-- ASN1START  aTCS-005-response-01 TTI-SERVICE-GATE-Responses ::= aTTI-ADMIN-UPDATE- ACL-Service-Response :{   aTTI-Service-Response eTTI-NOK } -- ASN1STOP</pre>	RU0602_007

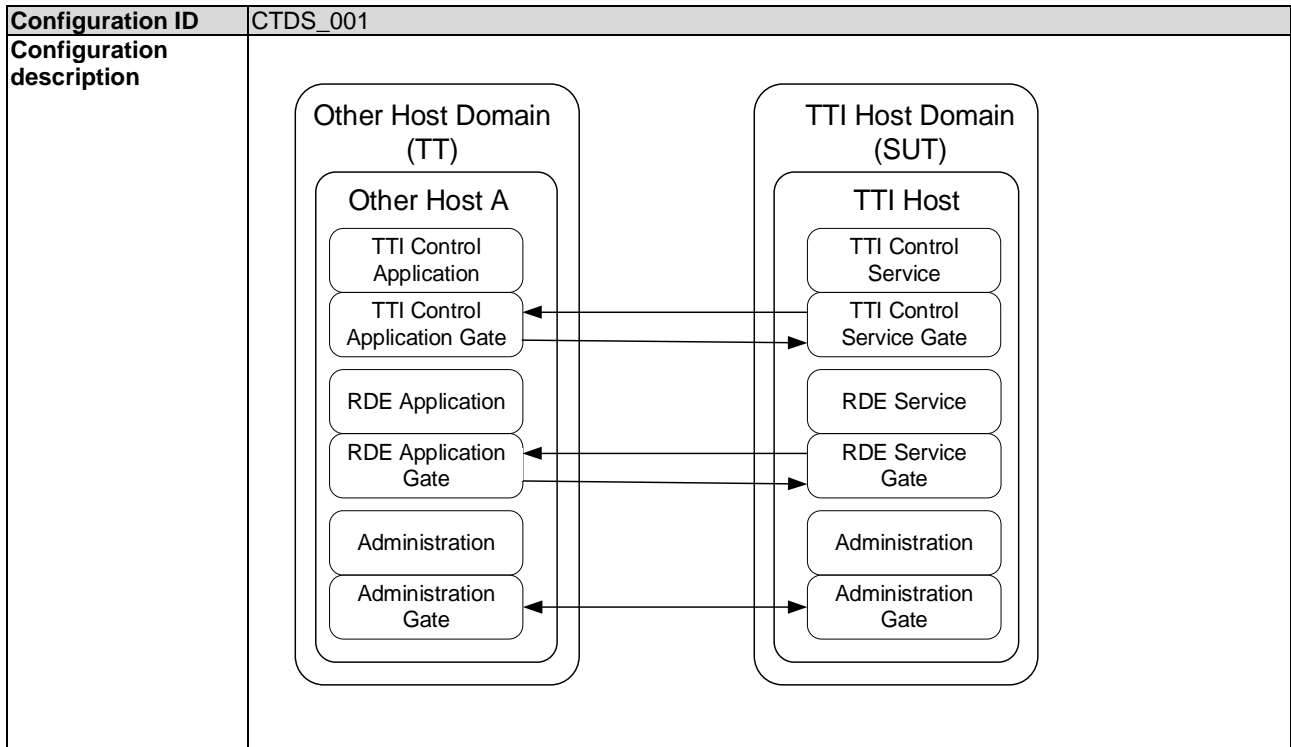
## 5.5.3.3.3 TCS\_006 - Successful ACL update (no APDU)

<b>Test ID</b>	TCS_006	
<b>Test objectives</b>	The TTI Control Application gate shall be able to update the ACL related to the TTI interface resource. The access to the APDU service of a host shall fail.	
<b>Configuration reference</b>	CTCS_003	
<b>Initial conditions</b>		
The procedure PTCS_012 shall be successfully executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	<pre> TTI Control Application gate sends aTCS-006-command-01 to TTI Control Service gate with: -- ASN1START  aTCS-006-command-01 TTI-SERVICE-GATE-Commands ::= aTTI-ADMIN-UPDATE- ACL-Service-Command : {   aACL {     {       aAccessorIdentity eTTI-ACC,       aAccessorRights '00000000000000000000000000000000'H     }   } } -- ASN1STOP </pre>	
2	<pre> TTI CONTROL Service gate sends eTTI-OK response to the TTI CONTROL Application gate with: -- ASN1START  aTCS-006-response-01 TTI-SERVICE-GATE-Responses ::= aTTI-ADMIN-UPDATE- ACL-Service-Response :{   aTTI-Service-Response eTTI-OK } -- ASN1STOP </pre>	RU0602_007

## 5.5.4 TTI Data Service

### 5.5.4.1 Configurations

#### 5.5.4.1.1 CTDS\_001



## 5.5.4.2 Procedures

## 5.5.4.2.1 PTDS\_001 - Access to TTI RDE Service for TTI1 with secure pipe

<b>Procedure ID</b>	PTDS_001
<b>Procedure objectives</b>	The Accessor Authentication Application on the behalf of TTI RDE Service 1 (TTI1) accessor shall be able to access the TTI Control Service from the Accessor Authentication Service using an aAAS-OP-ACCESS-SERVICE-Service-Command. The TTI RDE Service identifier is 09560b78-bed9-58b9-a5ff-6caa8384d556.
<b>Configuration reference</b>	CTDS_001
<b>Initial conditions</b>	
The procedure PTCS_003 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	AAA gate sends an AAS-OP-ACCESS-SERVICE-Service-Command to AAS gate with: -- ASN1START  aTCS-001-command-01 AAS-SERVICE-GATE-Commands ::= aAAS-OP-ACCESS-SERVICE-Service-Command : { aServiceIdentifier eTTI-ID-CS, aUseSecurePipe TRUE }  -- ASN1STOP
2	AAS gate sends an AAS-OP-ACCESS-SERVICE-Service-Response to AAA gate with: -- ASN1START  aTCS-0001-response-01 AAS-SERVICE-GATE-Responses ::= aAAS-OP-ACCESS-SERVICE-Service-Response : { aAAS-Service-Response eAAS-OK, aParameter { aGateIdentifier eAS-ID-ASS-GateID_2 /* <STORE(eAS-ID-ASS-GateID_2)> */ } }  -- ASN1STOP The AAS returns the gate identifier on which the authenticated TTI1 accessor can access the TTI RDE Service by using a secure pipe. The procedure is successful if the AAS returns eAAS-OK.

## 5.5.4.2.2 PTDS\_002 - Open a secure pipe session with the TTI Data Service for the TTI1 accessor

<b>Procedure ID</b>	PTDS_002
<b>Procedure objectives</b>	The other host shall be able to open a pipe session to the TTI Control Service gate of the TTI host on the behalf of the TTI1 accessor. If the procedure is successful, then a pipe session is open between the TTI Control Application in the other host and the TTI Control Service in the TTI host.
<b>Configuration reference</b>	CTDS_001
<b>Initial conditions</b>	
The procedure PTCS_001 shall be successfully executed.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate sends EVT_ADM_BIND to Administration gate in the TTI with: <ul style="list-style-type: none"> <li>PIPE<sub>BA</sub>: a dynamically assigned pipe identifier for the TTI Control Service gate.</li> <li>GATE<sub>TEST</sub>: the dynamically assigned UUID gate identifier returned by AAS in PTCS_001 (eAS-ID-ASS-GateID_2).</li> </ul>
2	Administration gate sends EVT_ADM_BIND to Administration gate in the other host with a binding parameter equal to: <ul style="list-style-type: none"> <li>PIPE<sub>AB</sub>: a dynamically assigned pipe identifier for the TTI Control Application gate.</li> <li>GATE<sub>TEST</sub>: the dynamically assigned UUID gate identifier returned by AAS in PTCS_001 (eAS-ID-ASS-GateID_2).</li> </ul> GATE <sub>TEST</sub> shall be present in one of the binding parameters (see VNP [5]). If present, then the procedure is successful. A secure pipe session is opened between the TTI1 accessor and the TTI Control Service gate.

### 5.5.4.3 Test Descriptions

#### 5.5.4.3.1 TDS\_001 - Tunnelling of SCL packets

<b>Test ID</b>	TDS_001	
<b>Test objectives</b>	The impersonated host shall be able to run any tests as defined in the ETSI TS 103 999-1 [3] and ETSI TS 103 999-2 [4].	
<b>Configuration reference</b>	CTDS_001	
<b>Initial conditions</b>		
The procedure PTDS_001 shall be executed. The procedure PTDS_002 shall be executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	Execution of the SCL_033 test description in the ETSI TS 103 999-1 [3] from the impersonated host.	RQ0608_002 RQ1008_001 RU0603_001

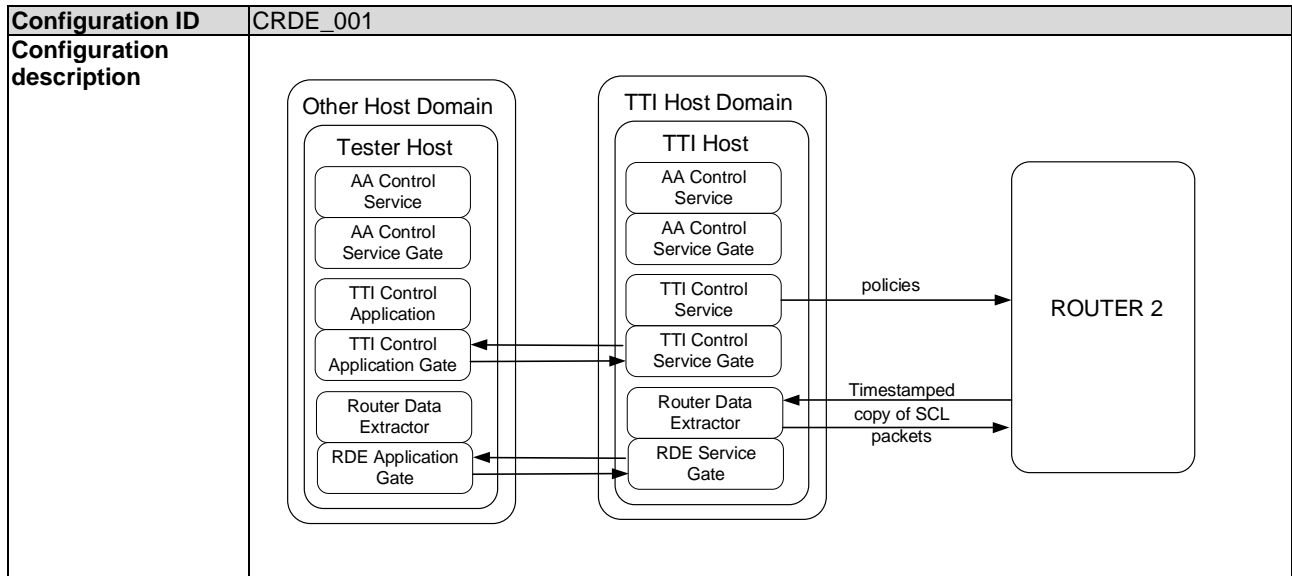
#### 5.5.4.3.2 TDS\_002 - Tunnelling of SCL packets

<b>Test ID</b>	TDS_002	
<b>Test objectives</b>	The impersonated host shall not be able to run any tests from the clause 10.2 as defined in the ETSI TS 103 999-1 [3].	
<b>Configuration reference</b>	CTDS_001	
<b>Initial conditions</b>		
The test description TCS_006 shall be executed. The procedure PTDS_001 shall be executed. The procedure PTDS_002 shall be executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	Execution of any test descriptions of the clause 10.2 in the ETSI TS 103 999-1 [3] from the impersonated host shall fail.	RQ0608_002 RQ1008_001 RQ0803_005 RQ0803_006 RQ0803_007 RQ0803_008 RQ0803_009 RQ0803_010 RQ0803_011 RQ0803_012 RQ0803_013 RQ0803_014 RQ0803_015 RQ0803_016

## 5.5.5 RDE Data Service

### 5.5.5.1 Configurations

#### 5.5.5.1.1 CRDE\_001



### 5.5.5.2 Procedures

#### 5.5.5.2.1 PRDE\_001 - Creation of an RDE Service gate

<b>Procedure ID</b>	PRDE_001
<b>Procedure objectives</b>	The TTI Host requests the creation of an RDE Service gate
<b>Configuration reference</b>	CRDE_001
<b>Initial conditions</b>	
The TT Host got confirmed the accessor authentication service capability by the TTI host. The TT Host got a confirmation for the authentication of an accessor.	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	The TT Host requests a session to a service by sending the command AAS-OP-ACCESS-SERVICE-Service-Command with the service identified by aServiceIdentifier containing the RDE Data Service Identifier.
2	The TTI Host dynamically creates a gate to the requested service and sends an answer to the TT Host with the identifier of the dynamically created gate.



## 5.5.5.2.2 PRDE\_0002 - Open RDE data service session

<b>Procedure ID</b>	PRDE_002
<b>Procedure objectives</b>	The TT Host requests the authentication service capability from the TTI host.
<b>Configuration reference</b>	CRDE_001
<b>Initial conditions</b>	
None	
<b>Procedure sequence</b>	
<b>Step</b>	<b>Description</b>
1	Administration gate in the other host sends EVT_ADM_BIND to Administration gate in the SSP host with: <ul style="list-style-type: none"> <li>• PIPE<sub>XY</sub>: a dynamically assigned pipe identifier for the identity service gate.</li> <li>• GATE<sub>IDENTITY</sub>: the UUID gate identifier of the identity gate (416B66AC-A134-5082-8160-FA1BA497F917).</li> </ul>
2	Administration gate in the SSP host sends EVT_ADM_BIND to Administration gate in the other host with: <ul style="list-style-type: none"> <li>• PIPE<sub>YX</sub>: a dynamically assigned pipe identifier for the identity application gate.</li> <li>• GATE<sub>IDENTITY</sub>: the UUID gate identifier of the identity gate (416B66AC-A134-5082-8160-FA1BA497F917).</li> </ul>

## 5.5.5.3 Test Descriptions

## 5.5.5.3.1 RDE\_001 - Extract Time Stamp

<b>Test ID</b>	RDE_001	
<b>Test objectives</b>	The other host shall be able to collect the timestamped EVT_RDE event from the RDE service. Any tests defined in the ETSI TS 103 999-1 [3] and ETSI TS 103 999-2 [4] can be executed from the impersonated host.	
<b>Configuration reference</b>	CDRE_001	
<b>Initial conditions</b>		
The procedure PDRE_001 shall be executed. The procedure PDRE_002 shall be executed.		
<b>Test sequence</b>		
<b>Step</b>	<b>Description</b>	<b>Req.ID</b>
1	Execution of the SCL_033 test description in the ETSI TS 103 999-1 [3] from the impersonated host. The collected events shall be consistent with the successful execution of the SCL_033 tests.	RU0604_001 RU0604_002 RS0402_001 RS0402_002 RS0402_003 RS0402_004 RS0402_005 RS0402_006

---

## Annex A (informative): References on ETSI forge

### A.1 ETSI forge repository for the TTI test specification

- <https://forge.etsi.org/rep/set/etsi-ts-103-834-part-2/tree/17.0.0>

---

### A.2 License information

- <https://forge.etsi.org/rep/set/etsi-ts-103-834-part-2/blob/17.0.0/LICENSE>

---

### A.3 ASN.1 coding

The complete ASN.1 coding is available on ETSI forge:

- <https://forge.etsi.org/rep/set/etsi-ts-103-834-part-2/tree/17.0.0/asn1>

---

## Annex B (informative): ASN.1 definition

### B.1 End of ASN.1

```
-- ASN1START  
END  
-- ASN1STOP
```

## Annex C (informative): Core specification version information

Unless otherwise specified, the versions of ETSI TS 103 666-1 [1] from which conformance requirements have been extracted are as follows.

Release	Version conformance requirements extracted from
15	V15.8.0
16	V16.4.0

Unless otherwise specified, the versions of ETSI TS 103 666-2 [2] from which conformance requirements have been extracted are as follows.

Release	Version conformance requirements extracted from
15	V15.3.0
16	V16.4.0

Unless otherwise specified, the versions of ETSI TS 103 999-1 [3] from which conformance requirements have been extracted are as follows.

Release	Version conformance requirements extracted from
15	V15.0.0

Unless otherwise specified, the versions of GlobalPlatform<sup>M</sup> Virtual Primary Platform - Network Protocol [5] from which conformance requirements have been extracted are as follows.

Release	Version conformance requirements extracted from
--	V2.0

Unless otherwise specified, the versions of ETSI TS 103 834-1 [7] from which conformance requirements have been extracted are as follows.

Release	Version conformance requirements extracted from
17	V17.0.0

---

## Annex D (informative): Change History

The table below indicates all changes that have been incorporated into the present document since it was placed under change control.

Change history								
Date	Meeting	Plenary Doc	CR	Rev	Cat	Subject/Comment	Old	New
08/12/2022	SET#108	SET(22)000231	-	-	-	Version 17.0.0 first publication	-	17.0.0

---

# History

<b>Document history</b>		
V17.0.0	December 2022	Publication