

ETSI TS 129 222 V15.14.0 (2026-03)



TECHNICAL SPECIFICATION

LTE;
5G;
Common API Framework for 3GPP Northbound APIs
(3GPP TS 29.222 version 15.14.0 Release 15)

get full document from standards.iteh.ai



Reference

RTS/TSGC-0329222vfe0

Keywords

5G,LTE

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 the
[ETSI Search & Browse Standards](#) application.

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 on [ETSI deliver](#) repository.

Users should be aware that the present document may be revised or have its status changed, this information is available in the [Milestones listing](#).

If you find errors in the present document, please send your comments to the relevant service listed under [Committee Support Staff](#).

If you find a security vulnerability in the present document, please report it through our [Coordinated Vulnerability Disclosure \(CVD\)](#) program.

Notice of disclaimer & limitation of liability

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

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

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

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

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

Copyright Notification

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

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2026.
All rights reserved.

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 IPR online database](#).

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™**, **LTE™** and **5G™** logo 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.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found at [3GPP to ETSI numbering cross-referencing](#).

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.

Contents

Intellectual Property Rights	2
Legal Notice	2
Modal verbs terminology.....	2
Foreword.....	10
1 Scope	11
2 References	11
3 Definitions and abbreviations.....	12
3.1 Definitions	12
3.2 Abbreviations	12
4 Overview	12
4.1 Introduction	12
4.2 Service Architecture	13
4.3 Functional Entities.....	13
4.3.1 API invoker.....	13
4.3.2 CAPIF core function.....	14
4.3.3 API exposing function	14
4.3.4 API publishing function.....	14
4.3.5 API management function	15
5 Services offered by the CAPIF Core Function.....	15
5.1 Introduction of Services	15
5.2 CAPIF_Discover_Service_API.....	16
5.2.1 Service Description.....	16
5.2.1.1 Overview	16
5.2.2 Service Operations.....	16
5.2.2.1 Introduction	16
5.2.2.2 Discover_Service_API.....	16
5.2.2.2.1 General	16
5.2.2.2.2 API invoker discovering service API using Discover_Service_API service operation.....	16
5.3 CAPIF_Publish_Service_API	16
5.3.1 Service Description.....	16
5.3.1.1 Overview.....	16
5.3.2 Service Operations.....	17
5.3.2.1 Introduction.....	17
5.3.2.2 Publish_Service_API	17
5.3.2.2.1 General	17
5.3.2.2.2 API publishing function publishing service APIs on CAPIF core function using Publish_Service_API service operation	17
5.3.2.3 Unpublish_Service_API.....	17
5.3.2.3.1 General	17
5.3.2.3.2 API publishing function un-publishing service APIs from CAPIF core function using Unpublish_Service_API service operation.....	18
5.3.2.4 Get_Service_API	18
5.3.2.4.1 General	18
5.3.2.4.2 API publishing function retrieving service APIs from CAPIF core function using Get_Service_API service operation.....	18
5.3.2.5 Update_Service_API.....	18
5.3.2.5.1 General	18
5.3.2.5.2 API publishing function updating published service APIs on CAPIF core function using Update_Service_API service operation.....	18
5.4 CAPIF_Events_API	19
5.4.1 Service Description.....	19
5.4.1.1 Overview.....	19
5.4.2 Service Operations.....	19

5.4.2.1	Introduction.....	19
5.4.2.2	Subscribe_Event.....	19
5.4.2.2.1	General	19
5.4.2.2.2	Subscribing to CAPIF events using Subscribe_Event service operation.....	19
5.4.2.3	Unsubscribe_Event	20
5.4.2.3.1	General	20
5.4.2.3.2	Unsubscribing from CAPIF events using Unsubscribe_Event service operation.....	20
5.4.2.4	Notify_Event.....	20
5.4.2.4.1	General	20
5.4.2.4.2	Notifying CAPIF events using Notify_Event service operation.....	20
5.5	CAPIF_API_Invoker_Management_API.....	20
5.5.1	Service Description.....	20
5.5.1.1	Overview.....	20
5.5.2	Service Operations.....	20
5.5.2.1	Introduction.....	20
5.5.2.2	Onboard_API_Invoker.....	21
5.5.2.2.1	General	21
5.5.2.2.2	API invoker on-boarding itself as a recognized user of CAPIF using Onboard_API_Invoker service operation.....	21
5.5.2.3	Offboard_API_Invoker	22
5.5.2.3.1	General	22
5.5.2.3.2	API invoker off-boarding itself as a recognized user of CAPIF using Offboard_API_Invoker service operation.....	22
5.5.2.4	Notify_Onboarding_Completion	22
5.5.2.4.1	General	22
5.5.2.4.2	Notifying Onboarding completion using Notify_Onboarding_Completion service operation.....	22
5.6	CAPIF_Security_API.....	23
5.6.1	Service Description.....	23
5.6.1.1	Overview	23
5.6.2	Service Operations.....	23
5.6.2.1	Introduction.....	23
5.6.2.2	Obtain_Security_Method.....	23
5.6.2.2.1	General	23
5.6.2.2.2	Request service API security method from CAPIF using Obtain_Security_Method service operation.....	23
5.6.2.3	Obtain_Authorization.....	24
5.6.2.3.1	General	24
5.6.2.3.2	Obtain authorization using Obtain_Authorization service operation.....	24
5.6.2.4	Obtain_API_Invoker_Info	24
5.6.2.4.1	General	24
5.6.2.4.2	Obtain API invoker's security information using Obtain_API_Invoker_Info service operation	25
5.6.2.5	Revoke_Authentication.....	25
5.6.2.5.1	General	25
5.6.2.5.2	Invalidate authorization using Revoke_Authorization service operation	25
5.7	CAPIF_Monitoring_API.....	25
5.8	CAPIF_Logging_API_Invocation_API	26
5.8.1	Service Description.....	26
5.8.1.1	Overview	26
5.8.2	Service Operations.....	26
5.8.2.1	Introduction.....	26
5.8.2.2	Log_API_Invocation_API	26
5.8.2.2.1	General	26
5.8.2.2.2	Logging service API invocations using Log_API_Invocation service operation	26
5.9	CAPIF_Auditing_API.....	26
5.9.1	Service Description.....	26
5.9.1.1	Overview.....	26
5.9.2	Service Operations.....	27
5.9.2.1	Introduction.....	27
5.9.2.2	Query_Invocation_Logs_API	27
5.9.2.2.1	General	27
5.9.2.2.2	Query API invocation information logs using Query_Invocation_Logs service operation.....	27
5.10	CAPIF_Access_Control_Policy_API.....	27

5.10.1	Service Description.....	27
5.10.1.1	Overview.....	27
5.10.2	Service Operations.....	27
5.10.2.1	Introduction.....	27
5.10.2.2	Obtain_Access_Control_Policy.....	28
5.10.2.2.1	General.....	28
5.10.2.2.2	API exposing function obtaining access control policy from the CAPIF core function using Obtain_Access_Control_Policy service operation.....	28
5.10.3	Related Events.....	28
6	Services offered by the API exposing function.....	28
6.1	Introduction of Services.....	28
6.2	AEF_Security_API.....	28
6.2.1	Service Description.....	28
6.2.1.1	Overview.....	28
6.2.2	Service Operations.....	29
6.2.2.1	Introduction.....	29
6.2.2.2	Initiate_Authentication.....	29
6.2.2.2.1	General.....	29
6.2.2.2.2	API invoker initiating authentication using Initiate_Authentication service operation.....	29
7	CAPIF Design Aspects Common for All APIs.....	30
7.1	General.....	30
7.2	Data Types.....	30
7.2.1	General.....	30
7.2.2	Referenced structured data types.....	30
7.2.3	Referenced Simple data types and enumerations.....	30
7.3	Usage of HTTP.....	31
7.4	Content type.....	31
7.5	URI structure.....	31
7.6	Notifications.....	32
7.7	Error handling.....	32
7.8	Feature negotiation.....	32
7.9	HTTP headers.....	32
7.10	Conventions for Open API specification files.....	32
8	CAPIF API Definition.....	33
8.1	CAPIF_Discover_Service_API.....	33
8.1.1	API URI.....	33
8.1.2	Resources.....	33
8.1.2.1	Overview.....	33
8.1.2.2	Resource: All published service APIs.....	33
8.1.2.2.1	Description.....	33
8.1.2.2.2	Resource Definition.....	33
8.1.2.2.3	Resource Standard Methods.....	34
8.1.2.2.4	Resource Custom Operations.....	34
8.1.3	Notifications.....	34
8.1.4	Data Model.....	35
8.1.4.1	General.....	35
8.1.4.2	Structured data types.....	35
8.1.4.2.1	Introduction.....	35
8.1.4.2.2	Type: DiscoveredAPIs.....	35
8.1.4.3	Simple data types and enumerations.....	35
8.1.5	Error Handling.....	35
8.1.6	Feature negotiation.....	35
8.2	CAPIF_Publish_Service_API.....	36
8.2.1	API URI.....	36
8.2.2	Resources.....	36
8.2.2.1	Overview.....	36
8.2.2.2	Resource: APF published APIs.....	37
8.2.2.2.1	Description.....	37
8.2.2.2.2	Resource Definition.....	37
8.2.2.2.3	Resource Standard Methods.....	37

8.2.2.2.4	Resource Custom Operations	38
8.2.2.3	Resource: Individual APF published API	38
8.2.2.3.1	Description	38
8.2.2.3.2	Resource Definition	38
8.2.2.3.3	Resource Standard Methods	38
8.2.2.3.4	Resource Custom Operations	40
8.2.3	Notifications	40
8.2.4	Data Model	40
8.2.4.1	General	40
8.2.4.2	Structured data types	41
8.2.4.2.1	Introduction	41
8.2.4.2.2	Type: ServiceAPIDescription	41
8.2.4.2.3	Type: InterfaceDescription	41
8.2.4.2.4	Type: AefProfile	42
8.2.4.2.5	Type: Version	42
8.2.4.2.6	Type: Resource	42
8.2.4.2.7	Type: CustomOperation	43
8.2.4.3	Simple data types and enumerations	43
8.2.4.3.1	Introduction	43
8.2.4.3.2	Simple data types	43
8.2.4.3.3	Enumeration: Protocol	43
8.2.4.3.4	Enumeration: DataFormat	43
8.2.4.3.5	Enumeration: CommunicationType	44
8.2.4.3.6	Enumeration: SecurityMethod	44
8.2.4.3.7	Enumeration: Operation	44
8.2.5	Error Handling	44
8.2.6	Feature negotiation	44
8.3	CAPIF_Events_API	44
8.3.1	API URI	44
8.3.2	Resources	45
8.3.2.1	Overview	45
8.3.2.2	Resource: CAPIF Events Subscriptions	45
8.3.2.2.1	Description	45
8.3.2.2.2	Resource Definition	45
8.3.2.2.3	Resource Standard Methods	46
8.3.2.2.4	Resource Custom Operations	46
8.3.2.3	Resource: Individual CAPIF Events Subscription	46
8.3.2.3.1	Description	46
8.3.2.3.2	Resource Definition	46
8.3.2.3.3	Resource Standard Methods	47
8.3.2.3.4	Resource Custom Operations	47
8.3.3	Notifications	47
8.3.3.1	General	47
8.3.3.2	Event Notification	47
8.3.3.2.1	Description	47
8.3.3.2.2	Notification definition	47
8.3.4	Data Model	48
8.3.4.1	General	48
8.3.4.2	Structured data types	49
8.3.4.2.1	Introduction	49
8.3.4.2.2	Type: EventSubscription	49
8.3.4.2.3	Type: EventNotification	49
8.3.4.3	Simple data types and enumerations	50
8.3.4.3.1	Introduction	50
8.3.4.3.2	Simple data types	50
8.3.4.3.3	Enumeration: CAPIFEvent	50
8.3.5	Error Handling	50
8.3.6	Feature negotiation	50
8.4	CAPIF_API_Invoker_Management_API	51
8.4.1	API URI	51
8.4.2	Resources	51
8.4.2.1	Overview	51

8.4.2.2	Resource: On-boarded API invokers.....	52
8.4.2.2.1	Description	52
8.4.2.2.2	Resource Definition.....	52
8.4.2.2.3	Resource Standard Methods	52
8.4.2.2.4	Resource Custom Operations	53
8.4.2.3	Resource: Individual On-boarded API Invoker.....	53
8.4.2.3.1	Description	53
8.4.2.3.2	Resource Definition.....	53
8.4.2.3.3	Resource Standard Methods	53
8.4.2.3.4	Resource Custom Operations	54
8.4.3	Notifications	54
8.4.3.1	General	54
8.4.3.2	Notify_Onboarding_Completion	54
8.4.3.2.1	Description	54
8.4.3.2.2	Notification definition	54
8.4.4	Data Model	55
8.4.4.1	General	55
8.4.4.2	Structured data types	56
8.4.4.2.1	Introduction	56
8.4.4.2.2	Type: APIInvokerEnrolmentDetails.....	56
8.4.4.2.3	Type: Void.....	56
8.4.4.2.4	Type: APIList.....	56
8.4.4.2.5	Type: OnboardingInformation.....	57
8.4.4.2.6	Type: Void.....	57
8.4.4.2.7	Type: OnboardingNotification	57
8.4.4.3	Simple data types and enumerations	57
8.4.5	Error Handling	57
8.4.6	Feature negotiation	57
8.5	CAPIF_Security_API.....	58
8.5.1	API URI.....	58
8.5.2	Resources.....	58
8.5.2.1	Overview.....	58
8.5.2.2	Resource: Trusted API invokers	59
8.5.2.2.1	Description	59
8.5.2.2.2	Resource Definition.....	59
8.5.2.2.3	Resource Standard Methods	59
8.5.2.2.4	Resource Custom Operations	59
8.5.2.3	Resource: Individual trusted API invokers	59
8.5.2.3.1	Description	59
8.5.2.3.2	Resource Definition.....	60
8.5.2.3.3	Resource Standard Methods	60
8.5.2.3.4	Resource Custom Operations	62
8.5.3	Notifications	64
8.5.3.1	General	64
8.5.3.2	Authorization revoked notification	64
8.5.3.2.1	Description	64
8.5.3.2.2	Notification definition	64
8.5.4	Data Model	65
8.5.4.1	General	65
8.5.4.2	Structured data types	66
8.5.4.2.1	Introduction	66
8.5.4.2.2	Type: ServiceSecurity	66
8.5.4.2.3	Type: SecurityInformation	66
8.5.4.2.4	Void.....	67
8.5.4.2.5	Type: SecurityNotification	67
8.5.4.2.6	Type: AccessTokenReq.....	67
8.5.4.2.7	Type: AccessTokenRsp.....	68
8.5.4.2.8	Type: AccessTokenClaims.....	68
8.5.4.3	Simple data types and enumerations	69
8.5.4.3.1	Introduction	69
8.5.4.3.2	Simple data types.....	69
8.5.4.3.3	Enumeration: Cause.....	69

8.5.5	Error Handling	69
8.5.6	Feature negotiation	69
8.6	CAPIF_Access_Control_Policy_API	69
8.6.1	API URI	69
8.6.2	Resources	70
8.6.2.1	Overview	70
8.6.2.2	Resource: Access Control Policy List	70
8.6.2.2.1	Description	70
8.6.2.2.2	Resource Definition	70
8.6.2.2.3	Resource Standard Methods	71
8.6.2.2.4	Resource Custom Operations	71
8.6.3	Notifications	71
8.6.4	Data Model	71
8.6.4.1	General	71
8.6.4.2	Structured data types	72
8.6.4.2.1	Introduction	72
8.6.4.2.2	Type: AccessControlPolicyList	72
8.6.4.2.3	Type: ApiInvokerPolicy	72
8.6.4.2.4	Type: TimeRangeList	72
8.6.4.3	Simple data types and enumerations	73
8.6.5	Error Handling	73
8.6.6	Feature negotiation	73
8.7	CAPIF_Logging_API_Invocation_API	73
8.7.1	API URI	73
8.7.2	Resources	73
8.7.2.1	Overview	73
8.7.2.2	Resource: Logs	74
8.7.2.2.1	Description	74
8.7.2.2.2	Resource Definition	74
8.7.2.2.3	Resource Standard Methods	74
8.7.2.2.4	Resource Custom Operations	75
8.7.3	Notifications	75
8.7.4	Data Model	75
8.7.4.1	General	75
8.7.4.2	Structured data types	76
8.7.4.2.1	Introduction	76
8.7.4.2.2	Type: InvocationLog	76
8.7.4.2.3	Type: Log	77
8.7.4.3	Simple data types and enumerations	77
8.7.5	Error Handling	78
8.7.6	Feature negotiation	78
8.8	CAPIF_Auditing_API	78
8.8.1	API URI	78
8.8.2	Resources	78
8.8.2.1	Overview	78
8.8.2.2	Resource: All service API invocation logs	79
8.8.2.2.1	Description	79
8.8.2.2.2	Resource Definition	79
8.8.2.2.3	Resource Standard Methods	79
8.8.2.2.4	Resource Custom Operations	80
8.8.3	Notifications	80
8.8.4	Data Model	80
8.8.4.1	General	80
8.8.4.2	Structured data types	80
8.8.4.3	Simple data types and enumerations	80
8.8.5	Error Handling	81
8.8.6	Feature negotiation	81
9	AEF API Definition	81
9.1	AEF_Security_API	81
9.1.1	API URI	81
9.1.2	Resources	81

9.1.2.1	Resource Custom Operations	81
9.1.2a	Custom Operations without associated resources	81
9.1.2a.1	Overview	81
9.1.2a.2	Operation: check-authentication	82
9.1.2a.2.1	Description	82
9.1.2a.2.2	Operation Definition	82
9.1.2a.3	Operation: revoke-authorization	82
9.1.2a.3.1	Description	82
9.1.2a.3.2	Operation Definition	82
9.1.3	Notifications	83
9.1.4	Data Model	83
9.1.4.1	General	83
9.1.4.2	Structured data types	83
9.1.4.2.1	Introduction	83
9.1.4.2.2	Type: CheckAuthenticationReq	84
9.1.4.2.3	Type: CheckAuthenticationRsp	84
9.1.4.2.4	Type: RevokeAuthorizationReq	84
9.1.4.2.5	Type: RevokeAuthorizationRsp	84
9.1.4.3	Simple data types and enumerations	84
9.1.5	Error Handling	84
9.1.6	Feature negotiation	84
10	Security	85
10.1	General	85
10.2	CAPIF-1/1e security	85
10.3	CAPIF-2/2e security and securely invoking service APIs	85
Annex A (normative): OpenAPI specification		85
A.1	General	85
A.2	CAPIF_Discover_Service_API	85
A.3	CAPIF_Publish_Service_API	87
A.4	CAPIF_Events_API	93
A.5	CAPIF_API_Invoker_Management_API	96
A.6	CAPIF_Security_API	99
A.7	CAPIF_Access_Control_Policy_API	104
A.8	CAPIF_Logging_API_Invocation_API	106
A.9	CAPIF_Auditing_API	108
A.10	AEF_Security_API	110
Annex B (informative): Change history		113
History		117

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

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

Sample Document

get full document from standards.iteh.ai

1 Scope

The present specification describes the protocol for the Common API Framework (CAPIF) for 3GPP Northbound APIs. The CAPIF and the related stage 2 architecture and functional requirements are defined in 3GPP TS 23.222 [2].

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

- [1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".
- [2] 3GPP TS 23.222: "Functional architecture and information flows to support Common API Framework for 3GPP Northbound APIs; Stage 2".
- [3] Open API Initiative, "OpenAPI 3.0.0 Specification", <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.0.md>.
- [4] IETF RFC 7230: "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing".
- [5] IETF RFC 7231: "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content".
- [6] IETF RFC 7232: "Hypertext Transfer Protocol (HTTP/1.1): Conditional Requests".
- [7] IETF RFC 7233: "Hypertext Transfer Protocol (HTTP/1.1): Range Requests".
- [8] IETF RFC 7234: "Hypertext Transfer Protocol (HTTP/1.1): Caching".
- [9] IETF RFC 7235: "Hypertext Transfer Protocol (HTTP/1.1): Authentication".
- [10] IETF RFC 7540: "Hypertext Transfer Protocol Version 2 (HTTP/2)".
- [11] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".
- [12] IETF RFC 7159: "The JavaScript Object Notation (JSON) Data Interchange Format".
- [13] IETF RFC 6455: "The WebSocket Protocol".
- [14] 3GPP TS 29.122: "T8 reference point for northbound Application Programming Interfaces (APIs)".
- [15] 3GPP TS 29.522: "5G System; Network Exposure Function Northbound APIs; Stage 3".
- [16] 3GPP TS 33.122: "Security Aspects of Common API Framework for 3GPP Northbound APIs".
- [17] IETF RFC 2617: "HTTP Authentication: Basic and Digest Access Authentication".
- [18] 3GPP TS 29.501: "5G System; Principles and Guidelines for Services Definition; Stage 3".
- [19] 3GPP TS 29.571: "5G System; Common Data Types for Service Based Interfaces Stage 3".
- [20] IETF RFC 7239: "Forwarded HTTP Extension".
- [21] 3GPP TS 29.500: "5G System; Technical Realization of Service Based Architecture; Stage 3". [22] W3C HTML 4.01 Specification, <https://www.w3.org/TR/2018/SPSD-html401-20180327/>.

- [23] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".
- [24] IETF RFC 7519: "JSON Web Token (JWT)".
- [25] IETF RFC 7515: "JSON Web Signature (JWS)".
- [26] IETF RFC 5280: "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

API registry: API registry is a registry maintained by the CAPIF core function to store information about the service APIs based on the data models defined in this specification. The structure of the API registry is out of scope of this specification.

CAPIF administrator: An authorized user with special permissions for CAPIF operations.

PLMN trust domain: The entities protected by adequate security and controlled by the PLMN operator or a trusted 3rd party of the PLMN.

Service API: The interface through which a component of the system exposes its services to API invokers by abstracting the services from the underlying mechanisms.

Subscriber: A functional entity that subscribes to another functional entity for notifications.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

AEF	API Exposing Function
AMF	API Management Function
APF	API Publishing Function
AS	Application Server
CAPIF	Common API Framework
CCF	CAPIF Core Function
JSON	JavaScript Object Notation
REST	Representational State Transfer
SCEF	Service Capability Exposure Function
SCS	Service Capability Server

4 Overview

4.1 Introduction

In 3GPP, there are multiple northbound API-related specifications. To avoid duplication and inconsistency of approaches between different API specifications and to specify common services (e.g. authorization), 3GPP has considered in 3GPP TS 23.222 [2] the development of a common API framework (CAPIF) that includes common aspects applicable to any northbound service APIs.

The present document specifies the APIs needed to support CAPIF.

4.2 Service Architecture

3GPP TS 23.222 [2], clause 6 specifies the functional entities and domains of the functional model, which is depicted in Figure 4.2-1, in detail.

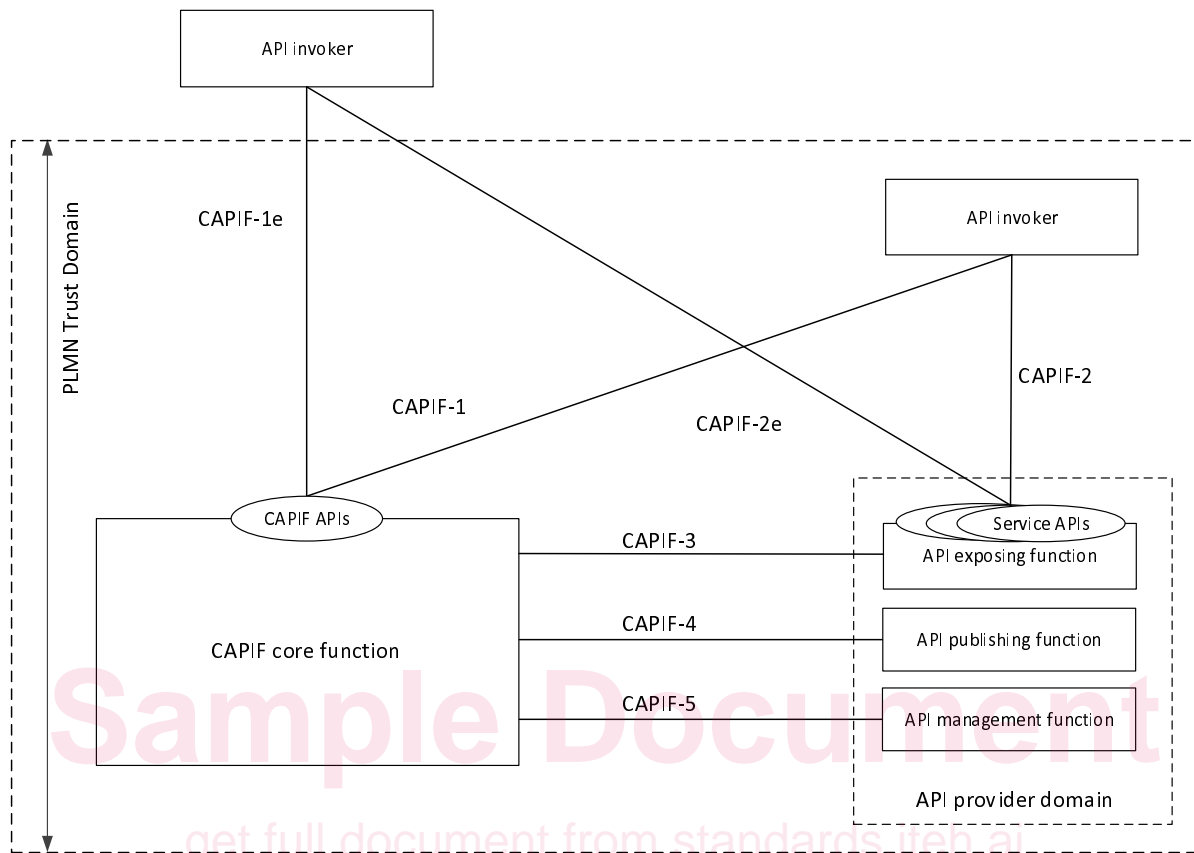


Figure 4.2-1: CAPIF Functional Model

CAPIF-1 and CAPIF-1e reference points connect an API invoker inside the PLMN Trust Domain and an API invoker outside the PLMN Trust Domain respectively, with the CAPIF core function.

CAPIF-2 and CAPIF-2e reference points connect an API invoker inside the PLMN Trust Domain and an API invoker outside the PLMN Trust Domain respectively, with the API exposing function.

CAPIF-3 reference point connects an API exposing function inside the PLMN Trust Domain with the CAPIF core function.

CAPIF-4 reference point connects an API publishing function inside the PLMN Trust Domain with the CAPIF core function.

CAPIF-5 reference point connects an API management function inside the PLMN Trust Domain with the CAPIF core function.

NOTE: The API exposing function, API publishing function and API management function are part the API provider domain which can be implemented by the Service Capability Exposure Function (SCEF) and/or the Network Exposure Function (NEF).

4.3 Functional Entities

4.3.1 API invoker

The API invoker is typically provided by a 3rd party application provider who has service agreement with PLMN operator. The API invoker may reside within the same trust domain as the PLMN operator network.

The API invoker supports several capabilities such as supporting

- the authentication and obtaining authorization and discovering using CAPIF-1/CAPIF-1e reference point as defined in 3GPP TS 23.222 [2]; and
- invoking the Service APIs using CAPIF-2/CAPIF-2e referenced point as defined in 3GPP TS 23.222 [2], e.g. the T8 interface as defined in 3GPP TS 29.122 [14] or the NEF Northbound interface as defined in 3GPP TS 29.522 [15].

4.3.2 CAPIF core function

The CAPIF core function (CCF) supports the following capabilities over CAPIF-1/CAPIF-1e reference point as defined in 3GPP TS 23.222 [2]:

- authenticating the API invoker;
- providing the authorization information; and
- service API discovery.

The CAPIF core function supports the following capabilities over CAPIF-3 reference point as defined in 3GPP TS 23.222 [2]:

- providing the service API access policy;
- providing the authentication and authorization information of API invoker for validation;
- logging of service API invocations and
- charging of service API invocations.

The CAPIF core function supports the following capabilities over CAPIF-4 reference point as defined in 3GPP TS 23.222 [2]:

- publishing and storing the service APIs information.

The CAPIF core function supports the following capabilities over CAPIF-5 reference point as defined in 3GPP TS 23.222 [2]:

- providing the service API invocation log for auditing;
- providing monitoring information the status of service APIs and
- storing configurations of the API provider policies.

4.3.3 API exposing function

The API exposing function (AEF) is the provider of the Service APIs and is also the service communication entry point of the Service API to the API invokers using CAPIF-2/CAPIF-2e reference point as defined in 3GPP TS 23.222 [2].

The API exposing function consists of capabilities such as authenticating the API invoker, validating the authorization provided by the CAPIF core function and logging the Service API invocations at the CAPIF core function using CAPIF-3 reference point as defined in 3GPP TS 23.222 [2].

According to the distributed deployment scenarios specified in 3GPP TS 23.222 [2], it is possible that the CAPIF can be deployed by splitting the functionality of the API exposing function among multiple API exposing function entities, of which one acts as the entry point. The source API exposing function takes the role of API invoker and communicates with the destination API exposing function over CAPIF-2.

4.3.4 API publishing function

The API publishing function (APF) enables the API provider to publish the Service APIs information using CAPIF-4 reference point as defined in 3GPP TS 23.222 [2] in order to enable the discovery of Service APIs by the API invoker.