ETSI GS MEC 014 V3.2.1 (2024-02)

GROUP SPECIFICATION

Multi-access Edge Computing (MEC);
UE Identity API

Reference

RGS/MEC-0014v321UEidentityAPI

Keywords

API, MEC, UE identity

ETSI

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

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

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

Important notice

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

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

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

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

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

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.
No recommendation as to products and services or vendors is made or should be implied.
No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.
In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

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

Contents

Intellectual Property Rights

Essential patents

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

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

Trademarks

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

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

Foreword

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

Modal verbs terminology

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

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

1   Scope

The present document focuses on the UE Identity functionality. It describes the related application policy information (including authorization, access control, support for authentication and key management, as well as traffic rule pattern format), information flows, required information. The present document specifies the necessary API, data model and data format, considering existing API(s) if applicable.

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]   Void.

[2]   ETSI GS MEC 002: "Multi-access Edge Computing (MEC); Use Cases and Requirements".

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

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

[5]   IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

NOTE: Obsoleted by IETF RFC 8446.

[6]   IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

[7]   IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

[8]   IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage".

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

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]   OpenAPI™ Specification.

[i.2]   ETSI GR MEC 001: "Multi-access Edge Computing (MEC); Terminology".

[i.3]   3GPP TS 23.558: "Architecture for enabling Edge Applications (Release 18)".

[i.4]   ETSI GR MEC 038: "Multi-access Edge Computing (MEC); MEC in Park enterprises deployment scenario".

[i.5]   3GPP TS 23.502: "Procedures for the 5G System (5GS) (Release 18)".

[i.6]   3GPP TS 33.535: "Authentication and Key Management for Applications (AKMA) based on 3GPP credentials in the 5G System (5GS) (Release 18)".

3   Definition of terms, symbols and abbreviations

3.1   Terms

For the purposes of the present document, the terms given in ETSI GR MEC 001 [i.2] apply.

3.2   Symbols

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

K_{AF} AKMA Application Key
K_{AKMA} AKMA Anchor Key

3.3   Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR MEC 001 [i.2] and the following apply:

AAnF   AKMA Anchor Function
A-KID   AKMA Key IDentifier
AKMA   Authentication and Key Management for Applications

4   Overview

The present document specifies the API for the UE Identity feature to support the corresponding requirements defined for the Multi-access Edge Computing in ETSI GS MEC 002 [2].

Clause 5 contains the description of the feature and the information flows of the procedures. Clause 6 describes the data model and data format applied in the UE Identity API. Clause 7 is the actual API definition of the UE Identity feature.

5   Description of the features (informative)

5.1   UE Identity tag registration/de-registration

5.1.1   Introduction

The purpose of the UE Identity feature is to allow UE specific traffic rules in the MEC system.

When the MEC system supports the UE Identity feature, the MEC platform provides the functionality for an MEC application instance to register a tag (representing a UE) or a list of tags. These tags can be included in a traffic rule descriptor for tag-based traffic rules; these tags can be included in the application's application descriptor (AppD), ETSI GS MEC 010-2 [9], as part of the application package. Traffic rules, with their traffic filters, associated with an application instance are also accessible via Mp1, as described in ETSI GS MEC 011 [4]. Each tag has been mapped into a specific UE in the mobile network operator's system. The MEC platform is provided with the mapping information. How the mapping is realized is outside the scope of the present document. The UE Identity tag registration triggers the MEC platform to activate the corresponding traffic rule(s) linked to the tag. Later, if the application instance does not wish to use the traffic rule for that user, it may de-register the UE Identity tag by invoking the de-registration procedure.

5.1.2   Sequence diagrams

5.1.2.1   General

The following clauses describe how MEC applications can register and de-register tags as part of UE Identity feature. The related sequence diagrams are presented.

5.1.2.2   UE Identity tag registration

Figure 5.1.2.2-1 illustrates the message flow for the UE Identity tag registration procedure. The tag is used in UE Identity feature.

Figure 5.1.2.2-1: UE Identity tag registration

The UE Identity tag registration procedure consists of the following steps:

1)   The MEC application instance sends a PUT request with the message body containing the UeIdentityTagInfo data structure with the state set to REGISTERED to the MEC platform. The variable {appInstanceId} is set to the application instance identifier assigned to the MEC application instance.

2)   The MEC platform sends "200 OK" response with the message body containing the UeIdentityTagInfo data structure with the state set to REGISTERED.

Once the tag or the list of tags, is successfully registered in the MEC platform the related traffic rules can then be activated.

5.1.2.3   UE Identity tag de-registration

Figure 5.1.2.3-1 illustrates the message flow for the UE Identity tag deregistration procedure.
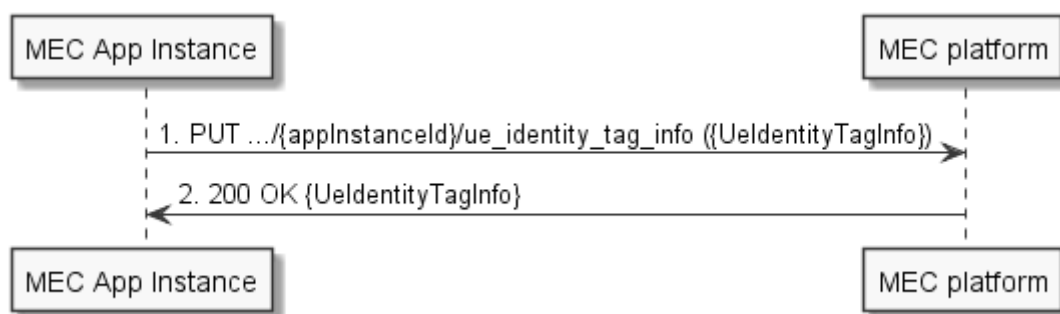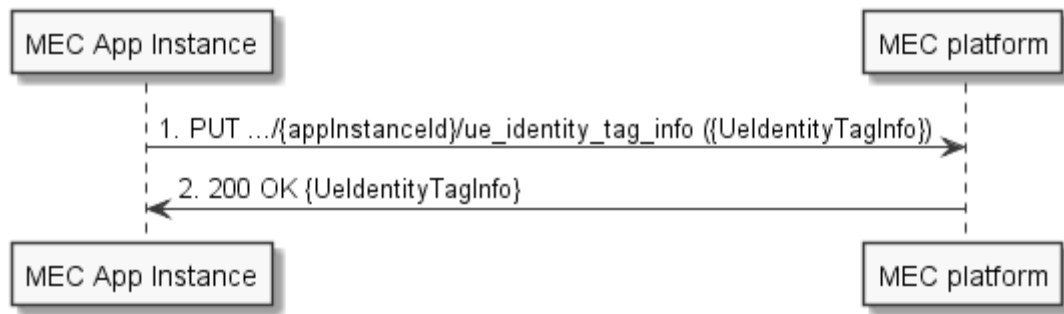
Figure 5.1.2.3-1: UE Identity tag de-registration

The UE Identity tag deregistration procedure consists of the following steps:

1) The MEC application instance sends a PUT request with the message body containing the UeIdentityTagInfo data structure with the state set to UNREGISTERED to the MEC platform. The variable {appInstanceId} is set to the application instance identifier assigned to the MEC application instance.

2) The MEC platform sends "200 OK" response with the message body containing the UeIdentityTagInfo data structure with the state set to UNREGISTERED.

Once the tag or the list of tags is successfully deregistered in the MEC platform the related traffic rules are then deactivated.

5.2 UE Identity tag acquisition

5.2.1 Introduction

If an MEC application instance provides Application Server (AS) capability to a UE hosted Application Client (AC), it may not know the mapping between the connection information associated with the AC (e.g. IP address of the UE hosting the AC) and a UE specific identifier.

In 3GPP TS 23.558 [i.3], clause 8.6.5, there is a procedure whereby an EES (Edge Enabler Server) is able to expose a UE Identifier API to an EAS (Edge Application Server) to provide it with an identifier uniquely identifying a UE. This identifier is called the UE ID and can be the 3GPP Core Network assigned UE ID (also known as AF-specific UE ID; see 3GPP TS 23.502 [i.5], clause 4.15.10) or the EES-generated Edge UE ID as defined in clause 7.2.9 of 3GPP TS 23.558 [i.3]. This approach to obtaining the UE ID was also discussed in clause 5.2.2 of ETSI GR MEC 038 [i.4].

The message flow sequence on how the 3GPP TS 23.558 [i.3] UE Identifier API can be used by an MEC application instance to determine the AC connection information (e.g. IP address) to UE ID mapping is described in the following subclauses. An MEC application instance may also wish to obtain the UE ID for other purposes, such as UE access authentication where IP address may be insufficient.

5.2.2 Sequence diagrams

5.2.2.1 General

The following clauses describe how an MEC application instance can obtain an 3GPP Network UE ID, 3GPP TS 23.558 [i.3].

5.2.2.2 UE Identifier acquisition

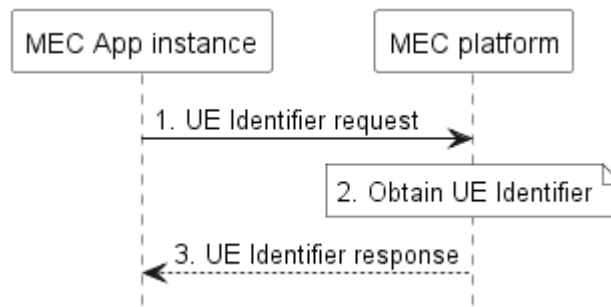Figure 5.2.2.2-1 illustrates the message flow for the UE Identifier acquisition procedure.

Figure 5.2.2.2-1: UE Identifier acquisition

1) The MEC App instance invokes the UE Identifier feature by making a request to the MEP.

2) The MEP uses the received AC connection information in the step 1 (e.g. IP address) and obtains the UE Identifier by interacting with NEF as specified in clause 4.15.10 of 3GPP TS 23.502 [i.5].

3) The MEP provides the obtained UE identifier as UE ID to the MEC App instance. The UE ID is specific to the given MEC App instance.

Once the MEC App instance has obtained the UE ID it is able to use that in any additional procedures that require a UE specific identifier to be provided.

5.3 AKMA authentication function

5.3.1 Introduction

AKMA is standardized in 3GPP TS 33.535 [i.6] to specify the security features and mechanisms to support authentication and key management aspects for applications based on subscription credential(s) in 5G system. After the primary authentication procedure, when AKMA is selected to be used between a UE and AF the associated key material (A-KID, K_{AKMA}) is generated in the UE and the AUSF. The UE stores the AKMA key material. On the contrary, the AUSF need not store the AKMA key material after it has been delivered to the AAnF. When UE initiates communication with an AF supporting AKMA, the AF will obtain the required AKMA Application key (K_{AF}) from the corresponding AAnF (if the AF does not already have an active context associated with the A-KID received from the UE).

The MEC Platform (MEP) may support the AAnF function as an Authentication Proxy (AP) which takes the role of an AF and delegates a group of Application Servers (ASs).Thus MEP as an AP helps the ASs behind the AP to execute AKMA procedures to save the consumption of signalling resources and AAnF computing resources. It may also relieve the AS of security tasks as defined in clause 4.7 of 3GPP TS 33.535 [i.6].

5.3.2 Architecture when MEP as an AKMA AP

Referenced figures 4.7.1-1 and 4.7.1-2 of 3GPP TS 33.535 [i.6], architecture when MEP as an AKMA AP is showed in figure 5.3.2-1.
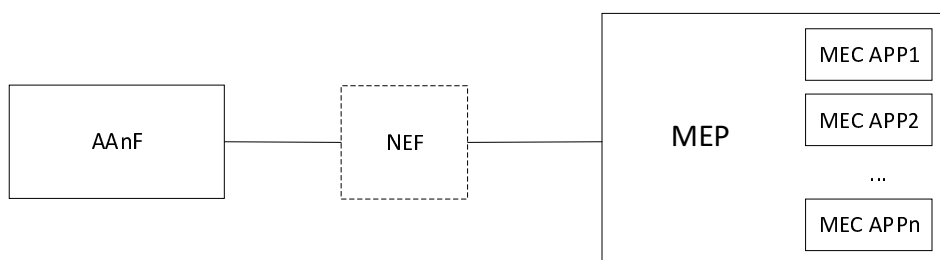

Figure 5.3.2-1: MEP as an AP to execute AKMA authentication

When MEP is in the MNO trusted domain, NEF is not needed. Otherwise NEF serves as a bridge for information exchange between AAnF and MEP.

6   Data model & Data format (normative)

6.1   Introduction

The following clauses specify the data types that are used to implement the UE Identity feature, for which the relevant sequence diagrams are described in clauses 5.1.2.2 and 5.1.2.3.

6.2   Resource data types

6.2.1   Introduction

This clause defines data structures to be used in resource representations.

6.2.2   Type: UeIdentityTagInfo

This type represents the information of UE Identity tag used in the UE Identity feature.

Table 6.2.2-1: Definition of type UeIdentityTagInfo

Attribute name	Data type	Cardinality	Description
ueIdentityTags	Structure (inlined)	1..N	1 to N tags presented by a MEC Application instance to a MEC Platform
>ueIdentityTag	String	1	Specific tag presented by a MEC Application instance to a MEC Platform
>state	Enum (inlined)	1	The following numeric values are defined: 0 = UNREGISTERED 1 = REGISTERED

6.3   Subscription types

In the present document, no subscription data types are defined.

6.4   Notification types

In the present document, no notifications data types are defined.

6.5   Referenced structured data types

In the present document, no referenced structured data types are defined.

6.6   Referenced simple data types

In the present document, no referenced simple data types are defined.

7 API definition (normative)

7.1 Introduction

This clause defines the resources and operations of the UE identity API.

7.2 Global definitions and resource structure

All resource URIs of this API shall have the following root:

{apiRoot}/{apiName}/{apiVersion}/

The "apiRoot" is discovered using the service registry. The "apiName" shall be set to "ui" and the "apiVersion" shall be set to "v1" for the present document. It includes the scheme ("https"), host and optional port, and an optional prefix string. The API shall support HTTP over TLS (also known as HTTPS) using TLS version 1.2 (as defined by IETF RFC 5246 [5]). TLS 1.3 (including the new specific requirements for TLS 1.2 implementation) defined by IETF RFC 8446 [6] should be supported. HTTP without TLS shall not be used. Versions of TLS earlier than 1.2 shall neither be supported nor used. All resource URIs in the clauses below are defined relative to the above root URI.

The content format JSON shall be supported.

The JSON format is signalled by the content type "application/json".

This API shall require the use of the OAuth 2.0 client credentials grant type according to IETF RFC 6749 [7] with bearer tokens according to IETF RFC 6750 [8]. See clause 6.16 of ETSI GS MEC 009 [3] for more information. The token endpoint can be discovered as part of the service availability query procedure defined in ETSI GS MEC 011 [4]. How the client credentials are provisioned into the MEC application is out of scope of the present document.

This API supports additional application-related error information to be provided in the HTTP response when an error occurs. See clause 6.15 of ETSI GS MEC 009 [3] for more information.

Figure 7.2-1 illustrates the resource URI structure of this API.

{apiRoot}/ui/v1

/{appInstanceId}

/ue_identity_tag_info
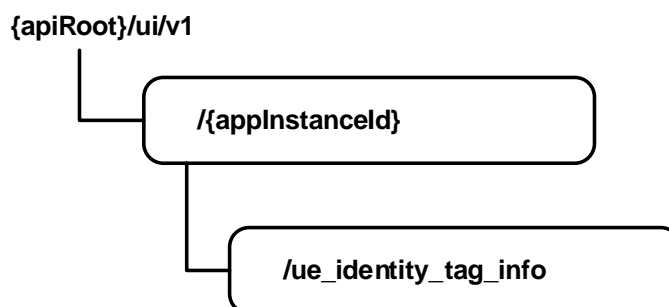
Figure 7.2-1: Resource URI structure of the UE Identity API

Table 7.2-1 provides an overview of the resources defined by the present document, and the applicable HTTP methods.

Table 7.2-1: Resources and methods overview

Resource name	Resource URI	HTTP method	Meaning
Individual UeIdentityTagInfo	…/{appInstanceId}/ue_identity_tag_info	GET	Retrieve information about specific UeIdentityTagInfo
		PUT	Register/De-register the information about specific UeIdentityTagInfo

7.3 Resource: individual UeIdentityTagInfo

7.3.1 Description

This resource is used to represent the information of a single UE Identity tag resource, which follows the resource data type of "UeIdentityTagInfo" as specified in clause 6.2.2.

7.3.2 Resource definition

Resource URI: {apiRoot}/ui/v1/{appInstanceId}/ue_identity_tag_info

Resource URI Variables for this resource are defined in table 7.3.2-1.

Table 7.3.2-1: Resource URI Variables for resource "individual UeIdentityTagInfo"

Name	Definition
apiRoot	See clause 7.2
appInstanceId	Represents a MEC application instance

7.3.3 Resource Methods

7.3.3.1 GET

This method retrieves information about a UeIdentityTagInfo resource.

This method shall comply with the URI query parameters, request and response data structures, and response codes, as specified in tables 7.3.3.1-1 and 7.3.3.1-2.

Table 7.3.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
ueIdentityTag	String	1..N	Represents one or more UEs

Table 7.3.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality		Remarks	
	n/a				
Response body	Data type	Cardinality	Response codes	Remarks	
	UeIdentityTagInfo	1	200 OK	It is used to indicate nonspecific success. The response body contains a representation of the UeIdentityTagInfo resource.	
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.	
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.	
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.	

7.3.3.2   PUT

This method registers/deregisters one or more UE Identity tags. It has "replace" semantics.

This method is typically used in "UE Identity tag registration" procedure as described in clause 5.2.2 and "UE Identity tag de-registration" procedure as described in clause 5.1.2.3.

This method shall comply with the URI query parameters, request and response data structures, and response codes, as specified in tables 7.3.3.2-1 and 7.3.3.2-2.

Table 7.3.3.2-1: URI query parameters supported by the PUT method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.3.3.2-2: Data structures supported by the PUT request/response on this resource

Request body	Data type	Cardinality		Remarks
	UeIdentityTagInfo	1		The updated "state" for each included UE Identity tag is included in the entity body of the request.
	Data type	**Cardinality**	**Response codes**	**Remarks**
Response body	UeIdentityTagInfo	1	200 OK	It is used to indicate success. The response body contains a representation of the UeIdentityTagInfo resource.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	412 Precondition Failed	It is used when a condition has failed during conditional requests, e.g. when using ETags to avoid write conflicts. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.3.3.3   PATCH

Not supported.

7.3.3.4   POST

Not supported.

7.3.3.5   DELETE

Not supported.

Annex A (informative): Complementary material for API utilization

To complement the definitions for each method and resource defined in the interface clauses of the present document, ETSI MEC ISG is providing for the *UE identity* API a supplementary description file compliant to the OpenAPI Specification [i.1].

In case of discrepancies between the supplementary description file and the related data structure definitions in the present document, the data structure definitions take precedence.

The supplementary description file, relating to the present document, is located at:

- https://forge.etsi.org/rep/mec/gs014-ue-identity-api.

Annex B (informative):
Change history

Version	Date	Information about changes
V3.0.1	September 2022	TB adoption of WI MEC, see contribution MEC(22)000335r1 in RC MEC(22)DEC210
V3.0.2	October 2022	Implements document MEC(22)000447r1,MEC(22)000473r1, MEC(22)000474r1
V3.0.3	December 2022	Implements document MEC(22)000536r1 and MEC(22)000551r1
V3.0.4	December 2022	Clean-up done by *editHelp!*
V3.0.5	January 2023	Stable draft v3.0.4 is moved to Final draft v3.0.5 as no objection raised on the MEC list. The two drafts are similar.
V3.0.6	February 2023	Implements document with MEC(23)000052r1
V3.0.7	May 2023	Implements document s with MEC(23)000173r1, MEC(23)000179r3 and MEC(23)000186r1
V3.0.8	June 2023	Implements document with MEC(23)000211r1
V3.1.1	July 2023	Publication
V3.1.2	July 2023	Early draft of V3.2.1
V3.1.3	October 2023	Implements document with MEC(23)000372
V3.1.4	November 2023	Implements document with MEC(23)000456r1 and MEC(23)000481
V3.1.5	December 2023	Implements document with MEC(23)000551r1 V3.1.5 is considered as a Stable draft

History

Document history		
V1.1.1	February 2018	Publication
V2.1.1	March 2021	Publication
V3.1.1	July 2023	Publication
V3.2.1	February 2024	Publication