ETSI GS MEC-DEC 032-2 V2.1.1 (2020-12)

Multi-access Edge Computing (MEC); API Conformance Test Specification; Part 2: Test Purposes (TP)

Disclaimer

The present document has been produced and approved by the Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
DGS/MEC-DEC32-2APIConformance

Keywords
API, conformance, MEC, 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 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

List of tables

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 IPR Policy, no investigation, 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.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Multi-access Edge Computing (MEC).

The present document is part 2 of a multi-part deliverable. Full details of the entire series can be found in part 1 [i.9].

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

Based on the testing methodology guidelines and framework specified in ETSI GR MEC-DEC 025 [i.8], the present document specifies part 2 of a multi-part deliverable test specification for the MEC service APIs (currently ETSI GS MEC 012 [5], ETSI GS MEC 013 [6], ETSI GS MEC 014 [7], ETSI GS MEC 015 [8], ETSI GS MEC 016 [9], ETSI GS MEC 021 [10] and ETSI GS MEC 029 [11]) and the MEC Application Enablement API (ETSI GS MEC 011 [4]).

The present document includes the Test Suite Structure (TSS) and Test Purposes (TPs) using the standardized notation Test Description Language - Test Objectives extension (TDL_TO).

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.

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 GS MEC 001 (V2.1.1) (01-2019): "Multi-access Edge Computing (MEC); Terminology".

[2] ETSI GS MEC 009 (V2.2.1) (10-2020): " Multi-access Edge Computing (MEC); General principles, patterns and common aspects of MEC Service APIs".

[3] ETSI GS MEC 010-2 (V2.1.1) (11-2019): "Multi-access Edge Computing (MEC); MEC Management; Part 2: Application lifecycle, rules and requirements management".

[4] ETSI GS MEC 011 (V2.1.1) (11-2019): "Multi-access Edge Computing (MEC); Edge Platform Application Enablement".

[5] ETSI GS MEC 012 (V2.1.1) (12-2019): "Multi-access Edge Computing (MEC); Radio Network Information API".

[6] ETSI GS MEC 013 (V2.1.1) (09-2019): "Multi-access Edge Computing (MEC); Location API".

[7] ETSI GS MEC 014 (V1.1.1) (02-2018): "Mobile Edge Computing (MEC); UE Identity API".

[8] ETSI GS MEC 015 (V1.1.1) (10-2017): "Mobile Edge Computing (MEC); Bandwidth Management API".

[9] ETSI GS MEC 016 (V2.1.1) (04-2019): "Multi-access Edge Computing (MEC); UE application interface".

[10] ETSI GS MEC 021 (V2.1.1) (01-2020): "Multi-access Edge Computing (MEC); Application Mobility Service API".

[11] ETSI GS MEC 029 (V2.1.1) (07-2019): "Multi-access Edge Computing (MEC); Fixed Access Information API".

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.

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] ETSI GS NFV-TST 002: "Network Functions Virtualisation (NFV); Testing Methodology; Report on NFV Interoperability Testing Methodology".

[i.2] ETSI EG 202 237: "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); Generic approach to interoperability testing".

[i.3] ETSI EG 202 568: "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); Testing: Methodology and Framework".

[i.4] ETSI GS MEC 003: "Multi-access Edge Computing (MEC); Framework and Reference Architecture".

[i.5] ISO/IEC 9646-7:1995: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 7: Implementation Conformance Statements".

NOTE: Available at https://www.iso.org/standard/3084.html.

[i.6] ISO/IEC 9646-1:1994: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 1: General concepts".

NOTE: Available at https://www.iso.org/standard/17473.html.

[i.7] TTCN-3 abstract test language.

NOTE: Available at http://www.ttcn-3.org/index.php/downloads/standards.

[i.8] ETSI GR MEC-DEC 025: "Multi-access Edge Computing (MEC); MEC Testing Framework".

[i.9] ETSI GS MEC-DEC 032-1: "Multi-access Edge Computing (MEC); API Conformance Test Specification; Part 1: Test Requirements and Implementation Conformance Statement (ICS)".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

certification/compliance assessment: major goal of a compliance assessment is to ensure the interoperability of implementations, and the conformance of implementations to the standard

conformance testing: purpose of conformance testing is to determine to what extent a single implementation of a particular standard conforms to the individual requirements of that standard

interoperability testing: purpose of interoperability testing is to prove that end-to-end functionality between (at least) two communicating systems is as required by the standard(s) on which those systems are based

Test Case (TC): complete and independent specification of the actions required to achieve a specific Test Purpose

NOTE: TCs are written in testing languages, e.g. TTCN-3.

Test Descriptions (TD): specification of the sequence of actions required to realize the verdict identified in the TP

NOTE: TDs are primarily intended for use in interoperability test specifications. However, in some instances, particularly where there is a considerable difference in complexity between the TPs and the TCs, it is worthwhile adding TDs as an extra design stage in a conformance test specification.

Test Purpose (TP): definition in broad terms of the goal of a particular test.

NOTE: A TP should be written for each potential test of each identified requirement. A TP is defined in prose, or in high level languages such as TDL-TO.

test suite: collection of Test Cases

testing framework: guidance for development of conformance and interoperability test strategies, test systems and the resulting test specifications

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS MEC 001 [1] apply.

4 Prerequisites and Test Configurations

4.1 Test Configurations

Test configurations capture and describe the components identified in the tests and the connections between them. In particular and as reported in ETSI GR MEC-DEC 025 [i.8], in the context of conformance testing the test configuration "defines how the test system connects to the SUT".

For the present test suite, six (6) configurations are identified and listed in the present clause.

For each test configuration two (2) main components are identified: the IUT implementing the API provider and the Tester implementing the API consumer. The IUT is part of a SUT (System Under Test), thus the component may be run together with other components of the MEC System that are required to enable the behaviour to be tested. The definition of the other components is out of scope.

Figure 4.1-1 depicts configuration Config_MEC_1 which includes the a MEC Platform as the IUT and a MEC App as the Tester. This configuration is applicable for all test purposes in all subgroups of the SRV Group.

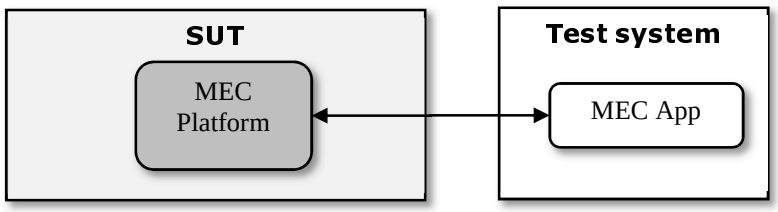

Figure 4.1-1: Config_MEC_1

Figure 4.1-2 depicts configuration Config_MEC_2 which includes a MEO as the IUT and OSS/BSS as the Tester. This configuration is applicable for group MEO, subgroup PKGM.

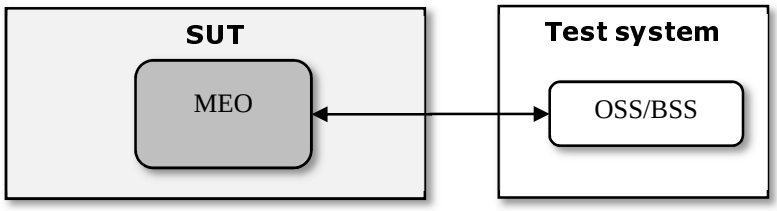

Figure 4.1-2: Config_MEC_2

Figure 4.1-3 depicts configuration Config_MEC_3 which includes the a MEO as the IUT and a MEPM as the Tester. This configuration is applicable for subgroup MEO/GRANT and in subgroup MEPM/PKGM.

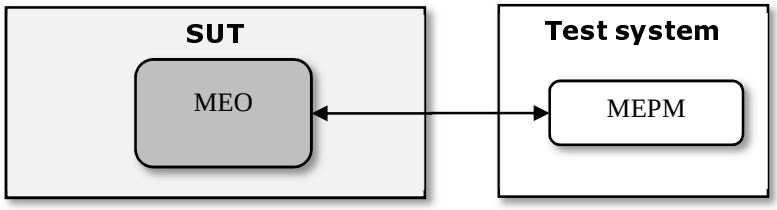

Figure 4.1-3: Config_MEC_3

Figure 4.1-4 depicts configuration Config_MEC_4 which includes the a UALCM Proxy as the IUT and a DEV App as the Tester. This configuration is applicable for group MEO subgroups UEAPPCTX and UEAPPS.

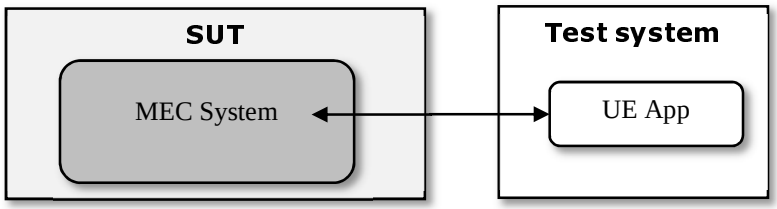

Figure 4.1-4: Config_MEC_4

Figure 4.1-5 depicts configuration Config_MEC_5 which includes the a MEPM as the IUT and a MEO as the Tester. This configuration is applicable for group MEPM, subgroup PKGM.

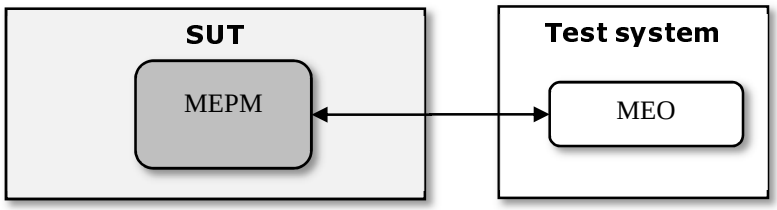

Figure 4.1-5: Config_MEC_5

Figure 4.1-6 depicts configuration Config_MEC_6 which includes the a generic MEC API Provider as the IUT and a generic MEC API consumer as the Tester. This configuration is applicable for test targeting generic API behaviours, thus group MEX , subgroup ANY and LCM.

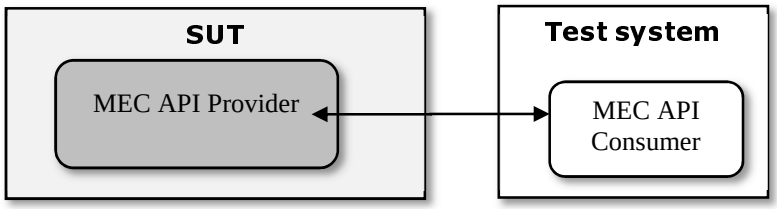

Figure 4.1-6: Config_MEC_6

5 Test Suite Structure (TSS)

5.1 Overview

The test suite structure identifies grouping of test purposes and serves a a base for grouping of Test Case in the ATS (Abstract Test Suite).

The Test Suite structure is used for the creation of identifiers of Test Purposes.

Table 5.1-1 identifies the Test Suite Structure for the MEC API Conformance test suites. Documentation on the groups and subgroups is provided in clause 5.2.

Table 5.1-1: Test Suite Structure for MEC API Conformance

TP_<root>_<gr>_<subgr>_<nnn>_<xx>		
<root> = root	MEC	MEC
<gr> = group	SRV	MEC Services
	MEO	MEC Orchestrator
	MEPM	MEC Platform Manager
	PLAT	MEC platform
	MEX	Generic MEC API Producer
<subgr> = subgroup	AMS	Application Mobility Service
	ANY	Un-specified feature, used for generic test purposes
	APPASQ	Application Service Availability Query
	APPSUB	Application Subscriptions
	BWA	Bandwidth Allocations
	DNS	DNS rules
	LCM	Lifecycle Management
	FAIS	Fixed Access Information Service
	GRANT	Granting
	PKGM	App Package Management
	SAQ	Service Availability Query
	RLOC	Radio Node Location Lookup
	RNIS	Radio Network InformationService
	SRVSUB	Service Subscriptions
	TIME	Timing capabilities
	TRAF	Traffic rules
	TRANS	Transport
	UEAPPCTX	Device Application Contexts
	UEAPPS	Device Applications
	UEAREASUB	UE Area Subscribe
	UEDISTLOOK	UE Distance Lookup
	UEDISTSUB	UE Distance Subscribe
	UEINFLOOK	UE Information Lookup
	UEINFSUB	UE Information Subscription
	UELOC	UE Location lookup
	UELOCSUB	UE Location Subscription
	UETAG	UE Identity Tag
	UETRACKSUB	UE Tracking Subscribe

TP_<root>_<gr>_<subgr>_<nnn>_<xx>		
<nnn> = sequential number		001 to 999
<xx> = type of testing	OK	Valid/Successful behaviour (200, 201, 202, 204)
	BR	Bad request
	NT	No token
	WT	Wrong Token
	NF	Missing (404)
	CO	Conflict (409)
	PF	Precondition Failed (412)
	E1	Generic error condition 1
	E2	Generic error condition 2
	E3	Generic error condition 3
	…	

5.2 Test groups and subgroups specifications

The test groups are organized on two (2) levels. The first level is the Group and identifies the MEC component that is providing the Service or Interface to be tested. The second is called Subroup and identifies a set of functionalities within an API. In general this is related to the entities and resources manipulated or served by the API.

Moreover, test purposes are identified and categorized by a sequential three-digits number (uniquely assigned upon definition of the each test purpose) and by the type of test performed. The type of test helps quickly identify the type of behaviour that is expected by the IUT in the test purpose.

5.3 Conventions

Conventions reported in ETSI GR MEC-DEC 025 [i.8], clauses 4.3.3.2.3 and 4.3.3.2.4 shall apply.

The test purposes are primarily developed in textual syntax of TDL-TO. The sources for the Test Purposes are available in https://forge.etsi.org/rep/mec/gs032p2-test-purposes/tree/v2.1.1.

The definitions of PICS, Entities, Events and data types are available in Domain section in the mec-common.tplan2 file.

6 Test Purposes (TP)

6.1 Multi-access Edge Orchestrator (MEO)

6.1.1 App Package Management (PKGM)

Table 6.1.1-1: TP_MEC_MEO_PKGM_001_OK

TP Id	"TP_MEC_MEO_PKGM_001_OK"
Test Objective	Check that MEO creates a new App Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.1.3.1, tables 6.2.3.3.2-1 and 6.2.3.2.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/apmi/v1/app_packages",
body containing
AppPkg containing
appPkgName set to APP_PKG_NAME,
appPkgVersion set to APP_PKG_VERSION,
checksum set to CHECKSUM,
appPkgPath set to APP_PKG_PATH
;
;
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 Created",
body containing
OnboardedAppPkgInfoList containing
OnboardedAppPkgInfo containing
appPkgId set to any_value,
appDId set to any_value,
appName set to APP_PKG_NAME,
appSoftwareVersion set to any_value,
appDVersion set to APP_PKG_VERSION,
checksum set to CHECKSUM,
attribute softwareImages,
operationalState set to "ENABLED",
usageState set to "NOT_IN_USE",
attribute _links
;
;
;
; to the MEC_OSS
}
}

Final Conditions

Table 6.1.1-2: TP_MEC_MEO_PKGM_001_BR

TP Id	"TP_MEC_MEO_PKGM_001_BR"
Test Objective	Check that MEO responds with an error when it receives a malformed request for creating a new App Package
Reference	ETSI GS MEC 010-2 [3], clause 7.3.1.3.1 and table 6.2.3.2.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/apmi/v1/app_packages",
body containing
AppPkg containing
"not" appPkgName //mandatory attribute
;
;
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-3: TP_MEC_MEO_PKGM_002_OK

TP Id	"TP_MEC_MEO_PKGM_002_OK"
Test Objective	Check that MEO returns the list of App Packages when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.1.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a App_Package containing
   appPkgId indicating value ON_BOARDED_APP_PKG_ID,
   appDId indicating value APPD_ID;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/apmi/v1/app_packages";
   from the MEC_OSS
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      OnboardedAppPkgInfoList containing
         OnboardedAppPkgInfo containing
            appPkgId set to ON_BOARDED_APP_PKG_ID,
            appDId set to APPD_ID,
            appName set to any_value,
            appSoftwareVersion set to any_value,
            appDVersion set to any_value,
            checksum set to any_value,
            attribute softwareImages,
            operationalState set to any_value,
            usageState set to any_value,
            attribute _links
         ;
      ;
   ;
; to the MEC_OSS
    }
}
```

Final Conditions

Table 6.1.1-4: TP_MEC_MEO_PKGM_002_BR

TP Id	"TP_MEC_MEO_PKGM_002_BR"
Test Objective	Check that MEO responds with an error when it receives a malformed request for retrieving the list of existing App Packages
Reference	ETSI GS MEC 010-2 [3], clause 7.3.1.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages",
query_parameters containing
operationalStatus indicating value any_value // the query parameter should be operationalState not operationalStatus
;
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-5: TP_MEC_MEO_PKGM_003_OK

TP Id	"TP_MEC_MEO_PKGM_003_OK"
Test Objective	Check that MEO returns the an App Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.2.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID,
appDId indicating value APPD_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{ON_BOARDED_APP_PKG_ID}";
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
OnboardedAppPkgInfo containing
appPkgId set to ON_BOARDED_APP_PKG_ID,
appDId set to APPD_ID,
appName set to any_value,
appSoftwareVersion set to any_value,
appDVersion set to any_value,
checksum set to any_value,
attribute softwareImages,
operationalState set to any_value,
usageState set to any_value,
attribute _links
;
;
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-6: TP_MEC_MEO_PKGM_003_NF

TP Id	"TP_MEC_MEO_PKGM_003_NF"
Test Objective	Check that MEO responds with an error when it receives a request for retrieving a App Package referred with a wrong ID
Reference	ETSI GS MEC 010-2 [3], clause 7.3.2.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Package containing
appPkgId indicating value NON_EXISTENT_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{NON_EXISTENT_APP_PKG_ID}" //wrong AppPackageID
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-7: TP_MEC_MEO_PKGM_004_OK

TP Id	"TP_MEC_MEO_PKGM_004_OK"
Test Objective	Check that MEO deletes an App Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.2.3.4
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/apmi/v1/app_packages/{ON_BOARDED_APP_PKG_ID}";
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "204 No Content"
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-8: TP_MEC_MEO_PKGM_004_NF

TP Id	"TP_MEC_MEO_PKGM_004_NF"
Test Objective	Check that MEO responds with an error when it receives a request for deleting an App Package referred with a wrong ID
Reference	ETSI GS MEC 010-2 [3], clause 7.3.2.3.4 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Package containing
appPkgId indicating value NON_EXISTENT_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/apmi/v1/app_packages/{NON_EXISTENT_APP_PKG_ID}" //wrong AppPackageID
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-9: TP_MEC_MEO_PKGM_005_OK

TP Id	"TP_MEC_MEO_PKGM_005_OK"
Test Objective	Check that MEO changes the status of an App Package from INITIAL_OP_STATE with an operation of type OPERATION_VALUE when requested, with the following possible combinations: • ENABLED, DISABLE • DISABLED, ENABLE • DELETION_PENDING, ABORT
Reference	ETSI GS MEC 010-2 [3], clause 7.3.2.3.3
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID,
operationalState set to INITIAL_OP_STATE;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "/apmi/v1/app_packages/{ON_BOARDED_APP_PKG_ID}",
uri_parameter appPkgOperation indicating value OPERATION_VALUE;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK";
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-10: TP_MEC_MEO_PKGM_005_NF

TP Id	"TP_MEC_MEO_PKGM_005_NF"
Test Objective	Check that MEO responds with an error when it receives a request for updating an App Package referred with a wrong ID
Reference	ETSI GS MEC 010-2 [3], clause 7.3.2.3.4 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Package containing
appPkgId indicating value NON_EXISTENT_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "/apmi/v1/app_packages/{NON_EXISTENT_APP_PKG_ID}", //wrong AppPackageID
uri_parameter appPkgOperation indicating value "ENABLE"
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-11: TP_MEC_MEO_PKGM_006_OK

TP Id	"TP_MEC_MEO_PKGM_006_OK"
Test Objective	Check that MEO returns the Application Descriptor contained on a on-boarded Application Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.6.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{ON_BOARDED_APP_PKG_ID}/appDId",
accept set to ACCEPTED_CONTENT_TYPE
;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE_ANY_CONTENT containing
status set to "200 OK",
content_type set to ACCEPTED_CONTENT_TYPE,
body set to FILE
;
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-12: TP_MEC_MEO_PKGM_006_NF

TP Id	"TP_MEC_MEO_PKGM_006_NF"
Test Objective	Check that MEO responds with an error when it receives a request to retrieve an application descriptor referred with a wrong app package ID
Reference	ETSI GS MEC 010-2 [3], clause 7.3.6.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Package containing
appPkgId indicating value NON_EXISTENT_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "/apmi/v1/app_packages/{NON_EXISTENT_APP_PKG_ID}/appDId"
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-13: TP_MEC_MEO_PKGM_007_OK

TP Id	"TP_MEC_MEO_PKGM_007_OK"
Test Objective	Check that MEO service sends a Application Package Subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.3.3.1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/apmi/v1/subscriptions"
body containing
AppPkgSubscription containing
callbackUri set to URI,
subscriptionType set to "ON_BOARDING"
;
;
;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 Created",
body containing
AppPkgSubscriptionInfo containing
subscriptionId set to any_value,
subscriptionType set to "ON_BOARDING",
callbackUri set to URI,
attribute _links
;
;
;
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-14: TP_MEC_MEO_PKGM_007_BR

TP Id	"TP_MEC_MEO_PKGM_007_BR"
Test Objective	Check that MEO service sends an error when it receives a malformed request for creating a new subscription on AppPackages
Reference	ETSI GS MEC 010-2 [3], clause 7.3.3.3.1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a POST containing
uri indicating value "/apmi/v1/subscriptions",
body containing
AppPkgSubscription containing
callbackUri set to URI,
subscriptionType set to "ONBOARDING" // Enum should be "ON-BOARDING"
;
;
;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
;
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-15: TP_MEC_MEO_PKGM_008_OK

TP Id	"TP_MEC_MEO_PKGM_008_OK"
Test Objective	Check that MEO service returns the list of Application Package Subscriptions when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.3.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a Subscription containing
   subscriptionId indicating value SUBSCRIPTION_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/apmi/v1/subscriptions"
;
from the MEC_OSS
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      AppPkgSubscriptionInfoList containing
         AppPkgSubscriptionInfo containing
            subscriptionId set to SUBSCRIPTION_ID,
            subscriptionType set to any_value,
            callbackUri set to any_value,
            _links containing
               self set to "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
            ;
         ;
      ;
   ;
;
to the MEC_OSS
    }
}
```

Final Conditions

Table 6.1.1-16: TP_MEC_MEO_PKGM_009_OK

TP Id	"TP_MEC_MEO_PKGM_009_OK"
Test Objective	Check that MEO service returns an Application Package Subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.4.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
AppPkgSubscriptionInfo containing
subscriptionId set to SUBSCRIPTION_ID,
subscriptionType set to any_value,
callbackUri set to any_value,
_links containing
self set to "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
;
;
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-17: TP_MEC_MEO_PKGM_009_NF

TP Id	"TP_MEC_MEO_PKGM_009_NF"
Test Objective	Check that MEO service sends an error when it receives a query for a subscription on AppPackages with a wrong identifier
Reference	ETSI GS MEC 010-2 [3], clause 7.3.4.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a Subscription containing
subscriptionId indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
;
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-18: TP_MEC_MEO_PKGM_010_OK

TP Id	"TP_MEC_MEO_PKGM_010_OK"
Test Objective	Check that MEO service deletes an Application Package Subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.4.3.4
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "204 No Content"
;
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-19: TP_MEC_MEO_PKGM_010_NF

TP Id	"TP_MEC_MEO_PKGM_010_NF"
Test Objective	Check that MEO service sends an error when it receives a deletion request for a subscription on AppPackages with a wrong identifier
Reference	ETSI GS MEC 010-2 [3], clause 7.3.4.3.4
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a Subscription containing
subscriptionId indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/apmi/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
;
to the MEC_OSS
}
}	

Final Conditions	

Table 6.1.1-20: TP_MEC_MEO_PKGM_011_OK

TP Id	"TP_MEC_MEO_PKGM_011_OK"
Test Objective	Check that the MEO service sends a application package notification if the MEO service has an associated subscription and the event is generated
Reference	ETSI GS MEC 010-2 [3], clause 7.3.5.1
PICS Selection	PIC_APP_PACKAGE_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a Subscriptions containing
subscriptionId set to SUBSCRIPTION_ID,
subscriptionType set to "ON_BOARDING",
callbackUri set to CALLBACK_URI
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a on_boarding_event containing
subscriptionId set to SUBSCRIPTION_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationId set to any_value,
notificationType indicating value "ON_BOARDING",
subscriptionId set to SUBSCRIPTION_ID,
timeStamp set to any_value,
appPkgId set to any_value,
appDId set to any_value,
_links containing
self set to "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
; to the MEC_SUB
}
}	

Final Conditions	

6.1.2 UE Application Contexts (UEAPPCTX)

Table 6.1.2-1: TP_MEC_MEO_UEAPPCTX_001_OK

TP Id	"TP_MEC_MEO_UEAPPCTX_001_OK"
Test Objective	Check that the IUT acknowledges the creation of the application context when requested by an UE Application
Reference	ETSI GS MEC 016 [9], clause 7.4.3.4
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "mx2/v2/app_contexts",
body containing
AppContext containing
appInfo containing
appName set to APP_NAME
;
;
;
;
from the DEV_APP
}
then {
// ETSI GS MEC 016, clause 7.4.3.4
the IUT sends a HTTP_RESPONSE containing
status_code set to "201 Created",
Location set to LOCATION,
body containing
AppContext containing
appInfo containing
appName set to APP_NAME
;
;
;
;
to the DEV_APP
}
}	

Final Conditions	

Table 6.1.2-2: TP_MEC_MEO_UEAPPCTX_001_BR

TP Id	"TP_MEC_MEO_UEAPPCTX_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 016 [9], clause 7.4.3.4
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "mx2/v2/app_contexts",
   body containing
      AppContext containing
         appInfo containing
            // Wrong parameter name should trigger an error response.
            app set to APP_NAME
         ;
      ;
   ;
;
from the DEV_APP
    }
    then {
            // ETSI GS MEC 016, clause 7.4.3.4
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the DEV_APP
    }
}
```

Final Conditions

Table 6.1.2-3: TP_MEC_MEO_UEAPPCTX_002_OK

TP Id	"TP_MEC_MEO_UEAPPCTX_002_OK"
Test Objective	Check that the IUT updates the application callback reference when commanded by an UE Application
Reference	ETSI GS MEC 016 [9], clause 7.5.3.2
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP having a appContext containing
contextId indicating value CONTEXT_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "mx2/v2/app_contexts/{CONTEXT_ID}"
body containing
AppContext containing
callbackReference indicating value some_uri
;
;
;
from the DEV_APP
}
then {
// ETSI GS MEC 016, clause 7.5.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the DEV_APP
}
}	

Final Conditions	

Table 6.1.2-4: TP_MEC_MEO_UEAPPCTX_002_BR

TP Id	"TP_MEC_MEO_UEAPPCTX_002_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 016 [9], clause 7.5.3.2
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP having a appContext containing
contextId indicating value CONTEXT_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "mx2/v2/app_contexts/{CONTEXT_ID}"
body containing
AppContext containing
// A parameter not legible for update should trigger an error response.
associateUeAppId indicating value some_value
;
;
;
from the DEV_APP
}
then {
// ETSI GS MEC 017, clause 7.5.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the DEV_APP
}
}	

Final Conditions	

Table 6.1.2-5: TP_MEC_MEO_UEAPPCTX_002_NF

TP Id	"TP_MEC_MEO_UEAPPCTX_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 016 [9], clause 7.5.3.2
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP "not" having a appContext containing
contextId indicating value NON_EXISTENT_CONTEXT_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "mx2/v2/app_contexts/{NON_EXISTENT_CONTEXT_ID}",
body containing
AppContext containing
callbackReference indicating value some_uri
;
;
;
from the DEV_APP
}
then {
// ETSI GS MEC 017, clause 7.5.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the DEV_APP
}
}	

Final Conditions	

Table 6.1.2-6: TP_MEC_MEO_UEAPPCTX_003_OK

TP Id	"TP_MEC_MEO_UEAPPCTX_003_OK"
Test Objective	Check that the IUT deletes the application context when commanded by an UE Application
Reference	ETSI GS MEC 016 [9], clause 7.5.3.5
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP having a appContext containing
contextId indicating value CONTEXT_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "mx2/v2/app_contexts/{CONTEXT_ID}"
;
from the DEV_APP
}
then {
// ETSI GS MEC 016, clause 7.5.3.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the DEV_APP
}
}	

Final Conditions	

Table 6.1.2-7: TP_MEC_MEO_UEAPPCTX_003_NF

TP Id	"TP_MEC_MEO_UEAPPCTX_003_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 016 [9], clause 7.5.3.5
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP "not" having a appContext containing
contextId indicating value NON_EXISTENT_CONTEXT_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "mx2/v2/app_contexts/{NON_EXISTENT_CONTEXT_ID}"
;
from the DEV_APP
}
then {
// ETSI GS MEC 016, clause 7.5.3.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the DEV_APP
}
}	

Final Conditions	

6.1.3 UE Applications (UEAPPS)

Table 6.1.3-1: TP_MEC_MEO_UEAPPS_001_OK

TP Id	"TP_MEC_MEO_UEAPPS_001_OK"
Test Objective	Check that the IUT responds with the list of user applications available when requested by an UE Application
Reference	ETSI GS MEC 016 [9], clause 7.3.3.1
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP having a appInfo containing
appName indicating value APP_NAME
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mx2/v2/app_list",
query_parameters containing
appName indicating value APP_NAME
;
;
from the DEV_APP
}
then {
// ETSI GS MEC 016, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
ApplicationList containing
appInfo containing
appName set to APP_NAME
;
;
;
to the DEV_APP
}
}	

Final Conditions	

Table 6.1.3-2: TP_MEC_MEO_UEAPPS_001_BR

TP Id	"TP_MEC_MEO_UEAPPS_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 016 [9], clause 7.3.3.1
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP having a appInfo containing
appName indicating value APP_NAME
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mx2/v2/app_list",
query_parameters containing
appName indicating value APP_NAME,
// Wrong name should trigger an error response.
serviceCont indicating value INVALID_VALUE
;
;
from the DEV_APP
}
then {
// ETSI GS MEC 016, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the DEV_APP
}
}	

Final Conditions	

Table 6.1.3-3: TP_MEC_MEO_UEAPPS_001_NF

TP Id	"TP_MEC_MEO_UEAPPS_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 016 [9], clause 7.3.3.1
PICS Selection	PIC_MEC_SYSTEM and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the DEV_APP "not" having a appInfo containing
appName indicating value NON_EXISTING_APP_NAME
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mx2/v2/app_list",
query_parameters containing
// Wrong value should trigger an error response.
appName indicating value NON_EXISTING_APP_NAME
;
;
from the DEV_APP
}
then {
// ETSI GS MEC 016, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the DEV_APP
}
}	

Final Conditions

6.1.4 Granting (GRANT)

Table 6.1.4-1: TP_MEC_MEO_GRANT_001_OK

TP Id	"TP_MEC_MEO_GRANT_001_OK"
Test Objective	Check that MEO sends a synchronous grant response when a grant request is requested
Reference	ETSI GS MEC 010-2 [3], clause 7.6.1.3.1, tables 6.2.4.2.2-1 and 6.2.4.4.2-1
PICS Selection	PIC_GRANTS_MANAGEMENT
Initial Conditions	

with {
the IUT having a app_instance containing
appInstanceID indicating value APP_INSTANCE_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/granting/v1/grants",
body containing
GrantRequest containing
appInstanceId set to APP_INSTANCE_ID,
appLcmOpOccId set to any_value,
appDId set to any_value,
operation set to OPERATION_TYPE //Shall be one from - INSTANTIATE - OPERATE - TERMINATE
attribute _links
;
;
;
from the MEPM
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 Created",
body containing
Grant containing
id set to any_value,
appInstanceId set to APP_INSTANCE_ID,
appLcmOpOccId set to any_value,
attribute _links
;
;
; to the MEPM
}
}	

Final Conditions	

Table 6.1.4-2: TP_MEC_MEO_GRANT_001_BR

TP Id	"TP_MEC_MEO_GRANT_001_BR"
Test Objective	Check that MEO responds with an error when it receives a malformed request when a new grant request is performed
Reference	ETSI GS MEC 010-2 [3], clause 7.6.1.3.2 and table 6.2.4.4.2-1
PICS Selection	PIC_GRANTS_MANAGEMENT
Initial Conditions	

with {
the IUT having a app_instance containing
appInstanceID indicating value APP_INSTANCE_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/granting/v1/grants",
body containing
GrantRequest containing
appInstanceId set to APP_INSTANCE_ID,
appLcmOpOccId set to any_value,
appDId set to any_value,
operationERROR set to OPERATION_TYPE //parameter should be named operation not operationType
attribute _links
;
;
; from the MEPM
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEPM
}
}	

Final Conditions	

Table 6.1.4-3: TP_MEC_MEO_GRANT_002_OK

TP Id	"TP_MEC_MEO_GRANT_002_OK"
Test Objective	Check that MEO sends a asynchronous grant response when a grant request is requested
Reference	ETSI GS MEC 010-2 [3], clause 7.6.1.3.1 and table 6.2.4.2.2-1
PICS Selection	PIC_GRANTS_MANAGEMENT
Initial Conditions	

with {
the IUT having a app_instance containing
appInstanceID indicating value APP_INSTANCE_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/granting/v1/grants",
body containing
GrantRequest containing
appInstanceId set to APP_INSTANCE_ID,
appLcmOpOccId set to any_value,
appDId set to any_value,
operation set to OPERATION_TYPE //Shall be one from - INSTANTIATE - OPERATE - TERMINATE
attribute _links
;
;
;
from the MEPM
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "202 Accepted",
headers containing
Location set to "/granting/v1/grants/{GRANTING_ID}"
;
; to the MEPM
}
}	

Final Conditions	

Table 6.1.4-4: TP_MEC_MEO_GRANT_003_OK

TP Id	"TP_MEC_MEO_GRANT_003_OK"
Test Objective	Check that MEO sends the status of a grant request when a query on a granting ID is performed
Reference	ETSI GS MEC 010-2 [3], clause 7.6.1.3.2
PICS Selection	PIC_GRANTS_MANAGEMENT
Initial Conditions	

with {
the IUT having a grant containing
id indicating value GRANTING_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/granting/v1/grants/{GRANTING_ID}"
;
from the MEPM
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "202 Accepted",
headers containing
Location set to "/granting/v1/grants/{GRANTING_ID}"
;
; to the MEPM
}
}	

Final Conditions	

Table 6.1.4-5: TP_MEC_MEO_GRANT_003_NF

TP Id	"TP_MEC_MEO_GRANTS_003_NF"
Test Objective	Check that MEO responds with an error when it receives a request for returning a grant referred with a wrong ID
Reference	ETSI GS MEC 010-2 [3], clause 7.6.1.3.2
PICS Selection	PIC_GRANTS_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a grant containing
id indicating value GRANTING_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/granting/v1/grants/{GRANTING_ID}"
; from the MEPM
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEPM
}
}	

Final Conditions	

Table 6.1.4-6: TP_MEC_MEO_GRANT_004_OK

TP Id	"TP_MEC_MEO_GRANT_004_OK"
Test Objective	Check that MEO sends the status of a grant request when a query on a granting ID is performed
Reference	ETSI GS MEC 010-2 [3], clause 7.6.1.3.2 and table 6.2.4.4.2-1
PICS Selection	PIC_GRANTS_MANAGEMENT
Initial Conditions	

with {
the IUT having a grant containing
id indicating value GRANTING_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/granting/v1/grants/{GRANTING_ID}"
;
from the MEPM
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
Grant containing
id set to GRANTING_ID,
appInstanceId set to any_value,
appLcmOpOccId set to any_value,
attribute _links
;
;
; to the MEPM
}
}	

Final Conditions	

6.2 Multi-access Edge Platform Manager (MEO)

6.2.1 App Package Management (PKGM)

Table 6.2.1-1: TP_MEC_MEPM_PKGM_001_OK

TP Id	"TP_MEC_MEPM_PKGM_001_OK"
Test Objective	Check that MEPM returns the list of App Packages when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.1.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID,
appDId indicating value APPD_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages";
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
OnboardedAppPkgInfoList containing
OnboardedAppPkgInfo containing
appPkgId set to ON_BOARDED_APP_PKG_ID,
appDId set to APPD_ID,
appName set to any_value,
appSoftwareVersion set to any_value,
appDVersion set to any_value,
checksum set to any_value,
attribute softwareImages,
operationalState set to any_value,
usageState set to any_value,
attribute _links
;
;
;
; to the MEO
}
}	

Final Conditions	

Table 6.2.1-2: TP_MEC_MEPM_PKGM_001_BR

TP Id	"TP_MEC_MEPM_PKGM_001_BR"
Test Objective	Check that MEPM responds with an error when it receives a malformed request for requesting the list of existing App Packages
Reference	ETSI GS MEC 010-2 [3], clause 7.4.1.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages",
query_parameters containing
operationalStatus indicating value ENABLED // the query parameter should be operationalState not operationalStatus
;
; from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEO
}
}	

Final Conditions	

Table 6.2.1-3: TP_MEC_MEPM_PKGM_002_OK

TP Id	"TP_MEC_MEPM_PKGM_002_OK"
Test Objective	Check that MEPM returns the an App Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.2.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{ON_BOARDED_APP_PKG_ID}";
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
OnboardedAppPkgInfo containing
appPkgId set to ON_BOARDED_APP_PKG_ID,
appDId set to any_value,
appName set to any_value,
appSoftwareVersion set to any_value,
appDVersion set to any_value,
checksum set to any_value,
attribute softwareImages,
operationalState set to any_value,
usageState set to any_value,
attribute _links
;
;
; to the MEO
}
}	

Final Conditions	

Table 6.2.1-4: TP_MEC_MEPM_PKGM_002_NF

TP Id	"TP_MEC_MEPM_PKGM_002_NF"
Test Objective	Check that MEPM responds with an error when it receives a request for returning a App Package referred with a wrong ID
Reference	ETSI GS MEC 010-2 [3], clause 7.4.2.3.2 and table 6.2.3.3.2-1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Package containing
appPkgId indicating value NON_EXISTENT_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{NON_EXISTENT_APP_PKG_ID}"
; from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEO
}
}	

Final Conditions	

Table 6.2.1-5: TP_MEC_MEPM_PKGM_003_OK

TP Id	"TP_MEC_MEPM_PKGM_003_OK"
Test Objective	Check that MEPM returns the Application Descriptor contained on a on-boarded Application Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.4.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{appPkgId}/app_descriptor",
accept set to ACCEPTED_CONTENT_TYPE
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body set to FILE
;
to the MEO
}
}	

Final Conditions	

Table 6.2.1-6: TP_MEC_MEPM_PKGM_003_NF

TP Id	"TP_MEC_MEPM_PKGM_003_NF"
Test Objective	Check that MEPM responds with an error when it receives a request for returning a App Descriptor referred with a wrong App Package ID
Reference	ETSI GS MEC 010-2 [3], clause 7.4.4.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Package containing
appPkgId indicating value NON_EXISTENT_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{NON_EXISTENT_APP_PKG_ID}/app_descriptor"
; from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEO
}
}	

Final Conditions	

Table 6.2.1-7: TP_MEC_MEPM_PKGM_004_OK

TP Id	"TP_MEC_MEPM_PKGM_004_OK"
Test Objective	Check that MEPM returns the content on a on-boarded Application Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.3.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Package containing
appPkgId indicating value ON_BOARDED_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{ON_BOARDED_APP_PKG_ID}/appPkgContent"
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
AppD set to FILE
;
;
to the MEO
}
}	

Final Conditions	

Table 6.2.1-8: TP_MEC_MEPM_PKGM_004_NF

TP Id	"TP_MEC_MEPM_PKGM_004_NF"
Test Objective	Check that MEPM responds with an error when it receives a request for returning a application package content referred with a wrong App Package ID
Reference	ETSI GS MEC 010-2 [3], clause 7.4.4.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Package containing
appPkgId indicating value NON_EXISTENT_APP_PKG_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/app_packages/{NON_EXISTENT_APP_PKG_ID}/appPkgContent"
; from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEO
}
}	

Final Conditions	

Table 6.2.1-9: TP_MEC_MEPM_PKGM_005_OK

TP Id	"TP_MEC_MEPM_PKGM_005_OK"
Test Objective	Check that MEPM service returns an application package subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.5.3.1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/apmi/v1/subscriptions"
body containing
AppPkgSubscription containing
callbackUri set to URI,
subscriptionType set to SUBSCRIPTION_TYPE
;
;
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 Created",
body containing
AppPkgSubscriptionInfo containing
subscriptionId set to any_value,
subscriptionType set to SUBSCRIPTION_TYPE,
callbackUri set to URI,
attribute _links
;
;
;
to the MEO
}
}	

Final Conditions	

Table 6.2.1-10: TP_MEC_MEPM_PKGM_005_BR

TP Id	"TP_MEC_MEPM_PKGM_005_BR"
Test Objective	Check that MEPM service sends an error when it receives a malformed request for creating a new subscription on AppPackages
Reference	ETSI GS MEC 010-2 [3], clause 7.4.5.3.1
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a POST containing
uri indicating value "/apmi/v1/subscriptions",
body containing
AppPkgSubscription containing
subscriptionType set to "ONBOARDING" // Enum should be "ON-BOARDING"
;
;
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
;
to the MEO
}
}	

Final Conditions	

Table 6.2.1-11: TP_MEC_MEPM_PKGM_006_OK

TP Id	"TP_MEC_MEPM_PKGM_006_OK"
Test Objective	Check that MEPM service returns the list of Application Package Subscriptions when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.5.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/subscriptions"
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
AppPkgSubscriptionInfoList containing
AppPkgSubscriptionInfo containing
subscriptionId set to SUBSCRIPTION_ID,
subscriptionType set to any_value,
callbackUri set to any_value,
_links containing
self set to "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
;
;
to the MEO
}
}	

Final Conditions

Table 6.2.1-12: TP_MEC_MEPM_PKGM_007_OK

TP Id	"TP_MEC_MEPM_PKGM_007_OK"
Test Objective	Check that MEPM service returns an Application Package Subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.6.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
AppPkgSubscriptionInfo containing
subscriptionId set to SUBSCRIPTION_ID,
subscriptionType set to any_value,
callbackUri set to any_value,
_links containing
self set to "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
;
;
to the MEO
}
}	

Final Conditions	

Table 6.2.1-13: TP_MEC_MEPM_PKGM_007_NF

TP Id	"TP_MEC_MEPM_PKGM_007_NF"
Test Objective	Check that MEPM service sends an error when it receives a query for a subscription on AppPackages with a wrong identifier
Reference	ETSI GS MEC 010-2 [3], clause 7.4.6.3.2
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a Subscription containing
subscriptionId indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/apmi/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
;
to the MEO
}
}	

Final Conditions

Table 6.2.1-14: TP_MEC_MEPM_PKGM_008_OK

TP Id	"TP_MEC_MEPM_PKGM_008_OK"
Test Objective	Check that MEPM service deletes an Application Package Subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.4.6.3.4
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "204 No Content"
;
to the MEO
}
}	

Final Conditions	

Table 6.2.1-15: TP_MEC_MEPM_PKGM_008_NF

TP Id	"TP_MEC_MEPM_PKGM_008_NF"
Test Objective	Check that MEPM service sends an error when it receives a deletion request for a subscription on AppPackages with a wrong identifier
Reference	ETSI GS MEC 010-2 [3], clause 7.4.6.3.4
PICS Selection	PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a Subscription containing
subscriptionId indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/apmi/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEO
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
;
to the MEO
}
}	

Final Conditions	

Table 6.2.1-16: TP_MEC_MEPM_PKGM_009_OK

TP Id	"TP_MEC_MEPM_PKGM_009_OK"
Test Objective	Check that the MEPM service sends a application package notification if the MEPM service has an associated subscription and the event is generated
Reference	ETSI GS MEC 010-2 [3], clause 7.4.7.3.1
PICS Selection	PIC_APP_PACKAGE_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId set to SUBSCRIPTION_ID,
subscriptionType set to SUBSCRIPTION_TYPE,
callbackUri set to CALLBACK_URI,
attribute _links
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a on_boarding_event containing
notificationId set to NOTIFICATION_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationId set to NOTIFICATION_ID,
notificationType indicating value any_value,
subscriptionId set to SUBSCRIPTION_ID,
timeStamp set to any_value,
appPkgId set to any_value,
appDId set to any_value,
_links containing
self set to "/apmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
; to the MEC_SUB
}
}	

Final Conditions	

6.3 Generic MEC API Producer (MEX)

6.3.1 Generic feature (Any)

Table 6.3.1-1: TP_MEC_MEX_ANY_001_NT

TP Id	"TP_MEC_MEX_ANY_001_NT"
Test Objective	Check that a MEC API provider responds with an error when it receives a request without token
Reference	ETSI GS MEC 009 [2], clause 6.16.1
PICS Selection	
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a HTTP_REQUEST containing
uri indicating value ACCEPTABLE_URI,
"not" authorization
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "401 Unauthorized"
; to the MEC_CONSUMER
}
}	

Final Conditions

Table 6.3.1-2: TP_MEC_MEX_ANY_001_WT

TP Id	"TP_MEC_MEX_ANY_001_WT"
Test Objective	Check that a MEC API provider responds with an error when it receives a request with a wrong token
Reference	ETSI GS MEC 009 [2], clause 6.16.1
PICS Selection	
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a HTTP_REQUEST containing
uri indicating value ACCEPTABLE_URI,
headers containing
authorization set to NOT_VALID_TOKEN
;
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "401 Unauthorized"
; to the MEC_CONSUMER
}
}	

Final Conditions	

6.3.2 Lifecycle management (LCM)

Table 6.3.2-1: TP_MEC_MEX_LCM_001_OK

TP Id	"TP_MEC_MEX_LCM_001_OK"
Test Objective	Check that MEC API provider creates a new App Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.1.3.1, tables 6.2.2.3.2-1 and 6.2.2.4.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT being_in idle_state
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "/alcmi/v1/app_instances",
   body containing
      CreateAppInstanceRequest containing
         appDId set to APP_D_ID
      ;
   ;
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "201 Created",
   body containing
      AppInstanceInfo containing
         appInstanceId set to any_value,
         appDId set to APP_D_ID,
         appProvider set to any_value,
         appName set to any_value,
         appSoftVersion set to any_value,
         appDVersion set to any_value,
         appPkgId set to any_value,
         instantiationState set to NOT_INSTANTIATED,
         attribute _links
      ;
   ;
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-2: TP_MEC_MEX_LCM_001_BR

TP Id	"TP_MEC_MEX_LCM_001_BR"
Test Objective	Check that MEC API providersends an error when it receives a malformed request for the creation of a new App Instance
Reference	ETSI GS MEC 010-2 [3], clause 7.5.1.3.1, tables 6.2.2.3.2-1 and 6.2.2.4.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/app_instances",
body containing
CreateAppInstanceRequest containing
appDDDId set to APP_D_ID
;
;
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-3: TP_MEC_MEX_LCM_002_OK

TP Id	"TP_MEC_MEX_LCM_002_OK"
Test Objective	Check that MEC API provider retrieves the list of App instances when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.1.3.2 and table 6.2.2.4.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a App_Instance containing
   appInstanceId indicating value APP_INSTANCE_ID;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/alcmi/v1/app_instances"
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      AppInstanceInfoList containing                        // Definition of AppInstanceInfoList is
missing. How to handle with arrays?
         AppInstanceInfo containing
            appInstanceId set to APP_INSTANCE_ID,
            appDId set to any_value,
            appProvider set to any_value,
            appName set to any_value,
            appSoftVersion set to any_value,
            appDVersion set to any_value,
            appPkgId set to any_value,
            instantiationState set to any_value,
            attribute _links
         ;
      ;
   ;
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-4: TP_MEC_MEX_LCM_003_OK

TP Id	"TP_MEC_MEX_LCM_003_OK"
Test Objective	Check that MEC API provider retrieves an App Package when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.2.3.2 and table 6.2.2.4.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a App_Instance containing
   appInstanceId indicating value APP_INSTANCE_ID;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}"
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      AppInstanceInfo containing
         appInstanceId set to APP_INSTANCE_ID,
         appDId set to any_value,
         appProvider set to any_value,
         appName set to any_value,
         appSoftVersion set to any_value,
         appDVersion set to any_value,
         appPkgId set to any_value,
         instantiationState set to any_value,
         attribute _links
      ;
   ;
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-5: TP_MEC_MEX_LCM_003_NF

TP Id	"TP_MEC_MEX_LCM_003_NF"
Test Objective	Check that MEC API providerfails on retrieving an App Instance when requested using wrong appInstanceId
Reference	ETSI GS MEC 010-2 [3], clause 7.5.2.3.2 and table 6.2.2.4.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT "not" having a App_Instance containing
   appInstanceId indicating value NON_EXISTENT_APP_INSTANCE_ID;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/alcmi/v1/app_instances/{NON_EXISTENT_APP_INSTANCE_ID}"
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "404 Not Found"
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-6: TP_MEC_MEX_LCM_004_OK

TP Id	"TP_MEC_MEX_LCM_004_OK"
Test Objective	Check that MEC API provider service deletes an App Instance when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.2.3.4
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Instance containing
appInstanceId indicating value APP_INSTANCE_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "204 No Content"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-7: TP_MEC_MEX_LCM_004_NF

TP Id	"TP_MEC_MEX_LCM_004_NF"
Test Objective	Check that MEC API providerfails on deletion of an App Instance when requested using wrong appInstanceId
Reference	ETSI GS MEC 010-2 [3], clause 7.5.2.3.2 and table 6.2.2.4.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Instance containing
appInstanceId indicating value NON_EXISTENT_APP_INSTANCE_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/alcmi/v1/app_instances/{NON_EXISTENT_APP_INSTANCE_ID}"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-8: TP_MEC_MEX_LCM_005_OK

TP Id	"TP_MEC_MEX_LCM_005_OK"
Test Objective	Check that MEC API provider service instantiates an App Instance when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.6.3.1 and table 6.2.2.7.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a App_Instance containing
   appInstanceId indicating value APP_INSTANCE_ID,
   instantiationState indicating value NOT_INSTANTIATED
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}/instantiate",
   body containing
      InstantiateAppRequest containing
         appInstanceId set to APP_INSTANCE_ID,
         selectedMECHostInfo set to SELECTED_MEC_HOST_INFO
      ;
   ;
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "202 Accepted",
   headers containing
      Location set to any_value
   ;
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-9: TP_MEC_MEX_LCM_005_BR

TP Id	"TP_MEC_MEX_LCM_005_BR"
Test Objective	Check that MEC API providerservice fails to instantiate an App Instance when it receives a malformed request
Reference	ETSI GS MEC 010-2 [3], clause 7.5.6.3.1 and table 6.2.2.7.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a App_Instance containing
   appInstanceId indicating value APP_INSTANCE_ID,
   instantiationState indicating value NOT_INSTANTIATED
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}/instantiate",
   body containing
      InstantiateAppRequest containing
         appERRORId set to APP_INSTANCE_ID        //wrong name of the parameter
      ;
   ;
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "400 Bad request"
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-10: TP_MEC_MEX_LCM_005_NF

TP Id	"TP_MEC_MEX_LCM_005_NF"
Test Objective	Check that MEC API providerservice fails to instantiate an App Instance when it receives a request related to a not existing App Instance
Reference	ETSI GS MEC 010-2 [3], clause 7.5.6.3.1 and table 6.2.2.7.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Instance containing
appInstanceId indicating value NON_EXISTENT_APP_INSTANCE_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/app_instances/{NON_EXISTENT_APP_INSTANCE_ID}/instantiate"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-11: TP_MEC_MEX_LCM_006_OK

TP Id	"TP_MEC_MEX_LCM_006_OK"
Test Objective	Check that MEC API provider service terminates an App Instance when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.7.3.1 and table 6.2.2.9.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Instance containing
appInstanceId indicating value APP_INSTANCE_ID,
instantiationState indicating value INSTANTIATED
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}/terminate",
body containing
TerminateAppRequest containing
terminationType set to TERMINATION_TYPE
;
;
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "202 Accepted",
headers containing
Location set to any_value
;
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-12: TP_MEC_MEX_LCM_006_BR

TP Id	"TP_MEC_MEX_LCM_006_BR"
Test Objective	Check that MEC API providerservice fails to terminate an App Instance when it receives a malformed request
Reference	ETSI GS MEC 010-2 [3], clause 7.5.7.3.1 and table 6.2.2.9.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a App_Instance containing
   appInstanceId indicating value APP_INSTANCE_ID,
   instantiationState indicating value INSTANTIATED
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}/terminate",
   body containing
      TerminateAppRequest containing
         terminationERRORType set to GRACEFULL        //wrong parameter
      ;
   ;
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "400 Bad request"
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-13: TP_MEC_MEX_LCM_006_NF

TP Id	"TP_MEC_MEX_LCM_006_NF"
Test Objective	Check that MEC API providerservice fails to terminate an App Instance when it receives a request related to a not existing App Instance
Reference	ETSI GS MEC 010-2 [3], clause 7.5.7.3.1 and table 6.2.2.9.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT "not" having a App_Instance containing
   appInstanceId indicating value NON_EXISTENT_APP_INSTANCE_ID;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "/alcmi/v1/app_instances/{NON_EXISTENT_APP_INSTANCE_ID}/terminate"
; from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "404 Not Found"
; to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-14: TP_MEC_MEX_LCM_007_OK

TP Id	"TP_MEC_MEX_LCM_007_OK"
Test Objective	Check that MEC API provider service changes the status of an App Instance from its INITIAL_STATE to a given FINAL_STATE, when requested. The following combinations INITIAL_STATE - FINAL_STATE are supported: - STARTED/STOP - STOPPED/START
Reference	ETSI GS MEC 010-2 [3], clause 7.5.8.3.1 and table 6.2.2.8.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Instance containing
appInstanceId indicating value APP_INSTANCE_ID,
instantiationState indicating value INSTANTIATED,
operationalState indicating value INITIAL_STATE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}/operate",
body containing
OperateAppRequest containing
changeStateTo set to FINAL_STATE
;
;
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "202 Accepted",
headers containing
Location set to any_value
;
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-15: TP_MEC_MEX_LCM_007_BR

TP Id	"TP_MEC_MEX_LCM_007_BR"
Test Objective	Check that MEC API providerservice fails to operate on an App Instance when it receives a malformed request
Reference	ETSI GS MEC 010-2 [3], clause 7.5.8.3.1 and table 6.2.2.8.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Instance containing
appInstanceId indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/app_instances/{APP_INSTANCE_ID}/operate",
body containing
OperateAppRequest containing
changeERRORStateTo set to any_value //wrong parameter
;
;
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad request"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-16: TP_MEC_MEX_LCM_007_NF

TP Id	"TP_MEC_MEX_LCM_007_NF"
Test Objective	Check that MEC API providerservice fails to change the status of an App Instance when it receives a request related to a not existing App Instance
Reference	ETSI GS MEC 010-2 [3], clause 7.5.8.3.1 and table 6.2.2.8.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Instance containing
appInstanceId indicating value NON_EXISTENT_APP_INSTANCE_ID;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/app_instances/{NON_EXISTENT_APP_INSTANCE_ID}/operate"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-17: TP_MEC_MEX_LCM_008_OK

TP Id	"TP_MEC_MEX_LCM_008_OK"
Test Objective	Check that MEC API provider service retrieves info about LCM Operation Occurrency on App Instances when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.9.1.3.2 and table 6.2.2.14.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Lcm_Op_Occ containing
appLcmOpOccId indicating value APP_LCM_OP_OCC_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/alcmi/v1/app_lcm_op_occs"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
AppLcmOpOccList containing
AppInstanceLcmOpOcc containing
appLcmOpOccId set to APP_LCM_OP_OCC_ID,
operationState set to any_value,
stateEnteredTime set to any_value,
startTime set to any_value,
lcmOperation set to any_value,
attribute _links containing
self set to "/alcmi/v1/app_lcm_op_occs/{APP_LCM_OP_OCC_ID}",
appInstance set to any_value
;
;
;
;
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-18: TP_MEC_MEX_LCM_009_OK

TP Id	"TP_MEC_MEX_LCM_009_OK"
Test Objective	Check that MEC API provider service retrieves info about LCM Operation Occurrency on an App Instance when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.10.1.3.2 and table 6.2.2.14.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a App_Lcm_Op_Occ containing
appLcmOpOccId indicating value APP_LCM_OP_OCC_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/alcmi/v1/app_lcm_op_occs/{APP_LCM_OP_OCC_ID}"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
AppLcmOpOccList containing
AppInstanceLcmOpOcc containing
appLcmOpOccId set to APP_LCM_OP_OCC_ID,
operationState set to any_value,
stateEnteredTime set to any_value,
startTime set to any_value,
lcmOperation set to any_value,
attribute _links containing
self set to "/alcmi/v1/app_lcm_op_occs/{APP_LCM_OP_OCC_ID}",
appInstance set to any_value
;
;
;
;
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-19: TP_MEC_MEX_LCM_009_NF

TP Id	"TP_MEC_MEX_LCM_009_NF"
Test Objective	Check that MEC API providerservice sends an error when it receives a query for a not existing LCM Operation Occurrency
Reference	ETSI GS MEC 010-2 [3], clause 7.5.10.1.3.2 and table 6.2.2.14.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Lcm_Op_Occ containing
appLcmOpOccId indicating value NON_EXISTENT_APP_LCM_OP_OCC_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/alcmi/v1/app_lcm_op_occs/{NON_EXISTENT_APP_LCM_OP_OCC_ID}"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-20: TP_MEC_MEX_LCM_010_OK

TP Id	"TP_MEC_MEX_LCM_010_OK"
Test Objective	Check that MEC API provider service creates a LCM Subscription when requested, where the subscription request can have SUBSCRIPTION_TYPE AppInstanceStateChangeSubscription or AppLcmOpOccStateChangeSubscription
Reference	ETSI GS MEC 010-2 [3], clause 7.5.3.3.1, tables 6.2.2.13.2-1 and 6.2.2.10.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/subscriptions"
body containing
SubscriptionRequest containing
callbackUri set to CALLBACK_URI,
subscriptionType set to SUBSCRIPTION_TYPE
;
;
;
from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 Created",
body containing
SubscriptionInfo containing
subscriptionId set to any_value,
subscriptionType set to SUBSCRIPTION_TYPE,
callbackUri set to CALLBACK_URI,
attribute _links
;
;
;
to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-21: TP_MEC_MEX_LCM_010_BR

TP Id	"TP_MEC_MEX_LCM_010_BR"
Test Objective	Check that MEC API providerservice sends an error when it receives a malformed request to create a LCM Subscription
Reference	ETSI GS MEC 010-2 [3], clause 7.5.3.3.1, tables 6.2.2.13.2-1 and 6.2.2.16.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/alcmi/v1/subscriptions"
body containing
AppInstSubscriptionRequest containing
callbackERRORUri set to CALLBACK_URI //wrong parameter
;
;
;
from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
;
to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-22: TP_MEC_MEX_LCM_011_OK

TP Id	"TP_MEC_MEX_LCM_011_OK"
Test Objective	Check that MEC API provider service sends the list of LCM Subscriptions when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.3.3.2, tables 6.2.2.13.2-1, 6.2.2.15.2-1, 6.2.2.10.2-1 and 6.2.2.16.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

```
with {
            the IUT having a Subscription containing
    subscriptionId indicating value SUBSCRIPTION_ID,
    subscriptionType indicating value SUBSCRIPTION_TYPE,
    callbackUri indicating value URI
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/alcmi/v1/subscriptions"
;
from the MEC_CONSUMER
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      SubscriptionInfo containing
        subscriptionId set to SUBSCRIPTION_ID,
            subscriptionType set to SUBSCRIPTION_TYPE,
            callbackUri set to URI,
            attribute _links
      ;
   ;
;
to the MEC_CONSUMER
    }
}
```

Final Conditions

Table 6.3.2-23: TP_MEC_MEX_LCM_012_OK

TP Id	"TP_MEC_MEX_LCM_012_OK"
Test Objective	Check that MEC API provider service sends the information about an existing LCM subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.5.4.3.2, tables 6.2.2.10.2-1 and 6.2.2.16.2-1
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId indicating value SUBSCRIPTION_ID,
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackUri indicating value URI
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/alcmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
SubscriptionInfo containing
subscriptionId set to SUBSCRIPTION_ID,
subscriptionType set to SUBSCRIPTION_TYPE,
callbackUri set to URI,
attribute _links
;
;
;
to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-24: TP_MEC_MEX_LCM_012_NF

TP Id	"TP_MEC_MEX_LCM_012_NF"
Test Objective	Check that MEC API providerservice sends an error when it receives a query for a not existing LCM Subscription
Reference	ETSI GS MEC 010-2 [3], clause 7.5.3.3.2
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Instance_Subscription containing
subscriptionId set to NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/alcmi/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-25: TP_MEC_MEX_LCM_013_OK

TP Id	"TP_MEC_MEX_LCM_013_OK"
Test Objective	Check that MEC API provider service delete an existing LCM Subscription when requested
Reference	ETSI GS MEC 010-2 [3], clause 7.3.4.3.4
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT having a Subscription containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/alcmi/v1/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "204 No Content"
;
to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-26: TP_MEC_MEX_LCM_013_NF

TP Id	"TP_MEC_MEX_LCM_013_NF"
Test Objective	Check that MEC API providerservice sends an error when it receives a deletion request for a not existing LCM Subscription
Reference	ETSI GS MEC 010-2 [3], clause 7.5.3.3.2
PICS Selection	PIC_APP_LCM_MANAGEMENT
Initial Conditions	

with {
the IUT "not" having a App_Instance_Subscription containing
subscriptionId set to NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/alcmi/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
; from the MEC_CONSUMER
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_CONSUMER
}
}	

Final Conditions	

Table 6.3.2-27: TP_MEC_MEX_LCM_014_OK

TP Id	"TP_MEC_MEX_LCM_014_OK"
Test Objective	Check that MEC API provider sends a notification to the subscriber when an application lcm change event occurs
Reference	ETSI GS MEC 010-2 [3], clause 7.5.5.3.1, tables 6.2.2.18.2-1 and 6.2.2.12.2-1
PICS Selection	PIC_APP_LCM_NOTIFICATIONS
Initial Conditions	

```
with {
            the IUT having a App_Instance_Subscription containing
   subscriptionType indicating value SUBSCRIPTION_TYPE,
   callbackUri indicating value CALLBACK_URI,
   attribute _links
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT generates a application_lcm_change_event containing
  notificationId set to NOTIFICATION_ID;
    }
    then {
            the IUT sends a vPOST containing
   uri indicating value CALLBACK_URI,
   body containing
      Notification containing
         notificationId set to NOTIFICATION_ID,
         notificationType set to NOTIFICATION_TYPE,
         subscriptionId set to any_value,
         timeStamp set to any_value,
         appInstanceId set to any_value,
         appDId set to any_value,
         attribute _links
      ;
   ;
; to the MEC_SUB
    }
}
```

Final Conditions

6.4 Services (SRV)

6.4.1 Application Mobility Service (AMS)

Table 6.4.1-1: TP_MEC_SRV_AMS_001_OK

TP Id	"TP_MEC_SRV_AMS_001_OK"
Test Objective	Check that the AMS service returns information about the registered application mobility services when requested
Reference	ETSI GS MEC 021 [10], clause 8.3.3.1
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT having a AppMobilityServiceInfo containing
appMobilityServiceId indicating value APP_MOBILITY_SERVICE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/amsi/v1/appMobilityServices",
query_parameters containing
appMobilityServiceId indicating value APP_MOBILITY_SERVICE_ID
;
; from the AMS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
AppMobilityServiceInfo containing
appMobilityServiceId indicating value APP_MOBILITY_SERVICE_ID,
attribute registeredAppMobilityService
;
;
; to the AMS_CLIENT
}
}	

Final Conditions	

Table 6.4.1-2: TP_MEC_SRV_AMS_001_BR

TP Id	"TP_MEC_SRV_AMS_001_BR"
Test Objective	Check that the AMS service returns an error when receives a query about a registered application mobility service with wrong parameters
Reference	ETSI GS MEC 021 [10], clause 8.3.3.1
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT having a AppMobilityServiceInfo containing
appMobilityServiceId indicating value APP_MOBILITY_SERVICE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/amsi/v1/appMobilityServices",
query_parameters containing
appMobilityService indicating value APP_MOBILITY_SERVICE_ID //wrong parameter, it should be appMobilityServiceId
;
; from the AMS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the AMS_CLIENT
}
}	

Final Conditions	

Table 6.4.1-3: TP_MEC_SRV_AMS_002_OK

TP Id	"TP_MEC_SRV_AMS_002_OK"
Test Objective	Check that the AMS service creates a new application mobility services when requested
Reference	ETSI GS MEC 021 [10], clause 8.3.3.4
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/amsi/v1/appMobilityServices",
body containing
RegistrationRequest containing //Note: at the moment the only element that is defined is RegistrationInfo. Not sure if they are the same.
serviceConsumerId containing
appInstanceId set to APP_INS_ID
;
;
;
; from the AMS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 Created",
body containing
AppMobilityServiceInfo containing
appMobilityServiceId set to any_value,
registeredAppMobilityService containing
serviceConsumerId containing
appInstanceId set to APP_INS_ID
;
;
;
; to the AMS_CLIENT
}
}	

Final Conditions	

Table 6.4.1-4: TP_MEC_SRV_AMS_002_BR

TP Id	"TP_MEC_SRV_AMS_002_BR"
Test Objective	Check that the AMS service sends an error when it receives a malformed request to create a new application mobility service
Reference	ETSI GS MEC 021 [10], clause 8.3.3.4
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/amsi/v1/appMobilityServices",
body containing
RegistrationRequest containing //Note: at the moment the only element that is defined is RegistrationInfo. Not sure if they are the same.
serviceConsumerId containing
appInstance set to APP_INS_ID //wrong parameter, it should be appInstanceId
;
;
;
; from the AMS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the AMS_CLIENT
}
}	

Final Conditions	

Table 6.4.1-5: TP_MEC_SRV_AMS_003_OK

TP Id	"TP_MEC_SRV_AMS_003_OK"
Test Objective	Check that the AMS service returns information about the available subscriptions when requested. Permitted SUBSCRIPTION_TYPE are: - MobilityProcedureSubscription - AdjacentAppInfoSubscription
Reference	ETSI GS MEC 021 [10], clause 8.6.3.1
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
_link containing
self indicating value SUBSCRIPTION_HREF_VALUE
;,
subscriptionType indicating value SUBSCRIPTION_TYPE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/amsi/v1/subscriptions/",
query_parameters containing
subscriptionType indicating value SUBSCRIPTION_TYPE
;
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
SubscriptionLinkList containing
subscription containing
_links containing
self indicating value SUBSCRIPTION_HREF_VALUE
;,
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackReference set to any_value
;
;
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-6: TP_MEC_SRV_AMS_003_BR

TP Id	"TP_MEC_SRV_AMS_003_BR"
Test Objective	Check that the AMS service sends an error when it receives a malformed query about the available subscriptions
Reference	ETSI GS MEC 021 [10], clause 8.6.3.1
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
_link containing
self indicating value SUBSCRIPTION_HREF_VALUE
;,
subscriptionType indicating value SUBSCRIPTION_TYPE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/amsi/v1/subscriptions/",
query_parameters containing
subscriptionTyp indicating value SUBSCRIPTION_TYPE //wrong parameter, it should be subscriptionType
;
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-7: TP_MEC_SRV_AMS_004_OK

TP Id	"TP_MEC_SRV_AMS_004_OK"
Test Objective	Check that the AMS service creates a notification subscriptions when requested. Permitted SUBSCRIPTION_TYPE are: - MobilityProcedureSubscription - AdjacentAppInfoSubscription
Reference	ETSI GS MEC 021 [10], clause 8.6.3.4
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/amsi/v1/subscriptions/",
body containing
NotificationSubscription containing
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackReference indicating value CALLBACK_REFERENCE
;
;
; from the AMS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 CREATED",
body containing
NotificationSubscription containing
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackReference indicating value CALLBACK_REFERENCE
;
;
; to the AMS_CLIENT
}
}	

Final Conditions

Table 6.4.1-8: TP_MEC_SRV_AMS_004_BR

TP Id	"TP_MEC_SRV_AMS_004_BR"
Test Objective	Check that the AMS service sends an error when it receives a malformed request to create a notification subscription Permitted SUBSCRIPTION_TYPE are: - MobilityProcedureSubscription - AdjacentAppInfoSubscription
Reference	ETSI GS MEC 021 [10], clause 8.6.3.4
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/amsi/v1/subscriptions/",
body containing
NotificationSubscription containing
subscriptionTyp indicating value SUBSCRIPTION_TYPE, //wrong parameter, it should be subscriptionType
callbackReference indicating value CALLBACK_REFERENCE
;
;
; from the AMS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the AMS_CLIENT
}
}	

Final Conditions	

Table 6.4.1-9: TP_MEC_SRV_AMS_005_OK

TP Id	"TP_MEC_SRV_AMS_005_OK"
Test Objective	Check that the AMS service returns information about a given subscription when requested. Permitted SUBSCRIPTION_TYPE are: - MobilityProcedureSubscription - AdjacentAppInfoSubscription
Reference	ETSI GS MEC 021 [10], clause 8.7.3.1
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
_link containing
self indicating value SUBSCRIPTION_HREF_VALUE
;,
callbackReference indicating value CALLBACK_REFERENCE,
subscriptionType indicating value SUBSCRIPTION_TYPE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value SUBSCRIPTION_HREF_VALUE
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
NotificationSubscription containing
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackReference indicating value CALLBACK_REFERENCE
;
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-10: TP_MEC_SRV_AMS_005_NF

TP Id	"TP_MEC_SRV_AMS_005_NF"
Test Objective	Check that the AMS service returns an error when receives a query about a not existing subscription
Reference	ETSI GS MEC 021 [10], clause 8.7.3.1
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT "not" having a AMS_subscription containing
_link containing
self indicating value NON_EXISTENT_SUBSCRIPTION_HREF_VALUE
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value NON_EXISTENT_SUBSCRIPTION_HREF_VALUE
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-11: TP_MEC_SRV_AMS_006_OK

TP Id	"TP_MEC_SRV_AMS_006_OK"
Test Objective	Check that the AMS service deletes a given subscription when requested
Reference	ETSI GS MEC 021 [10], clause 8.7.3.5
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
_link containing
self indicating value SUBSCRIPTION_HREF_VALUE
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value SUBSCRIPTION_HREF_VALUE
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "204 No Content"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-12: TP_MEC_SRV_AMS_006_NF

TP Id	"TP_MEC_SRV_AMS_006_NF"
Test Objective	Check that the AMS service sends an error when it receives a delete request for a not existing subscription
Reference	ETSI GS MEC 021 [10], clause 8.7.3.5
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT "not" having a AMS_subscription containing
_link containing
self indicating value NON_EXISTENT_SUBSCRIPTION_HREF_VALUE
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value NON_EXISTENT_SUBSCRIPTION_HREF_VALUE
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-13: TP_MEC_SRV_AMS_007_OK

TP Id	"TP_MEC_SRV_AMS_007_OK"
Test Objective	Check that the AMS service modifies a given subscription when requested. Permitted SUBSCRIPTION_TYPE are: - MobilityProcedureSubscription - AdjacentAppInfoSubscription
Reference	ETSI GS MEC 021 [10], clause 8.7.3.2
PICS Selection	PIC_AMS
Initial Conditions	

```
with {
            the IUT having a AMS_subscription containing
   _link containing
       self indicating value SUBSCRIPTION_HREF_VALUE
   ;,
   callbackReference indicating value CALLBACK_REFERENCE,
   subscriptionType indicating value SUBSCRIPTION_TYPE
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value SUBSCRIPTION_HREF_VALUE,
   body containing
      NotificationSubscription containing
         subscriptionType indicating value SUBSCRIPTION_TYPE,
         callbackReference indicating value NEW_CALLBACK_REFERENCE
      ;
   ;
; from the MEC_SUB
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      NotificationSubscription containing
         subscriptionType indicating value SUBSCRIPTION_TYPE,
         callbackReference indicating value NEW_CALLBACK_REFERENCE
      ;
   ;
; to the MEC_SUB
    }
}
```

Final Conditions

Table 6.4.1-14: TP_MEC_SRV_AMS_007_BR

TP Id	"TP_MEC_SRV_AMS_007_BR"
Test Objective	Check that the AMS service sends an error when it receives a malformed modify request for a given subscription. Permitted SUBSCRIPTION_TYPE are: - MobilityProcedureSubscription - AdjacentAppInfoSubscription
Reference	ETSI GS MEC 021 [10], clause 8.7.3.2
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
_link containing
self indicating value SUBSCRIPTION_HREF_VALUE
;,
callbackReference indicating value CALLBACK_REFERENCE,
subscriptionType indicating value SUBSCRIPTION_TYPE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value SUBSCRIPTION_HREF_VALUE,
body containing
NotificationSubscription containing
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackReferenc indicating value NEW_CALLBACK_REFERENCE //wrong field, it should be callbackReference
;
;
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-15: TP_MEC_SRV_AMS_007_NF

TP Id	"TP_MEC_SRV_AMS_007_NF"
Test Objective	Check that the AMS service sends an error when it receives a modify request for a not existing subscription
Reference	ETSI GS MEC 021 [10], clause 8.7.3.2
PICS Selection	PIC_AMS
Initial Conditions	

with {
the IUT "not" having a AMS_subscription containing
_link containing
self indicating value NON_EXISTENT_SUBSCRIPTION_HREF_VALUE
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value NON_EXISTENT_SUBSCRIPTION_HREF_VALUE
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-16: TP_MEC_SRV_AMS_008_OK

TP Id	"TP_MEC_SRV_AMS_008_OK"
Test Objective	Check that the AMS service sends an AMS notification about a mobility procedure if the AMS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 021 [10], clause 7.4.2
PICS Selection	PIC_AMS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
subscriptionType indicating value "MobilityProcedureSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteria containing
appInstanceId set to APP_INSTANCE_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a mobility_procedure_event containing
appInstanceId set to APP_INSTANCE_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "MobilityProcedureNotification",
targetAppInfo containing
appInstanceId set to APP_INSTANCE_ID
;,
mobilityStatus set to any_value
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-17: TP_MEC_SRV_AMS_009_OK

TP Id	"TP_MEC_SRV_AMS_009_OK"
Test Objective	Check that the AMS service sends an AMS notification about adjacent application instances if the AMS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 021 [10], clause 7.4.3
PICS Selection	PIC_AMS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
subscriptionType indicating value "AdjacentAppInfoSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteria containing
appInstanceId set to APP_INSTANCE_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a adjacent_app_info_event containing
appInstanceId set to APP_INSTANCE_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "AdjacentAppInfoNotification",
adjacentAppInfo containing
appInstanceId set to APP_INSTANCE_ID
;
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.1-18: TP_MEC_SRV_AMS_010_OK

TP Id	"TP_MEC_SRV_AMS_010_OK"
Test Objective	Check that the AMS service sends an AMS notification on subscription expiration if the AMS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 021 [10], clause 7.4.4
PICS Selection	PIC_AMS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a AMS_subscription containing
subscriptionType indicating value any_value, //VALID for all subscription types
callbackReference indicating value CALLBACK_URI,
_links containing
self set to SUBSCRIPTION_HREF_VALUE
;,
subscriptionId set to SUBSCRIPTION_ID,
expiryDeadline set to EXPIRY_DEADLINE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a expiry_notification_event containing
subscriptionId set to SUBSCRIPTION_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "ExpiryNotification", //this field is not in the spec.
expiryDeadline set to EXPIRY_DEADLINE
;
; to the MEC_SUB
}
}	

Final Conditions	

6.4.2 Application Service Availability Query (APPSAQ)

Table 6.4.2-1: TP_MEC_SRV_APPSAQ_001_OK

TP Id	"TP_MEC_SRV_APPSAQ_001_OK"
Test Objective	Check that the IUT responds with a list of available MEC services for a given application instance when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.15.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.15.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
ServiceInfoList containing
ServiceInfo set to any_value
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.2-2: TP_MEC_SRV_APPSAQ_001_BR

TP Id	"TP_MEC_SRV_APPSAQ_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.15.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services"
query_parameters containing
// Wrong parameter name should trigger an error response.
instance_id indicating value any_value
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.15.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.2-3: TP_MEC_SRV_APPSAQ_002_OK

TP Id	"TP_MEC_SRV_APPSAQ_002_OK"
Test Objective	Check that the IUT notifies the authorized relevant (subscribed) application instances when a new service for a given application instance is registered
Reference	ETSI GS MEC 011 [4], clause 7.15.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES and PIC_NOTIFICATIONS
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the MEC_APP_Registrant being_in idle_state and
the MEC_APP_Subscriber subscribed_to the MP1_SUBSCRIPTION_A
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services",
   body containing
      ServiceInfo containing
         serName indicating value SERVICE_NAME
      ;
   ;
;
from the MEC_APP_Registrant
    }
    then {
            // ETSI GS MEC 011, clause 7.15.3.4
the IUT sends a HTTP_RESPONSE containing
   status_code set to "201 Created",
   Location set to LOCATION,
   body containing
      ServiceInfo containing
         serName set to SERVICE_NAME
      ;
   ;
;
to the MEC_APP_Registrant and
// ETSI GS MEC 011, clause 6.4.2
the IUT sends a notification_message containing
   body containing
      notificationType set to "SerAvailabilityNotification",
      services containing
         serName set to SERVICE_NAME
      _links containing
         subscription set to MP1_SUBSCRIPTION_A
      ;
      ;
   ;
;
to the MEC_APP_Subscriber
    }
}
```

Final Conditions

Table 6.4.2-4: TP_MEC_SRV_APPSAQ_002_BR

TP Id	"TP_MEC_SRV_APPSAQ_002_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.15.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES and PIC_NOTIFICATIONS
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the MEC_APP_Registrant being_in idle_state and
the MEC_APP_Subscriber subscribed_to the MP1_SUBSCRIPTION_A
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services",
   body containing
      ServiceInfo containing
         // Wrong parameter name should trigger an error response.
         Name indicating value SERVICE_NAME
      ;
   ;
;
from the MEC_APP_Registrant
    }
    then {
            // ETSI GS MEC 011, clause 7.15.3.4
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the MEC_APP_Registrant
    }
}
```

Final Conditions

Table 6.4.2-5: TP_MEC_SRV_APPSAQ_002_NF

TP Id	"TP_MEC_SRV_APPSAQ_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.15.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a apps_instance containing
instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
; and
the MEC_APP_Registrant being_in idle_state and
the MEC_APP_Subscriber subscribed_to the MP1_SUBSCRIPTION_A
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value
"/mec_service_mgmt/v1/applications/{NON_EXISTENT_APP_INSTANCE_ID}/services",
body containing
ServiceInfo containing
serName indicating value SERVICE_NAME
;
;
;
from the MEC_APP_Registrant
}
then {
// ETSI GS MEC 011, clause 7.15.3.4
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP_Registrant
}
}	

Final Conditions	

Table 6.4.2-6: TP_MEC_SRV_APPSAQ_003_OK

TP Id	"TP_MEC_SRV_APPSAQ_003_OK"
Test Objective	Check that the IUT responds with the information on a specific service for a given application instance when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.14.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a services_running containing
   service_id indicating value SERVICE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value
"/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services/{SERVICE_ID}"
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.14.3.1
the IUT sends a HTTP_RESPONSE containing
   status_code set to "200 OK"
   body containing
      ServiceInfo containing
         serInstanceId set to SERVICE_ID
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.2-7: TP_MEC_SRV_APPSAQ_003_NF

TP Id	"TP_MEC_SRV_APPSAQ_003_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.14.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT "not" having a services_running containing
service_id indicating value NON_EXISTENT_SERVICE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services/{NON_EXISTENT_SERVICE_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.14.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.2-8: TP_MEC_SRV_APPSAQ_004_OK

TP Id	"TP_MEC_SRV_APPSAQ_004_OK"
Test Objective	Check that the IUT updates a service information for a given application instance when commanded by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.14.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a services_running containing
   service_id indicating value SERVICE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services/{SERVICE_ID}",
   if_match indicating value PROPER_ETAG,
   body containing
      ServiceInfo containing
         version indicating value NEW_VERSION
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.14.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "200 OK"
   body containing
      ServiceInfo containing
         version set to NEW_VERSION
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.2-9: TP_MEC_SRV_APPSAQ_004_BR

TP Id	"TP_MEC_SRV_APPSAQ_004_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.14.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a services_running containing
   service_id indicating value SERVICE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services/{SERVICE_ID}",
   if_match indicating value PROPER_ETAG,
   body containing
      ServiceInfo containing
         // Wrong parameter name should trigger an error response.
         a_version indicating value NEW_VERSION
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.14.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.2-10: TP_MEC_SRV_APPSAQ_004_NF

TP Id	"TP_MEC_SRV_APPSAQ_004_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.14.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT "not" having a services_running containing
   service_id indicating value NON_EXISTENT_SERVICE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services/{NON_EXISTENT_SERVICE_ID}",
   if_match indicating value PROPER_ETAG,
   body containing
      ServiceInfo containing
         version indicating value NEW_VERSION
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.14.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "404 Not Found"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.2-11: TP_MEC_SRV_APPSAQ_004_PF

TP Id	"TP_MEC_SRV_APPSAQ_004_PF"
Test Objective	Check that the IUT responds with an error when a request sent by a MEC Application does not comply with a required condition
Reference	ETSI GS MEC 011 [4], clause 7.14.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a services_running containing
   service_id indicating value SERVICE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"/mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/services/{SERVICE_ID}",
   if_match indicating value INVALID_ETAG,
   body containing
      ServiceInfo containing
         version indicating value NEW_VERSION
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.14.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "412 Precondition Failed"
;
to the MEC_APP
    }
}
```

Final Conditions

6.4.3 Application Subscriptions (APPSUB)

Table 6.4.3-1: TP_MEC_SRV_APPSUB_001_OK

TP Id	"TP_MEC_SRV_APPSUB_001_OK"
Test Objective	Check that the IUT responds with a list of subscriptions for notifications on services availability when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.6.3.1
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mec_app_support/v1/applications/{APP_INSTANCE_ID}/subscriptions"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.6.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
SubscriptionLinkList set to any_value
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.3-2: TP_MEC_SRV_APPSUB_001_NF

TP Id	"TP_MEC_SRV_APPSUB_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.6.3.1
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a apps_instance containing
instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_app_support/v1/applications/{NON_EXISTENT_APP_INSTANCE_ID}/subscriptions"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.6.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.3-3: TP_MEC_SRV_APPSUB_002_OK

TP Id	"TP_MEC_SRV_APPSUB_002_OK"
Test Objective	Check that the IUT acknowledges the subscription by a MEC Application to notifications on service availability events
Reference	ETSI GS MEC 011 [4], clause 7.6.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "mec_app_support/v1/applications/{APP_INSTANCE_ID}/subscriptions"
   body containing
      AppTerminationNotificationSubscription containing
         subscriptionType indicating value "AppTerminationNotificationSubscription",
         callbackReference indicating value some_uri
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.6.3.4
the IUT sends a HTTP_RESPONSE containing
   status_code set to "201 Created",
   Location set to LOCATION,
   body containing
      AppTerminationNotificationSubscription containing
         subscriptionType set to "AppTerminationNotificationSubscription",
         callbackReference set to some_uri
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.3-4: TP_MEC_SRV_APPSUB_002_BR

TP Id	"TP_MEC_SRV_APPSUB_002_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.6.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "mec_app_support/v1/applications/{APP_INSTANCE_ID}/subscriptions"
body containing
AppTerminationNotificationSubscription containing
// Unknown value should trigger an error response.
subscriptionType indicating value INVALID_SUBSCRIPTION,
callbackReference indicating value some_uri
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.6.3.4
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.3-5: TP_MEC_SRV_APPSUB_003_OK

TP Id	"TP_MEC_SRV_APPSUB_003_OK"
Test Objective	Check that the IUT responds with the information on a specific subscription when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.1
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a subscriptions containing
subscription_id indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.5.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
AppTerminationNotificationSubscription containing
subscriptionType set to "AppTerminationNotificationSubscription"
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.3-6: TP_MEC_SRV_APPSUB_003_NF

TP Id	"TP_MEC_SRV_APPSUB_003_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.1
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT "not" having a subscriptions containing
subscription_id indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.5.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.3-7: TP_MEC_SRV_APPSUB_004_OK

TP Id	"TP_MEC_SRV_APPSUB_004_OK"
Test Objective	Check that the IUT acknowledges the unsubscribe from service availability event notifications when commanded by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.5
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a subscriptions containing
subscription_id indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.5.3.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.3-8: TP_MEC_SRV_APPSUB_004_NF

TP Id	"TP_MEC_SRV_APPSUB_004_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.5
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a apps_instance containing
instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
; and
the IUT having a subscriptions containing
subscription_id indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value
"mec_app_support/v1/applications/{NON_EXISTENT_APP_INSTANCE_ID}/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.5.3.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.4 DNS rules (DNS)

Table 6.4.4-1: TP_MEC_SRV_DNS_001_OK

TP Id	"TP_MEC_SRV_DNS_001_OK"
Test Objective	Check that the IUT responds with a list of active DNS rules when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.12.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mec_app_support/v1/applications/{APP_INSTANCE_ID}/dns_rules"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.12.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
DnsRuleList containing
DnsRule set to any_value
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.4-2: TP_MEC_SRV_DNS_002_OK

TP Id	"TP_MEC_SRV_DNS_002_OK"
Test Objective	Check that the IUT responds with the information on a specific DNS rule when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.13.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a dns_rules containing
rule_id indicating value DNS_RULE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/dns_rules/{DNS_RULE_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.13.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
DnsRule containing
dnsRuleId set to DNS_RULE_ID
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.4-3: TP_MEC_SRV_DNS_002_NF

TP Id	"TP_MEC_SRV_DNS_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.13.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT "not" having a dns_rules containing
rule_id indicating value NON_EXISTENT_DNS_RULE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/dns_rules/{NON_EXISTENT_DNS_RULE_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.13.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.4-4: TP_MEC_SRV_DNS_003_OK

TP Id	"TP_MEC_SRV_DNS_003_OK"
Test Objective	Check that the IUT updates a specific DNS rule when commanded by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.13.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a dns_rules containing
   rule_id indicating value DNS_RULE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/dns_rules/{DNS_RULE_ID}",
   if_match indicating value PROPER_ETAG
   body containing
      dnsRuleId indicating value DNS_RULE_NAME,
      ipAddress indicating value SOME_IP_ADDRESS
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.13.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "200 OK"
   body containing
      DnsRule containing
         dnsRuleId set to DNS_RULE_NAME,
         ipAddress set to SOME_IP_ADDRESS
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.4-5: TP_MEC_SRV_DNS_003_BR

TP Id	"TP_MEC_SRV_DNS_003_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.13.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a dns_rules containing
   rule_id indicating value DNS_RULE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/dns_rules/{DNS_RULE_ID}",
   if_match indicating value PROPER_ETAG
   body containing
      dnsRuleId indicating value DNS_RULE_NAME,
      // Unknown parameter value should trigger an error response.
      state indicating value UNKNOWN_VALUE
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.13.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.4-6: TP_MEC_SRV_DNS_003_NF

TP Id	"TP_MEC_SRV_DNS_003_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.13.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT "not" having a dns_rules containing
   rule_id indicating value NON_EXISTENT_DNS_RULE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/dns_rules/{NON_EXISTENT_DNS_RULE_ID}",
   if_match indicating value PROPER_ETAG
   body containing
      dnsRuleId indicating value DNS_RULE_NAME,
      ipAddress indicating value SOME_IP_ADDRESS
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.13.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "404 Not Found"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.4-7: TP_MEC_SRV_DNS_003_PF

TP Id	"TP_MEC_SRV_DNS_003_PF"
Test Objective	Check that the IUT responds with an error when a request sent by a MEC Application does not comply with a required condition
Reference	ETSI GS MEC 011 [4], clause 7.13.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a dns_rules containing
   rule_id indicating value DNS_RULE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/dns_rules/{DNS_RULE_ID}",
   if_match indicating value INVALID_ETAG
   body containing
      dnsRuleId indicating value DNS_RULE_NAME,
      ipAddress indicating value SOME_IP_ADDRESS
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.13.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "412 Precondition Failed"
;
to the MEC_APP
    }
}
```

Final Conditions

6.4.5 Fixed Access Information Service (FAIS)

Table 6.4.5-1: TP_MEC_SRV_FAIS_001_OK

TP Id	"TP_MEC_SRV_FAIS_001_OK"
Test Objective	Check that the IUT responds with the current status of the fixed access information when queried by a MEC Application
Reference	ETSI GS MEC 029 [11], clause 7.3.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/fai/v1/queries/fa_info"
;
from the MEC_APP
}
then {
// ETSI GS MEC 029, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
FaInfo set to any_value
;
;
to the MEC_APP
}
}	

Final Conditions	

6.4.6 Radio Node Location Lookup (RLOC)

Table 6.4.6-2: TP_MEC_SRV_RLOC_001_OK

TP Id	"TP_MEC_SRV_RLOC_001_OK"
Test Objective	Check that the IUT responds with the list of radio nodes currently associated with the MEC host and the location of each radio node when queried by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.7
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a ue_location containing
zoneId indicating value ZONE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/zones/{ZONE_ID}/accessPoints"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.7
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
accessPointList containing
zoneId set to ZONE_ID
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.6-3: TP_MEC_SRV_RLOC_001_NF

TP Id	"TP_MEC_SRV_RLOC_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.7
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a ue_location containing
zoneId indicating value NON_EXISTENT_ZONE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/zones/{NON_EXISTENT_ZONE_ID}/accessPoints"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.7
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.7 Radio Network Information Service (RNIS)

Table 6.4.7-1: TP_MEC_SRV_RNIS_001_OK

TP Id	"TP_MEC_SRV_RNIS_001_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about cell change if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.2
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "CellChangeSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteriaAssocHo containing
associateIdList containing
associateId set to ASSOCIATE_ID
;
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a cell_change_event containing
associateId set to ASSOCIATE_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "CellChangeNotification",
associateId set to ASSOCIATE_ID,
srcEcgi set to any_value,
trgEcgi set to any_value,
hoStatus set to any_value
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-2: TP_MEC_SRV_RNIS_002_OK

TP Id	"TP_MEC_SRV_RNIS_002_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about RAB establishment if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.3
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "RabEstSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteriaQci containing
qci set to QCI
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a rab_establish_event containing
erabQosParameters containing
qci set to QCI;
;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "RabEstNotification",
ecgi set to any_value,
erabId set to any_value,
erabQosParameters containing
qci set to QCI,
attribute qosInformation
;
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-3: TP_MEC_SRV_RNIS_003_OK

TP Id	"TP_MEC_SRV_RNIS_003_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about RAB modification if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.4
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

```
with {
            the IUT having a RNIS_subscription containing
   subscriptionType indicating value "RabModSubscription",
   callbackReference indicating value CALLBACK_URI,
   filterCriteriaQci containing
      erabId set to E_RAB_ID
   ;
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT generates a rab_modification_event containing
   erabId set to E_RAB_ID;
    }
    then {
            the IUT sends a vPOST containing
   uri indicating value CALLBACK_URI,
   body containing
      notificationType indicating value "RabModNotification",
      ecgi set to any_value,
      erabId set to E_RAB_ID
   ;
; to the MEC_SUB
    }
}
```

Final Conditions

Table 6.4.7-4: TP_MEC_SRV_RNIS_004_OK

TP Id	"TP_MEC_SRV_RNIS_004_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about RAB release if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.5
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "RabRelSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteriaQci containing
erabId set to E_RAB_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a rab_release_event containing
erabId set to E_RAB_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "RabRelNotification",
ecgi set to any_value,
erabReleaseInfo containing
erabId set to E_RAB_ID
;
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-5: TP_MEC_SRV_RNIS_005_OK

TP Id	"TP_MEC_SRV_RNIS_005_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about UE measurement report if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.6
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "MeasRepUeSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteriaAssocTri containing
ecgi set to ECGI
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a measurement_repost_event containing
ecgi set to ECGI;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "MeasRepUeNotification",
ecgi set to ECGI,
rsrp set to any_value,
rsrq set to any_value,
trigger set to any_value
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-6: TP_MEC_SRV_RNIS_006_OK

TP Id	"TP_MEC_SRV_RNIS_006_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about UE timing advance if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.7
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "MeasTaSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteriaAssoc containing
ecgi set to ECGI
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a timing_advance_event containing
ecgi set to ECGI;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "MeasTaNotification",
ecgi set to ECGI,
timingAdvance set to any_value
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-7: TP_MEC_SRV_RNIS_007_OK

TP Id	"TP_MEC_SRV_RNIS_007_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about UE carrier aggregation reconfiguration if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.8
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "CaReconfSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteriaAssoc containing
associateId set to ASSOCIATE_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a carrier_aggregation_reconfiguration_event containing
associateId set to ASSOCIATE_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "CaReConfNotification",
ecgi set to any_value,
associateId set to ASSOCIATE_ID
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-8: TP_MEC_SRV_RNIS_008_OK

TP Id	"TP_MEC_SRV_RNIS_008_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about S1-U bearer if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.10
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "S1BearerSubscription",
callbackReference indicating value CALLBACK_URI,
s1BearerSubscriptionCriteria containing
associateId set to ASSOCIATE_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a S1_bearer_event containing
associateId set to ASSOCIATE_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "S1BearerNotification",
s1Event set to any_value,
s1UeInfo containing
associateId set to ASSOCIATE_ID,
ecgi set to any_value,
s1BearerInfo set to any_value
;
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-9: TP_MEC_SRV_RNIS_009_OK

TP Id	"TP_MEC_SRV_RNIS_009_OK"
Test Objective	Check that the RNIS service sends an RNIS notification about 5G NR UE measurement report if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.11
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "NrMeasRepUeSubscription",
callbackReference indicating value CALLBACK_URI,
filterCriteriaNrMrs indicating value FILTER_CRITERIA //the attributes of filterCriteriaNrMrs are FFS - ref. clause 6.3.11
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a nr_measurement_repost_event containing
associateId set to ASSOCIATE_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "NrMeasRepUeNotification",
trigger set to any_value,
associateId set to ASSOCIATE_ID
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-10: TP_MEC_SRV_RNIS_010_OK

TP Id	"TP_MEC_SRV_RNIS_010_OK"
Test Objective	Check that the RNIS service sends an RNIS notification on subscription expiration if the RNIS service has an associated subscription and the event is generated
Reference	ETSI GS MEC 012 [5], clause 6.4.9
PICS Selection	PIC_RNIS_NOTIFICATIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value any_value, //VALID for all subscription types
callbackReference indicating value CALLBACK_URI,
_links containing
self set to SUBSCRIPTION_HREF_VALUE
;,
subscriptionId set to SUBSCRIPTION_ID,
expiryDeadline set to EXPIRY_DEADLINE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT generates a expiry_notification_event containing
subscriptionId set to SUBSCRIPTION_ID;
}
then {
the IUT sends a vPOST containing
uri indicating value CALLBACK_URI,
body containing
notificationType indicating value "ExpiryNotification", //this field is not in the spec. - ref. clause 6.4.9
_link containing
self set to SUBSCRIPTION_HREF_VALUE
;,
expiryDeadline set to EXPIRY_DEADLINE
;
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-11: TP_MEC_SRV_RNIS_011_OK

TP Id	"TP_MEC_SRV_RNIS_011_OK"
Test Objective	Check that the RNIS service sends the list of links to the relevant RNIS subscriptions when requested. Acceptable SUBSCRIPTION_TYPE are the following: - cell_change Cell Change - rab_est RAB Establishment - rab_mod RAB Modification - rab_rel RAB Release - meas_rep_ue UE Measurement Report - nr_meas_rep_ue 5G NR UE Measurement Report - timing_advance_ue UE Timing Advance - ca_reconf Carrier Aggregation Reconfig - s1_bearer S1 Bearer Notification
Reference	ETSI GS MEC 012 [5], clause 7.6.3.1
PICS Selection	PIC_RNIS_ALL_SUBSCRIPTIONS
Initial Conditions	

```
with {
            the IUT having a RNIS_subscription containing
   subscriptionHref indicating value SUBSCRIPTION_HREF_VALUE,
   subscriptionType indicating value SUBSCRIPTION_TYPE
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/rni/v1/subscriptions",
   query_parameters containing
      subscription_type indicating value SUBSCRIPTION_TYPE
   ;
; from the MEC_SUB
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      subscriptionLinkList containing
         subscription containing
            href indicating value SUBSCRIPTION_HREF_VALUE,
            subscriptionType indicating value SUBSCRIPTION_TYPE
         ;
      ;
   ;
; to the MEC_SUB
    }
}
```

Final Conditions

Table 6.4.7-12: TP_MEC_SRV_RNIS_011_BR

TP Id	"TP_MEC_SRV_RNIS_011_BR"
Test Objective	Check that the RNIS service responds with an error when it receives a request to get all RNIS subscriptions with a wrong subscription type
Reference	ETSI GS MEC 012 [5], clause 7.6.3.1
PICS Selection	PIC_RNIS_ALL_SUBSCRIPTIONS
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionHref indicating value SUBSCRIPTION_HREF_VALUE,
subscriptionType indicating value SUBSCRIPTION_TYPE
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rni/v1/subscriptions",
query_parameters containing
subscriptionType indicating value "wrongSubscriptionType" //wrong subscription type
;
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-13: TP_MEC_SRV_RNIS_012_OK

TP Id	"TP_MEC_SRV_RNIS_012_OK"
Test Objective	Check that the RNIS service creates a new RNIS subscription. Acceptable SUBSCRIPTION_TYPE are the following: - CellChangeSubscription - RabEstSubscription - RabModSubscription - RabRelSubscription - MeasRepUeSubscription - MeasTaSubscription - CaReconfSubscription - S1BearerSubscription - NrMeasRepUeSubscription
Reference	ETSI GS MEC 012 [5], clause 7.6.3.4
PICS Selection	PIC_RNIS_ALL_SUBSCRIPTIONS
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/rni/v1/subscriptions",
body containing
CellChangeSubscription containing
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackReference indicating value CALLBACK_URI
;
;
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "201 Created",
body containing
CellChangeSubscription containing
subscriptionType indicating value SUBSCRIPTION_TYPE,
callbackReference indicating value CALLBACK_URI,
attribute _links
;
;
; to the MEC_SUB
}
}	

Final Conditions

Table 6.4.7-14: TP_MEC_SRV_RNIS_012_BR

TP Id	"TP_MEC_SRV_RNIS_012_BR"
Test Objective	Check that the RNIS service responds with an error when it receives a request to create a new RNIS subscription with a wrong format
Reference	ETSI GS MEC 012 [5], clause 7.6.3.4
PICS Selection	PIC_RNIS_ALL_SUBSCRIPTIONS
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "/rni/v1/subscriptions",
body containing
CellChangeSubscription containing
subscritionType indicating value "CelCangeSubscription", //wrong subscription type
callbackReference indicating value any_value,
attribute filterCriteriaAssocHo,
expiryDeadline set to any_value
;
;
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-15: TP_MEC_SRV_RNIS_013_OK

TP Id	"TP_MEC_SRV_RNIS_013_OK"
Test Objective	Check that the RNIS service sends a RNIS subscription when requested. Acceptable SUBSCRIPTION_TYPE are the following: - CellChangeSubscription - RabEstSubscription - RabModSubscription - RabRelSubscription - MeasRepUeSubscription - MeasTaSubscription - CaReconfSubscription - S1BearerSubscription - NrMeasRepUeSubscription
Reference	ETSI GS MEC 012 [5], clause 7.8.3.1
PICS Selection	PIC_RNIS_SPECIFIC_SUBSCRIPTION
Initial Conditions	

```
with {
            the IUT having a RNIS_subscription containing
   subscriptionType indicating value SUBSCRIPTION_TYPE,
   callbackReference indicating value CALLBACK_URI,
   _links containing
      self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
   ;
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
; from the MEC_SUB
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      subscription containing
         subscriptionType indicating value SUBSCRIPTION_TYPE,
         callbackReference indicating value CALLBACK_URI,
         _links containing
            self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
         ;
      ;
   ;
; to the MEC_SUB
    }
}
```

Final Conditions

Table 6.4.7-16: TP_MEC_SRV_RNIS_013_NF

TP Id	"TP_MEC_SRV_RNIS_013_NF"
Test Objective	Check that the RNIS service responds with error when a not existing RNIS subscription is requested
Reference	ETSI GS MEC 012 [5], clause 7.8.3.1
PICS Selection	PIC_RNIS_SPECIFIC_SUBSCRIPTION
Initial Conditions	

```
with {
            the IUT "not" having a RNIS_subscription containing
   _links containing
      self set to "/rni/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
   ;
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/rni/v1/subscription/{NON_EXISTENT_SUBSCRIPTION_ID}"
; from the MEC_SUB
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "404 Not Found"
; to the MEC_SUB
    }
}
```

Final Conditions

Table 6.4.7-17: TP_MEC_SRV_RNIS_014_OK

TP Id	"TP_MEC_SRV_RNIS_014_OK"
Test Objective	Check that the RNIS service modifies a RNIS subscription when requested
Reference	ETSI GS MEC 012 [5], clause 7.8.3.2
PICS Selection	PIC_RNIS_SPECIFIC_SUBSCRIPTION

Initial Conditions

```
with {
            the IUT having a RNIS_subscription containing
   subscriptionType indicating value SUBSCRIPTION_TYPE,
   callbackReference indicating value CALLBACK_URI,
   _links containing
      self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
   ;
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value "/rni/v1/subscriptions/{SUBSCRIPTION_ID}",
   body containing
      subscription containing
         subscriptionType indicating value SUBSCRIPTION_TYPE,
         callbackReference indicating value NEW_CALLBACK_URI,
         _links containing
            self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
         ;
      ;
   ;
; from the MEC_SUB
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      subscription containing
         subscriptionType indicating value SUBSCRIPTION_TYPE,
         callbackReference indicating value NEW_CALLBACK_URI,
         _links containing
            self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
         ;
      ;
   ;
; to the MEC_SUB
    }
}
```

Final Conditions

Table 6.4.7-18: TP_MEC_SRV_RNIS_014_BR

TP Id	"TP_MEC_SRV_RNIS_014_BR"
Test Objective	Check that the RNIS service sends an error when it receives a malformed modify request for a RNIS subscription
Reference	ETSI GS MEC 012 [5], clause 7.8.3.2
PICS Selection	PIC_RNIS_SPECIFIC_SUBSCRIPTION
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
subscriptionType indicating value "CellChangeSubscription",
callbackReference indicating value CALLBACK_URI,
_links containing
self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "/rni/v1/subscriptions/{SUBSCRIPTION_ID}",
body containing
subscription containing
callbackRefer indicating value NEW_CALLBACK_URI, //wrong field, it should be callbackReference
_links containing
self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
;
from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-19: TP_MEC_SRV_RNIS_014_NF

TP Id	"TP_MEC_SRV_RNIS_014_NF"
Test Objective	Check that the RNIS service responds with error when a modification for a not existing RNIS subscription is requested
Reference	ETSI GS MEC 012 [5], clause 7.8.3.2
PICS Selection	PIC_RNIS_SPECIFIC_SUBSCRIPTION
Initial Conditions	

with {
the IUT "not" having a RNIS_subscription containing
_links containing
self set to "/rni/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "/rni/v1/subscription/{NON_EXISTENT_SUBSCRIPTION_ID}"
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-20: TP_MEC_SRV_RNIS_015_OK

TP Id	"TP_MEC_SRV_RNIS_015_OK"
Test Objective	Check that the RNIS service deletes a RNIS subscription when requested
Reference	ETSI GS MEC 012 [5], clause 7.8.3.5
PICS Selection	PIC_RNIS_SPECIFIC_SUBSCRIPTION
Initial Conditions	

with {
the IUT having a RNIS_subscription containing
callbackReference indicating value CALLBACK_URI,
_links containing
self set to "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/rni/v1/subscriptions/{SUBSCRIPTION_ID}"
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "204 No Content"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-21: TP_MEC_SRV_RNIS_015_NF

TP Id	"TP_MEC_SRV_RNIS_015_NF"
Test Objective	Check that the RNIS service responds with error when the deletion of a not existing RNIS subscription is requested
Reference	ETSI GS MEC 012 [5], clause 7.8.3.5
PICS Selection	PIC_RNIS_SPECIFIC_SUBSCRIPTION
Initial Conditions	

with {
the IUT "not" having a RNIS_subscription containing
_links containing
self set to "/rni/v1/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "/rni/v1/subscription/{NON_EXISTENT_SUBSCRIPTION_ID}"
; from the MEC_SUB
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the MEC_SUB
}
}	

Final Conditions	

Table 6.4.7-22: TP_MEC_SRV_RNIS_016_OK

TP Id	"TP_MEC_SRV_RNIS_016_OK"
Test Objective	Check that the RNIS service returns the RAB information when requested
Reference	ETSI GS MEC 012 [5], clause 7.3.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

```
with {
            the IUT having a RabInfo containing
   cellUserInfo containing
       ecgi indicating value CELL_ID
   ;
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/rni/v1/queries/rab_info",
   query_parameters containing
      cell_id indicating value CELL_ID
   ;
; from the RNIS_CLIENT
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "200 OK",
   body containing
      RabInfo containing
        appInstanceId set to any_value,
        requestId set to any_value,
        cellUserInfo containing
          ecgi indicating value CELL_ID,
          attribute ueInfo
        ;
      ;
   ;
; to the RNIS_CLIENT
    }
}
```

Final Conditions

Table 6.4.7-23: TP_MEC_SRV_RNIS_016_BR

TP Id	"TP_MEC_SRV_RNIS_016_BR"
Test Objective	Check that the RNIS service returns an error when the RAB information is requested with a malformatted message
Reference	ETSI GS MEC 012 [5], clause 7.3.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT having a RabInfo containing
cellUserInfo containing
ecgi indicating value CELL_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rni/v1/queries/rab_info",
query_parameters containing
cId indicating value C_ID //wrong parameter
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the RNIS_CLIENT
}
}	

Final Conditions	

Table 6.4.7-24: TP_MEC_SRV_RNIS_016_NF

TP Id	"TP_MEC_SRV_RNIS_016_NF"
Test Objective	Check that the RNIS service returns an error when the RAB information for a not existing element is requested
Reference	ETSI GS MEC 012 [5], clause 7.3.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT "not" having a RabInfo containing
cellUserInfo containing
ecgi indicating value CELL_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rnis/v1/query/rab_info",
query_parameters containing
cell_id indicating value CELL_ID
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the RNIS_CLIENT
}
}	

Final Conditions	

Table 6.4.7-25: TP_MEC_SRV_RNIS_017_OK

TP Id	"TP_MEC_SRV_RNIS_017_OK"
Test Objective	Check that the RNIS service returns the PLMN information when requested
Reference	ETSI GS MEC 012 [5], clause 7.4.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT having a PlmnInfo containing
appInstanceId indicating value APP_INSTANCE_INFO
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rni/v1/queries/plmn_info",
query_parameters containing
app_ins_id indicating value APP_INS_ID
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
PlmnInfo containing
appInstanceId indicating value APP_INS_ID,
attribute plmn
;
;
; to the RNIS_CLIENT
}
}	

Final Conditions	

Table 6.4.7-26: TP_MEC_SRV_RNIS_017_BR

TP Id	"TP_MEC_SRV_RNIS_017_BR"
Test Objective	Check that the RNIS service returns an error when the PLMN information is requested with a malformatted message
Reference	ETSI GS MEC 012 [5], clause 7.4.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT having a PlmnInfo containing
appInstanceId indicating value APP_INSTANCE_INFO
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rni/v1/queries/plmn_info",
query_parameters containing
cId indicating value C_ID //wrong parameter
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the RNIS_CLIENT
}
}	

Final Conditions	

Table 6.4.7-27: TP_MEC_SRV_RNIS_017_NF

TP Id	"TP_MEC_SRV_RNIS_017_NF"
Test Objective	Check that the RNIS service returns an error when the PLMN information for a not existing element is requested
Reference	ETSI GS MEC 012 [5], clause 7.4.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT "not" having a PlmnInfo containing
appInstanceId indicating value NOT_EXISTENT_APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rnis/v1/query/plmn_info",
query_parameters containing
appInstanceId indicating value NOT_EXISTENT_APP_INSTANCE_ID
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the RNIS_CLIENT
}
}	

Final Conditions	

Table 6.4.7-28: TP_MEC_SRV_RNIS_018_OK

TP Id	"TP_MEC_SRV_RNIS_018_OK"
Test Objective	Check that the RNIS service returns the S1 bearer information
Reference	ETSI GS MEC 012 [5], clause 7.5.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT having a S1BearerInfo containing
s1UeInfo containing
ecgi indicating value CELL_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rni/v1/queries/s1_bearer_info",
query_parameters containing
cell_id indicating value CELL_ID
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
S1BearerInfo containing
s1UeInfo containing
attribute tempUeId,
attribute associateId,
ecgi indicating value CELL_ID,
attribute s1BearerInfoDetailed
;
;
;
; to the RNIS_CLIENT
}
}	

Final Conditions

Table 6.4.7-29: TP_MEC_SRV_RNIS_018_BR

TP Id	"TP_MEC_SRV_RNIS_018_BR"
Test Objective	Check that the RNIS service returns an error when the S1 bearer information is requested with a malformatted message
Reference	ETSI GS MEC 012 [5], clause 7.5.3.1
PICS Selection	PIC_RNIS_QUERY

Initial Conditions

```
with {
            the IUT having a S1BearerInfo containing
   s1UeInfo containing
     ecgi indicating value CELL_ID
   ;
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/rni/v1/queries/s1_bearer_info",
   query_parameters containing
      cId indicating value C_ID                    //wrong parameter
   ;
; from the RNIS_CLIENT
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "400 Bad Request"
; to the RNIS_CLIENT
    }
}
```

Final Conditions

Table 6.4.7-30: TP_MEC_SRV_RNIS_018_NF

TP Id	"TP_MEC_SRV_RNIS_018_NF"
Test Objective	Check that the RNIS service returns an error when the S1 bearer information for a not existing element is requested
Reference	ETSI GS MEC 012 [5], clause 7.5.3.1
PICS Selection	PIC_RNIS_QUERY

Initial Conditions

```
with {
            the IUT "not" having a S1BearerInfo containing
   s1UeInfo containing
     ecgi indicating value NOT_EXISTENT_CELL_ID
   ;
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "/rnis/v1/query/s1_bearer_info",
   query_parameters containing
      cell_id indicating value NOT_EXISTENT_CELL_ID
   ;
; from the RNIS_CLIENT
    }
    then {
            the IUT sends a HTTP_RESPONSE containing
   status set to "404 Not Found"
; to the RNIS_CLIENT
    }
}
```

Final Conditions

Table 6.4.7-31: TP_MEC_SRV_RNIS_019_OK

TP Id	"TP_MEC_SRV_RNIS_019_OK"
Test Objective	Check that the RNIS service returns the L2 measurements information
Reference	ETSI GS MEC 012 [5], clause 7.5a.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT having a L2_Meas_information containing
cellInfo containing
ecgi indicating value CELL_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rni/v1/queries/layer2_meas",
query_parameters containing
cell_id indicating value CELL_ID
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "200 OK",
body containing
L2_Meas_information containing
cellInfo containing
ecgi indicating value CELL_ID
;
;
;
; to the RNIS_CLIENT
}
}	

Final Conditions	

Table 6.4.7-32: TP_MEC_SRV_RNIS_019_BR

TP Id	"TP_MEC_SRV_RNIS_019_BR"
Test Objective	Check that the RNIS service returns an error when the L2 measurements information is requested with a malformatted message
Reference	ETSI GS MEC 012 [5], clause 7.5a.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT having a L2_Meas_information containing
cellInfo containing
ecgi indicating value CELL_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rni/v1/queries/layer2_meas",
query_parameters containing
cId indicating value C_ID //wrong parameter
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "400 Bad Request"
; to the RNIS_CLIENT
}
}	

Final Conditions	

Table 6.4.7-33: TP_MEC_SRV_RNIS_019_NF

TP Id	"TP_MEC_SRV_RNIS_019_NF"
Test Objective	Check that the RNIS service returns an error when the L2 measurements information for a not existing element is requested
Reference	ETSI GS MEC 012 [5], clause 7.5a.3.1
PICS Selection	PIC_RNIS_QUERY
Initial Conditions	

with {
the IUT "not" having a L2_Meas_information containing
cellInfo containing
ecgi indicating value NOT_EXISTENT_CELL_ID
;
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "/rnis/v1/query/layer2_meas",
query_parameters containing
cell_id indicating value NOT_EXISTENT_CELL_ID
;
; from the RNIS_CLIENT
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "404 Not Found"
; to the RNIS_CLIENT
}
}	

Final Conditions	

6.4.8 Service Availability Query (SAQ)

Table 6.4.8-1: TP_MEC_SRV_SAQ_001_OK

TP Id	"TP_MEC_SRV_SAQ_001_OK"
Test Objective	Check that the IUT responds with a list of available MEC services when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.4.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
Uri indicating value"/mec_service_mgmt/v1/services"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.4.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
ServiceInfoList containing
ServiceInfo set to any_value
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.8-2: TP_MEC_SRV_SAQ_001_BR

TP Id	"TP_MEC_SRV_SAQ_001_BR"
Test Objective	Check that the IUT responds with an error when incorrect parameters were sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.4.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
Uri indicating value"/mec_service_mgmt/v1/services"
query_parameters containing
// Wrong parameter name should trigger an error response.
instance_id indicating value any_value
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.4.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.8-3: TP_MEC_SRV_SAQ_002_OK

TP Id	"TP_MEC_SRV_SAQ_002_OK"
Test Objective	Check that the IUT responds with the information on a specific service when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.3.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a services_running containing
service_id indicating value SERVICE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
Uri indicating value"/mec_service_mgmt/v1/services/{SERVICE_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
ServiceInfo containing
serInstanceId set to SERVICE_ID
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.8-4: TP_MEC_SRV_SAQ_002_NF

TP Id	"TP_MEC_SRV_SAQ_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.3.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a services_running containing
service_id indicating value NON_EXISTENT_SERVICE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
Uri indicating value"/mec_service_mgmt/v1/services/{NON_EXISTENT_SERVICE_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.9 Service Subscriptions (SRVSUB)

Table 6.4.9-1: TP_MEC_SRV_SRVSUB_001_OK

TP Id	"TP_MEC_SRV_SRVSUB_001_OK"
Test Objective	Check that the IUT responds with a list of subscriptions for notifications on services availability when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause7.2.3.2
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/subscriptions"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.2.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
SubscriptionLinkList set to any_value
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.9-2: TP_MEC_SRV_SRVSUB_001_NF

TP Id	"TP_MEC_SRV_SRVSUB_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.6.3.1
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a apps_instance containing
instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_service_mgmt/v1/applications/{NON_EXISTENT_APP_INSTANCE_ID}/subscriptions"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.6.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.9-3: TP_MEC_SRV_SRVSUB_002_OK

TP Id	"TP_MEC_SRV_SRVSUB_002_OK"
Test Objective	Check that the IUT acknowledges the subscription by a MEC Application to notifications on service availability events
Reference	ETSI GS MEC 011 [4], clause 7.6.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/subscriptions"
   body containing
      SerAvailabilityNotificationSubscription containing
         subscriptionType indicating value "SerAvailabilityNotificationSubscription",
         callbackReference indicating value some_uri
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.6.3.4
the IUT sends a HTTP_RESPONSE containing
   status_code set to "201 Created",
   Location set to LOCATION,
   body containing
      SerAvailabilityNotificationSubscription containing
         subscriptionType set to "SerAvailabilityNotificationSubscription",
         callbackReference set to some_uri
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.9-4: TP_MEC_SRV_SRVSUB_002_BR

TP Id	"TP_MEC_SRV_SRVSUB_002_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.6.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/subscriptions"
   body containing
      SerAvailabilityNotificationSubscription containing
         // Unknown value should trigger an error response.
         subscriptionType indicating value INVALID_SUBSCRIPTION,
         callbackReference indicating value some_uri
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.6.3.4
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.9-5: TP_MEC_SRV_SRVSUB_003_OK

TP Id	"TP_MEC_SRV_SRVSUB_003_OK"
Test Objective	Check that the IUT responds with the information on a specific subscription when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.1
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a subscriptions containing
   subscription_id indicating value SUBSCRIPTION_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value
"mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.5.3.1
the IUT sends a HTTP_RESPONSE containing
   status_code set to "200 OK"
   body containing
      SerAvailabilityNotificationSubscription containing
         subscriptionType set to "SerAvailabilityNotificationSubscription"
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.9-6: TP_MEC_SRV_SRVSUB_003_NF

TP Id	"TP_MEC_SRV_SRVSUB_003_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.1
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT "not" having a subscriptions containing
subscription_id indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/subscriptions/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.5.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.9-7: TP_MEC_SRV_SRVSUB_004_OK

TP Id	"TP_MEC_SRV_SRVSUB_004_OK"
Test Objective	Check that the IUT acknowledges the unsubscribe from service availability event notifications when commanded by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.5
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a subscriptions containing
subscription_id indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value
"mec_service_mgmt/v1/applications/{APP_INSTANCE_ID}/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.5.3.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.9-8: TP_MEC_SRV_SRVSUB_004_NF

TP Id	"TP_MEC_SRV_SRVSUB_004_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.5.3.5
PICS Selection	PIC_MEC_PLAT and PIC_NOTIFICATIONS
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a apps_instance containing
instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
; and
the IUT having a subscriptions containing
subscription_id indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value
"mec_service_mgmt/v1/applications/{NON_EXISTENT_APP_INSTANCE_ID}/subscriptions/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.5.3.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.10 Timing capabilities (TIME)

Table 6.4.10-1: TP_MEC_SRV_TIME_001_OK

TP Id	"TP_MEC_SRV_TIME_001_OK"
Test Objective	Check that the IUT responds with timing capabilities when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.7.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mec_app_support/v1/timing/timing_caps"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.7.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
TimingCaps set to any_value
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.10-2: TP_MEC_SRV_TIME_002_OK

TP Id	"TP_MEC_SRV_TIME_002_OK"
Test Objective	Check that the IUT responds with current time when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.8.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mec_app_support/v1/timing/current_time"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.8.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
CurrentTime set to any_value
;
;
to the MEC_APP
}
}	

Final Conditions	

6.4.11 Traffic rules (TRAF)

Table 6.4.11-1: TP_MEC_SRV_TRAF_001_OK

TP Id	"TP_MEC_SRV_TRAF_001_OK"
Test Objective	Check that the IUT responds with a list of available traffic rules when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.10.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mec_app_support/v1/applications/{APP_INSTANCE_ID}/traffic_rules"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.10.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
TrafficRuleList containing
TrafficRule set to any_value
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.11-2: TP_MEC_SRV_TRAF_001_NF

TP Id	"TP_MEC_SRV_TRAF_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.10.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a apps_instance containing
instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_app_support/v1/applications/{NON_EXISTENT_APP_INSTANCE_ID}/traffic_rules"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.10.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.11-3: TP_MEC_SRV_TRAF_002_OK

TP Id	"TP_MEC_SRV_TRAF_002_OK"
Test Objective	Check that the IUT responds with the information on a specific traffic rule when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.11.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a traffic_rules containing
rule_id indicating value TRAFFIC_RULE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/traffic_rules/{TRAFFIC_RULE_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.11.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
TrafficRule containing
trafficRuleId set to TRAFFIC_RULE_ID
;
;
;
to the MEC_APP
}
}	

Final Conditions

Table 6.4.11-4: TP_MEC_SRV_TRAF_003_OK

TP Id	"TP_MEC_SRV_TRAF_003_OK"
Test Objective	Check that the IUT updates a specific traffic rule when commanded by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.11.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a traffic_rules containing
rule_id indicating value TRAFFIC_RULE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/traffic_rules/{TRAFFIC_RULE_ID}",
if_match indicating value PROPER_ETAG
body containing
TrafficRule containing
action indicating value "DROP"
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.11.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
TrafficRule containing
trafficRuleId set to TRAFFIC_RULE_ID,
action set to "DROP"
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.11-5: TP_MEC_SRV_TRAF_003_BR

TP Id	"TP_MEC_SRV_TRAF_003_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.11.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a traffic_rules containing
   rule_id indicating value TRAFFIC_RULE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/traffic_rules/{TRAFFIC_RULE_ID}",
   if_match indicating value PROPER_ETAG
   body containing
      TrafficRule containing
         // Invalid parameter value should trigger an error response.
         action indicating value UNKNOWN_VALUE
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.11.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.11-6: TP_MEC_SRV_TRAF_003_NF

TP Id	"TP_MEC_SRV_TRAF_003_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.11.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT "not" having a traffic_rules containing
   rule_id indicating value NON_EXISTENT_TRAFFIC_RULE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/traffic_rules/{NON_EXISTENT_TRAFFIC_RULE_ID}",
   if_match indicating value PROPER_ETAG
   body containing
      TrafficRule containing
         action indicating value "DROP"
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.11.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "404 Not Found"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.11-7: TP_MEC_SRV_TRAF_003_PF

TP Id	"TP_MEC_SRV_TRAF_003_PF"
Test Objective	Check that the IUT responds with an error when a request sent by a MEC Application does not comply with a required condition
Reference	ETSI GS MEC 011 [4], clause 7.11.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a apps_instance containing
   instance_id indicating value APP_INSTANCE_ID
; and
the IUT having a traffic_rules containing
   rule_id indicating value TRAFFIC_RULE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value
"mec_app_support/v1/applications/{APP_INSTANCE_ID}/traffic_rules/{TRAFFIC_RULE_ID}",
   if_match indicating value INVALID_ETAG
   body containing
      TrafficRule containing
         action indicating value "DROP"
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 011, clause 7.11.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "412 Precondition Failed"
;
to the MEC_APP
    }
}
```

Final Conditions

6.4.12 Transport (TRANS)

Table 6.4.12-1: TP_MEC_SRV_TRANS_001_OK

TP Id	"TP_MEC_SRV_TRANS_001_OK"
Test Objective	Check that the IUT responds with a list of available transports when queried by a MEC Application
Reference	ETSI GS MEC 011 [4], clause 7.9.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "mec_service_mgmt/v1/transports"
;
from the MEC_APP
}
then {
// ETSI GS MEC 011, clause 7.9.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
TransportInfoList containing
TransportInfo set to any_value
;
;
;
to the MEC_APP
}
}	

Final Conditions	

6.4.13 UE Area Subscribe (UEAREASUB)

Table 6.4.13-1: TP_MEC_SRV_UEAREASUB_001_OK

TP Id	"TP_MEC_SRV_UEAREASUB_001_OK"
Test Objective	Check that the IUT acknowledges the UE area change subscription request when commanded by a MEC Application and notifies it when the UE enters the specified circle
Reference	ETSI GS MEC 013 [6], clause 7.3.11
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "location/v2/subscriptions/area/circle",
body containing
circleNotificationSubscription containing
clientCorrelator indicating value CLIENT_ID,
callbackReference indicating value CALLBACK_URL,
address indicating value IP_ADDRESS
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.11.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "201 Created"
body containing
circleNotificationSubscription containing
clientCorrelator indicating value CLIENT_ID,
callbackReference indicating value CALLBACK_URL,
address indicating value IP_ADDRESS
;
;
;
to the MEC_APP and
// ETSI GS MEC 013, clause 7.3.11.3
the IUT sends a vPOST containing
uri indicating value CALLBACK_URL,
body containing
subscriptionNotification containing
terminalLocation containing
address set to IP_ADDRESS
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.13-2: TP_MEC_SRV_UEAREASUB_001_BR

TP Id	"TP_MEC_SRV_UEAREASUB_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.11
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "location/v2/subscriptions/area/circle",
body containing
circleNotificationSubscription containing
clientCorrelator indicating value CLIENT_ID,
// Wrong name should trigger an error response.
callback indicating value CALLBACK_URL,
address indicating value IP_ADDRESS
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.11.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bar Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.13-3: TP_MEC_SRV_UEAREASUB_002_OK

TP Id	"TP_MEC_SRV_UEAREASUB_002_OK"
Test Objective	Check that the IUT acknowledges the cancellation of UE area change notifications when commanded by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a subscriptions containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/area/circle/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.6
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.13-4: TP_MEC_SRV_UEAREASUB_002_NF

TP Id	"TP_MEC_SRV_UEAREASUB_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a subscriptions containing
subscriptionId indicating value NON_EXISTING_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/area/circle/{NON_EXISTING_SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.6
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.14 UE Distance Lookup (UEDISTLOOK)

Table 6.4.14-1: TP_MEC_SRV_UEDISTLOOK_001_OK

TP Id	"TP_MEC_SRV_UEDISTLOOK_001_OK"
Test Objective	Check that the IUT responds with the distance to a UE when queried by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.9
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/queries",
query_parameters containing
address indicating value any_value,
latitude indicating value any_value,
longitude indicating value any_value
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.9.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
terminalDistance set to any_value
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.14-2: TP_MEC_SRV_UEDISTLOOK_001_BR

TP Id	"TP_MEC_SRV_UEDISTLOOK_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.9
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/queries",
query_parameters containing
address indicating value any_value,
// Wrong name should trigger an error response.
lat indicating value any_value,
longitude indicating value any_value
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.9.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.15 UE Distance Subscribe (UEDISTSUB)

Table 6.4.15-1: TP_MEC_SRV_UEDISTSUB_001_OK

TP Id	"TP_MEC_SRV_UEDISTSUB_001_OK"
Test Objective	Check that the IUT acknowledges the UE distance subscription request when commanded by a MEC Application and notifies it when (all) the requested UE(s) is (are) within the specified distance
Reference	ETSI GS MEC 013 [6], clause 7.3.10
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "location/v2/subscriptions/distance",
   body containing
      distanceNotificationSubscription containing
         clientCorrelator indicating value CLIENT_ID,
         callbackReference indicating value CALLBACK_URL,
         monitoredAddress indicating value MONITORED_IP_ADDRESS,
         referenceAddress indicating value IP_ADDRESS
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 013, clause 7.3.10.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "201 Created"
   body containing
      distanceNotificationSubscription containing
         clientCorrelator indicating value CLIENT_ID,
         callbackReference indicating value CALLBACK_URL,
         monitoredAddress indicating value MONITORED_IP_ADDRESS,
         referenceAddress indicating value IP_ADDRESS
      ;
   ;
;
to the MEC_APP and
// ETSI GS MEC 013, clause 7.3.10.3
the IUT sends a vPOST containing
   Uri set to CALLBACK_URL
   body containing
      subscriptionNotification containing
         terminalLocation containing
            address set to IP_ADDRESS
         ;
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.15-2: TP_MEC_SRV_UEDISTSUB_001_BR

TP Id	"TP_MEC_SRV_UEDISTSUB_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.10
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "location/v2/subscriptions/distance",
body containing
distanceNotificationSubscription containing
clientCorrelator indicating value CLIENT_ID,
callbackReference indicating value CALLBACK_URL,
monitoredAddress indicating value MONITORED_IP_ADDRESS,
// Wrong name should trigger an error response.
reference indicating value IP_ADDRESS
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.10.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.15-3: TP_MEC_SRV_UEDISTSUB_002_OK

TP Id	"TP_MEC_SRV_UEDISTSUB_002_OK"
Test Objective	Check that the IUT acknowledges the cancellation of UE distance notifications when commanded by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a subscriptions containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/distance/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.15-4: TP_MEC_SRV_UEDISTSUB_002_NF

TP Id	"TP_MEC_SRV_UEDISTSUB_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a subscriptions containing
subscriptionId indicating value NON_EXISTING_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/distance/{NON_EXISTING_SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.5
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.16 UE Information Lookup (UEINFLOOK)

Table 6.4.16-1: TP_MEC_SRV_UEINFLOOK_001_OK

TP Id	"TP_MEC_SRV_UEINFLOOK_001_OK"
Test Objective	Check that the IUT responds with the information pertaining to one or more UEs in a particular location when queried by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.3
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a ue_location containing
address indicating value ACR_SOME_IP
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/users",
query_parameters containing
address indicating value ACR_SOME_IP
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.3
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
userList set to any_value
;
;
to the MEC_APP
}
}	

Final Conditions

Table 6.4.16-2: TP_MEC_SRV_UEINFLOOK_001_BR

TP Id	"TP_MEC_SRV_UEINFLOOK_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.3
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a ue_location containing
address indicating value ACR_SOME_IP
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/users",
query_parameters containing
// Wrong name should trigger an error response.
addr indicating value ACR_SOME_IP
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.3
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.16-3: TP_MEC_SRV_UEINFLOOK_001_NF

TP Id	"TP_MEC_SRV_UEINFLOOK_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.3
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a ue_location containing
address indicating value ACR_SOME_IP
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/users",
query_parameters containing
addr indicating value ACR_UNKNOWN_IP
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.3
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.17 UE Information Subscription (UEINFSUB)

Table 6.4.17-1: TP_MEC_SRV_UEINFSUB_001_OK

TP Id	"TP_MEC_SRV_UEINFSUB_001_OK"
Test Objective	Check that the IUT acknowledges the UE information change subscription request when commanded by a MEC Application and notifies it when the location changes
Reference	ETSI GS MEC 013 [6], clause 7.3.5
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "location/v2/subscriptions/zonalTraffic"
   body containing
      zonalTrafficSubscription containing
         clientCorrelator indicating value CLIENT_ID,
         callbackReference indicating value CALLBACK_URL,
         zoneId indicating value ZONE_ID
      ;
   ;
;
from the MEC_APP
    }
    then {
           // ETSI GS MEC 013, clause 7.3.5.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "201 Created"
   body containing
      zonalTrafficSubscription containing
         clientCorrelator indicating value CLIENT_ID,
         callbackReference indicating value CALLBACK_URL,
         zoneId indicating value ZONE_ID
      ;
   ;
;
to the MEC_APP and
// ETSI GS MEC 013, clause 7.3.5.3
the IUT sends a vPOST containing
   Uri set to CALLBACK_URL
   body containing
      zonalPresenceNotification containing
         clientCorrelator set to CLIENT_ID,
         zoneId indicating value ZONE_ID
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.17-2: TP_MEC_SRV_UEINFSUB_001_BR

TP Id	"TP_MEC_SRV_UEINFSUB_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.5
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "location/v2/subscriptions/zonalTraffic"
body containing
zonalTrafficSubscription containing
clientCorrelator indicating value CLIENT_ID,
callbackReference indicating value CALLBACK_URL,
// Wrong name should trigger an error response.
zone indicating value ZONE_ID
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.5.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.17-3: TP_MEC_SRV_UEINFSUB_002_OK

TP Id	"TP_MEC_SRV_UEINFSUB_002_OK"
Test Objective	Check that the IUT acknowledges the cancellation of UE information change notifications when commanded by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a subscriptions containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/zonalTraffic/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.3
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.17-4: TP_MEC_SRV_UEINFSUB_002_NF

TP Id	"TP_MEC_SRV_UEINFSUB_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a subscriptions containing
subscriptionId indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/zonalTraffic/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.3
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.18 UE Location Lookup (UELOC)

Table 6.4.18-1: TP_MEC_SRV_UELOC_001_OK

TP Id	"TP_MEC_SRV_UELOC_001_OK"
Test Objective	Check that the IUT responds with a list for the location of User Equipments when queried by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a ue_location containing
zoneId indicating value ZONE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/users",
query_parameters containing
zoneId indicating value ZONE_ID
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
userInfo containing
zoneId set to ZONE_ID
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.18-2: TP_MEC_SRV_UELOC_001_BR

TP Id	"TP_MEC_SRV_UELOC_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a ue_location containing
zoneId indicating value ZONE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/users",
query_parameters containing
// Wrong name should trigger an error response.
zone indicating value ZONE_ID
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.18-3: TP_MEC_SRV_UELOC_001_NF

TP Id	"TP_MEC_SRV_UELOC_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a ue_location containing
zoneId indicating value NON_EXISTENT_ZONE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "location/v2/users",
query_parameters containing
zoneId indicating value NON_EXISTENT_ZONE_ID
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.19 UE Location Subscription (UELOCSUB)

Table 6.4.19-1: TP_MEC_SRV_UELOCSUB_001_OK

TP Id	"TP_MEC_SRV_UELOCSUB_001_OK"
Test Objective	Check that the IUT acknowledges the UE location change subscription request when commanded by a MEC Application and notifies it when the location changes
Reference	ETSI GS MEC 013 [6], clause 7.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "location/v2/subscriptions/userTracking"
body containing
userTrackingSubscription containing
clientCorrelator indicating value CLIENT_ID,
callbackReference indicating value CALLBACK_URL,
address indicating value IP_ADDRESS
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.4.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "201 Created"
body containing
userTrackingSubscription containing
clientCorrelator indicating value CLIENT_ID,
callbackReference indicating value CALLBACK_URL,
address indicating value IP_ADDRESS
;
;
;
to the MEC_APP and
// ETSI GS MEC 013, clause 7.3.4.3
the IUT sends a vPOST containing
Uri set to CALLBACK_URL
body containing
zonalPresenceNotification containing
clientCorrelator set to CLIENT_ID,
address set to IP_ADDRESS
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.19-2: TP_MEC_SRV_UELOCSUB_001_BR

TP Id	"TP_MEC_SRV_UELOCSUB_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPOST containing
uri indicating value "location/v2/subscriptions/userTracking"
body containing
userTrackingSubscription containing
// Wrong name should trigger an error response.
client indicating value CLIENT_ID,
callbackReference indicating value CALLBACK_URL,
address indicating value IP_ADDRESS
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.4.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.19-3: TP_MEC_SRV_UELOCSUB_002_OK

TP Id	"TP_MEC_SRV_UELOCSUB_002_OK"
Test Objective	Check that the IUT acknowledges the cancellation of UE location change notifications when commanded by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a subscriptions containing
subscriptionId indicating value SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/userTracking/{SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "204 No Content"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.19-4: TP_MEC_SRV_UELOCSUB_002_NF

TP Id	"TP_MEC_SRV_UELOCSUB_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a subscriptions containing
subscriptionId indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/userTracking/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.20 UE Identity tag (UETAG)

Table 6.4.20-1: TP_MEC_SRV_UETAG_001_OK

TP Id	"TP_MEC_SRV_UETAG_001_OK"
Test Objective	Check that the IUT responds with the information on a UE Identity tag when queried by a MEC Application
Reference	ETSI GS MEC 014 [7], clause 7.3.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the MEC_APP having a app_instance containing
   instance_id indicating value APP_INSTANCE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vGET containing
   uri indicating value "ui/v1/{APP_INSTANCE_ID}/ue_id_tag_info",
   query_parameters containing
      ueIdTag indicating value UE_IDENTITY_TAG
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 014, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
   status_code set to "200 OK"
   body containing
      UeIdTagInfo containing
         ueIdTagsList containing
            ueIdTags containing
               ueIdTag set to UE_IDENTITY_TAG
            ;
         ;
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.20-2: TP_MEC_SRV_UETAG_001_BR

TP Id	"TP_MEC_SRV_UETAG_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 014 [7], clause 7.3.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the MEC_APP having a app_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "ui/v1/{APP_INSTANCE_ID}/ue_id_tag_info",
query_parameters containing
// Wrong parameter name should trigger an error response.
ueTag indicating value UE_IDENTITY_TAG
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 014, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.20-3: TP_MEC_SRV_UETAG_001_NF

TP Id	"TP_MEC_SRV_UETAG_001_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 014 [7], clause 7.3.3.1
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the MEC_APP "not" having a app_instance containing
instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
uri indicating value "ui/v1/{NON_EXISTENT_APP_INSTANCE_ID}/ue_id_tag_info",
query_parameters containing
ueIdTag indicating value UE_IDENTITY_TAG
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 014, clause 7.3.3.1
the IUT sends a HTTP_RESPONSE containing
status_code set to "404 Not Found"
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.20-4: TP_MEC_SRV_UETAG_002_OK

TP Id	"TP_MEC_SRV_UETAG_002_OK"
Test Objective	Check that the IUT registers a tag (representing a UE) or a list of tags when commanded by a MEC Application
Reference	ETSI GS MEC 014 [7], clause 7.3.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the MEC_APP having a app_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "ui/v1/{APP_INSTANCE_ID}/ue_id_tag_info",
if_match indicating value PROPER_ETAG,
body containing
ueIdTagsList containing
ueIdTags containing
ueIdTag set to UE_IDENTITY_TAG,
state set to REGISTERED
;
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 014, clause 7.3.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "200 OK"
body containing
UeIdTagInfo containing
ueIdTagsList containing
ueIdTags containing
ueIdTag set to UE_IDENTITY_TAG,
state set to REGISTERED
;
;
;
;
to the MEC_APP
}
}	

Final Conditions	

Table 6.4.20-5: TP_MEC_SRV_UETAG_002_BR

TP Id	"TP_MEC_SRV_UETAG_002_BR"
Test Objective	Check that the IUT responds with an error when an unauthorized request is sent by a MEC Application
Reference	ETSI GS MEC 014 [7], clause 7.3.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the MEC_APP having a app_instance containing
   instance_id indicating value APP_INSTANCE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value "ui/v1/{APP_INSTANCE_ID}/ue_id_tag_info",
x                  if_match indicating value PROPER_ETAG,
   body containing
      ueIdTagsList containing
         ueIdTags containing
            ueIdTag set to UE_IDENTITY_TAG,
            state set to INVALID_STATE
         ;
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 014, clause 7.3.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.20-6: TP_MEC_SRV_UETAG_002_NF

TP Id	"TP_MEC_SRV_UETAG_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an unknown URI is sent by a MEC Application
Reference	ETSI GS MEC 014 [7], clause 7.3.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the MEC_APP "not" having a app_instance containing
   instance_id indicating value NON_EXISTENT_APP_INSTANCE_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPUT containing
   uri indicating value "ui/v1/{NON_EXISTENT_APP_INSTANCE_ID}/ue_id_tag_info",
x                  if_match indicating value PROPER_ETAG,
   body containing
      ueIdTagsList containing
         ueIdTags containing
            ueIdTag set to UE_IDENTITY_TAG,
            state set to REGISTERED
         ;
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 014, clause 7.3.3.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "404 Not Found"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.20-7: TP_MEC_SRV_UETAG_002_PF

TP Id	"TP_MEC_SRV_UETAG_002_PF"
Test Objective	Check that the IUT responds with an error when a request sent by a MEC Application does not comply with a required condition
Reference	ETSI GS MEC 014 [7], clause 7.3.3.2
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the MEC_APP having a app_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vPUT containing
uri indicating value "ui/v1/{APP_INSTANCE_ID}/ue_id_tag_info",
x if_match indicating value INVALID_ETAG,
body containing
ueIdTagsList containing
ueIdTags containing
ueIdTag set to UE_IDENTITY_TAG,
state set to UNREGISTERED
;
;
;
;
from the MEC_APP
}
then {
// ETSI GS MEC 014, clause 7.3.3.2
the IUT sends a HTTP_RESPONSE containing
status_code set to "412 Precondition Failed"
;
to the MEC_APP
}
}	

Final Conditions	

6.4.21 UE Tracking Subscribe (UETRACKSUB)

Table 6.4.21-1: TP_MEC_SRV_UETRACKSUB_001_OK

TP Id	"TP_MEC_SRV_UETRACKSUB_001_OK"
Test Objective	Check that the IUT acknowledges the UE location change subscription request when commanded by a MEC Application and notifies it when the UE changes location
Reference	ETSI GS MEC 013 [6], clause 7.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "location/v2/subscriptions/periodic"
   body containing
      periodicNotificationSubscription containing
         clientCorrelator indicating value CLIENT_ID,
         callbackReference indicating value CALLBACK_URL,
         address indicating value IP_ADDRESS
      ;
   ;
;
from the MEC_APP
    }
    then {
           // ETSI GS MEC 013, clause 7.3.8.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "201 Created"
   body containing
      periodicNotificationSubscription containing
         clientCorrelator indicating value CLIENT_ID,
         callbackReference indicating value CALLBACK_URL,
         address indicating value IP_ADDRESS
      ;
   ;
;
to the MEC_APP and
// ETSI GS MEC 013, clause 7.3.8.3
the IUT sends a vPOST containing
   uri indicating value CALLBACK_URL
   body containing
      subscriptionNotification containing
         terminalLocation containing
            address set to IP_ADDRESS
         ;
      ;
   ;
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.21-2: TP_MEC_SRV_UETRACKSUB_001_BR

TP Id	"TP_MEC_SRV_UETRACKSUB_001_BR"
Test Objective	Check that the IUT responds with an error when a request with incorrect parameters is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.4
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vPOST containing
   uri indicating value "location/v2/subscriptions/periodic"
   body containing
      periodicNotificationSubscription containing
         // Wrong name should trigger an error response.
         client indicating value CLIENT_ID,
         callbackReference indicating value CALLBACK_URL,
         address indicating value IP_ADDRESS
      ;
   ;
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 013, clause 7.3.8.2
the IUT sends a HTTP_RESPONSE containing
   status_code set to "400 Bad Request"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.21-3: TP_MEC_SRV_UETRACKSUB_002_OK

TP Id	"TP_MEC_SRV_UETRACKSUB_002_OK"
Test Objective	Check that the IUT acknowledges the cancellation of UE tracking notifications when commanded by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

```
with {
            the IUT being_in idle_state and
the IUT having a subscriptions containing
   subscriptionId indicating value SUBSCRIPTION_ID
;
}
```

Expected Behaviour

```
ensure that {
    when {
            the IUT receives a vDELETE containing
   uri indicating value "location/v2/subscriptions/periodic/{SUBSCRIPTION_ID}"
;
from the MEC_APP
    }
    then {
            // ETSI GS MEC 013, clause 7.3.6.4
the IUT sends a HTTP_RESPONSE containing
   status_code set to "204 No Content"
;
to the MEC_APP
    }
}
```

Final Conditions

Table 6.4.21-4: TP_MEC_SRV_UETRACKSUB_002_NF

TP Id	"TP_MEC_SRV_UETRACKSUB_002_NF"
Test Objective	Check that the IUT responds with an error when a request for an URI that cannot be mapped to a valid resource URI is sent by a MEC Application
Reference	ETSI GS MEC 013 [6], clause 7.3.6
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT "not" having a subscriptions containing
subscriptionId indicating value NON_EXISTENT_SUBSCRIPTION_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vDELETE containing
uri indicating value "location/v2/subscriptions/periodic/{NON_EXISTENT_SUBSCRIPTION_ID}"
;
from the MEC_APP
}
then {
// ETSI GS MEC 013, clause 7.3.6.4
the IUT sends a HTTP_RESPONSE containing
status_code set to "400 Bad Request"
;
to the MEC_APP
}
}	

Final Conditions	

Annex A (informative): Information on the tools to generate the present document

The Test Purposes have been first developed in textual syntax of TDL-TO, following recommendation and process reported in ETSI GR MEC-DEC 025 [i.8].

The sources are available at https://forge.etsi.org/rep/mec/gs032p2-test-purposes/tree/v2.1.1 via web access or using the Git versioning system.

The repository is structured according to groups and subgroups of the TSS and test purposes are grouped into files according to their type. Files are marked BV (Valid behaviour) when they contain test purposes for test of successful behaviour and BO_BI (Invalid vehaviour) when they contain tests on errors or expecitons managements behaviours.

The reader may make use of the IDE available as part of the TDL Open Source project (TOP), freely available at https://top.etsi.org.

Annex B (informative): Change History

Date	Version	Information about changes
June 2019	0.0.1	First proposal.
Aug 2019	0.0.2	Included changes approved in the following contributions: - MECDECODE(19)000030 - MECDECODE(19)000033 - MECDECODE(19)000034 - MECDECODE(19)000035 - MECDECODE(19)000036 - MECDECODE(19)000037 - MECDECODE(19)000038r1 - MECDECODE(19)000039 - MECDECODE(19)000040 - MECDECODE(19)000041 - MECDECODE(19)000042 - MECDECODE(19)000043 - MECDECODE(19)000044 - MECDECODE(19)000045 - MECDECODE(19)000047 - MECDECODE(19)000048 - MECDECODE(19)000049 - MECDECODE(19)000050 - MECDECODE(19)000051 - MECDECODE(19)000052 - MECDECODE(19)000053 - MECDECODE(19)000054 - MECDECODE(19)000055 - MECDECODE(19)000056 - MECDECODE(19)000057 - MECDECODE(19)000058 - MECDECODE(19)000059 - MECDECODE(19)000060 - MECDECODE(19)000061
Aug 2019	0.0.3	Incorporated changes proposed by editHelp
Oct 2020	0.0.4	Included changes approved in the following contributions: - MECDECODE(20)000050 - MECDECODE(20)000051 - MECDECODE(20)000052r1 - MECDECODE(20)000054 - MECDECODE(20)000055

History

Document history		
V2.1.1	December 2020	Publication