

ETSI TS 129 500 V19.6.0 (2026-03)



TECHNICAL SPECIFICATION

**5G;
5G System;
Technical Realization of Service Based Architecture;
Stage 3
(3GPP TS 29.500 version 19.6.0 Release 19)**



Reference

RTS/TSGC-0429500vj60

Keywords

5G

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.....	8
1 Scope	10
2 References	10
3 Definitions and abbreviations.....	12
3.1 Definitions	12
3.2 Abbreviations	13
3.3 Special characters, operators and delimiters.....	13
3.3.1 General.....	13
3.3.2 ABNF operators.....	13
3.3.3 URI – reserved and special characters	13
3.3.4 SBI specific usage of delimiters	13
4 Service Based Architecture Overview.....	14
4.1 NF Services	14
4.2 Service Based Interfaces	14
4.3 NF Service Framework	14
4.3.1 General.....	14
4.3.2 NF Service Advertisement URI.....	14
5 Protocols Over Service Based Interfaces	15
5.1 Protocol Stack Overview.....	15
5.2 HTTP/2 Protocol	15
5.2.1 General.....	15
5.2.2 HTTP standard headers.....	15
5.2.2.1 General	15
5.2.2.2 Mandatory to support HTTP standard headers.....	16
5.2.3 HTTP custom headers.....	19
5.2.3.1 General	19
5.2.3.2 Mandatory to support custom headers.....	20
5.2.3.2.1 General	20
5.2.3.2.2 3gpp-Sbi-Message-Priority.....	25
5.2.3.2.3 3gpp-Sbi-Callback.....	25
5.2.3.2.4 3gpp-Sbi-Target-apiRoot.....	26
5.2.3.2.5 3gpp-Sbi-Routing-Binding	26
5.2.3.2.6 3gpp-Sbi-Binding	27
5.2.3.2.7 3gpp-Sbi-Discovery.....	31
5.2.3.2.8 3gpp-Sbi-Producer-Id	32
5.2.3.2.9 3gpp-Sbi-Oci	33
5.2.3.2.10 3gpp-Sbi-Lci.....	35
5.2.3.2.11 3gpp-Sbi-Client-Credentials	37
5.2.3.2.12 3gpp-Sbi-Nrf-Uri	38
5.2.3.2.13 3gpp-Sbi-Target-Nf-Id	39
5.2.3.2.14 3gpp-Sbi-Max-Forward-Hops	39
5.2.3.2.15 3gpp-Sbi-Originating-Network-Id.....	40
5.2.3.2.16 3gpp-Sbi-Access-Scope.....	40
5.2.3.2.17 3gpp-Sbi-Access-Token	41
5.2.3.2.18 Void	41
5.2.3.2.19 3gpp-Sbi-Target-Nf-Group-Id	41
5.2.3.2.20 3gpp-Sbi-Nrf-Uri-Callback	41
5.2.3.2.21 3gpp-Sbi-NF-Peer-Info.....	42
5.2.3.2.22 3gpp-Sbi-Source-NF-Client-Credentials	43
5.2.3.2.23 3gpp-Sbi-Scp-apiRoot	43

5.2.3.3	Optional to support custom headers	44
5.2.3.3.1	General	44
5.2.3.3.2	3gpp-Sbi-Sender-Timestamp	47
5.2.3.3.3	3gpp-Sbi-Max-Rsp-Time	47
5.2.3.3.4	3gpp-Sbi-Correlation-Info	48
5.2.3.3.5	3gpp-Sbi-Alternate-Chf-Id	49
5.2.3.3.6	3gpp-Sbi-Notif-Accepted-Encoding	49
5.2.3.3.7	3gpp-Sbi-Consumer-Info	49
5.2.3.3.8	3gpp-Sbi-Response-Info	51
5.2.3.3.9	Void	52
5.2.3.3.10	3gpp-Sbi-Selection-Info	52
5.2.3.3.11	3gpp-Sbi-Interplmn-Purpose	53
5.2.3.3.12	3gpp-Sbi-Request-Info	54
5.2.3.3.13	3gpp-Sbi-Retry-Info	55
5.2.3.3.14	3gpp-Sbi-Other-Access-Scopes	56
5.2.3.3.15	3gpp-Sbi-Binding-Indication-Notify-Restricted	56
5.2.4	HTTP error handling	57
5.2.5	HTTP/2 server push	57
5.2.6	HTTP/2 connection management	57
5.2.7	HTTP status codes	58
5.2.7.1	General	58
5.2.7.2	NF as HTTP Server	59
5.2.7.3	NF as HTTP Client	64
5.2.7.4	SCP/SEPP	65
5.2.8	HTTP/2 request retries	68
5.2.9	Handling of unsupported query parameters	69
5.2.10	URL Encoding of data	70
5.2.10.1	General	70
5.2.10.2	URL Encoding of URI components	70
5.2.10.3	URL Encoding of HTTP/2 request bodies	71
5.3	Transport Protocol	71
5.4	Serialization Protocol	72
5.5	Interface Definition Language	72
6	General Functionalities in Service Based Architecture	72
6.1	Routing Mechanisms	72
6.1.1	General	72
6.1.2	Identifying a target resource	72
6.1.3	Connecting inbound	72
6.1.4	Pseudo-header setting	73
6.1.4.1	General	73
6.1.4.2	Routing within a PLMN	73
6.1.4.3	Routing across PLMN	73
6.1.4.3.1	General	73
6.1.4.3.2	Use of telescopic FQDN between NFs and SEPP within a PLMN	74
6.1.4.3.3	Use of 3gpp-Sbi-Target-apiRoot between NFs and SEPP within a PLMN	74
6.1.4.3.4	Routing between SEPPs	75
6.1.5	Host header	75
6.1.6	Message forwarding	76
6.2	Server-Initiated Communication	76
6.2.1	General	76
6.2.2	Subscription on behalf of NF Service Consumer	76
6.2.3	Notification error handling	77
6.3	Load Control	77
6.3.1	General	77
6.3.2	Load Control based on load signalled via the NRF	77
6.3.3	Load Control based on LCI Header	78
6.3.3.1	General	78
6.3.3.2	Conveyance of Load Control Information	78
6.3.3.3	Frequency of Conveyance	79
6.3.3.4	Load Control Information	79
6.3.3.4.1	General Description	79

6.3.3.4.2	Load Control Timestamp	79
6.3.3.4.3	Load Metric	80
6.3.3.4.4	Scope of LCI	80
6.3.3.4.5	S-NSSAI/DNN Relative Capacity	82
6.3.3.5	LC-H feature support	82
6.3.3.5.1	Indicating supports for the LC-H feature	82
6.3.3.5.2	Utilizing the LC-H feature support indication	82
6.4	Overload Control	82
6.4.1	General	82
6.4.2	Overload Control based on HTTP status codes	83
6.4.2.1	General	83
6.4.2.2	HTTP Status Code "503 Service Unavailable"	83
6.4.2.3	HTTP Status Code "429 Too Many Requests"	84
6.4.2.4	HTTP Status Code "307 Temporary Redirect"	84
6.4.3	Overload Control based on OCI Header	84
6.4.3.1	General	84
6.4.3.2	Conveyance of Overload Control Information	85
6.4.3.3	Frequency of Conveyance	85
6.4.3.4	Overload Control Information	85
6.4.3.4.1	General Description	85
6.4.3.4.2	Overload Control Timestamp	86
6.4.3.4.3	Overload Reduction Metric	87
6.4.3.4.4	Overload Control Period of Validity	87
6.4.3.4.5	Scope of OCI	87
6.4.3.4.6	Extension of registration timer in OCI	91
6.4.3.5	Overload Control Enforcement	91
6.4.3.5.1	Message Throttling	91
6.4.3.5.2	Loss Algorithm	92
6.4.3.6	OLC-H feature support	92
6.4.3.6.1	Indicating supports for the OLC-H feature	92
6.5	Support of Stateless NFs	92
6.5.1	General	92
6.5.2	Stateless AMFs	92
6.5.2.1	General	92
6.5.2.2	AMF as service consumer	93
6.5.2.3	AMF as service producer	94
6.5.3	Stateless NFs (for any 5GC NF type)	94
6.5.3.1	General	94
6.5.3.2	Stateless NF as service consumer	94
6.5.3.3	Stateless NF as service producer	96
6.6	Extensibility Mechanisms	97
6.6.1	General	97
6.6.2	Feature negotiation	97
6.6.3	Vendor-specific extensions	98
6.6.4	Extensibility for Query parameters	99
6.7	Security Mechanisms	100
6.7.1	General	100
6.7.2	Transport layer security protection of messages	100
6.7.3	Authorization of NF service access	100
6.7.4	Application layer security across PLMN	101
6.7.4.1	General	101
6.7.4.2	N32 Procedures	102
6.7.5	Client credentials assertion and authentication	102
6.7.5.1	General	102
6.7.5.2	Authorization of NF Service Consumers for data access via DCCF	102
6.7.5.3	Authorization of requesting ML models on behalf of another ML model consumer	103
6.8	SBI Message Priority Mechanism	103
6.8.1	General	103
6.8.2	Message level priority	104
6.8.3	Stream priority	104
6.8.4	Recommendations when defining SBI Message Priorities	104
6.8.5	HTTP/2 client behaviour	105

6.8.6	HTTP/2 server behaviour.....	105
6.8.7	HTTP/2 proxy behaviour.....	106
6.8.8	DSCP marking of messages.....	106
6.9	Discovering the supported communication options	106
6.9.1	General.....	106
6.9.2	Discoverable communication options.....	106
6.9.2.1	Content-encodings supported in HTTP requests.....	106
6.9.2.2	Content-encodings supported in HTTP responses	107
6.10	Support of Indirect Communication	107
6.10.1	General.....	107
6.10.2	Routing Mechanisms with SCP Known to the NF.....	108
6.10.2.1	General.....	108
6.10.2.2	HTTP/2 connection management.....	108
6.10.2.3	Connecting inbound	108
6.10.2.4	Pseudo-header setting.....	108
6.10.2.5	3gpp-Sbi-Target-apiRoot header setting	110
6.10.2.6	Cache key (ck) query parameter.....	110
6.10.2A	Routing Mechanism with SCP Unknown to the NF	111
6.10.2A.1	Connecting inbound	111
6.10.2A.2	HTTP/2 connection management.....	111
6.10.2A.3	Pseudo-header setting.....	111
6.10.3	NF Discovery and Selection for indirect communication with Delegated Discovery	111
6.10.3.1	General.....	111
6.10.3.2	Conveyance of NF Discovery Factors.....	111
6.10.3.3	Notifications corresponding to default notification subscriptions.....	113
6.10.3.4	Returning the Producer's NF Instance ID and NF Group ID to the NF Service Consumer	113
6.10.3.5	Returning an Alternate CHF instance ID to the NF Service Consumer	114
6.10.4	Authority and/or deployment-specific string in apiRoot of resource URI.....	115
6.10.5	NF / NF service instance selection for Indirect Communication without Delegated Discovery.....	115
6.10.5.1	General.....	115
6.10.5.2	Notifications corresponding to default notification subscriptions.....	116
6.10.6	Feature negotiation for Indirect Communication with Delegated Discovery	117
6.10.7	Notification and callback requests sent with Indirect Communication.....	118
6.10.8	Error Handling.....	118
6.10.8.1	General.....	118
6.10.8.2	Requirements for the originator of an HTTP error response.....	119
6.10.8.3	Requirements for an SCP or SEPP relaying an HTTP error response	119
6.10.9	HTTP redirection.....	120
6.10.9.1	General.....	120
6.10.10	Detection and handling of loop path when relaying message with indirect communication	121
6.10.10.1	General.....	121
6.10.10.2	Message Forwarding Depth Control	121
6.10.10.3	Loop Detection with Via header	121
6.10.11	Authorization of NF service access	122
6.10.11.1	General.....	122
6.10.11.2	Authorization for indirect communication with delegated discovery	122
6.10.11.2.1	General	122
6.10.11.2.2	Error handling when the SCP fails to obtain an access token.....	123
6.10.11.2.2A	Error handling when the SCP obtains an access token without all the scopes required for access authorization of the incoming service request.....	124
6.10.11.2.3	Error handling when SCP receives a "401 Unauthorized" response or a "403 Forbidden" response with a "WWW-Authenticate" header.....	124
6.10.11.3	Authorization for indirect communication without delegated discovery	125
6.10.12	Indirect communication with or without delegated discovery between different PLMNs with possible NF selection at target PLMN	125
6.10.12.1	General.....	125
6.10.12.2	NF Discovery and Selection.....	125
6.10.12.2.1	NF Discovery and Selection for indirect communication with delegated discovery.....	125
6.10.12.2.2	NF Discovery and Selection for indirect communication without delegated discovery.....	126
6.10.12.3	Routing mechanisms	126
6.10.12.3.1	General	126
6.10.12.3.2	Routing within the source PLMN.....	126

6.10.12.3.3	Routing between SEPPs	126
6.10.12.3.4	Routing within the target PLMN	127
6.10.12.4	Authorization of NF service access	127
6.10.12.4.1	Authorization for indirect communication with delegated discovery	127
6.10.12.4.2	Authorization for indirect communication without delegated discovery	128
6.11	Detection and handling of late arriving requests	128
6.11.1	General	128
6.11.2	Detection and handling of requests which have timed out at the HTTP client	128
6.11.2.1	General	128
6.11.2.2	Principles	128
6.12	Binding between an NF Service Consumer and an NF Service Resource	129
6.12.1	General	129
6.12.2	Binding created as part of a service response	132
6.12.3	Binding created as part of a service request	132
6.12.4	Binding for explicit or implicit subscription requests	133
6.12.5	Binding for service requests creating a callback resource	135
6.13	SBI messages correlation for network troubleshooting	135
6.13.1	General	135
6.13.2	SBI messages correlation using UE identifier	135
6.13.2.1	General	135
6.13.2.2	Principles	136
6.13.3	SBI messages correlation using NF Peer Information	136
6.13.3.1	General	136
6.13.3.2	Principles	136
6.14	Indicating the purpose of Inter-PLMN signaling	137
6.14.1	General	137
6.14.2	Inclusion of the intended purpose	137
6.14.3	Evaluating the intended purpose	137
Annex A (informative): Client-side Adaptive Throttling for Overload Control		138
Annex B (normative): 3gpp-Sbi-Callback Types		139
Annex C (informative): Internal NF Routing of HTTP Requests		140
Annex D (Normative): ABNF grammar for 3GPP SBI HTTP custom headers		141
D.1	General	141
D.2	ABNF definitions (Filename: "TS29500_CustomHeaders.abnf")	142
Annex E (informative): Change history		152
History		160

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.

In the present document, modal verbs have the following meanings:

- shall** indicates a mandatory requirement to do something
- shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

- should** indicates a recommendation to do something
- should not** indicates a recommendation not to do something
- may** indicates permission to do something
- need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

- can** indicates that something is possible
- cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

- will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document
- will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document
- might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

might not indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

is (or any other verb in the indicative mood) indicates a statement of fact

is not (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

1 Scope

The present document specifies the technical realization of the 5GC Service Based Architecture, protocols supported over the Service Based Interfaces, and the functionalities supported in the Service Based Architecture.

The service requirements for the 5G system are defined in TS 22.261 [2]. The system architecture requirements are defined in TS 23.501 [3] and the procedures and flows in TS 23.502 [4].

The design principles and documentation guidelines for 5GC SBI APIs are specified in TS 29.501 [5].

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 22.261: "Service requirements for the 5G system; Stage 1".
- [3] 3GPP TS 23.501: "System Architecture for the 5G System; Stage 2".
- [4] 3GPP TS 23.502: "Procedures for the 5G System; Stage 2".
- [5] 3GPP TS 29.501: "5G System; Principles and Guidelines for Services Definition; Stage 3".
- [6] IETF RFC 793: "Transmission Control Protocol".
- [7] IETF RFC 9113: "HTTP/2".
- [8] 3GPP TS 29.510: "5G System; Network Function Repository Services; Stage 3".
- [9] OpenAPI: "OpenAPI Specification Version 3.0.0", <https://spec.openapis.org/oas/v3.0.0>.
- [10] IETF RFC 8259: "The JavaScript Object Notation (JSON) Data Interchange Format".
- [11] IETF RFC 9110: "HTTP Semantics".
- [12] Void.
- [13] 3GPP TS 29.571: "5G System; Common Data Types for Service Based Interfaces Stage 3".
- [14] IETF RFC 3986: "Uniform Resource Identifier (URI): Generic Syntax".
- [15] 3GPP TS 23.003: "Numbering, addressing and identification".
- [16] IETF RFC 5681: "TCP Congestion Control".
- [17] 3GPP TS 33.501: "Security Architecture and Procedures for 5G System".
- [18] IANA: "SMI Network Management Private Enterprise Codes", <http://www.iana.org/assignments/enterprise-numbers>.
- [19] IETF RFC 7944: "Diameter Routing Message Priority".
- [20] IETF RFC 9111: "HTTP Caching".

- [21] Void.
- [22] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".
- [23] IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage".
- [24] Void.
- [25] IETF RFC 7516: "JSON Web Encryption (JWE)".
- [26] IETF RFC 7515: "JSON Web Signature (JWS)".
- [27] 3GPP TS 29.573: "5G System: Public Land Mobile Network (PLMN) Interconnection; Stage 3".
- [28] 3GPP TS 29.502: "5G System; Session Management Services; Stage 3".
- [29] 3GPP TS 29.503: "5G System; Unified Data Management Services; Stage 3".
- [30] Void.
- [31] 3GPP TS 29.518: "5G System; Access and Mobility Management Services; Stage 3".
- [32] 3GPP TS 29.531: "5G System; Network Slice Selection Services; Stage 3".
- [33] Void.
- [34] IETF RFC 1952: "GZIP file format specification version 4.3".
- [35] 3GPP TS 29.525: "5G System; UE Policy Control Service; Stage 3".
- [36] IETF RFC 3040: "Internet Web Replication and Caching Taxonomy".
- [37] IETF RFC 5322: "Internet Message Format".
- [38] 3GPP TS 23.527: "5G System; Restoration Procedures".
- [39] 3GPP TS 29.303: "Domain Name System Procedures; Stage 3".
- [40] 3GPP TS 29.515: "5G System; GMLC Services; Stage 3".
- [41] IETF RFC 7519: "JSON Web Token (JWT)".
- [42] 3GPP TS 32.291: "5G System; charging service; Stage 3".
- [43] IETF RFC 5234: "Augmented BNF for Syntax Specifications: ABNF".
- [44] 3GPP TS 29.526: "5G System; Network Slice-Specific Authentication and Authorization (NSSAA) Services; Stage 3".
- [45] 3GPP TS 29.562: "5G System; Home Subscriber Server (HSS) Services for interworking with the IP Multimedia Subsystem (IMS); Stage 3".
- [46] 3GPP TS 29.555: "5G System; 5G Direct Discovery Name Management Services; Stage 3".
- [47] IETF RFC 9562: "Universally Unique IDentifiers (UUIDs)".
- [48] IETF RFC 1866: "Hypertext Markup Language - 2.0".
- [49] IETF RFC 1738: "Uniform Resource Locators (URL)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1], TS 23.501 [3] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

Binding indication (consumer): Binding can be used by the NF Service Consumer to indicate suitable NF Service Consumer instance(s) for notification target instance selection, reselection and routing of subsequent notification requests associated with a specific notification subscription. Binding indication needs to be stored by the NF Service Producer. Binding indication may also be used later if the NF Service Consumer starts acting as NF Service Producer, so that service requests can be sent to this NF Service Producer. See clauses 3.1 and 6.3.1.0 in TS 23.501 [3]. See also Routing binding indication.

Binding indication (producer): Binding can be used to indicate suitable target NF Service Producer instance(s) for an NF service instance selection, reselection and routing of subsequent requests associated with a specific NF Service Producer resource (context) and NF service. Binding allows the NF Service Producer to indicate to the NF Service Consumer if a particular context should be bound to an NF service instance, NF instance, NF service set or NF set. Binding indication needs to be stored by the NF Service Consumer. See clauses 3.1 and 6.3.1.0 in TS 23.501 [3]. See also Routing binding indication.

Binding entity: Either of the following identifiers: NF Service Instance, NF Service Set, NF Instance or an NF Set. The relation between these are explained below.

Binding entity ID: An identification of a binding entity, i.e. NF Service Instance ID, NF Service Set ID, NF Instance ID or an NF set ID.

Binding level: A parameter (bl) in "3gpp-Sbi-Routing-Binding" and "3gpp-Sbi-Binding" HTTP custom headers, which indicates the binding entity towards which a preferred binding exists (i.e. either to NF Service Instance, NF Service Set, NF Instance or an NF Set). Other binding entities in these headers, which do not correspond to the binding level indicate alternative binding entities that can be reselected and that share the same resource contexts (see Table 6.3.1.0-1 in TS 23.501 [3]).

Callback URI: URI to be used by an NF Service Producer to send notification or callback requests.

consumer's SEPP (cSEPP): The SEPP residing in the PLMN where the NF service consumer is located.

Endpoint address: An address in the format of an IP address, transport and port information, or FQDN, which is used to determine the host/authority part of the target URI. This Target URI is used to access an NF service (i.e. to invoke service operations) of an NF service producer or for notifications to an NF service consumer. See clauses 3.1 and 6.3.1.0 of TS 23.501 [3].

NF Instance: An identifiable instance of the NF. An NF Instance may provide services offered by one or more NF Service instances.

NF Service Instance: An identifiable instance of the NF service.

NF Service Set: A group of interchangeable NF service instances of the same service type within an NF instance. The NF service instances in the same NF Service Set have access to the same context data.

NF Set: A group of interchangeable NF instances of the same type, supporting the same services and the same Network Slice(s). The NF instances in the same NF Set may be geographically distributed but have access to the same context data.

Notification endpoint: Notification endpoint is a destination URI of the network entity where the notification is sent. See clause 6.3.1.0 in TS 23.501 [3].

producer's SEPP (pSEPP): The SEPP residing in the PLMN where the NF service producer is located.

Routing binding indication: Information included in a request or notification and that can be used by the SCP for discovery and associated selection to of a suitable target. See clauses 3.1, 6.3.1.0 and 7.1.2 in TS 23.501 [3]. Routing binding indication has similar syntax as a binding indication, but it has different purpose. Routing binding indication

provides the receiver (i.e. SCP) with information enabling to route an HTTP request to an HTTP server that can serve the request. Routing binding indication is not stored by the receiver.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and TS 23.501 [3] and the following apply. An abbreviation defined in the present document or in TS 23.501 [3] takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1]. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TS 23.501 [3].

GZIP	GNU ZIP
LC-H	Load Control based on LCI Header
LCI	Load Control Information
MCX	Mission Critical Service
MPS	Multimedia Priority Service
OCI	Overload Control Information
OLC-H	Overload Control based on OCI Header
SCP	Service Communication Proxy
SEPP	Security and Edge Protection Proxy
SMP	SBI Message Priority

3.3 Special characters, operators and delimiters

3.3.1 General

A number of characters have special meaning and are used as delimiters in this document and also in other stage 3 SBI specifications. Below clauses specify the usage of a selected set of the special characters. Full set of these special characters are specified in the respective IETF specifications.

3.3.2 ABNF operators

/	Operator. The forward slash character separates alternatives. See clause 3.2 in IETF RFC 5234 [43].
#	Operator. The number sign character allows for compact definition of comma-separated lists, similar to the "*" operator. See clause 2.1 in IETF RFC 9110 [11].
=	Special character. The equal sign character separates an ABNF rule name from the rule elements. See clause 2.2 in IETF RFC 5234 [43].
[]	Operator. The square bracket characters enclose an optional element sequence. See clause 3.8 in IETF RFC 5234 [43].
<>	Special characters. The angle bracket characters typically enclose an ABNF rule element (they are optional). See clause 2.1 in IETF RFC 5234 [43].
*	Operator. The star character precedes an element and indicates the elements repetition. See clause 3.6 in IETF RFC 5234 [43].
;	Operator. Semicolon character indicates the start of a comment that continues to the end of line. See clause 3.9 in IETF RFC 5234 [43].

NOTE: The same characters, like "/", "#", etc. lead to different processing in ABNF and URI grammars. For instance, in URI syntax, ";" character separates parameter and its value, while in ABNF ";" starts a comment. Besides, unlike URI syntax, neither "?", nor ":" operators are specified for ABNF.

3.3.3 URI – reserved and special characters

Special characters that are used as delimiters in URI syntax have somewhat different purpose from the same characters when used by ABNF syntax. See clause 3.3.3 in TS 29.501 [5].

3.3.4 SBI specific usage of delimiters

See clause 3.3.4 in TS 29.501 [5].

4 Service Based Architecture Overview

4.1 NF Services

TS 23.501 [3] defines the 5G System Architecture as a Service Based Architecture, i.e. a system architecture in which the system functionality is achieved by a set of NFs providing services to other authorized NFs to access their services.

Control Plane (CP) Network Functions in the 5G System architecture shall be based on the service based architecture.

A NF service is one type of capability exposed by a NF (NF Service Producer) to other authorized NF (NF Service Consumer) through a service based interface. A NF service may support one or more NF service operation(s). See clause 7 of TS 23.501 [3].

Network Functions may offer different functionalities and thus different NF services. Each of the NF services offered by a Network Function shall be self-contained, acted upon and managed independently from other NF services offered by the same Network Function (e.g. for scaling, healing).

4.2 Service Based Interfaces

A service based interface represents how the set of services is provided or exposed by a given NF. This is the interface where the NF service operations are invoked.

The service based Control Plane interfaces within the 5G Core Network are specified in TS 23.501 [3].

4.3 NF Service Framework

4.3.1 General

The Service Based Architecture shall support the NF Service Framework that enable the use of NF services as specified in clause 7.1 of TS 23.501 [3].

The NF Service Framework includes the following mechanisms:

- NF service registration and de-registration: to make the NRF aware of the available NF instances and supported services (see clause 7.1.5 of TS 23.501 [3]);
- NF service discovery: to enable a NF Service Consumer to discover NF Service Producer instance(s) which provide the expected NF service(s) (see clause 7.1.3 of TS 23.501 [3]);
- NF service authorization: to ensure the NF Service Consumer is authorized to access the NF service provided by the NF Service Producer (see clause 7.1.4 of TS 23.501 [3]);
- Inter service communication: NF Service Consumers and NF Service Producers may communicate directly or indirectly via a Service Communication Proxy (SCP). Whether a NF uses Direct Communication or Indirect Communication via an SCP is based on configuration of the NF.

The stage 3 procedures for NF service registration and de-registration, NF service discovery and NF service authorization are defined in TS 29.510 [8].

4.3.2 NF Service Advertisement URI

When invoking a service operation of a NF Service Producer that use HTTP methods with a message body (i.e. PUT, POST and PATCH), the NF Service Consumer may provide NF Service Advertisement URI(s) in the service operation request, based on operator policy, if it expects that the NF Service Producer may subsequently consume NF service(s) which the NF Service Consumer can provide (as a NF Service Producer).

When receiving NF Service Advertisement URI(s) in a service operation request, the NF Service Producer may store and use the Service Advertisement URI(s) to discover NF services produced by the NF Service Consumer in subsequent procedures, based on operator policy.

The NF Service Advertisement URI identifies the nfInstance resource(s) in the NRF which are registered by NF Service Producer(s).

An example of NF Service Advertisement URI could be represented as:

"{apiRoot}/nnrf-disc/nf-instances?nfInstanceId={nfInstanceId}".

NOTE: The NF Service Advertisement URI can be used e.g. when different NRFs are deployed in the PLMN.

When applicable, the NF Service Advertisement URI(s) shall be carried in HTTP message body.

5 Protocols Over Service Based Interfaces

5.1 Protocol Stack Overview

The protocol stack for the service-based interfaces is shown on Figure 5.1-1.

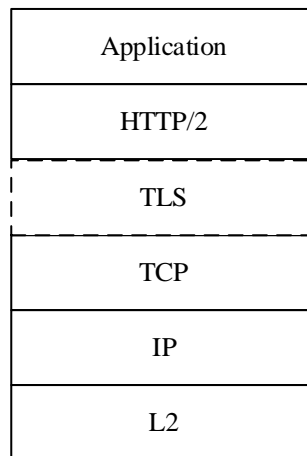


Figure 5.1-1: SBI Protocol Stack

The service-based interfaces use HTTP/2 protocol (see clause 5.2) with JSON (see clause 5.4) as the application layer serialization protocol. For the security protection at the transport layer, all 3GPP NFs shall support TLS and TLS shall be used within a PLMN if network security is not provided by other means, as specified in TS 33.501 [17].

5.2 HTTP/2 Protocol

5.2.1 General

HTTP/2 as described in IETF RFC 9113 [7] shall be used in Service based interface.

5.2.2 HTTP standard headers

5.2.2.1 General

This clause lists the HTTP standard headers that shall be supported on SBI, other HTTP standard headers defined in IETF RFCs may be supported by NF.

5.2.2.2 Mandatory to support HTTP standard headers

The HTTP request standard headers and the HTTP response standard headers that shall be supported on SBI are defined in Table 5.2.2.2-1 and in Table 5.2.2.2-2 respectively. Mandatory to support HTTP standard headers does not mean all the HTTP requests and responses carry the identified request and response headers respectively. It only means it is mandatory to support the processing of the identified headers in request and response message.

Table 5.2.2.2-1: Mandatory to support HTTP request standard headers

Name	Reference	Description
Accept	IETF RFC 9110 [11]	This header is used to specify response media types that are acceptable.
Accept-Encoding	IETF RFC 9110 [11]	This header may be used to indicate what response content-encodings (e.g. gzip) are acceptable in the response.
Content-Length	IETF RFC 9110 [11]	This header is used to provide the anticipated size, as a decimal number of octets, for a potential content.
Content-Type	IETF RFC 9110 [11]	This header is used to indicate the media type of the associated representation.
Content-Encoding	IETF RFC 9110 [11]	This header may be used in some requests to indicate the content encodings (e.g. gzip) applied to the resource representation beyond those inherent in the media type.
User-Agent	IETF RFC 9110 [11]	<p>This header shall be mainly used to identify the NF type of the HTTP/2 client. This header should be included in every HTTP/2 request sent over any SBI; This header shall be included in every HTTP/2 request sent using indirect communication when target NF (re-)selection is to be performed at SCP.</p> <p>For Indirect communications, the User-Agent header in a request that is:</p> <ul style="list-style-type: none"> - forwarded by the SCP (with or without delegated discovery) shall identify the NF type of the original NF that issued the request (i.e. the SCP shall forward the header received in the incoming request); - originated by the SCP towards the NRF (e.g. NF Discovery or Access Token Request) shall identify the SCP. <p>The pattern of the content should start with the value of NF type (e.g. "UDM", see NOTE 1) or "SCP" (for a request originated by an SCP) and followed by a "-" and any other specific information if needed afterwards.</p>
Cache-Control	IETF RFC 9111 [20]	This header may be used in some HTTP/2 requests to provide the HTTP cache-control directives that the client is willing to accept from the server.
If-Modified-Since	IETF RFC 9110 [11]	This header may be used in a conditional GET request, for server revalidation. This is used in conjunction with the Last-Modified server response header, to fetch content only if the content has been modified from the cached version.
If-None-Match	IETF RFC 9110 [11]	This header may be used in a conditional GET request. This is used in conjunction with the ETag server response header, to fetch content only if the tag value of the resource on the server differs from the tag value in the If-None-Match header.
If-Match	IETF RFC 9110 [11]	This header may be used in a conditional POST or PUT or DELETE or PATCH request. This is used in conjunction with the ETag server response header, to update / delete content only if the tag value of the resource on the server matches the tag value in the If-Match header.
Via	IETF RFC 9110 [11]	This header shall be inserted by HTTP proxies and it shall be inserted by an SCP and SEPP when relaying an HTTP request. When inserted by an SCP or SEPP, the header field value should be formatted as defined for the Via header in Table 5.2.2.2-2.
Authorization	IETF RFC 9110 [11]	This header shall be used if OAuth 2.0 based access authorization with "Client Credentials" grant type is used as specified in clause 13.4.1 of TS 33.501 [17], clause 7 of IETF RFC 6749 [22] and IETF RFC 6750 [23].
NOTE 1: The value of NF type in the User-Agent header shall comply with the enumeration value (case insensitive) of Table 6.1.6.3.3-1 in TS 29.510 [8].		

Table 5.2.2.2-2: Mandatory to support HTTP response standard headers

Name	Reference	Description
Content-Length	IETF RFC 9110 [11]	This header may be used to provide the anticipated size, as a decimal number of octets, for a potential content.
Content-Type	IETF RFC 9110 [11]	This header shall be used to indicate the media type of the associated representation.
Location	IETF RFC 9110 [11]	This header may be used in some responses to refer to a specific resource in relation to the response.
Retry-After	IETF RFC 9110 [11]	This header may be used in some responses to indicate how long the user agent ought to wait before making a follow-up request.
Content-Encoding	IETF RFC 9110 [11]	This header may be used in some responses to indicate to the HTTP/2 client the content encodings (e.g. gzip) applied to the resource representation beyond those inherent in the media type.
Cache-Control	IETF RFC 9111 [20]	This header may be used in some responses (e.g. NRF responses to queries) to provide HTTP response cache control directives. The cache directives "no-cache", "no-store", "max-age" and "must-revalidate" values shall be supported.
Age	IETF RFC 9111 [20]	This header may be inserted by HTTP proxies when returning a cached response. The "Age" header field conveys the sender's estimate of the amount of time since the response was generated or successfully validated at the origin server. The presence of an Age header field implies that the response was not generated or validated by the origin server for this request.
Last-Modified	IETF RFC 9110 [11]	This header may be sent to allow for conditional GET with the If-Modified-Since header.
ETag	IETF RFC 9110 [11]	This header may be sent to allow for conditional GET with the If-None-Match header or a conditional POST / PUT / PATCH / DELETE with the If-Match header.
Via	IETF RFC 9110 [11]	This header shall be inserted by HTTP proxies. This header shall be inserted by an SCP or SEPP when relaying an HTTP error response (see clause 6.10.8). It may be inserted when relaying other HTTP responses. When inserted by an SCP or SEPP, the received-protocol portion of the header field value should be set to "HTTP/2.0" or "2.0" and the received-by portion of the header field value should be formatted as follows: - "SCP-<SCP FQDN>" for an SCP - "SEPP-<SEPP FQDN>" for a SEPP See examples in clause 6.10.8.
Allow	IETF RFC 9110 [11]	This header field shall be used to indicate methods supported by the target resource.
WWW-Authenticate	IETF RFC 9110 [11]	This header field shall be included when an NF service producer rejects a request with a "401 Unauthorized" status code (e.g. when a request is sent without an OAuth 2.0 access token or with an invalid OAuth 2.0 access token).
Accept-Encoding	IETF RFC 9110 [11]	See clause 6.9 for the use of this header.
Server	IETF RFC 9110 [11]	This header should be inserted by the originator of an HTTP error response (see clause 6.10.8). It may be inserted otherwise. When inserted by an NF, an SCP or a SEPP, the pattern of the header should be formatted as follows: - "SCP-<SCP FQDN>" for an SCP - "SEPP-<SEPP FQDN>" for a SEPP - "<NFtype>-<NF Instance ID>" for an NF
NOTE:	The value of NF type in the Server header shall comply with the enumeration value (case insensitive) of Table 6.1.6.3.3-1 in TS 29.510 [8].	

5.2.3 HTTP custom headers

5.2.3.1 General

The list of custom HTTP headers applicable to 3GPP Service Based NFs are specified below.

The ABNF definition of these custom headers is expressed in the following clauses using common syntax components defined in IETF RFC 9110 [11], clauses 5.6.2 - 5.6.5, such as <token> and <tchar>. As indicated there, the following characters (expressed by their UNICODE name) shall not be used as part of a <token>, or as a <tchar>:

- QUOTATION MARK (U+0022): "
- LEFT PARENTHESIS (U+0028): (
- RIGHT PARENTHESIS (U+0029):)
- COMMA (U+002C): ,
- SOLIDUS (U+002F): /
- COLON (U+003A): :
- SEMICOLON (U+003B): ;
- LESS-THAN SIGN (U+003C): <
- EQUALS SIGN (U+003D): =
- GREATER-THAN SIGN (U+003E): >
- QUESTION MARK (U+003F): ?
- COMMERCIAL AT (U+0040): @
- LEFT SQUARE BRACKET (U+005B): [
- REVERSE SOLIDUS (U+005C): \
- RIGHT SQUARE BRACKET (U+005D):]
- LEFT CURLY BRACKET (U+007B): {
- RIGHT CURLY BRACKET (U+007D): }

If a 3GPP custom HTTP header, whose ABNF syntax definition uses the <token> or <tchar> components, needs to include a value containing a character outside of the character set allowed for <token> or <tchar>, such character shall be encoded using percent-encoding, as follows:

pct-encoded = "%" HEXDIG HEXDIG

The HEXDIG ABNF production rule, originally defined in IETF RFC 5234 [43], shall be considered as if the uppercase hexadecimal digits 'A' through 'F' are equivalent to the lowercase digits 'a' through 'f', respectively.

The literal "%" character shall also be encoded as above (i.e. %25).

Percent encoding shall not be used for characters that are in the <token> or <tchar> allowed character set.

EXAMPLE: 3GPP Custom Header "3gpp-Sbi-Oci" (see clause 5.2.3.2.9) can include an optional parameter "snsai". If this parameter takes the value:

```
{ "sst": 1, "sd": "A08923" }
```

it will be formatted, when included in this custom header, as:

```
S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D
```

The ABNF definitions of 3GPP custom HTTP headers follow a uniform structure in the grammar production rules; this structure is based on HTTP versions prior to HTTP/2, where HTTP headers were sent as character strings between client and server. The structure is as follows:

```
<header-ABNF-rule-name> = <header-name> ":" OWS <header-value> OWS
```

with the following restrictions:

- <header-ABNF-rule-name>: it shall adhere to the naming restrictions defined in ABNF (IETF RFC 5234 [43], section 2); e.g., rule names shall not start with a digit, and they shall be considered as case-insensitive strings. As part of 3GPP naming conventions, SBI custom header ABNF rule names should start with prefix "Sbi-" and should end with string "-Header".
- <header-name>: as part of the specific 3GPP naming conventions, header names shall be prefixed with the string "3gpp-Sbi-".
- <header-value>: its content shall not start or end with white spaces.

It should be noted that, in HTTP/2 (IETF RFC 9113 [7]), headers are sent in binary format on the HEADERS frame type, with <header-name> and <header-value> sent on specific protocol elements, i.e., they don't contain a colon (":") character, plus optional white spaces, as separators between <header-name> and <header-value>.

5.2.3.2 Mandatory to support custom headers

5.2.3.2.1 General

The 3GPP NF Services shall support the HTTP custom headers specified in Table 5.2.3.2.1-1 below. A description of each custom header and the normative requirements on when to include them are also provided in Table 5.2.3.2-1.

Table 5.2.3.2.1-1: Mandatory HTTP custom headers

Name	Reference	Description
3gpp-Sbi-Message-Priority	Clause 5.2.3.2.2	This header is used to specify the HTTP/2 message priority for 3GPP service based interfaces. This header shall be included in HTTP/2 messages when a priority for the message needs to be conveyed (e.g. HTTP/2 messages related to Multimedia Priority Sessions).
3gpp-Sbi-Callback	Clause 5.2.3.2.3	<p>This header is used to indicate if a HTTP/2 message is a callback (e.g. notification).</p> <p>This header shall be included in HTTP POST messages for callbacks towards NF service consumer(s) in another PLMN via the SEPP (See TS 29.573 [27]).</p> <p>This header shall also be included in HTTP POST messages for callbacks in indirect communication (See clause 6.10.7).</p> <p>This header should also be included in the HTTP POST message of any event notification request for direct communications.</p> <p>If the header is included in received HTTP request, the SEPP or SCP shall include this header in the HTTP request forwarded to next hop. (NOTE 1)</p>
3gpp-Sbi-Target-apiRoot	Clause 5.2.3.2.4	<p>This header is used by an HTTP client to indicate the apiRoot of the target URI when communicating indirectly with the HTTP server via an SCP. This header is also used by SCP to indicate the apiRoot of the target URI, if a new HTTP server is selected or reselected and there is no Location header included in the response.</p> <p>This header may also be used by an HTTP client towards its local SEPP to indicate the apiRoot of the target URI towards HTTP server in another PLMN.</p> <p>This header may also be used between SEPPs to indicate the apiRoot of the target URI towards HTTP server in another PLMN, when TLS security with the 3gpp-Sbi-Target-apiRoot header is used between the SEPPs.</p>
3gpp-Sbi-Scp-apiRoot	Clause 5.2.3.2.23	This header is used by an SCP in a source PLMN to indicate the apiRoot of an SCP in a target PLMN when using indirect communication with or without delegated discovery between different PLMNs with possible NF selection at target PLMN (see clause 6.10.12). The authority of the 3gpp-Sbi-Scp-apiRoot header may correspond to an authority of a specific SCP, or to an authority that can resolve into several SCPs for scalability, load sharing and resiliency purposes.
3gpp-Sbi-Routing-Binding	Clause 5.2.3.2.5	This header is used in a service request to signal binding information to direct the service request to an HTTP server which has the targeted NF Service Resource context (see clause 6.12).
3gpp-Sbi-Binding	Clause 5.2.3.2.6	This header is used to signal binding information related to an NF Service Resource to a future consumer (HTTP client) of that resource (see clause 6.12).
3gpp-Sbi-Discovery-*	Clause 5.2.3.2.7	Headers beginning with the prefix 3gpp-Sbi-Discovery- are used in indirect communication mode to allow the discovery and selection of a suitable NF service producer (e.g. in case of service requests) or NF service consumer (e.g. in case of notifications or callbacks) by the SCP, as specified in clause 5.2.3.2.7, clause 6.5.3 and clause 6.10. Such headers may be included in any SBI message and include information allowing an SCP to find a suitable NF service producer or NF service consumer, as per the discovery and selection parameters provided respectively by the NF service consumer or the NF service producer.

3gpp-Sbi-Producer-Id	Clause 5.2.3.2.8	<p>This header is used in a service response from the SCP to the NF Service Consumer, when using indirect communication, to identify the NF service producer. See clause 6.10.3.4.</p> <p>This header may also be used in a resource creation response from the NF Service Producer to the NF consumer (or SCP), when the resource is created in a different NF Service Producer (e.g. UE Context Create with AMF relocation during inter-PLMN N2 handover procedure).</p>
3gpp-Sbi-Oci	Clause 5.2.3.2.9	<p>This header may be used by an overloaded NF Service Producer in a service response, or in a notification request to signal Overload Control Information (OCI) to the NF Service Consumer.</p> <p>This header may also be used by an overloaded NF Service Consumer in a notification response or in a service request to signal Overload Control Information (OCI) to the NF Service Producer.</p>
3gpp-Sbi-Lci	Clause 5.2.3.2.10	This header may be used by a NF Service Producer to send Load Control Information (LCI) to the NF Service Consumer.
3gpp-Sbi-Client-Credentials	Clause 5.2.3.2.11	This header may be used by an NF Service Consumer to send Client Credentials Assertion to the NRF or to the NF Service Producer. See clause 6.7.5.
3gpp-Sbi-Source-NF-Client-Credentials	Clause 5.2.3.2.22	<p>This header may be used by an NF Service Consumer (e.g., DCCF, NWDAF containing MTLF) to send Client Credentials Assertion of the source NF (e.g. NWDAF, NWDAF containing AnLF) to the NF Service Producer (e.g. AMF, SMF, NWDAF containing MTLF).</p> <p>The purpose is one of the followings:</p> <ul style="list-style-type: none"> - to enable the authorization of NF service consumers for data access via DCCF as specified in Annex X of TS 33.501 [17]; - to enable the authorization of the ML model consumer when the NF service consumer requests ML models on behalf of the ML model consumer as specified in Annex X.10 of TS 33.501 [17]. <p>See clause 6.7.5.</p>
3gpp-Sbi-Nrf-Uri	Clause 5.2.3.2.12	<p>This header may be used to indicate the NRF API URIs to be used for a given service request, e.g. in indirect communication with delegated discovery as a result of an NSSF query. It may also indicate whether OAuth2 based authorization is required for accessing the NRF services.</p> <p>This header may also be used to indicate the NRF API URI to be used for a given notification request, e.g. if the NF service producer has received NRF API URI from the NF service consumer and the NF producer delegates NF consumer reselection to the SCP in indirect communication,</p>
3gpp-Sbi-Target-Nf-Id	Clause 5.2.3.2.13	This header is used in a 307 Temporary Redirect or 308 Permanent Redirect response, to identify the target NF (service) instance towards which the request is redirected. See clause 6.10.9.1.
3gpp-Sbi-Max-Forward-Hops	Clause 5.2.3.2.14	<p>This header may be used to indicate the maximum number of allowed hops with specified node type to relay the request message to the target HTTP server.</p> <p>If node type is "scp", its value indicates the maximum number of allowed SCP hops to relay the request message to the target NF as HTTP server when indirect communication is used.</p>

3gpp-Sbi-Originating-Network-Id	Clause 5.2.3.2.15	<p>This header shall be inserted by an NF service consumer or an NF service producer originating an HTTP request message towards a different PLMN or SNPN.</p> <p>It should be inserted by the sending SCP in SBI HTTP request messages towards the SEPP, only if the header is not present in the SBI HTTP request message and the SCP can determine which PLMN-ID value should be included in the header.</p> <p>It shall be inserted by the sending SEPP or the receiving SEPP in SBI HTTP request messages towards the target PLMN or SNPN, only if the header is not present in the SBI HTTP request message and the sending SEPP or the receiving SEPP (respectively) can determine the PLMN ID or SNPN ID of the source PLMN or SNPN.</p> <p>If the SEPP cannot uniquely determine the PLMN-ID or SNPN-ID, it is a configuration/deployment aspect to determine which PLMN-ID or SNPN-ID value should be included in the header by these entities. In such case, the message should either be rejected, or the SEPP shall indicate to the peer that the header is derived based on configuration.</p> <p>It shall indicate the PLMN-ID or the SNPN-ID of the source PLMN or SNPN of the HTTP request message (i.e., the PLMN ID or the SNPN ID of the NF Service Consumer or NF Service Producer).</p> <p>See clause 5.9.3.2 of TS 33.501 [17] for the handling of this header by the sending NF, the sending SCP, the sending SEPP and the receiving SEPP. (NOTE 2)</p>
3gpp-Sbi-Access-Scope	Clause 5.2.3.2.16	<p>This header is used in a service request for Indirect Communication to indicate the access scope of the service request for NF service access authorization. See clauses 6.7.3 and 6.10.11.</p>
3gpp-Sbi-Access-Token	Clause 5.2.3.2.17	<p>This header is used in a service response forwarded by the SCP to an NF service consumer to provide an access token for possible re-use in subsequent service requests. See clause 6.10.1.</p>
3gpp-Sbi-Target-Nf-Group-Id	Clause 5.2.3.2.19	<p>This header is used in a service response from the SCP to the NF Service Consumer, when using indirect communication with delegated discovery, to indicate the NF Group ID of the NF service producer selected by the SCP. See clause 6.10.3.4.</p>
3gpp-Sbi-Nrf-Uri-Callback	Clause 5.2.3.2.20	<p>This header may be included in service request (e.g. subscription creation request) from the NF service consumer to the NF service producer, to indicate:</p> <ul style="list-style-type: none"> - the NRF NFDISCOVERY API URI to be used to discover an alternative NF service consumer for callback, e.g. during NF service consumer reselection for callback when the original NF service consumer is no longer available; and - if available, the NRF NFMANAGEMENT API URI to be used to subscribe to NF status change of the NF service consumer. <p>For indirect communication, if the NF service producer delegates NF service consumer reselection to the SCP, the NF service producer should include 3gpp-Sbi-Nrf-Uri header with received NRF API URI (which was received in the 3gpp-Sbi-Nrf-Uri-Callback from the NF service consumer) in the notification requests to the NF service consumer.</p>

3gpp-Sbi-NF-Peer-Info	Clause 5.2.3.2.21	<p>This header is used in HTTP requests and responses to indicate the sender and receiver of the message.</p> <p>The HTTP client and server should include this header in every HTTP request and response messages.</p> <p>HTTP intermediaries (e.g. SCP) should forward this header, when relaying HTTP messages to next hop, and may update the destination in the header if the receiver NF of the message is (re)selected. The parameters defined for the source and destination of SCPs or SEPPs (as defined in clause 5.2.3.2.21) may also need to be updated according to the source and destination of the HTTP message.</p>
<p>NOTE 1: The callback URI for event subscription may receive event notifications from different NF producers, e.g. UDM may subscribe to AMF/SMF on behalf of NEF with directly reporting mode for certain UDM events in the subscription, which should be inspected with corresponding OpenAPI schema where the notification is defined. For both direct and indirect communications, to include this header in all event notification requests can help NF consumer to identify the type of event notification and select corresponding schema to perform OpenAPI inspection.</p> <p>NOTE 2: The value of this header shall be verified by the sending SEPP and receiving SEPP (see clause 5.9.3.2 of TS 33.501 [17]) . If the PLMN ID or SNPN ID in this header does not match with any of the PLMN IDs that the sending SEPP represents, the sending SEPP or the receiving SEPP shall reject the received signalling message using the error cause "ORIGINATING_NETWORK_ID_MISMATCH".</p>		

5.2.3.2.2 3gpp-Sbi-Message-Priority

The header contains the HTTP/2 message priority value from 0 to 31, as defined in clause 6.8.4.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Message-Priority-Header = "3gpp-Sbi-Message-Priority:" OWS ( DIGIT / %x31-32 DIGIT / "3" %x30-31 ) OWS
```

A message with 3gpp-Sbi-Message-Priority "0" has the highest priority.

EXAMPLE: 3gpp-Sbi-Message-Priority: 10

5.2.3.2.3 3gpp-Sbi-Callback

The header contains the type of notification. The value for the notification type is a string used identifying a particular type of callback (e.g. a notification, typically the name of the notify service operation).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Callback-Header = "3gpp-Sbi-Callback:" OWS cbtype *1( ";" OWS "apiversion=" majorversion ) OWS
cbtype = 1*cbchar
cbchar = "-" / "_" / DIGIT / ALPHA
majorversion = *DIGIT
```

EXAMPLE 1: 3gpp-Sbi-Callback: Nnrf_NFManagement_NFStatusNotify

EXAMPLE 2: 3gpp-Sbi-Callback: Nudm_SDM_Notification; apiversion=2

The list of valid values for the cbtype is specified in Annex B.

The apiversion parameter should be present if the major version is higher than 1.

NOTE: The apiversion parameter can be used by the SEPP to identify the protection and modification policies applicable to the API version of a notification or callback request, or by the SCP to select a notification endpoint of a NF Service Consumer that supports the API version when forwarding a notification request issued for a default notification subscription.

5.2.3.2.4 3gpp-Sbi-Target-apiRoot

The header contains the apiRoot of the target URI (see clause 4.4 of TS 29.501 [5]) in a request sent to an SCP when using Indirect Communication. This header contains the apiRoot of the selected or changed target URI in a response sent to an HTTP client, when the SCP selected or reselected a new HTTP server to route the request (or when the SCP sends the request to an alternative HTTP Server upon receiving a 3xx redirect message, see clause 6.10.9.1) and no Location HTTP header is included in the HTTP response. It may also be used in a request sent to a SEPP and in a request between SEPPs (see clause 6.1.4.3.2).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Target-apiRoot-Header = "3gpp-Sbi-Target-apiRoot" ":" OWS sbi-scheme "://" sbi-authority
[ prefix ] OWS
```

```
sbi-scheme = "http" / "https"
```

```
sbi-authority = host [ ":" port ]
```

```
port = *DIGIT
```

```
prefix = path-absolute ; path-absolute production rule from IETF RFC 3986, clause 3.3
```

EXAMPLE: 3gpp-Sbi-Target-apiRoot: https://example.com/a/b/c

5.2.3.2.5 3gpp-Sbi-Routing-Binding

This header contains a Routing Binding Indication used to direct a service request to an HTTP server which has the targeted NF service resource context (see clause 6.12).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Routing-Binding-Header = "3gpp-Sbi-Routing-Binding:" OWS "bl=" blvalue 1*( ";" OWS parameter )
[ ";" OWS callback-uri-prefix ] OWS
blvalue = "nf-instance" / "nf-set" / "nfservice-instance" / "nfservice-set"
```

```
parameter = parametername "=" token
```

```
parametername = "nfinst" / "nfset" / "nfservinst" / "nfserviceset" / "servname" / "backupamfinst" /
"backuppnf"
```

The following parameters are defined:

- bl (binding level): the value of this parameter (blvalue) indicates a preferred binding to a binding entity, i.e. either to an NF Instance, an NF set, an NF Service Instance or an NF Service Set. If the binding level is set to an NF Service Instance (nfservice-instance), then either NF Service Set ID or NF Instance ID shall also be present to unambiguously identify the NF Service Instance.
- nfinst (NF instance): indicates an NF Instance ID, as defined in clause 5.2.2.2.2 in TS 29.510 [8]. This parameter shall be present if the binding level is set to "nf-instance", or if the binding level is set to "nfservice-instance" and the nfserviceset parameter is not included.
- nfset (NF set): indicates an NF Set ID, as defined in clause 28.12 in TS 23.003 [15]. This parameter shall be present if the binding level is set to "nf-set". It may be present otherwise (see clause 6.12.1).
- nfservinst (NF service instance): indicates an NF Service Instance ID. This parameter shall be present if the binding level is set to "nfservice-instance".
- nfserviceset (NF service set): indicates an NF Service Set ID as defined in clause 28.13 in TS 23.003 [15]. This parameter shall be present if the binding level is set to "nfservice-set". It shall also be present if the binding level is set to "nfservice-instance" and the NF service instance indicated by the nfservinst parameter is part of an NF service set (see clause 6.12.1).
- servname (service name): indicates the name of a service, as defined in TS 29.510 [8], or a custom service that handles a notification or a callback request. It may be present in a Routing Binding Indication in a notification or a callback request.

- backupamfinst (backup NF Instance): indicates the NF Instance ID (as defined in clause 5.2.2.2.2 in TS 29.510 [8]) of the backup NF, i.e. the backup AMF as specified in TS 23.501 [3]. The backupamfinst may be present only when the binding level is nf-instance or nf-service-instance or nf-service-set. When backupamfinst is present, no binding entity corresponding to NF set shall be present. When the binding level is nf-set, backupamfinst shall not be present.
- for the definition and encoding of the backupnf see clause 5.2.3.2.6.
- for the definition and encoding of the callback-uri-prefix see clause 5.2.3.2.6. The ABNF is defined in clause 5.2.3.3.7.

See clause 5.6.2 of IETF RFC 9110 [11] for the "token" type definition. A token's value is a string, which contains a binding entity ID or a service name.

EXAMPLE 1: Binding to SMF set 1 of MCC 345 and MNC 012:

```
3gpp-Sbi-Routing-Binding: bl=nf-set; nfset=set1.smfset.5gc.mnc012.mcc345
```

EXAMPLE 2: Binding to an SMF instance within SMF set of Example 1:

```
3gpp-Sbi-Routing-Binding: bl=nf-instance; nfirst=54804518-4191-46b3-955c-ac631f953ed8;
nfset=set1.smfset.5gc.mnc012.mcc345
```

EXAMPLE 3: Binding to a SMF Service Set "xyz" within an SMF instance within SMF set of Example 1:

```
3gpp-Sbi-Routing-Binding: bl=nf-service-set; nfserviceset=setxyz.snnsmf-
pdusession.nfi54804518-4191-46b3-955c-ac631f953ed8.5gc.mnc012.mcc345;
nfset=set1.smfset.5gc.mnc012.mcc345
```

EXAMPLE 4: Binding to AMF set 1 within AMF region 48 (hexadecimal):

```
3gpp-Sbi-Routing-Binding: bl=nf-set; nfset=set1-region48.amfset.5gc.mnc012.mcc345
```

EXAMPLE 5: Binding for a subscription (i.e. notification requests) to AMF set 1 within AMF region 48 (hexadecimal) and Namf_Communication service:

```
3gpp-Sbi-Routing-Binding: bl=nf-set; nfset=set1-region48.amfset.5gc.mnc012.mcc345;
servname=namf-comm
```

EXAMPLE 6: Binding to the AMF Instance in addition with backup AMF, where the nfirst carries the Identity of the AMF to which the resource is bound and whose backup AMF is indicated in backupamfinst:

```
3gpp-Sbi-Routing-Binding: bl=nf-instance; nfirst=54804518-4191-46b3-955c-ac631f953ed7;
backupamfinst=54804518-4191-46b3-955c-ac631f953ed8
```

5.2.3.2.6 3gpp-Sbi-Binding

This header contains a comma-delimited list of Binding Indications from an HTTP server for storage and subsequent use by an HTTP client (see clause 6.12).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Binding-Header = "3gpp-Sbi-Binding:" OWS 1#( "bl=" blvalue 1*( ";" OWS parameter ) [ ";" OWS
recoverytime ] [ ";" OWS notif-receiver ] [ ";" OWS "group=" groupvalue ] [ 1*( ";" OWS
groupparameter ) ] [ ";" OWS "no-redundancy=" no-red-value ] [ ";" OWS "callback-uri-prefix" ]) OWS
```

```
blvalue = "nf-instance" / "nf-set" / "nf-service-instance" / "nf-service-set"
```

```
parameter = parametername "=" token
```

```
parametername = "nfirst" / "nfset" / "nfservinst" / "nfserviceset" / "servname" / "scope" /
"backupamfinst" / "backupnf"
```

```
recoverytime = "recoverytime=" OWS DQUOTE date-time DQUOTE
```

```
notif-receiver = "nr=" URI ; URI production rule from IETF RFC 3986, Appendix A
```

```
groupvalue = "true" / "false"
```

```
groupparameter = groupparametername "=" token
```

```

groupparametername = "oldgroupid" / "groupid" / "uribase" / "oldnfinst" / "oldservset" /
"oldservinst" / "guami"

no-red-value = "true"

```

The following parameters are defined:

- scope: indicates the applicability of a Binding Indication in a service request other than a notification request, or in a notification or callback response. This may take one of the following values:
 - "other-service": the binding information applies to other service(s) that the NF Service Consumer may later on provide as an NF Service Producer (see clause 6.12.3);
 - "subscription-events": the binding information applies to subscription change event notifications (see clause 6.12.4);
 - "callback": the binding information applies to notification or callback requests (see clauses 6.12.4 and 6.12.5).

The absence of this parameter in a Binding Indication in a service request other than a notification request, or in a notification or callback response, shall be interpreted as "callback".

Two scope parameters may be present in a Binding Indication if the binding information applies to notification/callback requests and to other services.

- servname (service name): indicates the name of a service, as defined in TS 29.510 [8], or a custom service, i.e.:
 - the name of the service that handles a notification or a callback request, when present in a Binding Indication for a subscription or a callback, i.e. with a scope parameter absent or set to "callback"; or
 - the name of the other service(s) for which the binding applies, when present in a Binding Indication in a service request for the other services the NF Service Consumer can provide later on as an NF Service Producer, i.e. with the scope parameter set to "other-service". More than one servname parameter may be present to represent multiple such services. The absence of this parameter in a Binding Indication with the scope parameter set to "other-service" shall be interpreted as binding information that applies to all the services that the NF Service Consumer may provide later as an NF Service Producer.
- recoverytime: indicates the recovery timestamp of the entity corresponding to the highest resiliency level supported for the resource, that is, the higher level binding entity indicated in the Binding Indication. See Table 6.3.1.0-1 of TS 23.501 [3] and clause 6.1 of TS 23.527 [38]. The date-time type is specified in IETF RFC 5322 [37] and clause 5.6.7 of IETF RFC 9110 [11].
- nr: indicates the URI of the notification endpoint when this binding information is applicable; it applies to callback requests (see clause 6.12.4); if the notification URI does not contain a correlationID in the path (i.e. it is a common notification URI for multiple subscriptions), the correlationID shall be added as a fragment component of the URI (i.e. following the "#" character) at the end of the URI.
- for the definition and encoding of the blvalue, nfinst, backupamfinst, nfset, nfservinst and nfserviceset see clause 5.2.3.2.5.
- backupnf: indicates the backup NF service instance identifier and/or the backup NF identifier as defined in clause 5.2.2.2.2 or in TS 29.510 [8], which shall be used when preferred binding entity is not reachable if supported.
- group: it is a boolean indicating if the binding indication is for a group of resource/session contexts.
- groupid (group id): indicates the group identifier allocated by the NF (service) instance, one or more resource/session contexts are sharing the same groupid. The groupid is optional and it may be allocated when the resource/session context is created and then be updated afterwards. The groupid is global unique and it may be encoded using the same mechanism for the NfInstanceId as specified in TS 29.571 [13].
- oldgroupid (old group id): indicates the group identifier allocated by the NF (service) instance previously and to be replaced by the groupid, hence it shall only be present when to update a Binding Indication for multiple

contexts. When the if the oldgroupid is present, the groupid shall also be present to indicate the new groupid allocated.

- uribase: identify the ariroot and path segments part in the resource URI or notification/callback URI which is common to multiple contexts. This parameter may only be present when to update a Binding Indication for multiple contexts and when the "group" is set to "true". When included, it indicates that all resources or notification contexts with this uribase will use the updated Binding Indication subsequently. More than one uribase may be present.
- oldnfnst: indicates the NF Instance ID of the NF instance where the group of resource/session contexts are currently served (i.e. the Binding Indication allocated previously for the group of resource/session contexts includes the information of the NF instance), as defined in clause 5.2.2.2.2 in TS 29.510 [8]. When included, it indicates that all the resource/session contexts served by this NF instance will use the updated Binding Indication subsequently.
- oldsrvset: indicates the NF Service Set ID of the NF Service Set where the group of resource/session contexts are currently served (i.e. the Binding Indication allocated previously for the group of resource/session contexts includes the information of the NF Service Set), as defined in clause 5.2.2.2.2 in TS 29.510 [8]. When included, it indicates that all the resource/session contexts served by this NF Service Set will use the updated Binding Indication subsequently.
- oldsrvinst: indicates the NF Service Instance ID of the NF service instance where the group of resource/session contexts are currently served (i.e. the Binding Indication allocated previously for the group of resource/session contexts includes the information of the NF service instance), as defined in clause 5.2.2.2.2 in TS 29.510 [8]. When included, it indicates that all the resource/session contexts served by this NF service instance will use the updated Binding Indication subsequently.
- guami (GUAMI): indicates the GUAMI of the AMF currently serving UE contexts, as defined in clause 5.3.4.1 of TS 29.571 [13]. When included, it indicates that all the UE contexts associated with the GUMAI will use the updated Binding Indication subsequently.
- no-redundancy: it is a boolean set to true indicating that the resource is exclusively bound to the NF service instance identified in the binding indication. It may be present in a binding with any scope, i.e. "other-service", "subscription-events" or "callback", or with no scope parameter. When this parameter is present, the blvalue shall be set to "nfservice-instance", the nfsrvinst parameter shall be present and either the nfserviceset parameter or the nfnst parameter shall be present. The nfserviceset or nfnst parameter included in the binding indication shall only be used to identify the NF service instance and shall not be considered as a binding entity for reselection. The no-redundancy parameter shall only be signaled if the receiver of this information is known to support this parameter (see clause 6.12.1). Subsequently, when sending further requests targeting a resource with no-redundancy, the HTTP client shall not include any routing binding indication in the request message (to prevent the SCP from performing any reselection).
- callback-uri-prefix: The NF service consumer may include this parameter when providing a Callback URI when the Callback URI contains a prefix. When present, the "callback-uri-prefix" shall be a path-absolute as specified IETF RFC 3986 [14] (i.e. the first path segment(s) after the authority) which is part of the Callback URI provided by a NF service consumer in the corresponding service request message sent to a NF service producer. The authority and "callback-uri-prefix" in the Callback URI shall uniquely identify a consumer service instance. This parameter is relevant when the "scope" parameter is either "subscription-events" or "callback". See clauses 6.12.4 and 6.12.5 for the usage of this parameter. The ABNF is defined in clause 5.2.3.3.7.

EXAMPLES 1 to 5: Same as EXAMPLES 1 to 5 defined in clause 5.2.3.2.5, with the header name "3gpp-Sbi-Binding" instead of "3gpp-Sbi-Routing-Binding".

EXAMPLE 6: Subscription request from one NF on behalf of another NF, with 2 binding indications:

```
3gpp-Sbi-Binding: bl=nf-set; nfset=set1.udmset.5gc.mnc012.mcc345; servname=nudm-ee;scope=subscription-events
3gpp-Sbi-Binding: bl=nf-set; nfset=set1.nefset.5gc.mnc012.mcc345; servname=nnef-event-exposure
```

- EXAMPLE 7: Service request with 2 binding indications, for callback requests and for other services the NF Service Consumer may provide later as an NF Service Producer:
- ```
3gpp-Sbi-Binding: bl=nf-instance; nfirst=54804518-4191-46b3-955c-ac631f953ed8;
nfset=set1.smfset.5gc.mnc012.mcc345; servname=nsmf-pdusession
3gpp-Sbi-Binding: bl=nf-instance; nfirst=54804518-4191-46b3-955c-ac631f953ed8;
nfset=set1.smfset.5gc.mnc012.mcc345; scope=other-service; servname=nsmf-event-exposure
```
- EXAMPLE 8: Service request with one binding indication applying to notification/callback requests and to any other services the NF Service Consumer may provide later as an NF Service Producer:
- ```
3gpp-Sbi-Binding: bl=nf-set; nfset=set1-region48.amfset.5gc.mnc012.mcc345; scope=callback;
scope=other-service
```
- EXAMPLE 9: Service request with one binding indication applying to notification/callback requests together with a recovery time stamp associated with the NF Set indicated in the binding indication and with the binding level set to "nfset":
- ```
3gpp-Sbi-Binding: bl=nf-set; nfset=set1-region48.amfset.5gc.mnc012.mcc345; scope=callback;
recoverytime="Tue, 04 Feb 2020 08:49:37 GMT"
```
- EXAMPLE 10: Service response with one binding indication applying to the session context with a recovery time stamp associated with the NF Set indicated in "nfset" in the binding indication and with the binding level set to "ninstance":
- ```
3gpp-Sbi-Binding: bl=nf-instance; nfirst=54804518-4191-46b3-955c-ac631f953ed8;
nfset=set1.smfset.5gc.mnc012.mcc345; recoverytime="Tue, 04 Feb 2020 08:49:37 GMT"
```
- EXAMPLE 11: Service response with one binding indication applying to the session context with a recovery time stamp associated with the NF Instance included the binding indication and with the binding level set to nserviceinstance:
- ```
3gpp-Sbi-Binding: bl=nservice-instance; nservinst=xyz; nfirst=54804518-4191-46b3-955c-
ac631f953ed8; recoverytime="Tue, 04 Feb 2020 08:49:37 GMT"
```
- EXAMPLE 12: Service response with one binding indication applying to the resource context pertaining to a group identified by "54804518-4191-46b3-955c-ac631f953ed1" together with a backup nf:
- ```
3gpp-Sbi-Binding: bl=nf-instance; nfirst=54804518-4191-46b3-955c-ac631f953ed0;
nfset=set1.smfset.5gc.mnc012.mcc345; backupnf=54804519-4191-46b3-955c-ac631f953ed2;
groupid=54804518-4191-46b3-955c-ac631f953ed1
```
- EXAMPLE 13: A notification request message with one binding indication applying to the resource contexts with the oldgroup identifier "54804518-4191-46b3-955c-ac631f953ed1", where the preferred binding entity is changed to "nfirst=54804519-4191-46b3-955c-ac631f953ed0" together with a new group identifier "54804519-4191-46b3-955c-ac631f953ed3" allocated.
- ```
3gpp-Sbi-Binding: bl=nf-instance; nfirst=54804519-4191-46b3-955c-ac631f953ed0;
nfset=set1.smfset.5gc.mnc012.mcc345; group=true; oldgroupid=54804518-4191-46b3-955c-
ac631f953ed1; groupid=54804519-4191-46b3-955c-ac631f953ed3
```
- EXAMPLE 14: A notification request message with one binding indication applying to the resource contexts identified by an uribase, where the preferred binding entity is changed to "nfirst=54804519-4191-46b3-955c-ac631f953ed0":
- ```
3gpp-Sbi-Binding: bl=nf-instance; nfirst=54804519-4191-46b3-955c-ac631f953ed0;
nfset=set1.smfset.5gc.mnc012.mcc345; group=true;
uribase=http%3A%2F%2F10.10.10.10%2Fstringxyz
```

EXAMPLE 15: A notification request message with one binding indication applying to the resource contexts served by the NF instance identified by "64804518-4191-46b3-955c-ac631f953ed8" where the preferred binding entity is changed to "nfinst=74804519-4191-46b3-955c-ac631f953ed0".

```
3gpp-Sbi-Binding: bl=nf-instance; nfinst=74804519-4191-46b3-955c-ac631f953ed0;
nfset=set1.smfset.5gc.mnc012.mcc345; group=true; oldnfinst=64804518-4191-46b3-955c-
ac631f953ed8
```

EXAMPLE 16: Service request message with an updated binding indication applying to the UE contexts for GUAMI " <mnc(012)><mcc(345)><AmfId("abcd12")> where the backupamfinst is changed.

```
3gpp-Sbi-Binding: bl=nf-instance; nfinst=54804518-4191-46b3-955c-ac631f953ed7;
backupamfinst=54804520-4191-46b3-955c-ac631f953ed8; scope=other-service; group=true;
guami=%7B%22plmnId%22%3A%7B%22mnc%22%3A%22012%22%2C%22mcc%22%3A%22
345%22%7D%2C%22amfId%22%3A%22abcd12%22%7D
```

EXAMPLE 17: Service response with a binding indication applying to the resource context which is exclusively bound to a specific NF service instance.

```
3gpp-Sbi-Binding: bl=nf-service-instance; nfservinst=xyz; nfinst=54804518-4191-46b3-955c-
ac631f953ed8; no-redundancy=true
```

EXAMPLE 18: Subscription request with a Callback URI containing a prefix "/abc":

```
3gpp-Sbi-Binding: bl=nf-set; nfset=set1.nefset.5gc.mnc012.mcc345; servname=nnef-event-
exposure; callback-uri-prefix="/abc"
```

NOTE: Examples 6 and 7 are formatted as two distinct headers (which improves the readability), but they can also be formatted as a single header with two Binding Indication values separated by a comma.

5.2.3.2.7 3gpp-Sbi-Discovery

These headers shall be used to convey NF service discovery factors to the SCP in indirect communication models. They contain discovery parameters to be conveyed by an NF service consumer or an NF service producer to the SCP or by an SCP to the next hop SCP and they shall be used by the SCP to select or reselect a suitable NF service producer instance to create or update a (existing) resource context, or a suitable NF service consumer instance towards which to send a notification or a callback request, e.g. by performing the NF service discovery procedure with the NRF on behalf of the NF consumer in case of indirect communication with delegated discovery model.

The name of each NF service discovery factors header shall be constructed by concatenating the string "3gpp-Sbi-Discovery-" with the name of the conveyed discovery parameter, i.e. "3gpp-Sbi-Discovery-<discovery parameter>".

The discovery headers shall be used to include any of the discovery query parameters listed in TS 29.510 [8], Table 6.2.3.2.3.1-1. The value of each NF service discovery header shall be encoded in the same way as the corresponding discovery parameter (i.e. with the same data type, cardinality and format). Thus, the values of these headers may be validated with the same data model as that of the corresponding discovery parameters. The discovery headers shall comply with the condition of presence of the discovery parameters defined in Table 6.2.3.2.3.1-1 of TS 29.510 [8], e.g. discovery headers shall be included for discovery parameters defined as mandatory in this table. Table 5.2.3.2.7-1 lists examples of NF service discovery headers.

Table 5.2.3.2.7-1: NF service discovery factors headers examples

Header name	Discovery parameter	Header value	Data Model
3gpp-Sbi-Discovery-target-nf-type	target-nf-type (TS 29.510 [8], Table 6.2.3.2.3.1-1)	AMF	NFType: Enumeration as of TS 29.510 [8], Table 6.1.6.3.3- 1.
3gpp-Sbi-Discovery-snssais	snssais (TS 29.510 [8], Table 6.2.3.2.3.1-1)	{{"sst": 1, "sd": "A08923"}, {"sst": 1, "sd": "0023F1"}}	array(Snssai), where Snssai is a structured data type as of TS 29.571 [13], Table 5.4.4.2- 1
3gpp-Sbi-Discovery-target-nf- instance-id	target-nf-instance-id (TS 29.510 [8], Table 6.2.3.2.3.1-1)	e553cf50-f32b-4638- 8a7e-0d416cc60952	NfInstanceId: simple data type as of TS 29.571 [13], Table 5.3.2-1
3gpp-Sbi-Discovery-pdu-session- types	pdu-session-types (TS 29.510 [8], Table 6.2.3.2.3.1-1)	IPV6,IPV4V6	array(PduSessionType), where PduSessionType is a simple data type as of TS 29.571 [13], Table 5.4.3.3- 1.

The 3gpp-Sbi-Discovery-* header is not documented in OpenAPI specification files. It shall comply with the following OpenAPI definition:

- for parameters defined with a "content:" block and specifying the "application/json" media type" in clause A.3 of TS 29.510 [8]:

```
parameters:
  - name: 3gpp-Sbi-Discovery-<Discovery parameter name>:
    in: header
    description: Discovery parameter defined in Table 6.2.3.2.3.1-1 of 3GPP TS 29.510
    content:
      application/json:
        schema:
          type: <Data type defined in Table 6.2.3.2.3.1-1 of 3GPP TS 29.510>
```

- for other parameters:

```
parameters:
  - name: 3gpp-Sbi-Discovery-<Discovery parameter name>:
    in: header
    description: Discovery parameter defined in Table 6.2.3.2.3.1-1 of 3GPP TS 29.510
    schema:
      type: <Data type defined in Table 6.2.3.2.3.1-1 of 3GPP TS 29.510>
```

3gpp-Sbi-Discovery-* headers corresponding to query parameters defined with the "style" keyword set to "form" and the "explode" keyword set to false in clause A.3 of TS 29.510 [8] shall be formatted using the style "simple", i.e. as comma-separated values.

NOTE 1: The style "form" is not available for header parameters, thus the corresponding 3gpp-Sbi-Discovery-* headers use the default style "simple" defined for header parameters in the OpenAPI Specification [9].

NOTE 2: The percent-encoding described in clause 5.2.3.1 is not applicable to the 3gpp-Sbi-Discovery-* headers since their syntax is not defined using ABNF; such encoding is only applicable to headers whose ABNF syntax is defined in terms of <token> and <tchar> common components.

5.2.3.2.8 3gpp-Sbi-Producer-Id

This header contains the NF Service Producer Instance ID (see clause 6.10.3.4).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Producer-Id-Header = "3gpp-Sbi-Producer-Id:" OWS "nfinst=" nfinst [ OWS ";" OWS "nfservinst="
nfservinst ] [ OWS ";" OWS "nfset=" nfset ] [ OWS ";" OWS "nfserviceset=" nfserviceset ] OWS
```

```

nfinst = 8HEXDIG "-" 4HEXDIG "-" 4HEXDIG "-" 4HEXDIG "-" 12HEXDIG
nfservinst = token
nfset = token
nfserviceset = token

```

The following parameters are defined:

- nfinst (NF instance): indicates a NF Instance ID, as defined in TS 29.510 [8];
- nfservinst (NF service instance): indicates a NF Service Instance ID, as defined in TS 29.510 [8];
- nfset (NF set): indicates an NF Set ID, as defined in clause 28.12 in TS 23.003 [15];
- nfserviceset (NF service set): indicates an NF Service Set ID as defined in clause 28.13 in TS 23.003 [15].

EXAMPLE 1: 3gpp-Sbi-Producer-Id: nfinst=54804518-4191-46b3-955c-ac631f953ed8

EXAMPLE 2: 3gpp-Sbi-Producer-Id: nfinst=54804518-4191-46b3-955c-ac631f953ed8; nfservinst=xyz

EXAMPLE 3: 3gpp-Sbi-Producer-Id: nfinst=54804518-4191-46b3-955c-ac631f953ed8; nfservinst=xyz;
nfset=set1.smfset.5gc.mnc012.mcc345

5.2.3.2.9 3gpp-Sbi-Oci

The header contains a comma-delimited list of Overload Control Information (OCI). See clause 6.4.3.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```

Sbi-Oci-Header = "3gpp-Sbi-Oci:" OWS 1#( timestamp ";" RWS validityPeriod ";" RWS olcMetric ";" RWS
olcScope [ ";" RWS olcExtRegTimer ] ) OWS

timestamp = "Timestamp:" RWS DQUOTE date-time DQUOTE

validityPeriod = "Period-of-Validity:" RWS 1*DIGIT "s"

olcMetric = "Overload-Reduction-Metric:" RWS ( DIGIT / %x31-39 DIGIT / "100" ) "%"

olcScope = nfProducerScope / nfConsumerScope / scpScope / seppScope

nfProducerScope = ( ( "NF-Instance:" RWS nfinst )
/ ( "NF-Set:" RWS nfset )
/ ( "NF-Service-Instance:" RWS nfservinst [ ";" RWS "NF-Inst:" RWS nfinst ] )
/ ( "NF-Service-Set:" RWS nfserviceset )
) [ ";" RWS sNssaiList ";" RWS dnnList ]

nfConsumerScope = ( "NF-Instance:" RWS nfinst [ ";" RWS "Service-Name:" RWS servname ] )
/ ( "NF-Set:" RWS nfset [ ";" RWS "Service-Name:" RWS servname ] )
/ ( "NF-Service-Instance:" RWS nfservinst [ ";" RWS "NF-Inst:" RWS nfinst ] )
/ ( "NF-Service-Set:" RWS nfserviceset )
/ ( "Callback-Uri:" RWS DQUOTE URI DQUOTE *( RWS "&" RWS DQUOTE URI DQUOTE ) )

scpScope = "SCP-FQDN:" RWS fqdn

seppScope = "SEPP-FQDN:" RWS fqdn

dnnList = "DNN:" RWS 1*tchar *( RWS "&" RWS 1*tchar )

sNssaiList = "S-NSSAI:" RWS snssai *( RWS "&" RWS snssai )

snssai = 1*tchar

olcExtRegTimer = "Extend-Registration-Timer:" RWS ert-value

ert-value = "true" / "false"

```

NOTE 1: Implementations complying with earlier releases of the specification can require the presence of a leading white space after "3gpp-Sbi-Oci:".

"Timestamp" (Mandatory parameter): The date-time type is specified in IETF RFC 5322 [37] and clause 5.6.7 of IETF RFC 9110 [11]. It indicates the timestamp at which the overload control information was generated.

"Period-of-Validity" (Mandatory parameter): Period of validity is a timer that is measured in seconds. Once the timer expires, the OCI becomes invalid.

"Overload-Reduction-Metric" (Mandatory parameter): Overload-Reduction-Metric up to 3 digits long decimal string and the value range shall be from 0 to 100.

The Overload Control Scope is a mandatory header component, and it shall contain one of the parameters: "NF-Instance", "NF-Set", "NF-Service-Instance" or "NF-Service-Set" (for NF consumers or NF producers), "Callback-URI" (for NF consumers), "SCP-FQDN" (for SCP), or "SEPP-FQDN" (for SEPP).

The Extend-Registration-Timer is an optional header component applicable to the IMS network; if present, it shall contain an indication ("true" or "false") requiring the receiving entity (e.g., a "serving" Network Function handling the access control to the network, such as I-CSCF, or S-CSCF) to extend the registration timers for UEs trying to get access to the IMS network. Other NFs may ignore this header parameter.

See clause 6.4.3.4.5. The `nfinst`, `nfset`, `nfservinst`, `nfserviceset` and `servname` parameters are defined in clause 5.2.3.2.8. `fqdn` shall encode an FQDN. URI is defined in clause 3 of IETF RFC 3986 [14].

When signaling overload control information of an NF service instance, the "NF-Inst" parameter shall be present to identify the NF service instance unambiguously. If the "NF-Inst" parameter is absent, the receiving NF should assume the last known NF instance ID of NF service producer or consumer, if available.

NOTE 2: Implementations complying with earlier versions of the specification can signal overload control information of an NF service instance without including the NF-Inst parameter.

"DNN" (Optional parameter): Used for S-NSSAI/DNN based overload control by SMF, see clause 6.4.3.4.5.2.2, that refers to one or more specific DNN(s). DNN format is defined in TS 23.003 [15].

"S-NSSAI" (Optional parameter): Used for S-NSSAI/DNN based overload control by SMF, see clause 6.4.3.4.5.2.2, that refers to one or more specific S-NSSAI(s).

S-NSSAI format is defined in clause 5.4.4.2 of TS 29.571 [13]. It shall be encoded as the object format (i.e. not converted to the string pattern defined in clause 5.4.4.2 of TS 29.571 [13]).

EXAMPLE 1: Overload Control Information for an NF Instance:

```
3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 75s; Overload-Reduction-Metric: 50%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8
```

EXAMPLE 2: Overload Control Information for an NF Service Set:

```
3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 120s; Overload-Reduction-Metric: 50%; NF-Service-Set: setxyz.snnsmf-pdusession.nfi54804518-4191-46b3-955c-ac631f953ed8.5gc.mnc012.mcc345
```

EXAMPLE 3: Overload Control Information for an SMF instance related to a particular DNN of an S-NSSAI:

```
3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 600s; Overload-Reduction-Metric: 50%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8; S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN: internet.mnc012.mcc345.gprs
```

NOTE 3: The S-NSSAI parameter corresponds to the JSON encoding: {"sst": 1, "sd": "A08923"} (see clause 5.2.3.1)

EXAMPLE 4: Overload Control Information for an SMF instance related to a particular DNN shared by two S-NSSAIs:

```
3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 240s; Overload-Reduction-Metric: 50%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8; S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D & %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08924%22%7D; DNN: internet.mnc012.mcc345.gprs
```

NOTE 4: The S-NSSAI parameter corresponds to the JSON encoding: {"sst": 1, "sd": "A08923"} & {"sst": 1, "sd": "A08924"} (see clause 5.2.3.1)

EXAMPLE 5: Overload Control Information sent by a NF service consumer with a scope set to a Callback-Uri:

3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 120s;
Overload-Reduction-Metric: 25%; Callback-Uri: "https://pcf12.operator.com/serviceY"

EXAMPLE 6: Overload Control Information sent by a NF service consumer with a scope set to a specific NF Instance and service:

3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 120s;
Overload-Reduction-Metric: 25%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8;
Service-Name: nsmf-pdusession

EXAMPLE 7: Overload Control Information sent by an SCP:

3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 120s;
Overload-Reduction-Metric: 25%; SCP-FQDN: scp1.example.com

EXAMPLE 8: Example with two OCI values, one for an SMF Instance and another one for a specific DNN of an S-NSSAI for the same SMF Instance:

3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 75s; Overload-Reduction-Metric: 50%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8
3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 600s;
Overload-Reduction-Metric: 40%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8; S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN: internet.mnc012.mcc345.gprs

NOTE 5: The S-NSSAI parameter corresponds to the JSON encoding: {"sst": 1, "sd": "A08923"} (see clause 5.2.3.1)

EXAMPLE 9: Overload Control Information sent by an SEPP:

3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 120s;
Overload-Reduction-Metric: 25%; SEPP-FQDN: sepp1.example.com

NOTE 6: Example 8 is formatted as two distinct headers (which improves the readability), but it can also be formatted as a single header with two OCI values separated by a comma.

EXAMPLE 10: Overload Control Information for an NF Service Instance:

3gpp-Sbi-Oci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Period-of-Validity: 75s; Overload-Reduction-Metric: 50%; NF-Service-Instance: xyz; NF-Inst: 54804518-4191-46b3-955c-ac631f953ed8

5.2.3.2.10 3gpp-Sbi-Lci

The header contains a comma-delimited list (see IETF RFC 9110 [11]) of Load Control Information (LCI). See clause 6.3.3.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Lci-Header = "3gpp-Sbi-Lci:" OWS 1#( timestamp ";" RWS lcMetric ";" RWS lcScope ) OWS
timestamp = "Timestamp:" RWS DQUOTE date-time DQUOTE
lcMetric = "Load-Metric:" RWS ( DIGIT / %x31-39 DIGIT / "100" ) "%"
lcScope = lcNfProducerScope / scpScope / seppScope
lcNfProducerScope = ( ( "NF-Instance:" RWS nfinst )
/ ( "NF-Set:" RWS nfset )
/ ( "NF-Service-Instance:" RWS nfservinst [ ";" RWS "NF-Inst:" RWS nfinst ] )
/ ( "NF-Service-Set:" RWS nfserviceset )
[ ";" RWS sNssaiList ";" RWS dnnList ";" RWS relativeCapacity ]
```

```
dnnList = "DNN:" RWS 1*tchar *( RWS "&" RWS 1*tchar )
sNssaiList = "S-NSSAI:" RWS snssai *( RWS "&" RWS snssai )
snssai = 1*tchar
relativeCapacity = "Relative-Capacity:" RWS ( 1*2DIGIT / "100" ) "%"
```

NOTE 1: Implementations complying with earlier releases of the specification can require the presence of a leading white space after "3gpp-Sbi-Lci".

"Timestamp" (Mandatory parameter): The date-time type is specified in IETF RFC 5322 [37] and clause 5.6.7 of IETF RFC 9110 [11]. It indicates the timestamp associated with the load control information.

"Load-Metric" (Mandatory parameter): Load-Metric is up to 3 digits long decimal string and the value range shall be from 0 to 100.

The Load Control Scope is a mandatory header component, and it shall contain one of the parameters: "NF-Instance", "NF-Set", "NF-Service-Instance" or "NF-Service-Set" (for NF producers), "SCP-FQDN" (for SCP), or "SEPP-FQDN" (for SEPP).

See clause 6.3.3.4.4. The *nfinst*, *nfset*, *nfservinst* and *nfserviceset* parameters are defined in clause 5.2.3.2.5. *fqdn* shall encode an FQDN.

When signaling load control information of an NF service instance, the *NF-Inst* parameter shall be present to identify the NF service instance unambiguously. If the *NF-Inst* parameter is absent, the receiving NF should assume the last known NF instance ID of the NF service producer, if available.

NOTE 2: Implementations complying with earlier versions of the specification can signal load control information of an NF service instance without including the *NF-Inst* parameter.

"DNN" (Optional parameter): Used for S-NSSAI/DNN based load control by SMF, see clause 6.3.3.4.4.2.2, that refers to one or more specific DNN(s). DNN format is defined in TS 23.003 [15].

"S-NSSAI" (Optional parameter): Used for S-NSSAI/DNN based load control by SMF, see clause 6.3.3.4.4.2.2, that refers to one or more specific S-NSSAI(s).

S-NSSAI format is defined in clause 5.4.4.2 of TS 29.571 [13]. It shall be encoded as the object format (i.e. not converted to the string pattern defined in clause 5.4.4.2 of TS 29.571 [13]).

"Relative-Capacity" (Optional parameter): Used for S-NSSAI/DNN based load control by SMF, see clause 6.3.3.4.5. Up to 3 digits long decimal string with value range from 0 to 100. The value applies to all combinations of S-NSSAIs and DNNs indicated in the LCI.

EXAMPLE 1: Load Control Information for an NF Instance:

```
3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 25%; NF-Instance:
54804518-4191-46b3-955c-ac631f953ed8
```

EXAMPLE 2: Load Control Information for an NF Service Set:

```
3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 25%; NF-Service-
Set: setxyz.snsmf-pdusession.nfi54804518-4191-46b3-955c-ac631f953ed8.5gc.mnc012.mcc345
```

EXAMPLE 3: Load Control Information for an SMF instance related to a particular DNN of an S-NSSAI (SST=1, SD="A08923"):

```
3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 25%; NF-Instance:
54804518-4191-46b3-955c-ac631f953ed8; S-
NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN:
internet.mnc012.mcc345.gprs; Relative-Capacity: 20%
```

EXAMPLE 4: Load Control Information for an SMF instance related to a particular S-NSSAI (SST=1, SD="A08923"):

3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 25%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8; S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN: internet.mnc012.mcc345.gprs; Relative-Capacity: 20%

NOTE 3: The S-Nssai parameter corresponds to the JSON encoding: {"sst": 1, "sd": "A08923"} (see clause 5.2.3.1)

EXAMPLE 5: Load Control Information for SCP:

3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 25%; SCP-FQDN: scp1.example.com

EXAMPLE 6: Example with two LCI values, for different DNNs of a same S-NSSAI (SST=1, SD="A08923"):

3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 40%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8; S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN: internet.mnc012.mcc345.gprs; Relative-Capacity: 30%

3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 70%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8; S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN: ciot.mnc012.mcc345.gprs; Relative-Capacity: 20%

NOTE: The S-Nssai parameter corresponds to the JSON encoding: {"sst": 1, "sd": "A08923"} (see clause 5.2.3.1)

EXAMPLE 7: Load Control Information for SEPP:

3gpp-Sbi-Lci: Timestamp: "Tue, 04 Apr 2021 08:36:42 GMT"; Load-Metric: 25%; SEPP-FQDN: sepp1.example.com

NOTE 4: Example 6 is formatted as two distinct headers (which improves the readability), but it can also be formatted as a single header with two LCI values separated by a comma.

EXAMPLE 8: Load Control Information for an NF Service Instance:

3gpp-Sbi-Lci: Timestamp: "Tue, 04 Feb 2020 08:49:37 GMT"; Load-Metric: 25%; NF-Service-Instance: xyz; NF-Inst: 54804518-4191-46b3-955c-ac631f953ed8

5.2.3.2.11 3gpp-Sbi-Client-Credentials

The header contains client credentials assertion (see clause 13.3.8.1 of TS 33.501 [17]).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Client-Credentials-Header = "3gpp-Sbi-Client-Credentials:" OWS jwt OWS
jwt = 1*b64urlchar "." 1*b64urlchar "." 1*b64urlchar
b64urlchar = ALPHA / DIGIT / "-" / "_"
```

The client credentials assertion shall be a JSON Web Token (JWT) as specified in IETF RFC 7519 [41], digitally signed using JWS as specified in IETF RFC 7515 [26] and in clause 13.3.8.1 of TS 33.501 [15]. It shall include:

- the claims defined in Table 5.2.3.2.11-1 encoded as a JSON object; and
- one of the following JOSE headers:
 - the X.509 URL (x5u) header (see clause 4.1.5 of IETF RFC 7515 [26]) referring to a resource for the X.509 public key certificate or certificate chain used for signing the client authentication assertion, or
 - the X.509 Certificate Chain (x5c) header (see clause 4.1.5 of IETF RFC 7515 [26]) including the X.509 public key certificate or certificate chain used for signing the client authentication assertion.

The digitally signed client credentials assertion shall be converted to the JWS Compact Serialization encoding as a string as specified in clause 7.1 of IETF RFC 7515 [26].

Table 5.2.3.2.11 -1: Definition of type ClientCredentialsAssertion

Attribute name	Data type	P	Cardinality	Description
sub	NfInstanceId	M	1	This IE shall contain the NF instance ID of the NF service consumer, corresponding to the standard "Subject" claim described in IETF RFC 7519 [41], clause 4.1.2.
iat	integer	M	1	This IE shall indicate the time at which the JWT was issued, corresponding to the standard "Issued At" claim described in IETF RFC 7519 [41], clause 4.1.6. This claim may be used to determine the age of the JWT.
exp	integer	M	1	This IE shall contain the expiration time after which the client credentials assertion is considered to be expired, corresponding to the standard "Expiration Time" claim described in IETF RFC 7519 [41], clause 4.1.4.
aud	array(NFType)	M	1..N	This IE shall contain the NF type of the NF service producer and/or "NRF", for which the claim is applicable, corresponding to the standard "Audience" claim described in IETF RFC 7519 [41], clause 4.1.3.

The JSON object containing the client credentials assertion shall comply with the following OpenAPI definition:

```
ClientCredentialsAssertion:
  description: The data structure for the client credentials assertion
  type: object
  required:
    - sub
    - iat
    - exp
    - aud
  properties:
    sub:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/NfInstanceId'
    iat:
      type: integer
    exp:
      type: integer
    aud:
      type: array
      items:
        $ref: 'TS29510_Nnrf_NFManagement.yaml#/components/schemas/NFType'
      minItems: 1
```

5.2.3.2.12 3gpp-Sbi-Nrf-Uri

The header contains a list of NRF API URIs. See clauses 6.10.3.2 and 6.10.5.1.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Nrf-Uri-Header = "3gpp-Sbi-Nrf-Uri:" OWS nrfUriParam *( OWS ";" OWS nrfUriParam ) OWS
nrfUriParam = nrfUriParamName ":" RWS ( nrfUriParamValue1 / nrfUriParamValue2 )
nrfUriParamName = "nnrf-disc" / "nnrf-nfm" / "nnrf-oauth2" / "oauth2-requested-services" / token
nrfUriParamValue1 = DQUOTE URI DQUOTE
nrfUriParamValue2 = ( nrfServiceName *( RWS "&" RWS nrfServiceName ) )
```

```
nrfServiceName = "nnrf-disc" / "nnrf-nfm"
```

NOTE: token is defined for future extensibility.

- for the nnrf-disc, nnrf-nfm and nnrf-oauth2 parameters:

URI shall comply with the URI definition in IETF RFC 3986 [14].

- for the oauth2-requested-services parameter:

nrfServiceName shall encode an NRF API name, e.g. "nnrf-disc" or "nnrf-nfm".

When present, the oauth2-requested-services parameter shall indicate the list of NRF services for which OAuth2 based authorization is required for accessing the respective NRF services.

If OAuth2 based authorization is required for accessing the respective NRF services, the nnrf-oauth2 parameter shall be present and shall be used to request access token for NRF services.

The absence of the oauth2-requested-services parameter means that no indication is provided about the potential usage of OAuth2 for authorization.

EXAMPLE 1: Header with NRF NF Discovery, NF Management and Access Token API URIs, without indication on whether OAuth2-based authorization is required to access the NRF services:

```
3gpp-Sbi-Nrf-Uri: nnrf-disc: "https://nrf1.operator.com/nnrf-disc/v1"; nnrf-nfm:
"https://nrf1.operator.com/nnrf-nfm/v1"; nnrf-oauth2: "https://nrf1.operator.com/oauth2"
```

EXAMPLE 2: Header with NRF NF Discovery, NF Management and Access Token API URIs, indication on whether OAuth2-based authorization is required to access the NRF services:

```
3gpp-Sbi-Nrf-Uri: nnrf-disc: "https://nrf1.operator.com/nnrf-disc/v1"; nnrf-nfm:
"https://nrf1.operator.com/nnrf-nfm/v1"; nnrf-oauth2: "https://nrf1.operator.com/oauth2"; oauth2-
requested-services: nnrf-disc & nnrf-nfm
```

5.2.3.2.13 3gpp-Sbi-Target-Nf-Id

This header contains the target NF (Service) Instance ID in an HTTP 307/308 response (see clause 6.10.9).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Target-Nf-Id-Header = "3gpp-Sbi-Target-Nf-Id:" OWS "nfinst=" nfinst [ ";" OWS "nfservinst="
nfservinst ] OWS
```

The following parameters are defined:

- nfinst (NF instance): indicates a NF Instance ID, as defined in TS 29.510 [8]; the ABNF is defined in clause 5.2.3.2.8.
- nfservinst (NF service instance): indicates a NF Service Instance ID, as defined in TS 29.510 [8]; the ABNF is defined in clause 5.2.3.2.8.

EXAMPLE: 3gpp-Sbi-Target-Nf-Id: nfinst=54804518-4191-46b3-955c-ac631f953ed8; nfservinst=xyz

5.2.3.2.14 3gpp-Sbi-Max-Forward-Hops

The header contains the value of maximum number of allowed hops with specified node type to relay the request message to the target NF.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Max-Forward-Hops-Header = "3gpp-Sbi-Max-Forward-Hops:" OWS (DIGIT / %x31-39 DIGIT) ";" OWS
"nodetype=" nodetypevalue OWS
```

nodetypevalue = "scp"

EXAMPLE: Allowed up to 5 SCP hops to relay the request:

3gpp-Sbi-Max-Forward-Hops: 5; nodetype=scp

5.2.3.2.15 3gpp-Sbi-Originating-Network-Id

The header contains the PLMN Identity (MCC-MNC) of the source PLMN or the SNPN ID (MCC-MNC-NID) of the source SNPN of the received HTTP messages.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Originating-Network-Id-Header = "3gpp-Sbi-Originating-Network-Id:" OWS 3DIGIT "-" 2*3DIGIT [ "-"
11HEXDIG ] [ ";" OWS srcinfo ] OWS
```

```
srcinfo = "src" ":" RWS srctype "-" srcfqdn
```

```
srctype = "SCP" / "SEPP"
```

```
srcfqdn = 4*( ALPHA / DIGIT / "-" / "." )
```

NOTE 1: Implementations complying with earlier releases of the specification can require the presence of a leading white space after "3gpp-Sbi-Originating-Network-Id".

The srcinfo shall only be present when SCP or SEPP was unable to uniquely determine the value, i.e. PLMN ID, and has decided to insert the header with the value derived by configuration as described in Table 5.2.3.2.1-1.

The srcfqdn shall indicate FQDN of SCP or SEPP that inserted the header when srcinfo is present.

EXAMPLE 1: For a source PLMN:

3gpp-Sbi-Originating-Network-Id: 123-45

EXAMPLE 2: For a source PLMN and the header included by SEPP under the condition when the value of the header is derived based on the configuration and inserted by the SEPP:

3gpp-Sbi-Originating-Network-Id: 123-45; src: SEPP-sepp001.sepp.5gc.mnc045.mcc123.3gppnetwork.org

EXAMPLE 3: For a source SNPN:

3gpp-Sbi-Originating-Network-Id: 123-45-000007ed9d5

5.2.3.2.16 3gpp-Sbi-Access-Scope

The header indicates the access scope of the service request for NF service access authorization, as defined in clauses 6.7.3 and 6.10.11.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Access-Scope-Header = "3gpp-Sbi-Access-Scope:" OWS scope-token *(SP scope-token) OWS
```

```
scope-token = 1*NQCHAR
```

Scope-token shall consist of a list of space-delimited, case-sensitive strings, containing the NF service name of the NF service producer corresponding to the service request and, if defined for the specific resource/operation in the corresponding API, the additional resource/operation-level scope.

NQCHAR is defined in Appendix A of IETF RFC 6749 [22].

NOTE 1: This corresponds to the "scope" syntax defined for OAuth in clauses 3.3 and A.4 of IETF RFC 6749 [22] and also to the syntax of the "scope" parameter in AccessTokenReq in TS 29.510 [8]. This enables the SCP to set the scope parameter in the Nnrf_Get Access Token Request to the value of the 3gpp-Sbi-Access-Scope header received in an incoming service request, or to a list of scopes that is the intersection of the scopes indicated in the 3gpp-Sbi-Access-Scope header and the scopes expected by the NF Service producer (as registered in its NF profile).

NOTE 2: For indirect communication with delegated discovery (see clause 6.10.11.2), for a specific resource / operation for which the API defines a resource/operation-level scope, the NF service consumer does not and need not know whether the NF service producer is configured to require the resource/operation level scope or not. The setting of the 3gpp-Sbi-Access-Scope header is the same regardless of whether the NF service producer is configured to require the resource/operation level scope or not.

EXAMPLE: 3gpp-Sbi-Access-Scope: nhss-ims-uecm nhss-ims-uecm:authorize:invoke

5.2.3.2.17 3gpp-Sbi-Access-Token

The header contains an Access Token in a service response, for possible re-use in subsequent service requests, as defined in clause 6.10.11.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Access-Token = "3gpp-Sbi-Access-Token:" OWS credentials OWS
```

See clause 11.4 of IETF RFC 9110 [11] for the definition of "credentials".

NOTE: The 3gpp-Sbi-Access-Token header is encoded as the Authorization header.

5.2.3.2.18 Void

5.2.3.2.19 3gpp-Sbi-Target-Nf-Group-Id

This header contains the NF Group ID (e.g. UDM, HSS, AUSF, UDR, CHF, PCF Group ID) of the NF service producer.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Target-Nf-Group-Id-Header = "3gpp-Sbi-Target-Nf-Group-Id:" OWS "nfgid=" nfGroupIdValue OWS
nfGroupIdValue = DQUOTE token DQUOTE
```

The following parameter is defined:

- nfgid (NF Group ID): indicates a NF Group ID, as defined in TS 29.571 [13].

EXAMPLE: 3gpp-Sbi-Target-Nf-Group-Id: nfgid="udm-group-15"

5.2.3.2.20 3gpp-Sbi-Nrf-Uri-Callback

The header contains the NRF API URI of the NF discovery service and may contain the NRF API URI of the NF management service. See clauses 6.5.3.2.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Nrf-Uri-Callback-Header = "3gpp-Sbi-Nrf-Uri-Callback:" OWS nrfUriCallbackParam *( OWS ";"
nrfUriCallbackParam ) OWS
```

```
nrfUriCallbackParam = nrfUriCallbackParamName ":" RWS nrfUriCallbackParamValue
```

```
nrfUriCallbackParamName = "nnrf-disc" / "nnrf-nfm" / token ; token is defined for future
extensibility
```

```
nrfUriCallbackParamValue = DQUOTE URI DQUOTE
```

URI shall comply with the URI definition in IETF RFC 3986 [14].

EXAMPLE 1: Header with NRF NF Discovery Service:

```
3gpp-Sbi-Nrf-Uri-Callback: nrf-disc: "https://nrf1.operator.com/nnrf-disc/v1"
```

EXAMPLE 2: Header with NRF NF Discovery and NF Management Services:

```
3gpp-Sbi-Nrf-Uri-Callback: nrf-disc: "https://nrf1.operator.com/nnrf-disc/v1"; nrf-nfm:
"https://nrf1.operator.com/nnrf-nfm/v1"
```

5.2.3.2.21 3gpp-Sbi-NF-Peer-Info

This header contains the IDs of the NF (service) instance as HTTP client and the NF (service) instance as HTTP server.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-NF-Peer-Info-Header = "3gpp-Sbi-NF-Peer-Info" ":" OWS peerinfo *( ";" OWS peerinfo ) OWS
```

```
peerinfo = peertype "=" token
```

```
peertype = "srcinst" / "srcservinst" / "srcscp" / "srcsepp" / "dstinst" / "dstservinst" / "dstscp" /
"dstsepp" / "dstfqdn" / "srcfqdn"
```

The following peertype are defined:

- srcinst (Source NF instance): indicates the Source NF Instance ID, as defined in TS 29.510 [8];
- srcservinst (Source NF service instance): indicates the Source NF Service Instance ID, as defined in TS 29.510 [8]; if this parameter is present, srcinst shall also be present;
- srcscp (Source SCP): indicates the FQDN of the Source SCP, the format is "SCP-<SCP FQDN>"; this parameter shall only be included by an SCP, i.e. when the HTTP request or response message is originated or relayed by an SCP;
- srcsepp (Source SEPP): indicates the FQDN of the Source SEPP, the format is "SEPP-<SEPP FQDN>"; this parameter shall only be included by a SEPP, i.e. when the HTTP request or response message is originated or relayed by a SEPP;
- dstinst (Destination NF instance): indicates the Destination NF Instance ID, as defined in TS 29.510 [8];
- dstservinst (Destination NF service instance): indicates the Destination NF Service Instance ID, as defined in TS 29.510 [8]; if this parameter is present, dstinst shall also be present;
- dstscp (Destination SCP): indicates the FQDN of the Destination SCP, the format is "SCP-<SCP FQDN>"; this parameter shall contain the next-hop SCP of the HTTP request or response message to be included by an SCP or SEPP or by clients/servers sending requests/responses to an SCP;
- dstsepp (Destination SEPP): indicates the FQDN of the Destination SEPP, the format is "SEPP-<SEPP FQDN>"; this parameter shall be included by an SCP or by clients/servers sending requests/responses to a SEPP; it may also be included by a SEPP, based on operator's policy;
- dstfqdn (Destination FQDN): indicates the FQDN of the Destination NF, the format is "<DESTINATION FQDN>"; this parameter shall only be included in HTTP requests by the client if the client knows the FQDN of the destination NF but the NF (Service) instance ID of the destination NF is unknown;
- srcfqdn (Source FQDN): indicates the FQDN of the Source NF, the format is "<SOURCE FQDN>"; this parameter shall only be included in a HTTP response originated by the NF when answering to a HTTP request including the "dstfqdn" parameter, i.e. as a result of swapping the source and destination.

NOTE: When the NF consumer communicates with NRF and the NF consumer does not know the NRF NF instance ID, the NF consumer can use the dstfqdn to indicate the destination NRF in the header in the request to the NRF, and the NRF can use srcfqdn parameter in the response to the NF consumer.

The header shall contain the source peer information, and should contain the destination peer information if available.

EXAMPLE 1: 3gpp-Sbi-NF-Peer-Info: srcinst=54804518-4191-46b3-955c-ac631f953ed8; dstinst=54804518-4191-4453-569c-ac631f74765cd

EXAMPLE 2: 3gpp-Sbi-NF-Peer-Info: srcinst=54804518-4191-46b3-955c-ac631f953ed8; dstfqdn=inst1.nrf.5gc.mnc123.mcc456.3gppnetwork.org

5.2.3.2.22 3gpp-Sbi-Source-NF-Client-Credentials

The header contains client credentials assertion of a source NF instance (e.g. NWDAF, NWDAF containing AnLF) in a service request that is sent from an NF Service Consumer (e.g., DCCF, NWDAF containing MTLF) to an NF Service Producer (e.g. AMF, SMF, NWDAF containing MTLF). The purpose is to enable the authorization of NF service consumers for data access via DCCF or NWDAF containing MTLF (see clause 13.3.8.1 and Annex X of TS 33.501 [17]).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Source-NF-Client-Credentials-Header = "3gpp-Sbi-Source-NF-Client-Credentials:" OWS jwt OWS
```

The client credentials assertion shall be a JSON Web Token (JWT) as defined in clause 5.2.3.2.11, with the sub claim identifying the source NF instance, i.e. corresponding to the sourceNfInstanceId claim specified in Table 6.3.5.2.4-1 of TS 29.510 [8].

The ABNF of the JSON Web Token (JWT) is defined in clause 5.2.3.2.11.

NOTE 1: The 3gpp-Sbi-Source-NF-Client-Credentials header in the service request sent from the NF Service Consumer (e.g., DCCF) to an NF Service Producer (e.g. AMF, SMF) has the same contents as the 3gpp-Sbi-Client-Credentials header received by the NF Service Consumer (e.g. DCCF) from the source NF instance (e.g. NWDAF).

NOTE 2: In the scenario of the authorization of a first MTLF to request ML models to a second MTLF on behalf of an AnLF, the 3gpp-Sbi-Source-NF-Client-Credentials header in the service request sent from the NF Service Consumer (e.g. NWDAF containing the first MTLF) to the NF Service Producer (e.g. NWDAF containing the second MTLF) has the same contents as the 3gpp-Sbi-Client-Credentials header received by the NF Service Consumer (i.e. the first MTLF) from the source NF instance i.e. ML model consumer (e.g. AnLF).

5.2.3.2.23 3gpp-Sbi-Scp-apiRoot

The header contains the apiRoot of an SCP in the target PLMN (see clause 4.4 of TS 29.501 [5]) in a request sent from a source PLMN to a target PLMN when using indirect communication with or without delegated discovery between different PLMNs with possible NF selection at target PLMN and the apiRoot of the SCP in the target PLMN is known by the source PLMN (see clause 6.10.12). The authority of the 3gpp-Sbi-Scp-apiRoot header may correspond to an authority of a specific SCP, or to an authority that can resolve into several SCPs for scalability, load sharing and resiliency purposes.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Scp-ApiRoot-Header = "3gpp-Sbi-Scp-apiRoot" ":" OWS sbi-scheme "://" sbi-authority [ prefix ] OWS
```

The ABNF of sbi-scheme, sbi-authority and prefix is defined in clause 5.2.3.2.4.

EXAMPLE: 3gpp-Sbi-Scp-apiRoot: https://scp12.operator.com/a/b/c

5.2.3.3 Optional to support custom headers

5.2.3.3.1 General

The 3GPP NF Services may support the HTTP custom headers specified in Table 5.2.3.3-1 below. A description of each custom header and the normative requirements on when to include them are also provided in Table 5.2.3.3-1.

Table 5.2.3.3-1: Optional HTTP custom headers

Name	Reference	Description
3gpp-Sbi-Sender-Timestamp	Clause 5.2.3.3.2	This header may be used to indicate the date and time (with a millisecond granularity) at which an HTTP request or response is originated. This may be used e.g. for measuring signalling delays between different NF service instances.
3gpp-Sbi-Max-Rsp-Time	Clause 5.2.3.3.3	This header may be used in a HTTP request to indicate the duration during which the HTTP client waits for a response. See clause 6.11.2.
3gpp-Sbi-Correlation-Info	Clause 5.2.3.3.4	This header may be used to contain correlation information (e.g. UE identity), that may be used by an operator in various offline network management, performance analysis and troubleshooting tools/applications to identify messages (requests, responses, subscriptions, notifications) related to a particular subscriber. See clause 6.13.
3gpp-Sbi-Alternate-Chf-Id	Clause 5.2.3.3.5	This header may be used to indicate a primary or secondary CHF instance, e.g. when using indirect communication with delegated discovery. See clause 6.10.3.5.
3gpp-Sbi-Request-Info	Clause 5.2.3.3.12	<p>This header may be used to indicate additional information related to a HTTP request, e.g. if the request is involving a reselection towards an alternative NF, and/or if the request is a retransmission of a request towards an (alternative) NF.</p> <p>This header may be used in a non-idempotent HTTP request message to include an idempotency key to enable the receiver to detect possible duplicated request messages. See clause 5.2.8.</p> <p>This header may also be used in notification requests to indicate to the SCP a prefix of the Callback URI when binding procedures are not supported.</p>
3gpp-Sbi-Notif-Accepted-Encoding	Clause 5.2.3.3.6	This header may be used to indicate the content encodings supported by the NF service Consumer when receiving notifications related to the subscriptions data conveyed by the HTTP request in which the header is included. See clause 6.9.2.1.
3gpp-Sbi-Consumer-Info	Clause 5.2.3.3.7	<p>This header is used in a service request to create a subscription to indicate the API version(s) and feature(s) of the corresponding NF service(s) for the subscribed event(s) and the accepted encodings for notifications of the subscribed event(s), which are supported by the NF consumer.</p> <p>The NF consumer may include this header when subscribing to an intermediate NF for event(s) which may be detected and reported directly by a target NF, e.g. subscribe to Location Reporting event at AMF via UDM with AMF directly reporting the notifications to the NF consumer. See clause 6.2.2.</p> <p>The NF service consumer may include this header when providing a Callback URI when the authority part of the Callback URI is shared by several NF service consumer instances. See clause 6.12.1 for the usage of this parameter. The NF service consumer may also include this header when providing a Callback URI including a prefix, for use during NF service consumer reselection, when binding procedures are not supported.</p>
3gpp-Sbi-Response-Info	Clause 5.2.3.3.8	This header may be used to provide additional information related to an HTTP response, e.g. in a 4xx or 5xx response sent: <ul style="list-style-type: none"> - by an SCP to indicate whether it attempted to retransmit the request to alternative HTTP server instances (see clause 6.10.8.1); or - by an alternative HTTP server instance to indicate whether the resource/context has been transferred to the instance sending the response, or by an HTTP server instance to indicate that the failed request shall not be retried (see clause 6.10.3.4, 6.10.5.1 and 6.10.8.1).

3gpp-Sbi-Selection-Info	Clause 5.2.3.3.10	This header may be included in a HTTP request message for indirect communication and may be used by the SCP when performing the (re)selection of the target NF. See clauses 6.10.3.2 and 6.10.5.1.
3gpp-Sbi-Interplmn-Purpose	Clause 5.2.3.3.11	This header is used in HTTP request to indicate the intended purpose for inter-PLMN signaling. The HTTP client may include this header in HTTP request when the target NF is in a different PLMN, and if included shall set the intended purpose of the HTTP request. SEPP shall evaluate the contents of this header against the local policy and continue or reject the request if received. (see clause 6.14.3)
3gpp-Sbi-Retry-Info	Clause 5.2.3.3.13	This header may be included in a HTTP request message for indirect communication to indicate that the request shall only be sent once and shall not be retried.
3gpp-Sbi-Other-Access-Scopes	Clause 5.2.3.3.14	This header may be included in a service request for Indirect communication with delegated discovery to indicate other access scopes that are desired to be obtained for the access token, in addition to the scopes indicated in the 3gpp-Sbi-Access-Scope, that are not required for the service request itself but that may be required for further service requests. See clauses 5.2.3.3.14 and 6.10.11.
3gpp-Sbi-Binding-Indication-Notify-Restricted	Clause 5.2.3.3.15	This header may be inserted by an intermediate NF in a subscription creation request to indicate that updated binding indication shall not be sent via notifications, i.e. for a subscription by intermediate NF on behalf of the NF consumer and the notification requests are sent directly to the NF consumer. See clauses 5.2.3.3.3 and 6.12.4.

5.2.3.3.2 3gpp-Sbi-Sender-Timestamp

The header contains the date and time (with a millisecond granularity) at which an HTTP request or response is originated.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Sender-Timestamp-Header = "3gpp-Sbi-Sender-Timestamp:" OWS day-name "," SP date1 SP time-of-day
                              "." milliseconds SP "GMT" OWS
```

```
milliseconds = 3DIGIT
```

day-name, date1, time-of-day shall comply with the definition in clause 5.6.7 of IETF RFC 9110 [11].

When a 3gpp-Sbi-Sender-Timestamp header field is generated, the sender should generate its field value as the best available approximation of the date and time of message generation.

NOTE: This is the same format as the Date header of clause 5.6.7 of IETF RFC 9110 [11], but with the time expressed with a millisecond granularity.

EXAMPLE: 3gpp-Sbi-Sender-Timestamp: Sun, 04 Aug 2019 08:49:37.845 GMT

5.2.3.3.3 3gpp-Sbi-Max-Rsp-Time

The header indicates the duration, expressed in milliseconds since the request was originated, during which the HTTP client waits for a response. See clause 6.11.2.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Max-Rsp-Time-Header = "3gpp-Sbi-Max-Rsp-Time:" OWS 1*5DIGIT OWS
```

EXAMPLE: 3gpp-Sbi-Max-Rsp-Time: 10000

5.2.3.3.4 3gpp-Sbi-Correlation-Info

The header contains correlation information e.g. UE identifier related to the HTTP request or response.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Correlation-Info-Header = "3gpp-Sbi-Correlation-Info:" OWS correlationinfo *( ";" OWS correlationinfo ) OWS
```

```
correlationinfo = ctype "-" cvalue
```

```
ctype = "imsi" / "impi" / "suci" / "nai" / "gci" / "gli" / "impu" / "msisdn" / "extid" / "imei" / "imeisv" / "mac" / "eui" / extension-token
```

```
extension-token = 1*( "!" / "#" / "$" / "%" / "&" / "'" / "*" / "+" / "." / "^" / "_" / "`" / "|" / "~" / DIGIT / ALPHA )
```

```
cvalue = 1*( tchar / "@" )
```

NOTE 1: Only one of each ctype can be included in the 3gpp-Sbi-Correlation-Info header; the possibility to include more than one of the same ctype is kept for future extensibility.

NOTE 2: extension-token is defined for future extensibility, defined as one or more occurrences of any of the characters allowed for "tchar", except for the HYPHEN-MINUS ("-") character. See clause 5.6.2 of IETF RFC 9110 [11] for the definition of "tchar".

The format of cvalue shall comply with the data type description provided in Table 5.2.3.3.4-1.

Table 5.2.3.3.4-1: cvalue format

ctype	Description
imsi	VarUeld format defined in Table 5.2.2-1 of TS 29.571 [13] for IMSI and starting after the string "imsi-"
impi	imsUeld format defined in Table 6.1.3.2.2-1 of TS 29.562 [45] for IMPI and starting after the string "impi-"
suci	SupiOrSuci format defined in Table 5.3.2-1 of TS 29.571 [13] for SUCI and starting after the string "suci-"
nai	VarUeld format defined in Table 5.2.2-1 of TS 29.571 [13] for NAI and starting after the string "nai-"
gci	VarUeld format defined in Table 5.2.2-1 of TS 29.571 [13] for GCI and starting after the string "gci-"
gli	VarUeld format defined in Table 5.2.2-1 of TS 29.571 [13] for GLI and starting after the string "gli-"
impu	imsUeld format defined in Table 6.1.3.2.2-1 of TS 29.562 [45] for IMPU and starting after the string "impu-". Depending on whether the IMPU contains a SIP URI or a TEL URI, the corresponding pattern from the definition of imsUeld in Table 6.1.3.2.2-1 of TS 29.562 [45] shall be used.
msisdn	VarUeld format defined in Table 5.2.2-1 of TS 29.571 [13] for MSISDN and starting after the string "msisdn-"
extid	VarUeld format defined in Table 5.2.2-1 of TS 29.571 [13] for External Identifier and starting after the string "extid-"
imei	Pei format defined in Table 5.3.2-1 of TS 29.571 [13] for IMEI and starting after the string "imei-"
imeisv	Pei format defined in Table 5.3.2-1 of TS 29.571 [13] for IMEISV and starting after the string "imeisv-"
mac	Pei format defined in Table 5.3.2-1 of TS 29.571 [13] for MAC address and starting after the string "mac-"
eui	Pei format defined in Table 5.3.2-1 of TS 29.571 [13] for IEEE Extended Unique Identifier (EUI-64) and starting after the string "eui-"

EXAMPLE 1: When UE identifier used is SUPI and SUPI type is an IMSI:

```
3gpp-Sbi-Correlation-Info: imsi-345012123123123
```

EXAMPLE 2: When UE identifier used is PEI and PEI type is an IMEISV:

3gpp-Sbi-Correlation-Info: imeisv-3550121231231230

EXAMPLE 3: When UE identifier used is PEI and PEI type is a MAC address:

3gpp-Sbi-Correlation-Info: mac-00-00-5E-00-53-00

EXAMPLE 4: When UE identifier used is GPSI and GPSI type is an MSISDN:

3gpp-Sbi-Correlation-Info: msisdn-1234567890

EXAMPLE 5: When UE identifier used is GPSI and GPSI type is an External Identifier:

3gpp-Sbi-Correlation-Info: extid-123456789@domain.com

EXAMPLE 6: When UE identifiers used are SUPI and GPSI where SUPI type is an IMSI and GPSI type is an MSISDN:

3gpp-Sbi-Correlation-Info: imsi-345012123123123; msisdn-1234567890

5.2.3.3.5 3gpp-Sbi-Alternate-Chf-Id

The header indicates a primary or a secondary CHF Instance ID. See clause 6.10.3.5.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Alternate-Chf-Id-Header = "3gpp-Sbi-Alternate-Chf-Id:" OWS "nfinst=" nfinst ";" OWS ( "primary" / "secondary" ) OWS
```

Parameter "nfinst" shall indicate an NF Instance ID, as defined in clause 5.2.2.2.2 in TS 29.510 [8]; the ABNF is defined in clause 5.2.3.2.8.

EXAMPLE 1: Service response from a primary CHF instance signalling a secondary CHF instance Id:

3gpp-Sbi-Alternate-Chf-Id: nfinst=54804518-4191-46b3-955c-ac631f953ed8; secondary

EXAMPLE 2: Service response from a secondary CHF instance signalling a primary CHF instance Id:

3gpp-Sbi-Alternate-Chf-Id: nfinst=54804518-4191-46b3-955c-ac631f953ed8; primary

5.2.3.3.6 3gpp-Sbi-Notif-Accepted-Encoding

The header indicates the content encodings supported when receiving notifications related to the subscriptions data conveyed by the HTTP request in which the header is included.

This header shall be compliant with Accept-Encoding header defined in IETF RFC 9110 [11] clause 12.5.3.

Example: 3gpp-Sbi-Notif-Accepted-Encoding: gzip;q=1.0, identity;q=0.5, *;q=0

5.2.3.3.7 3gpp-Sbi-Consumer-Info

This header contains a comma-delimited list of NF service consumer information from an HTTP client (as NF service consumer).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Consumer-Info-Header = "3gpp-Sbi-Consumer-Info:" OWS 1#( supportedService ";" OWS supportedVersions [ ";" OWS supportedFeatures ] [ ";" OWS acceptEncoding ] [ ";" OWS callback-uri-prefix ] [ ";" OWS intraPlmnCallbackRoot ";" OWS interPlmnCallbackRoot ] [ ";" OWS intermediateNfIndication ] ) OWS
```

```
supportedService = "service=" servicename
```

```
servicename = 1*( "-" / %x30-39 / %x41-5A / "_" / %x61-7A )
```

```

supportedVersions = "apiversion=" "(" OWS [ apimajorversion *( RWS apimajorversion ) OWS ] ")"
apimajorversion = %x31-39 [ *DIGIT ]
supportedFeatures = "supportedfeatures=" features
features = *HEXDIG
acceptEncoding = "acceptencoding=" %x22 encodingList %x22
encodingList = #( codings [ weight ] )
intraPlmnCallbackRoot = "intraPlmnCallbackRoot=" DQUOTE sbi-scheme "://" sbi-authority
[ prefix ] DQUOTE
interPlmnCallbackRoot = "interPlmnCallbackRoot=" DQUOTE sbi-scheme "://" sbi-authority
[ prefix ] DQUOTE
sbi-scheme = "http" / "https"
sbi-authority = host [ ":" port ]
port = *DIGIT
prefix = path-absolute ; path-absolute production rule from IETF RFC 3986, clause 3.3
callback-uri-prefix = "callback-uri-prefix=" DQUOTE prefix DQUOTE
intermediateNfIndication = "intermediate-nf=" intermediateNfValue
intermediateNfValue = "true"

```

"service" (Mandatory parameter): Supported Service parameter indicates the name of a service, as defined in TS 29.510 [8], which is supported by the sender as NF service consumer.

"apiversion" (Mandatory parameter): Supported Versions parameter indicates the major version(s) of the service API that are supported by the sender as NF service consumer.

"supportedfeatures" (Optional parameter): Supported Features parameter carries a string containing a bitmask in hexadecimal representation, as specified for SupportedFeatures data type in TS 29.571 [13], to indicate the feature(s) of the service API that are supported by the sender as NF service consumer.

"acceptencoding" (Optional parameter): Accept Encoding carries a string indicating the accepted content encodings supported by the sender as NF service consumer, when receiving notifications defined by the service. In the ABNF definition, "codings" and "weight" are defined in IETF RFC 9110 [11] clauses 12.5.3 and 12.4.2.

"intraPlmnCallbackRoot", "interPlmnCallbackRoot" (Optional parameters): intra plmn callback root and inter plmn callback root supported by the sender as NF service consumer, for the indicated service.

"callback-uri-prefix" (Optional parameter): The NF service consumer may include this parameter when providing a Callback URI when the authority part of the Callback URI is shared by several NF service consumer instances. The NF service consumer may also include this header when providing a Callback URI including a prefix, for use during NF service consumer reselection, when binding procedures are not supported. When present, the "callback-uri-prefix" shall be a path-absolute as specified IETF RFC 3986 [14] (i.e. the first path segment(s) after the authority) which is part of the Callback URI provided by a NF service consumer in the corresponding service request message sent to a NF service producer. The authority and "callback-uri-prefix" in the Callback URI shall uniquely identify a consumer service instance. See clause 6.12.1 for the usage of this parameter.

"intermediate-nf" (Optional parameter): The intermediate NF indication parameter is a boolean set to true indicating that the header is provided by an intermediate NF as the NF service consumer.

EXAMPLE 1: The NF consumer supports Namf_EventExposure OpenAPI "v1" without any optional feature:
3gpp-Sbi-Consumer-Info: service=namf-evts; apiversion=(1)

EXAMPLE 2: The NF consumer supports Nsmf_EventExposure OpenAPI "v1" and "v2" with optional feature number 1 and accepted encoding "gzip":

3gpp-Sbi-Consumer-Info: service=nsmf-event-exposure; apiversion=(1 2); supportedfeatures=01; acceptencoding="gzip; q=1.0, *,q=0.5"

EXAMPLE 3: The NF consumer supports both Namf_EventExposure OpenAPI "v1" and Nsmf_EventExposure OpenAPI "v2":

3gpp-Sbi-Consumer-Info: service=namf-evts; apiversion=(1), service=nsmf-event-exposure; apiversion=(2)

EXAMPLE 4: An AMF service instance supports Nsmf_PDUSession OpenAPI "v1", provides the callback URI <https://amf45.operator.com/servinst123/pdusession>, whereby the authority is shared by more than one AMF service instance, while the prefix "/servinst123" uniquely identifies a specific AMF service instance:

3gpp-Sbi-Consumer-Info: service=nsmf-pdusession; apiversion=(1); callback-uri-prefix="/servinst123"

EXAMPLE 5: The NF consumer supports Namf_EventExposure OpenAPI "v1" and sends intra PLMN callback root "<https://operator.com>" and inter PLMN callback root "<https://5gc.mnc012.mcc345.3gppnetwork.org>" in the header:

3gpp-Sbi-Consumer-Info: service=namf-evts; apiversion=(1); intraPlmnCallbackRoot="https://operator.com"; interPlmnCallbackRoot=https://5gc.mnc012.mcc345.3gppnetwork.org

EXAMPLE 6: The intermediate NF service consumer (e.g. UDM) supports Namf_EventExposure OpenAPI "v1" and sends intra PLMN callback root "<https://intermediate-nf-authority.com>" and inter PLMN callback root "<https://5gc.mnc012.mcc345.3gppnetwork.org>" in the header:

3gpp-Sbi-Consumer-Info: service=namf-evts; apiversion=(1); intraPlmnCallbackRoot="https://intermediate-nf-authority.com"; interPlmnCallbackRoot="https://5gc.mnc012.mcc345.3gppnetwork.org";intermediate-nf=true

5.2.3.3.8 3gpp-Sbi-Response-Info

The header contains a list of additional information related to an HTTP response. It may be included e.g. in a 4xx or 5xx response sent:

- by an SCP to indicate whether it attempted to retransmit the request to alternative HTTP server instances; or
- by an alternative HTTP server instance to indicate whether the corresponding resource or context has been transferred to the alternative HTTP server instance, or by an HTTP server instance to indicate that the failed request shall not be retried.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Response-Info-Header = "3gpp-Sbi-Response-Info:" OWS resp-info-param *( ";" OWS resp-info-param ) OWS
```

```
resp-info-param = resp-info-param-name "=" OWS resp-info-param-value
```

```
resp-info-param-name = "request-retransmitted" / "nfinst" / "nfset" / "nfservinst" / "nfserviceset" / "context-transferred" / "no-retry" / token
```

```
resp-info-param-value = token
```

The following parameters are defined:

- "request-retransmitted": this parameter indicates, in an error response, whether the SCP attempted to (re)transmit the request to alternative HTTP server instances. When present, it shall be set to "true" if so, and to "false" otherwise. See clause 6.10.8.1.
- "nfinst", "nfset", "nfservinst", "nfserviceset": one or more of these parameters may be present in an error response, when the request-retransmitted is set to "true". When present, it shall indicate the NF Instances, NF Sets, NF Service Instances or NF Service Sets that were attempted to serve the request. See clause 6.10.8.1. The

value of the `nfirst`, `nfset`, `nfservinst` and `nfserviceset` parameters shall be encoded as defined for the corresponding parameters in clause 5.2.3.2.5.

- "context-transferred": this parameter indicates, in an error response, whether the corresponding resource or context has been transferred to the HTTP server instance sending the response. When present, it shall be set to "true" if the request has been transferred, i.e. the subsequent requests towards the resource or context shall be sent to the HTTP server instance sending the response, and to "false" otherwise.
- "no-retry": this parameter indicates, in an error response, whether the failed request can be retried at other alternative HTTP server instance or not. When present, it shall be set to "true" if the failed request shall not be retried at other alternative NF instances, and to "false" otherwise.

NOTE: Additional parameters can be defined in future versions of the specification.

EXAMPLE 1: `3gpp-Sbi-Response-Info: request-retransmitted=true`

EXAMPLE 2: `3gpp-Sbi-Response-Info: request-retransmitted=true; nfirst=54804518-4191-46b3-955c-ac631f953ed8; nfirst=54804518-4191-46b3-955c-ac631f953456; nfirst=54804518-4191-46b3-955c-ac631f953780`

EXAMPLE 3: `3gpp-Sbi-Response-Info: context-transferred=false; no-retry=true`

5.2.3.3.9 Void

5.2.3.3.10 3gpp-Sbi-Selection-Info

The header contains a comma-delimited list of additional (re)selection information for an HTTP request message. It may be included by a NF service consumer or a NF service producer in a HTTP request message for indirect communication. If the header is received by the SCP and the SCP supports the header, the SCP shall:

- avoid forwarding the request message to the target NF as indicated in the `3gpp-Sbi-Target-apiRoot` (if present in the request) or the request URI (otherwise) if reselection is set "true", i.e., the SCP shall perform a reselection; and
- use the selection-criteria included in this header together with `3gpp-Sbi-Routing-Binding` or `3gpp-Sbi-Discovery-*` headers whichever available, when the SCP performs the (re)selection of the target NF.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Selection-Info-Header = "3gpp-Sbi-Selection-Info:" OWS 1#( selection-info-element ) OWS
selection-info-element = ( "reselection=" reselectionvalue *( ";" OWS selection-criteria ) )
                        / ( selection-criteria *( ";" OWS selection-criteria ) )
reselectionvalue = "true" / "false"
selection-criteria = selection-action "=" token
selection-action = "not-select-nfservinst" / "not-select-nfserviceset" / "not-select-nfirst" / "not-select-nfset"
```

- reselection: it is a boolean and set to "false" by default. When it is set to "true", it indicates that the SCP shall perform a reselection, i.e., the SCP shall not forward the request message towards the target as indicated in the target uri or in the `3gpp-Sbi-Target-ApiRoot`. When this parameter occurs multiple times in the comma-delimited list, all parameters shall have the same value.
- not-select-nfservinst (the NF service instance(s) that shall not be selected): indicates an NF Service Instance ID. This parameter shall be present if the sender of the request message knows that the target NF or other potential target NF service instance that shall not be selected, e.g., when the target NF service instance is overloaded, or some NF service instances are out of service. (see also clause 6.4.3.4.5.2.1) When this parameter is present, one of not-select-nfserviceset or not-select-nfirst shall be present to enable the SCP to identify the nfservinst.
- not-select-nfserviceset (the NF service instance pertaining to a NF service set in a NF instance that shall not be selected): indicates an NF Service Set ID as defined in clause 28.13 in TS 23.003 [15]. This parameter shall be present if the sender of the request message knows that all NF service instances in the NF service set shall not be

selected, e.g., when target NF service instance has indicated its overload and the overload scope is NF service set level, in this case, not-select-nfservinst shall not be present. (see also clause 6.4.3.4.5.2.1)

- not-select-nfinst (the NF instance(s) that shall not be selected): indicates an NF Instance ID, as defined in clause 5.2.2.2.2 in TS 29.510 [8]. This parameter shall be present if the sender of the request message knows the target NF instance or other potential target NF instance that shall not be selected, e.g., when the target NF instance is overloaded, or other NF instance(s) is out of service, in this case, not-select-nfservinst shall not be present. (see also clause 6.4.3.4.5.2.1)
- not-select-nfset (the NF set that shall not be selected): indicates an NF Set ID, as defined in clause 28.12 in TS 23.003 [15]. This parameter may be present, e.g., during an initial resource creation with Delegated Discovery (Indirect Communication Mode D), the NF service consumer knows certain NF set shall not be selected.

EXAMPLE 1: The SCP may or may not perform reselection, but when doing reselection, it shall not select NF instance as identified by 87654321-4191-46b3-955c-ac631f953ed8.

3gpp-Sbi-Selection-Info: not-select-nfinst=87654321-4191-46b3-955c-ac631f953ed8

EXAMPLE 2: The SCP may or may not perform reselection, but when doing reselection, it shall not select NF service set in the NF instance (as identified in nfi87654321-4191-46b3-955c-ac631f953ed8).

3gpp-Sbi-Selection-Info: not-select-nfserviceset=setxyz.snmf-pdusession.nfi87654321-4191-46b3-955c-ac631f953ed8.5gc.mnc012.mcc345

EXAMPLE 3: The SCP shall perform reselection; and when doing reselection, it shall not select NF instance as identified by 87654321-4191-46b3-955c-ac631f953ed8.

3gpp-Sbi-Selection-Info: reselection=true; not-select-nfinst=87654321-4191-46b3-955c-ac631f953ed8

EXAMPLE 4: The SCP shall perform reselection; and when doing reselection, the SCP shall not select NF service instance xyz1 and xyz2 in the NF instance identified by 87654321-4191-46b3-955c-ac631f953ed8, and NF service instance abc1 and abc2 in the NF instance identified by 12345678-4191-46b3-955c-ac631f953ed8.

3gpp-Sbi-Selection-Info: reselection=true; not-select-nfservinst=xyz1; not-select-nfservinst=xyz2; not-select-nfinst=87654321-4191-46b3-955c-ac631f953ed8, reselection=true; not-select-nfservinst=abc1; not-select-nfservinst=abc2; not-select-nfinst=12345678-4191-46b3-955c-ac631f953ed8

5.2.3.3.11 3gpp-Sbi-Interplmn-Purpose

The header contains the intended purpose for inter-PLMN signaling. See clauses 6.14.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Interplmn-Purpose-Header = "3gpp-Sbi-Interplmn-Purpose:" OWS N32Purpose ":" OWS additional-info OWS
```

```
N32Purpose = "ROAMING"
             / "INTER_PLMN_MOBILITY"
             / "SMS_INTERCONNECT"
             / "ROAMING_TEST"
             / "INTER_PLMN_MOBILITY_TEST"
             / "SMS_INTERCONNECT_TEST"
             / "SNPN_INTERCONNECT"
             / "SNPN_INTERCONNECT_TEST"
             / "DISASTER_ROAMING"
             / "DISASTER_ROAMING_TEST"
             / "DATA_ANALYTICS_EXCHANGE"
             / "DATA_ANALYTICS_EXCHANGE_TEST"
             / "INDIRECT_NETWORK_SHARING"
             / token
```

```
additional-info = token
```

- N32Purpose: The parameter for N32Purpose indicates the intended purpose of inter-PLMN signaling, and values specified in TS 29.573 [27] clause 6.1.5.3.9 are used.

EXAMPLE: 3gpp-Sbi-Interplmn-Purpose: ROAMING: usecaseA

5.2.3.3.12 3gpp-Sbi-Request-Info

The header contains a list of additional information related to a HTTP request which may be included by a NF or a SCP, to indicate e.g.:

- whether the HTTP request message is involving a reselection of an alternative NF;
- whether the HTTP request message is a retransmission of the message, i.e. the request message has been sent but being rejected with a temporary failure or timeout;

When the header is included by a NF acting as a HTTP client, an idempotency-key may be included for a non-idempotent request to enable the receiver to detect possible duplicated request messages as specified in clause 5.2.8.

The receiving NF may use the header, e.g. to determine whether to accept the request.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Request-Info-Header = "3gpp-Sbi-Request-Info:" OWS req-param *( ";" OWS req-param ) OWS
req-param = req-param-name "=" OWS req-param-value

req-param-name = "retrans" / "redirect" / "reason" / "idempotency-key" / "receivedrejectioncause" /
"callback-uri-prefix" / "nfinst" / "nfservinst" / "redirection-cause" / token

req-param-value = token
```

The following parameters are defined:

- "reason": indicates the reason for which the NF resends or redirects the HTTP request message. This may take one of the following values:
 - "unreachable": indicates that the HTTP request is redirected to an alternative NF due to the request URI (e.g. the resource URI or Notification/callback URI) is not reachable;
 - "overloaded": indicates that the HTTP request is redirected to an alternative NF as result of overload control enforcement, by doing redirection towards an alternative NF (see clause 6.4.3.5.1);
 - "3xx-redirect": indicates that the HTTP request is redirected to an alternative NF as result of receiving a 3xx status code.
 - "temporary-rejection-cause": indicates the HTTP request is retransmitted towards the same or alternative NF due to a temporary rejection.
- "receivedrejectioncause": indicates a temporary rejection application cause received from the NF or SCP (for last attempt) as defined in clause 5.2.7.2, when the "retrans" parameter is set to "true" and the reason is set to "temporary-rejection-cause". The cause data type is specified in clause 5.2.4.1 of TS 29.571 [13].
- "retrans": it is a boolean and shall be set to "true" to indicate that the request message has been retransmitted e.g. when the request didn't get any response or get a temporary failure cause, otherwise the "retrans" shall not be present.
- "redirect": it is a boolean and shall be set to "true" to indicate that the request message has been redirected to an alternative NF.
- "nfinst": indicates the NF Instance ID of the NF instance towards which the request was sent previously, as defined in TS 29.510 [8]; the ABNF is defined in clause 5.2.3.2.8. This parameter may be present if the "reason" parameter is present to indicate the NF instance which was not reachable, or which redirected or rejected the request.

- "nfservinst": indicates the NF Service Instance ID of the NF service instance towards which the request was sent previously, as defined in TS 29.510 [8]; the ABNF is defined in clause 5.2.3.2.8. This parameter may be present if the "reason" parameter is present to indicate the NF service instance which was not reachable, or which redirected or rejected the request.
- "redirection-cause": indicates the redirection cause received in the 3xx redirect, when the "reason" parameter is set to "3xx-redirect". The redirection cause is defined as the cause attribute received in the RedirectResponse data type as specified in clause 5.2.4.21 of TS 29.571 [13].
- "idempotency-key": it is a string and may be encoded using Universally Unique Identifier (UUID), as described in IETF RFC 9562 [47], to uniquely identify a request message (to be received) in the target NF. See clause 5.2.8.
- "callback-uri-prefix": path-absolute as specified IETF RFC 3986 [14] (i.e. the first path segment(s) after the authority) which is part of the Callback URI. The ABNF is defined in clause 5.2.3.3.7.

NOTE 1: The "nfinst" and "nfservinst" parameters can be used for trouble shooting, KPIs, to avoid redirecting the request back to the previous NF (service) instance if the request needs to be redirected again (thus avoiding a redirection loop) and/or to refrain from redirecting further requests to an overloaded NF (service) instance.

EXAMPLE 1: For a request retransmitted to an alternative NF due to the rejection by the original target NF with a temporary rejection cause:

```
3gpp-Sbi-Request-Info: retrans=true; redirect=true; reason=temporary-rejection-cause;
receivedrejectioncause=INSUFFICIENT_RESOURCES
```

EXAMPLE 2: For a request sent towards an alternative NF due to the original target NF not reachable:

```
3gpp-Sbi-Request-Info: redirect=true; reason=unreachable
```

EXAMPLE 3: For a non-idempotent request:

```
3gpp-Sbi-Request-Info: idempotency-key=54804518-4191-46b3-955c-ac631f953ed8
```

EXAMPLE 4: For a notification request with a callback URI containing the prefix "/abc":

```
3gpp-Sbi-Request-Info: callback-uri-prefix="/abc"
```

EXAMPLE 5: For a request retransmitted towards an alternative NF due to the original target NF redirecting the request:

```
3gpp-Sbi-Request-Info: retrans=true; redirect=true; reason=3xx-redirect; nfinst= 54804518-4191-
46b3-955c-ac631f953ed8; nfservinst=xyz; redirection-cause="NF service instance shutting down".
```

EXAMPLE 6: For a request sent towards an alternative NF due to the original target NF being overloaded:

```
3gpp-Sbi-Request-Info: redirect=true; reason=overloaded; nfinst= 54804518-4191-46b3-955c-
ac631f953ed8.
```

5.2.3.3.13 3gpp-Sbi-Retry-Info

The header may be included in a HTTP request message for indirect communication to indicate that the request shall only be sent once and shall not be retried.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Retry-Info-Header = "3gpp-Sbi-Retry-Info:" OWS retriesindication OWS
retriesindication = "no-retries"
```

The following value is defined:

- "no-retries" indicates that the request shall only be sent once and shall not be retried to the same nor alternative endpoints of the same target NF service instance nor towards another target NF service instance once the request has been forwarded once.

EXAMPLE 1: NF service consumer instructing the SCP to not perform any retries:

3gpp-Sbi-Retry-Info: no-retries

EXAMPLE 2: NF service consumer instructing the SCP to perform an NF reselection, not reselecting the NF instance identified by 87654321-4191-46b3-955c-ac631f953ed8, and to not perform any retries then if no successful response is received from the reselected NF instance.

3gpp-Sbi-Selection-Info: reselection=true; not-select-nfinst=87654321-4191-46b3-955c-ac631f953ed8
3gpp-Sbi-Retry-Info: no-retries

5.2.3.3.14 3gpp-Sbi-Other-Access-Scopes

The header indicates other access scopes that are desired to be obtained for the access token, in addition to the scopes indicated in the 3gpp-Sbi-Access-Scope, that are not required for the service request itself but that may be required for further service requests. It enables the SCP to request access tokens that can be reused in further service requests, for NF service access authorization as defined in clauses 6.7.3 and 6.10.11.

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Other-Access-Scopes-Header = "3gpp-Sbi-Other-Access-Scopes:" OWS scope-token *(SP scope-token)
OWS
```

Scope-token shall consist of a list of space-delimited, case-sensitive strings, containing one or more NF service name(s) of the NF service producer and/or additional resource/operation-level scope(s) for these API(s) that are not already contained in the 3gpp-Sbi-Access-Scope header. The ABNF is defined in clause 5.2.3.2.16.

NQCHAR is defined in Appendix A of IETF RFC 6749 [22].

NOTE: This corresponds to the "scope" syntax defined for OAuth in clauses 3.3 and A.4 of IETF RFC 6749 [22] and also to the syntax of the "scope" parameter in AccessTokenReq in TS 29.510 [8]. This enables the SCP to include in the scope parameter in the Nnrf_Get Access Token Request, in addition to the scopes required for an incoming service request determined from the 3gpp-Sbi-Access-Scope header (see clause 5.2.3.2.16), the value of the 3gpp-Sbi-Other-Access-Scopes header received in the service request, or a list of scopes that is the intersection of the scopes indicated in the 3gpp-Sbi-Other-Access-Scopes header and the scopes expected by the NF Service producer (as registered in its NF profile).

EXAMPLE 1: 3gpp-Sbi-Access-Scope: nudm-uecm nudm-uecm:amf-registration:write
3gpp-Sbi-Other-Access-Scopes: nudm-sdm nudm-sdm:am-data:read nudm-sdm:smf-select-data:read

EXAMPLE 2: 3gpp-Sbi-Access-Scope: namf-comm namf-comm:n1-n2-messages
3gpp-Sbi-Other-Access-Scopes: namf-comm:ue-contexts:assign-ebi

5.2.3.3.15 3gpp-Sbi-Binding-Indication-Notify-Restricted

This header restricts sending updated Binding Indication in notification requests (see clause 6.5.3.3 and 6.12.4).

The encoding of the header follows the ABNF as defined in IETF RFC 9110 [11].

```
Sbi-Binding-Indication-Notify-Restricted-Header = "3gpp-Sbi-Binding-Indication-Notify-Restricted:"
OWS restrict-value OWS [";" OWS callback-root ] [ ";" OWS callback-uri-prefix ] OWS
```

```
restrict-value = "true"
```

```
callback-root = "callback-root=" DQUOTE sbi-scheme "://" sbi-authority [ prefix ] DQUOTE
```

The encoding of the parameters "callback-uri-prefix", "sbi-scheme", "sbi-scheme" and "prefix" see clause 5.2.3.3.7.

"callback-root" (Optional parameters): the callback root that can be used to uniquely identify the notification callback URI restricted for sending the updated binding indication for subscription resource. This parameter may be provided when multiple notification callback URIs are provided in the subscription.

"callback-uri-prefix" (Optional parameter): the callback prefix that can be used to uniquely identify the notification callback URI restricted for sending the updated binding indication for subscription resource. This parameter may be provided when multiple notification callback URIs are provided in the subscription.

If neither the "callback-root" parameter nor "callback-uri-prefix" parameter is included, all notifications for the subscription are restricted to carry the updated binding indication.

5.2.4 HTTP error handling

HTTP/2 connection error and stream error shall be supported as specified in clause 5.4 of IETF RFC 9113 [7].

Guidelines for error responses to the invocation of APIs of NF services are specified in clause 4.8 of TS 29.501 [3]. API specific error responses are specified in the respective technical specifications.

5.2.5 HTTP/2 server push

HTTP/2 Server Push as specified in clause 8.4 of IETF RFC 9113 [7] may be supported and may be used by a NF Service Producer to proactively push resources to a NF Service Consumer, see clause 4.9.5 of TS 29.501 [5].

A NF Service Consumer may choose to disable HTTP/2 Server Push by setting SETTINGS_ENABLE_PUSH to 0, as specified in clause 8.4 of IETF RFC 9113 [7].

5.2.6 HTTP/2 connection management

The HTTP request / response exchange mechanism as specified in clause 8.1 of IETF RFC 9113 [7] shall be supported between the 3GPP NFs. An HTTP/2 endpoint shall support establishing multiple HTTP/2 connections (at least two) towards a peer HTTP/2 endpoint. The peer HTTP/2 endpoint is identified by host and port pair where the host is derived from the target URI (see clause 6.1.1).

NOTE 1: HTTP/2 connection redundancy allows transporting messages through diverse IP paths and improve 5GC resiliency.

As per clause 8.1 of IETF RFC 9113 [7] a HTTP request / response exchange fully consumes a single stream. When the HTTP/2 Stream IDs on a given HTTP/2 connection is exhausted, an HTTP/2 endpoint, shall establish another HTTP/2 connection towards that peer HTTP/2 endpoints.

NOTE 2: As per IETF RFC 9113 [7], a stream ID once closed cannot be reused on the same HTTP/2 connection.

The 3GPP NF shall take care to avoid simultaneous stream ID exhaustion on all the available HTTP/2 connections towards each peer.

The 3GPP NF shall support gracefully shutdown of a HTTP/2 connection by sending a GOAWAY frame with "Error Code" field set to "NO_ERROR (0x0)". The HTTP connection should remain "open" (by the sender and receiver of GOAWAY frame) until all in-progress streams numbered lower or equal to the last stream identifier indicated by the "Last-Stream-Id" field in the GOAWAY frame are completed. See clause 6.8 of IETF RFC 9113 [7].

An NF acting as an HTTP/2 client shall support testing whether a connection is still active by sending a PING frame. An NF acting as an HTTP/2 server may test whether a connection is still active by sending a PING frame. An NF acting as an HTTP/2 client or server shall respond to received PING frames as specified in clause 6.7 of IETF RFC 9113 [7]. When and how often a PING frame may be sent is implementation specific but shall be configurable by operator policy. When HTTP server detects the connection failure, it shall follow connection error handling as defined in clause 5.4.1 of RFC 9113 [7].

NOTE 1: The above requirement also applies to network entities such as SCP and SEPP.

A PING frame shall not be sent more often than every 60 s on each path.

5.2.7 HTTP status codes

5.2.7.1 General

This clause describes the HTTP status codes usage on SBI.

HTTP status codes are carried in ":status" pseudo header field in HTTP/2, as defined in clause 8.1.2.4 in IETF RFC 9113 [7].

Table 5.2.7.1-1 specifies HTTP status codes per HTTP method which shall be supported on SBI. Support of an HTTP status code shall be:

- mandatory, which is marked in table as "M". This means that all 3GPP NFs shall support the processing of the specific HTTP status code for the specific HTTP method, when received in a HTTP response message. In such cases the 3GPP NF shall also support the handling of the "ProblemDetails" JSON object with the Content-Type header field set to the value "application/problem+json" for HTTP status codes 4xx and 5xx, if the corresponding API definition in the related technical specification does not specify another response body for the corresponding status code;
- service specific, which is marked in table as "SS" and means that the requirement to process the HTTP status code depends on the definition of the specific API; or
- not applicable, which is marked in table as "N/A". This means that the specific HTTP status code shall not be used for the specific HTTP method within the 3GPP NFs.

Table 5.2.7.1-1: HTTP status code supported on SBI

HTTP status code	HTTP method					
	DELETE	GET	PATCH	POST	PUT	OPTIONS
100 Continue	N/A	N/A	N/A	N/A	N/A	N/A
200 OK (NOTE 1, NOTE 2)	SS	M	SS	SS	SS	M
201 Created	N/A	N/A	N/A	SS	SS	N/A
202 Accepted	SS	N/A	SS	SS	SS	N/A
204 No Content (NOTE 2)	M	N/A	SS	SS	SS	SS
300 Multiple Choices	N/A	N/A	N/A	N/A	N/A	N/A
303 See Other	SS	SS	N/A	SS	SS	N/A
307 Temporary Redirect	SS	SS	SS	SS	SS	SS
308 Permanent Redirect	SS	SS	SS	SS	SS	SS
400 Bad Request	M	M	M	M	M	M
401 Unauthorized	M	M	M	M	M	M
403 Forbidden	M	M	M	M	M	M
404 Not Found	M	M	M	M	M	M
405 Method Not Allowed	SS	SS	SS	SS	SS	SS
406 Not Acceptable	N/A	M	N/A	N/A	N/A	SS
408 Request Timeout	SS	SS	SS	SS	SS	SS
409 Conflict	N/A	SS	SS	SS	SS	N/A
410 Gone	SS	SS	SS	SS	SS	SS
411 Length Required	N/A	N/A	M	M	M	SS
412 Precondition Failed	SS	SS	SS	SS	SS	N/A
413 Content Too Large	N/A	N/A	M	M	M	SS
414 URI Too Long	N/A	SS (NOTE 3)	N/A	N/A	SS	N/A
415 Unsupported Media Type	N/A	N/A	M	M	M	SS
422 Unprocessable Entity	N/A	N/A	SS	N/A	N/A	N/A
429 Too Many Requests	M	M	M	M	M	M
500 Internal Server Error	M	M	M	M	M	M
501 Not Implemented	SS	SS	SS	SS	SS	SS
502 Bad Gateway	M	M	M	M	M	M
503 Service Unavailable	M	M	M	M	M	M
504 Gateway Timeout (NOTE 4)	SS	SS	SS	SS	SS	SS
NOTE 1: "200 OK" response used on SBI shall contain body. NOTE 2: If the NF acting as an HTTP Client receives 2xx response code not appearing in table, the NF shall treat the received 2xx response: - as "204 No Content" if 2xx response does not contain body; and - as "200 OK" if 2xx response contains body. NOTE 3: If GET method includes any query parameter, the NF acting as an HTTP Client shall support "414 URI Too Long" status code. NOTE 4: A 5GC Network Function acting as an HTTP Client and supporting indirect communications shall support "504 Gateway Timeout" status code with "ProblemDetails" (see clause 6.10.8.2).						

5.2.7.2 NF as HTTP Server

A NF acting as an HTTP server shall be able to generate HTTP status codes specified in clause 5.2.7.1 per indicated HTTP method.

A request using an HTTP method which is not supported by any resource of a given 5GC SBI API shall be rejected with the HTTP status code "501 Not Implemented".

NOTE 1: In this case, the NF does not need to include in the HTTP response the "cause" attribute indicating corresponding error since the HTTP status code "501 Not Implemented" itself provides enough information of the error, i.e. the NF does not recognize the HTTP method.

If the specified target resource does not exist, the NF shall reject the HTTP method with the HTTP status code "404 Not Found".

If the NF supports the HTTP method for several resources in the API, but not for the target resource of a given HTTP request, the NF shall reject the request with the HTTP status code "405 Method Not Allowed" and shall include in the response an Allow header field containing the supported method(s) for that resource.

NOTE 2: In this case, the NF does not need to include in the HTTP response the "cause" attribute indicating corresponding error since the HTTP status code "405 Method Not Allowed" itself provides enough information of the error and hence the Allow header field lists HTTP method(s) supported by the target resource.

If a received HTTP request contains unknown IEs, i.e. Information Elements within the JSON body, the NF may discard such IEs and shall process the rest of the request message, unless the schema definition of the received message prohibits the presence of additional IEs or constrains their types. There are cases (e.g. Nnrf_NFManagement API) where the receiver of certain HTTP requests needs to process unknown IEs (e.g. to store in NRF an NF Profile containing vendor-specific attributes, and send them in NFDisccovery results).

If a received HTTP request contains IEs or query parameters not compliant with the schema defined in the corresponding OpenAPI specification, the NF should reject the request with the appropriate error code, e.g. "400 Bad Request (INVALID_MSG_FORMAT)", even when the failed IEs are defined as optional by the schema.

If a received HTTP PATCH request contains a body with modification instruction(s) for unknown attribute(s) in addition to modification instruction(s) for known attribute(s), the NF shall:

- a) implement all the modification(s) for known attribute(s) and unknown attribute(s) if explicitly specified in the corresponding specification of the API; or
- b) otherwise, implement the modification(s) for known attribute(s) and discard those modification instruction(s) for unknown attribute(s). The NF may additionally indicate in the response the result of the execution of the PATCH request, i.e. which modification(s) are implemented and/or which modification(s) are discarded, using the "PatchResult" JSON object as defined in TS 29.571 [13].

If the NF supports the HTTP method by a target resource but the NF cannot successfully fulfil the received request, the following requirements apply.

A NF as HTTP Server should map status codes to the most similar 3xx/4xx/5xx HTTP status code specified in table 5.2.7.1-1. If no such code is applicable, it should use "400 Bad Request" status code for errors caused by client side or "500 Server Internal Error" status code for errors caused on server side.

If the received HTTP request contains unsupported content format, the NF shall reject the HTTP request with the HTTP status code "415 Unsupported Media Type". If the HTTP PATCH method is rejected due to unsupported patch document, the NF shall include the Accept-Patch header field set to the value of supported patch document media types for a target resource i.e. to "application/merge-patch+json" if the NF supports "JSON Merge Patch" and to "application/json-patch+json" if the NF supports "JSON Patch". If the received HTTP PATCH request contains both "JSON Merge Patch" and "JSON Patch" documents and the NF supports only one of them, the NF shall ignore unsupported patch document.

NOTE 3: The format problem might be due to the request's indicated Content-Type or Content-Encoding header fields, or as a result of inspecting the content directly.

If the received HTTP request contains content larger than the NF is able to process, the NF shall reject the HTTP request with the HTTP status code "413 Content Too Large".

If the result of the received HTTP POST request used for a resource creation would be equivalent to the existing resource, the NF shall reject the HTTP request with the HTTP status code "303 See Other" and shall include in the HTTP response a Location header field set to the URI of the existing resource.

Protocol and application errors common to several 5GC SBI API specifications for which the NF shall include in the HTTP response content a "ProblemDetails" data structure or application specific error data structure with the "cause" attribute indicating corresponding error are listed in table 5.2.7.2-1.

Table 5.2.7.2-1: Protocol and application errors common to several 5GC SBI API specifications (HTTP server)

Protocol or application Error	HTTP status code	Description
INVALID_API	400 Bad Request	The HTTP request contains an unsupported API name or API version in the URI.
INVALID_MSG_FORMAT	400 Bad Request	The HTTP request has an invalid format.
INVALID_QUERY_PARAM	400 Bad Request	The HTTP request contains an unsupported query parameter in the URI. (NOTE 1)
MANDATORY_QUERY_PARAM_INCORRECT	400 Bad Request	A mandatory query parameter, or a conditional query parameter but mandatory required, for an HTTP method was received in the URI with semantically incorrect value. (NOTE 1)
OPTIONAL_QUERY_PARAM_INCORRECT	400 Bad Request	An optional query parameter for an HTTP method was received in the URI with a semantically incorrect value that prevents successful processing of the service request. (NOTE 1)
MANDATORY_QUERY_PARAM_MISSING	400 Bad Request	Query parameter which is defined as mandatory, or as conditional but mandatory required, for an HTTP method is not included in the URI of the request. (NOTE 1)
MANDATORY_IE_INCORRECT	400 Bad Request	A mandatory IE (within the JSON body or within a variable part of an "apiSpecificResourceUriPart" or within an HTTP header), or conditional IE but mandatory required, for an HTTP method was received with a semantically incorrect value. (NOTE 1)
OPTIONAL_IE_INCORRECT	400 Bad Request	An optional IE (within the JSON body or within an HTTP header) for an HTTP method was received with a semantically incorrect value that prevents successful processing of the service request. (NOTE 1)
MANDATORY_IE_MISSING	400 Bad Request	A mandatory IE (within the JSON body or within the variable part of an "apiSpecificResourceUriPart" or within an HTTP header), or conditional IE but mandatory required, for an HTTP method is not included in the request. (NOTE 1)
UNSPECIFIED_MSG_FAILURE	400 Bad Request	The request is rejected due to unspecified client error. (NOTE 2)
ACCESS_TOKEN_CLAIM_MISSING	401 Unauthorized	The request is rejected due to one or more missing claim(s) in the OAuth2.0 access token.
RESOURCE_CONTEXT_NOT_FOUND	400 Bad Request	The notification request is rejected because the callback URI still exists in the receiver of the notification, but the specific resource context identified within the notification content is not found in the NF service consumer.
ACCESS_TOKEN_CLAIM_MISMATCH	403 Forbidden	The request is rejected due to one or more claim(s) (e.g. slice, NSI, etc.) in the OAuth2.0 access token does not match the attributes of the NF service producer (e.g. the NF service producer does not serve any slice indicated in the producerSnsaiList or producerNsiList), as specified in clause 13.4.1.1.2 of TS 33.501 [17].
CCA_VERIFICATION_FAILURE	403 Forbidden	The request is rejected due to a failure to verify the 3gpp-Sbi-Client-Credentials at the receiving entity (e.g. NRF or NF service producer).
SOURCE_NF_CCA_VERIFICATION_FAILURE	403 Forbidden	The request is rejected due to a failure to verify the 3gpp-Sbi-Source-NF-Client-Credentials at the receiving entity (e.g. NRF or NF service producer).

TOKEN_CCA_MISMATCH	403 Forbidden	The request is rejected due to a mismatch between the subject claim in the access token and subject claim in the 3gpp-Sbi-Client-Credentials .
TOKEN_SOURCE_NF_CCA_MISMATCH	403 Forbidden	The request is rejected due to a mismatch between the sourceNFInstanceId claim in the access token and subject claim in the 3gpp-Sbi-Source-NF-Client-Credentials.
MODIFICATION_NOT_ALLOWED	403 Forbidden	The request is rejected because the contained modification instructions attempt to modify IE which is not allowed to be modified.
MISSING_PARAMETER	403 Forbidden	The request is rejected because one or more information elements of the requester required to authorize the requester are missing. (NOTE 1)
SUBSCRIPTION_NOT_FOUND	404 Not Found	The request for modification or deletion of a subscription, or the notification request, is rejected because the subscription is not found in the NF.
RESOURCE_URI_STRUCTURE_NOT_FOUND	404 Not Found	The request is rejected because a path segment of an "apiSpecificResourceUriPart" (as defined in clause 4.4.1 of TS 29.501 [5]) neither matches the format of an expected known variable part nor matches the name of an expected fixed part, and therefore is not found in the NF. (NOTE 5)
INCORRECT_LENGTH	411 Length Required	The request is rejected due to incorrect value of a Content-length header field.
MAX_JSON_SIZE_EXCEEDED	413 Content Too Large	The request is rejected because the JSON payload size exceeds the allowed maximum size (see clause 6.2 of TS 29.501 [5]).
NF_CONGESTION_RISK	429 Too Many Requests	The request is rejected due to excessive traffic which, if continued over time, may lead to (or may increase) an overload situation of the NF instance. (NOTE 7)
NF_SERVICE_CONGESTION_RISK	429 Too Many Requests	The request is rejected due to excessive traffic which, if continued over time, may lead to (or may increase) an overload situation of the NF service instance. (NOTE 7)
INSUFFICIENT_RESOURCES	500 Internal Server Error	The request is rejected due to insufficient resources.
UNSPECIFIED_NF_FAILURE	500 Internal Server Error	The request is rejected due to unspecified reason at the NF. (NOTE 3)
SYSTEM_FAILURE	500 Internal Server Error	The request is rejected due to generic error condition in the NF.
NF_FAILOVER	500 Internal Server Error	The request is rejected due to the unavailability of the NF, and the requester may trigger an immediate re-selection of an alternative NF based on this information. (NOTE 6) (NOTE 8).
NF_SERVICE_FAILOVER	500 Internal Server Error	The request is rejected due to the unavailability of the NF service, and the requester may trigger an immediate re-selection of an alternative NF service based on this information. (NOTE 6) (NOTE 8).
INBOUND_SERVER_ERROR	502 Bad Gateway	The request is rejected due to the receipt of an 5xx error from an inbound server that the NF Service Producer accessed while attempting to fulfil the request (see clause 6.4.2.1).
NF_CONGESTION	503 Service Unavailable	The NF instance experiences congestion and performs overload control, which does not allow the request to be processed. (NOTE 4) (NOTE 7)

NF_SERVICE_CONGESTION	503 Service Unavailable	The NF service instance experiences congestion and performs overload control, which does not allow the request to be processed. (NOTE 4) (NOTE 7)
TARGET_NF_NOT_REACHABLE	504 Gateway Timeout	The request is not served as the target NF is not reachable. (NOTE 6)
TIMED_OUT_REQUEST	504 Gateway Timeout	The request is rejected due a request that has timed out at the HTTP client (see clause 6.11.2).
<p>NOTE 1: "invalidParams" attribute shall be included in the "ProblemDetails" data structure indicating unsupported, missing or incorrect IE(s) or query parameter(s) or 3gpp-Sbi-Discovery-* header(s).</p> <p>NOTE 2: This application error indicates error in the HTTP request and there is no other application error value that can be used instead.</p> <p>NOTE 3: This application error indicates error condition in the NF and there is no other application error value that can be used instead.</p> <p>NOTE 4: If the reason for rejection is a temporary overload, the NF may include in the response a Retry-After header field to indicate how long the service is expected to be unavailable.</p> <p>NOTE 5: If the request is rejected because of an error in an URI before the first variable part of an "apiSpecificResourceUriPart", the "404 Not Found" HTTP status code may be sent without "ProblemDetails" data structure indicating protocol or application error.</p> <p>NOTE 6: The NF service consumer (as receiver of the cause code) should stop sending subsequent requests addressing the resource contexts in the producer's NF instance (for NF_FAILOVER) or NF service instance (for NF_SERVICE_FAILOVER) to avoid massive rejections. The NF service consumer may reselect an alternative NF service producer as specified clause 6.5 of TS 23.527 [38], e.g. using the Binding Indication of resource context. It is implementation specific for the NF service consumer to determine when and whether the NF producer becomes available again, e.g. when there is no other alternative available or at expiry of a local configured timer.</p> <p>NOTE 7: When a NF service producer receives NF_CONGESTION_RISK, NF_SERVICE_CONGESTION_RISK, NF_CONGESTION and NF_SERVICE_CONGESTION from a NF service consumer when sending a request message towards a callback/notification URI, the NF service producer shall identify the NF service consumer that is congested using either the authority of the notification/callback URI or together with the "callback-uri-prefix" if it is provided in 3gpp-Sbi-consumer-info as specified in clause 5.2.3.3.7.</p> <p>NOTE 8: The NF service producer (as receiver of the cause code) should stop sending subsequent notification requests addressing the session contexts towards the consumer NF (service) instance to avoid massive rejections, where the consumer NF (service) instance shall be identified by either the authority of the notification/callback URI or together with the "callback-uri-prefix" if it is provided in 3gpp-Sbi-consumer-info as specified in clause 5.2.3.3.7. The NF service producer may reselect an alternative NF service consumer as specified in clause 6.6 of TS 23.527 [38], e.g. using the Binding Indication of the session context. It is implementation specific for the NF service producer to determine when and whether the NF consumer becomes available again, e.g. when there is no other alternative available or at expiry of a local configured timer. Note that if a consumer NF service instance complying with an earlier version of the specification shares the same authority with other consumer NF service instances and sends the NF_FAILOVER and NF_SERVICE_FAILOVER causes to a NF service producer while not supporting the new callback-uri-prefix parameter in 3gpp-Sbi-consumer-info, this can result in the NF service producer no longer sending traffic to these consumer NF service instances sharing the same authority.</p>		

5.2.7.3 NF as HTTP Client

Besides the HTTP Status Codes defined in the API specification, a NF as HTTP client should support handling of 1xx, 3xx, 4xx and 5xx HTTP Status Codes specified in table 5.2.7.1-1, following the client behaviour in corresponding IETF RFC where the received HTTP Status Code is defined.

When receiving a not recommended or not recognized 1xx, 3xx, 4xx or 5xx HTTP Status Code, a NF as HTTP client should treat it as x00 status code of the class, as described in clause 15 of IETF RFC 9110 [11].

If 100, 200/204, 300, 400 or 500 response code is not defined by the API specification, the client may follow guidelines below:

- a) For 1xx (Informational):
 - 1) Discard the response and wait for final response.
- b) For 2xx (Successful):
 - 1) Consider the service operation is successful if no mandatory information is expected from the response content in subsequent procedure.

- 2) If mandatory information is expected from response content in subsequent procedure, parse the content following description in clause 15.2.1 of IETF RFC 9110 [11]. If parse is successful and mandatory information is extracted, continue with subsequent procedure.
 - 3) Otherwise, consider service operation has failure and start failure handling.
- c) For 3xx (Redirection):
- 1) Retry the request towards the directed resource referred in the Location header, using same request method.
- d) For 4xx (Client Error):
- 1) Validate the request message and make correction before resending. Otherwise, stop process and go to error handling procedure.
- e) For 5xx (Server Error):
- 1) Stop process and go to error handling process.

The handling of unknown, unexpected or erroneous HTTP request message IEs shall provide for the forward compatibility of the HTTP APIs used for the service-based interfaces. Therefore, the sending HTTP entity shall be able to safely include in a message a new optional IE. Such an IE may also have a new type. A receiving HTTP entity shall behave as specified in clause 5.2.7.2.

If a received HTTP response message contains unknown IEs (Information Elements within the JSON body), the NF may discard those IEs and it shall process the rest of the response message, as long as it is compliant with the OpenAPI schema definition of such response message.

If a received HTTP response message contains IEs not compliant with the schema defined in the corresponding OpenAPI specification (e.g., because the schema of the response body prohibits the presence of additional IEs or constrains their types), the NF shall stop processing such response message and go to error handling process.

5.2.7.4 SCP/SEPP

The SCP or SEPP shall be able to forward the HTTP status codes defined in Table 5.2.7.1-1 and Table 5.2.7.2-1 from HTTP Server to HTTP client. In addition, it shall be able to generate HTTP status codes to indicate failures during indirect communication (e.g. see clauses 6.10.3.2 and 6.10.6), error handling (see clause 6.10.8), detection and handling of loop path (see clause 6.10.10) and SCP or SEPP overload control (see clause 6.4) as defined in Table 5.2.7.4-1 and Table 5.2.7.4-2.

If the SCP or SEPP detects a loop in the routing path of an HTTP request, it should reject the request with the HTTP status code "400 Bad Request (MSG_LOOP_DETECTED)".

If the received HTTP request contains content larger than the SCP or SEPP is able to process, the SCP or SEPP shall reject the HTTP request with the HTTP status code "413 Content Too Large".

An HTTP status code "429 Too Many Requests (NF_CONGESTION_RISK)" is sent, when the SCP or SEPP detects that a given NF Service Consumer is sending excessive traffic which, if continued over time, may lead to (or may increase) an overload situation in the SCP or SEPP. If the SCP or SEPP decides to redirect HTTP requests to another less loaded SCP or SEPP, it may send the HTTP status code "307 Temporary Redirect" or "308 Permanent Redirect" with the cause attribute set to "SCP_REDIRECTION" (see clause 6.10.9) / "SEPP_REDIRECTION" as defined in Table 5.2.7.4-2.

The SCP or SEPP should map status codes to the most similar 3xx/4xx/5xx HTTP status code specified in Table 5.2.7.4-1 and Table 5.2.7.4-2. If no such code is applicable, it should use "400 Bad Request" status code for errors caused by client side or "500 Server Internal Error" status code for errors caused on server side.

Table 5.2.7.4-1: Protocol and application errors generated by the SCP/SEPP

Protocol or application Error	HTTP status code	Description
INVALID_API	400 Bad Request	The HTTP request contains an unsupported API name or API version in the URI.
INVALID_MSG_FORMAT	400 Bad Request	The HTTP request has an invalid format.
INVALID_QUERY_PARAM	400 Bad Request	The HTTP request contains an unsupported query parameter in the URI. (NOTE 1)
MANDATORY_QUERY_PARAM_INCORRECT	400 Bad Request	A mandatory query parameter, or a conditional query parameter but mandatory required, for an HTTP method was received in the URI with semantically incorrect value. (NOTE 1)
OPTIONAL_QUERY_PARAM_INCORRECT	400 Bad Request	An optional query parameter for an HTTP method was received in the URI with a semantically incorrect value that prevents successful processing of the service request. (NOTE 1)
MANDATORY_QUERY_PARAM_MISSING	400 Bad Request	Query parameter which is defined as mandatory, or as conditional but mandatory required, for an HTTP method is not included in the URI of the request. (NOTE 1)
MANDATORY_IE_INCORRECT	400 Bad Request	A mandatory IE (within a variable part of an "apiSpecificResourceUriPart" or within an HTTP header), or conditional IE but mandatory required, for an HTTP method was received with a semantically incorrect value. (NOTE 1)
OPTIONAL_IE_INCORRECT	400 Bad Request	An optional IE (within an HTTP header) for an HTTP method was received with a semantically incorrect value that prevents successful processing of the service request. (NOTE 1)
MANDATORY_IE_MISSING	400 Bad Request	A mandatory IE (within the variable part of an "apiSpecificResourceUriPart" or within an HTTP header), or conditional IE but mandatory required, for an HTTP method is not included in the request. (NOTE 1)
UNSPECIFIED_MSG_FAILURE	400 Bad Request	The request is rejected due to unspecified client error. (NOTE 2)
NF_DISCOVERY_FAILURE	400 Bad Request	The request is rejected by the SCP because no NF Service Producer can be found matching the NF service discovery factors (see clause 6.10.6).
INVALID_DISCOVERY_PARAM	400 Bad Request	The request is rejected by the SCP because it contains an unsupported discovery parameter (i.e. unknown 3gpp-Sbi-Discovery-* header) (see clause 6.10.3.2). (NOTE 1)
MSG_LOOP_DETECTED	400 Bad Request	The request is rejected because message loop is detected.
MISSING_ACCESS_TOKEN_INFORMATION	400 Bad Request	The request is rejected due to missing information in the service request that prevents the SCP from requesting an access token to the Authorization Server. See clause 6.10.3.5.
ACCESS_TOKEN_DENIED	403 Forbidden	The request is rejected due to the Authorization Server rejecting to grant an access token to the SCP. See clause 6.10.3.5.
PLMNID_MISMATCH	403 Forbidden	The request is rejected by the SEPP due to the PLMN ID in the bearer token carried in the "Authorization" header of the reconstructed message does not match the PLMN ID of the N32-f context.
REQUESTED_PURPOSE_NOT_ALLOWED	403 Forbidden	The request is rejected due to requested purpose provided in the HTTP request is not allowed by the policy. See clause 6.14.
ORIGINATING_NETWORK_ID_MISMATCH	403 Forbidden	The request is rejected because the PLMN ID or the SNPN ID received in the 3gpp-Sbi-Originating-Network-Id header does not match with any of the PLMN IDs that the sending SEPP represents.
INCORRECT_LENGTH	411 Length Required	The request is rejected due to incorrect value of a Content-length header field.

MAX_JSON_SIZE_EXCEEDED	413 Content Too Large	The request is rejected because the JSON payload size exceeds the allowed maximum size (see clause 6.2 of TS 29.501 [5]).
NF_CONGESTION_RISK	429 Too Many Requests	The request is rejected due to excessive traffic which, if continued over time, may lead to (or may increase) an overload situation.
INSUFFICIENT_RESOURCES	500 Internal Server Error	The request is rejected due to insufficient resources.
UNSPECIFIED_NF_FAILURE	500 Internal Server Error	The request is rejected due to unspecified reason at the SCP or SEPP. (NOTE 3)
SYSTEM_FAILURE	500 Internal Server Error	The request is rejected due to generic error condition in the SCP or SEPP.
NF_FAILOVER	500 Internal Server Error	The request is rejected by the SCP due to the unavailability of the NF, and the requester may trigger an immediate re-selection of an alternative NF based on this information.
NF_SERVICE_FAILOVER	500 Internal Server Error	The request is rejected by the SCP due to the unavailability of the NF service, and the requester may trigger an immediate re-selection of an alternative NF service based on this information.
MAX_SCP_HOPS_REACHED	502 Bad Gateway	The request is rejected due to the maximum number of allowed SCP hops has been reached when relaying the request message to the target NF.
NF_DISCOVERY_ERROR	502 Bad Gateway	The request is rejected due to the receipt of an 5xx or 429 response from the NRF during an NF Discovery procedure the SCP initiated to fulfil the request.
NF_CONGESTION	503 Service Unavailable	The SCP or SEPP experiences congestion and performs overload control, which does not allow the request to be processed. (NOTE 4)
TIMED_OUT_REQUEST	504 Gateway Timeout	The request is rejected due a request that has timed out at the HTTP client (see clause 6.11.2).
TARGET_NF_NOT_REACHABLE	504 Gateway Timeout	The request is not served as the target NF is not reachable (see clause 6.10.8.2).
NRF_NOT_REACHABLE	504 Gateway Timeout	The request is not served due to the NRF being unreachable (see clause 6.10.8.2).
TARGET_PLMN_NOT_REACHABLE	504 Gateway Timeout	The request is not delivered due to issues on interconnect with another PLMN (e.g. issues on N32 connection including contractual reasons).
NOTE 1: "invalidParams" attribute shall be included in the "ProblemDetails" data structure indicating unsupported, missing or incorrect IE(s) or 3gpp-Sbi-Discovery-* header(s).		
NOTE 2: This application error indicates error in the HTTP request and there is no other application error value that can be used instead.		
NOTE 3: This application error indicates error condition in the SCP/SEPP and there is no other application error value that can be used instead.		
NOTE 4: If the reason for rejection is a temporary overload, the SCP/SEPP may include in the response a Retry-After header field to indicate how long the service is expected to be unavailable.		

Table 5.2.7.4-2: Redirect responses generated by the SCP/SEPP

Cause value	HTTP status code	Description
SCP_REDIRECTION	307 Temporary Redirect 308 Permanent Redirect	The request is redirected to a different SCP (see clause 6.10.9).
SEPP_REDIRECTION	307 Temporary Redirect 308 Permanent Redirect	The request is redirected to a different SEPP (see clause 6.10.9).
SEPP_REDIRECTION_WITH_DISCOVERY	307 Temporary Redirect 308 Permanent Redirect	The request is redirected over N32-c to different SEPPs to be discovered (see clause 6.1.8.2 of TS 29.573 [27]).

5.2.8 HTTP/2 request retries

All NF services expose APIs across the service based interfaces and the APIs operate on resources. Invocation of an API through a HTTP method may result in the change of state of a resource depending of the request type. When a

HTTP/2 client sends a request and it does not receive a response or it experiences a delay, it does not guarantee that the HTTP/2 request has not been processed by the HTTP/2 server.

A HTTP/2 client may retry the same request that uses an idempotent method any time (see IETF RFC 9110 [11] clause 9.2.2).

Retrying a non-idempotent HTTP/2 request on the same resource before a response for the previous request is received may lead to state changes on the resource with unspecified behaviour. HTTP conditional requests, as specified in IETF RFC 9110 [11] may be used to avoid such situations.

An NF acting as an HTTP/2 client should also retry non-idempotent request if the request has not been processed, i.e. if the identifier of the stream corresponding to the request is larger than the Last-Stream-Id in a GOAWAY frame, or the REFUSED_STREAM error code is included in a RST_STREAM frame for the stream corresponding to the request as specified in clause 8.7 of IETF RFC 9113 [7]. API specific mechanisms as specified in respective technical specifications may be used for reconciling the state of resources, if the retry is attempted through a new TCP connection after a TCP connection failure.

The number of retry shall be limited. A client should always prefer to retry requests to an alternative server if the initial server is overloaded. In case of general overload situation where all possible servers are overloaded retry mechanisms should be disabled automatically.

The support of "detection of duplicated request message" is optional for HTTP clients and servers. When it is supported:

- the NF acting as an HTTP/2 client shall:
 - include an idempotency key (which shall uniquely identify the request message towards the target NF) in the 3gpp-Sbi-Request-Info header for a non-idempotent request message, e.g. a POST request;
 - include the same idempotency key in the 3gpp-Sbi-Request-Info header when subsequently the NF acting as an HTTP/2 client decides to retry the same request towards the same NF acting as an HTTP/2 server or an alternative NF (e.g. from the same NF (service) Set);
- the NF acting as an HTTP/2 server, upon receiving a request message containing an idempotency key in the 3gpp-Sbi-Request-Info header, may:
 - use the idempotency key to determine if it is a duplicated request message; and if so
- produce a proper response based on the current state of the resource/session context considering the original request has been processed. The SCP shall forward the idempotency key received from the HTTP client unmodified towards the target NF, regardless of whether the SCP performs (re)selection of the target NF.

The idempotency key that is supplied as part of every non-idempotent request shall be unique and shall not be reused with another request other than a retransmission of the same request. The server may consider an idempotency key as expired after an operator configurable timer.

5.2.9 Handling of unsupported query parameters

Unless specified otherwise for an API, a NF Service Producer that receives an HTTP request with one or more unsupported (i.e. not comprehended) query parameters shall:

- a) for safe HTTP methods (e.g. HTTP GET request):
 - ignore the unsupported query parameters and respond to the request based on the rest of the request (e.g. other supported query parameters); or
 - reject the HTTP request as specified below for non-safe HTTP methods, e.g. based on other query parameters in the request or based on a response becoming very large;
- b) for non-safe HTTP methods:
 - reject the request with a 400 Bad Request including a ProblemDetails IE with:
 - the cause attribute set to INVALID_QUERY_PARAM;

- the `invalidParams` attribute indicating the unsupported query parameters;
- the `supportedFeatures` attribute listing the features supported by the NF Service Producer, if any, set as specified for HTTP responses in clause 6.6.2.

5.2.10 URL Encoding of data

5.2.10.1 General

As indicated in IETF RFC 3986 [14], the URI syntax defines a set of characters (a subset of the URI allowed characters) as delimiters of syntax components; those characters are called "reserved" and should not be used in URI fields intended to convey generic data (e.g., in the value part of a query parameter, or in the URI path segments), since this would interfere with the original meaning (syntax) of those reserved characters.

In addition, HTTP/2 request body parts encoded with media type "application/x-www-form-urlencoded" shall also escape reserved and unsafe characters, as described in OpenAPI Specification [9].

5.2.10.2 URL Encoding of URI components

When a URI is composed in the 3GPP 5G APIs, the different components (e.g., path segments, values of query parameters, etc.) shall percent-encode the following "reserved" characters (see IETF RFC 3986 [14], section 2.1):

- EXCLAMATION MARK (U+0021): !
- NUMBER SIGN (U+0023): #
- DOLLAR SIGN (U+0024): \$
- AMPERSAND (U+0026): &
- APOSTROPHE (U+0027): '
- LEFT PARENTHESIS (U+0028): (
- RIGHT PARENTHESIS (U+0029):)
- ASTERISK (U+002A): *
- PLUS SIGN (U+002B): +
- COMMA (U+002C): ,
- SOLIDUS (U+002F): /
- COLON (U+003A): :
- SEMICOLON (U+003B): ;
- EQUALS SIGN (U+003D): =
- QUESTION MARK (U+003F): ?
- COMMERCIAL AT (U+0040): @
- LEFT SQUARE BRACKET (U+005B): [
- RIGHT SQUARE BRACKET (U+005D):]

The following characters (not listed as "reserved" in IETF RFC 3986 [14]) shall be percent-encoded:

- QUOTATION MARK (U+0022): "
- PERCENT SIGN (U+0025): %
- LEFT CURLY BRACKET (U+007B): {

- RIGHT CURLY BRACKET (U+007D): }

NOTE: Implementations complying with an earlier version of the specification can send curly brackets in URI without percent-encoding. Implementations can support receiving curly brackets with and without percent-encoding.

SPACE (U+0020) character shall be escaped, either by percent-encoding it (as %20), or by replacing it with character PLUS SIGN (U+002B).

The encoding of query parameters consisting of arrays of strings shall follow the guidelines indicated in TS 29.501 [5], clause 5.3.13, for the escaping of the COMMA (U+002C) characters.

In addition, implementations may percent-encode other characters.

The receiving entity shall percent-decode the received URI as specified in IETF RFC 3986 [14], section 2.4.

Percent-encoding of reserved characters in the URI fields as described in this clause shall also apply to JSON attributes defined as URI and to HTTP header parameters whose ABNF definition uses production rules defined as URI or path-absolute (e.g. prefix parameter of the 3gpp-Sbi-Target-apiRoot header).

5.2.10.3 URL Encoding of HTTP/2 request bodies

When composing an HTTP/2 request body with media type "application/x-www-form-urlencoded", the OpenAPI Specification [9] requires that the encoding shall follow IETF RFC 1866 [48], section 8.2.1, which indicates:

- a) the "reserved" character set described in IETF RFC 1738 [49], section 2.2, shall be percent-encoded:

- AMPERSAND (U+0026): &
- SOLIDUS (U+002F): /
- COLON (U+003A): :
- SEMICOLON (U+003B): ;
- EQUALS SIGN (U+003D): =
- QUESTION MARK (U+003F): ?
- COMMERCIAL AT (U+0040): @

- b) SPACE (U+0020) character shall be escaped by replacing it with character PLUS SIGN (U+002B).

The following characters (not listed as "reserved" in IETF RFC 1738 [49]) shall be percent-encoded:

- QUOTATION MARK (U+0022): "
- PERCENT SIGN (U+0025): %
- COMMA (U+002C): ,
- LEFT SQUARE BRACKET (U+005B): [
- RIGHT SQUARE BRACKET (U+005D):]
- LEFT CURLY BRACKET (U+007B): {
- RIGHT CURLY BRACKET (U+007D): }

In addition, implementations may also percent-encode any of the characters listed in clause 5.2.10.2.

5.3 Transport Protocol

The Transmission Control Protocol as described in IETF RFC 793 [6] shall be used as transport protocol as required by HTTP/2 (see IETF RFC 9113 [7]).

NOTE: When using TCP as the transport protocol, an HTTP/2 connection is mapped to a TCP connection.

If a Network Function does not register any port number to the NRF then it shall be prepared to receive connections on default port numbers, i.e. TCP port 80 for "http" URIs and TCP port 443 for "https" URIs as specified in IETF RFC 9113 [7].

5.4 Serialization Protocol

The JavaScript Object Notation (JSON) format as described in IETF RFC 8259 [10] shall be used as serialization protocol.

The content of JSON attributes of string type shall be encoded as UTF-8.

For transmitting large parts of opaque binary data along with JSON format, multipart messages shall be supported using:

- A multipart/related media type;
- 3gpp vendor specific content subtype; and
- Cross-referencing from the JSON content using the Content-ID field.

Use of multipart messages is documented in specific specifications.

5.5 Interface Definition Language

OpenAPI Specification [9] shall be used as Interface Definition Language (IDL) of SBI.

6 General Functionalities in Service Based Architecture

6.1 Routing Mechanisms

6.1.1 General

This clause specifies the generic routing mechanisms in the 5GC. Specific requirements to support Indirect communication are further defined in clause 6.10.

For HTTP message routing between Network Functions, the message routing mechanism as specified in clause 7 of IETF RFC 9110 [11] is almost followed with some differences due to the adoption of HTTP/2 and to some 5G system specificities.

NOTE: The term "inbound" are defined in clause 3.7 of IETF RFC 9110 [11]. It describes a directional requirement in relation to the request route: "inbound" means toward the origin server.

6.1.2 Identifying a target resource

The target resource is identified by a target URI (e.g. a Resource URI, a Custom operation URI or a Callback URI as defined in clause 4.4 of TS 29.501 [5]).

6.1.3 Connecting inbound

If the request is not satisfied by a local cache, then the client shall connect to an authority server for the target resource or to a proxy.

If a proxy is applicable for the target URI, the client connects inbound by establishing (or reusing) a connection to that proxy as defined in clause 7.3.2 of IETF RFC 9110 [11]. For connecting inbound to an authority not in the same PLMN, the client connects to the Security Edge Protection Proxy.

If no proxy is applicable, then the client connects directly to an authority server for the target resource as defined in IETF RFC 9110 [11].

6.1.4 Pseudo-header setting

6.1.4.1 General

Once an inbound connection is obtained, the client sends a request message over the wire. The message starts with a HEADERS frame containing the Pseudo-Header Fields identifying the request target. The ":method" pseudo-header is always present.

When sending a request directly to an origin server or to a proxy, other than a CONNECT or server-wide OPTIONS request, a client shall include the below pseudo-headers:

- ":scheme".
- ":authority".
- ":path" includes the path and query components of the target URI. The path includes the optional deployment-specific string of the Resource URI or Custom operation URI "apiRoot" part.

When sending a CONNECT request to a proxy, a client shall include the ":authority" pseudo-header. The ":scheme" and ":path" ones shall be absent.

When sending a server-wide OPTIONS request to an origin server or to a proxy, a client shall include the below pseudo-headers:

- ":scheme".
- ":authority".
- ":path" set with the value "*".

6.1.4.2 Routing within a PLMN

For HTTP/2 request messages where the target URI authority component designates an origin server in the same PLMN as the client, the ":authority" HTTP/2 pseudo-header field shall be set to:

" :authority" = uri-host [":" port] as specified in clause 8.3.1 of IETF RFC 9113 [7], excluding the [userinfo "@"] information as specified in clause 3.2 of IETF RFC 3986 [14].

Where the uri-host shall be:

- FQDN of the target NF service; or
- IP address of the target NF service

The FQDN of the target NF service need not contain the PLMN identifier.

6.1.4.3 Routing across PLMN

6.1.4.3.1 General

This clause specifies how to route requests across PLMNs when the target NF instance from the target PLMN has been selected by the source PLMN (i.e. PLMN of the NF service consumer). Specific requirements to support indirect communication with or without delegated discovery between different PLMNs with NF selection at target PLMN are defined in clause 6.10.12.

In order to reach the correct target NF service in the right PLMN and for HTTP/2 request messages where the target URI authority component designates an origin server not in the same PLMN as the client, the ":authority" HTTP/2 pseudo-header shall contain the FQDN including the PLMN ID.

The ":authority" pseudo-header field in the HTTP/2 request message shall be set to:

":authority" = uri-host [":" port] as specified in clause 8.3.1 of IETF RFC 9113 [7], excluding the [userinfo "@"] information as specified in clause 3.2 of IETF RFC 3986 [14].

Where the uri-host shall be:

- FQDN of the target NF service or the FQDN (authority) part of a callback URI or a specified link relation

The FQDN of the target NF service or the FQDN (authority) part of a callback URI or a specified link relation shall contain the PLMN identifier.

The format of the FQDN of target NF service is specified in clause 28.5 of TS 23.003 [15].

To allow for TLS protection between the SEPP and Network Functions within a PLMN, the SEPP shall support:

- TLS wildcard certificate for its domain name and generation of telescopic FQDN, as specified in clause 13.1 of TS 33.501 [17] and in clause 6.1.4.3.2; and
- forwarding HTTP requests originated by NFs within the SEPP's PLMN towards the remote PLMN using the 3gpp-Sbi-Target-apiRoot header as specified in clause 6.1.4.3.3.

NOTE: Whether the SEPP and NFs within the SEPP's PLMN use telescopic FQDN or the 3gpp-Sbi-Target-apiRoot header is based on PLMN operator's policy and is independent from the method supported and used in the remote PLMN.

Both solutions for TLS protection between the SEPP and Network Functions within a PLMN may be used concurrently in a PLMN, e.g. in the transient phase where not all NFs of the PLMN have been upgraded to support the 3gpp-Sbi-Target-apiRoot header but when the PLMN operator would like to use the solution based on the 3gpp-Sbi-Target-apiRoot header with upgraded NFs. In this case, the SEPP should skip converting URIs into telescopic FQDNs (and use the solution based on 3gpp-Sbi-Target-apiRoot header) in:

- HTTP responses received from the remote PLMN (e.g. including the FQDN of the target NF service) when the corresponding HTTP request contains a 3gpp-Sbi-Target-apiRoot header;
- HTTP requests received from the remote PLMN (e.g. including callback URIs) using SEPP policies based on the target URI (i.e. target FQDN).

6.1.4.3.2 Use of telescopic FQDN between NFs and SEPP within a PLMN

When using TLS wildcard certificate and telescopic FQDN between the SEPP and NFs within the SEPP's PLMN, the SEPP on the HTTP/2 client side shall form the telescopic FQDN, as specified in TS 23.003 [15], for the following cases:

- FQDN of the target NF service in HPLMN is modified into a telescopic FQDN by the SEPP in the VPLMN;
- FQDN of the target NF service in VPLMN is modified into a telescopic FQDN by the SEPP in the HPLMN;
- FQDN (authority) part of callback URI of NF service resources in VPLMN is modified into a telescopic FQDN by the SEPP in the HPLMN;
- FQDN (authority) part of callback URI of NF service resources in HPLMN is modified into a telescopic FQDN by the SEPP in the VPLMN;
- FQDN (authority) part of link relation URI of NF service resources in VPLMN is modified into a telescopic FQDN by the SEPP in the HPLMN;
- FQDN (authority) part of link relation URI of NF service resources in HPLMN is modified into a telescopic FQDN by the SEPP in the VPLMN.

6.1.4.3.3 Use of 3gpp-Sbi-Target-apiRoot between NFs and SEPP within a PLMN

When using the 3gpp-Sbi-Target-apiRoot header between the SEPP and NFs within the SEPP's PLMN, HTTP requests between the NFs and the SEPP shall be routed as specified in clause 6.10.2 for indirect communications, with the SEPP taking the role of the SCP.

When sending an HTTP request targeting a URI with an authority of a remote PLMN, NFs shall include the 3gpp-Sbi-Target-apiRoot header in the HTTP request, containing the apiRoot of the target URI in the remote PLMN, and shall set the apiRoot in the request URI to the apiRoot of the SEPP (or to the apiRoot of the SCP if the communication between the NF and SEPP goes through an SCP). The apiRoot of the SEPP (or SCP) may include an optional deployment-specific string of the SEPP (or SCP).

An SCP that receives an HTTP request targeting a URI with an authority of a remote PLMN shall route the HTTP request towards the SEPP as specified in clause 6.10.2 for indirect communications, i.e. the SCP shall forward the 3gpp-Sbi-Target-apiRoot header in the HTTP request it forwards to the SEPP, containing the apiRoot of the target URI in the remote PLMN, and it shall set the apiRoot in the request URI to the apiRoot of the SEPP.

If the SEPP receives an HTTP request from a NF with a request URI containing a telescopic FQDN and with a 3gpp-Sbi-Target-apiRoot header, the SEPP shall ignore the 3gpp-Sbi-Target-apiRoot header and route the request using the telescopic FQDN.

NOTE 1: This is to address the case of a potentially malicious or misbehaving NF that would include the 3gpp-Sbi-Target-apiRoot header and a request URI containing a telescopic FQDN when communicating with the SEPP.

NOTE 2: This solution does not require the SEPP to support TLS wildcard certificate for its domain name, nor the SEPP to modify URI attributes in HTTP request and response contents with telescopic FQDNs.

NOTE 3: The communication between the NF and SEPP can be direct or go through an SCP.

6.1.4.3.4 Routing between SEPPs

The 3gpp-Sbi-Target-apiRoot header shall not be used between SEPPs if PRINS security is negotiated between the SEPPs. The apiRoot of the Request URI of the HTTP request encapsulating the protected message shall be set to the apiRoot of the remote SEPP. See clause 5.3.2.4 of TS 29.573 [27].

If TLS security is negotiated between the SEPPs and at least one SEPP does not indicate support of the 3gpp-Sbi-Target-apiRoot header when negotiating the security policy, the SEPP shall use a pre-established TLS connection towards the other SEPP to forward the HTTP/2 messages sent by the NF service producers and NF service consumers, as is without reformatting. Additionally,

- if the NF uses the 3gpp-Sbi-Target-apiRoot HTTP header in the HTTP Request to convey the target apiRoot to the sending SEPP, the sending SEPP shall remove the 3gpp-Sbi-Target-apiRoot header and set the apiRoot of the request URI it forwards on the N32-f interface to the apiRoot received in the 3gpp-Sbi-Target-apiRoot header from the HTTP client;
- if the NF uses a telescopic FQDN in the HTTP Request to convey the target apiRoot to the sending SEPP, or if TLS is not used between the NF and the sending SEPP, the sending SEPP shall set the apiRoot of the Request URI in the HTTP Request towards the remote SEPP to the apiRoot of the target NF derived from the telescopic FQDN or from the request URI respectively.

If TLS security is negotiated between the SEPPs and both SEPPs indicate support of the 3gpp-Sbi-Target-apiRoot header when negotiating the security policy, HTTPS shall be used to forward messages between SEPPs. The sending SEPP shall replace the apiRoot of the Request URI in the HTTP Request with the apiRoot of the receiving SEPP before forwarding the HTTP Request on the N32 interface. Additionally,

- if the NF uses the 3gpp-Sbi-Target-apiRoot HTTP header in the HTTP Request to convey the target apiRoot to the sending SEPP, the sending SEPP shall forward the 3gpp-Sbi-Target-apiRoot header unmodified in the HTTP request towards the remote SEPP;
- if the NF uses a telescopic FQDN in the HTTP Request to convey the target apiRoot to the sending SEPP, or if TLS is not used between the NF and the sending SEPP, the sending SEPP shall insert the 3gpp-Sbi-Target-apiRoot header in the HTTP request towards the remote SEPP and set it to the apiRoot of the target NF derived from the telescopic FQDN or from the request URI respectively.

NOTE: Rel-15 compliant NFs and SEPP do not support the 3gpp-Sbi-Target-apiRoot header.

6.1.5 Host header

Clients that generate HTTP/2 requests shall use the ":authority" pseudo-header field instead of the Host header field.

6.1.6 Message forwarding

An HTTP/2 proxy shall use the ":authority" pseudo-header field to connect inbound to the origin server or another proxy if the request cannot be satisfied by the proxy cache.

An HTTP/2 proxy may also use other headers and/or content to connect inbound to the origin server or another proxy if the request cannot be satisfied by the proxy cache.

6.2 Server-Initiated Communication

6.2.1 General

The Subscribe-Notify service operations shall be supported between NFs as specified in clause 7.1.2 of TS 23.501 [3].

Subscribe-Notify service operations require bidirectional communication between the NFs when the server needs to initiate communication with the client.

Subscribe-Notify service operations shall be supported with two TCP connections, one per direction, as follows:

- NF service consumer acts as an HTTP client and NF service producer acts as an HTTP server when NF service consumer subscribes to NF service producer's notifications;
- NF service producer acts as an HTTP client and NF service consumer acts as an HTTP server when NF service producer delivers notifications to NF service consumer.

As described in TS 23.501 [3], the Subscribe-Notify interaction requires the NF Service Consumer to provide a "notification endpoint" and a "notification correlation ID"; those are also called "callback URI" in the context of the present Technical Specification, and the authority of the "callback URI" is the HTTP endpoint where the notifications shall be delivered by the NF Service Producer. As indicated in TS 23.501 [3], the interaction between NF Service Consumer and NF Service Producer may not occur, or may occur via interactions on a different service or API, prior to the notification sent by the NF Service Producer (e.g. for Implicit Subscriptions, or for Default Notifications); in that case, the notification is called "standalone notification", and shall be specified as described in TS 29.501 [5], clause 5.3.7.

For notifications sent in Direct Communication scenarios, if the authority of the callback URI contains an FQDN, the NF Service Producer shall use DNS as resolution mechanism in order to setup the TCP connection with the NF Service Consumer; for notifications sent in Indirect Communication scenarios, see clause 6.10.7.

6.2.2 Subscription on behalf of NF Service Consumer

When an NF service consumer subscribes to an intermediate NF for events which may be detected and reported directly by target NF (e.g. an NEF subscribes to Location Reporting event at AMF via UDM and AMF directly reports to the NEF), the NF service consumer may include the "3gpp-Sbi-Consumer-Info" header in the subscription creation request to indicate the supported API versions, features and accepted encodings of the service on the target NF.

When subscribing to the target NF and requiring the target NF to directly report to NF service consumer, the intermediate NF shall invoke the highest API version at the target NF which is supported by the NF service consumer (if indicated) and the intermediate NF. The intermediate NF shall include all received "3gpp-Sbi-Consumer-Info" header(s) in the subscription creation request sent to the target NF.

If the target NF received this header in event subscription creation, the target NF shall generate the request body according to the supported feature(s) and accepted encodings of the NF service consumer for notifications directly sent to the NF service consumer.

Based on operator policy, the NF service consumer may provide a default inter-PLMN or intra-PLMN callback URI in a subscription request to the intermediate NF.

The NF Service Consumer may also include, for each provided service, the following information in the "3gpp-Sbi-Consumer-Info" header(s):

- the `intraPlmnCallbackRoot` parameter containing the callback root for receiving intra-PLMN notifications, and

- the `interPlmnCallbackRoot` parameter containing the callback root for receiving inter-PLMN notifications.

Upon receiving a subscription request including the above information in the "3gpp-Sbi-Consumer-Info" header, the intermediate NF should check if the target NF is in VPLMN or HPLMN and adapt the callback Root of the callback URI according to the information received from the NF service consumer. For instance, if the NF service consumer included an inter-PLMN callback URI in the subscription request:

- if the target NF is in HPLMN, then the intermediate NF should replace the callback Root of the callback URI in the subscription request with the provided `intraPlmnCallbackRoot` while sending the subscription creation request to the target NF; and
- if the target NF is in VPLMN, then the intermediate NF shall not change the notification/callback URI.

In either case, the Intermediate NF should then forward the "3gpp-Sbi-Consumer-Info" header in the subscription creation request sent to the target NF.

If the intermediate NF expects to receive notifications from the target NF (e.g. the UDM may provide the callback URI to receive subscription change event notifications from the AMF when subscribing to AMF events on behalf of the NEF), to enable the target NF to use, e.g., the proper callback URI, the supported features, the intermediate NF may provide the "3gpp-Sbi-Consumer-Info" header for itself in the subscription creation request to the target NF. If it does so, the intermediate NF shall include the intermediate NF indication parameter in the header.

When the target NF is an AMF, the source AMF should forward the information in the received "3gpp-Sbi-Consumer-Info" header(s) to the target AMF during inter-AMF mobility. If the target AMF received `intraPlmnCallbackRoot` and `interPlmnCallbackRoot` parameters in the "3gpp-Sbi-Consumer-Info" header information from the source AMF, the target AMF should determine the PLMN of the NF Service Consumer and adapt the callback root of the callback URI correspondingly.

6.2.3 Notification error handling

The following requirements apply unless specified otherwise for a given API,

An NF Service Producer that sends a notification request should consider that the subscription is no longer valid and terminate the subscription, if it receives any of the following response codes and application errors:

- "400 Bad Request" with the application error "RESOURCE_CONTEXT_NOT_FOUND"; or
- "404 Not Found" with the application error "SUBSCRIPTION_NOT_FOUND".

An NF Service Producer should not keep on sending notification requests to an NF service consumer and may consider that the subscription is no longer valid and terminate the subscription, if it receives one or more "404 Not Found" responses without application errors or with other application errors.

6.3 Load Control

6.3.1 General

Load control enables an NF Service Producer to signal its load information to NF Service Consumers, either via the NRF (as defined in clause 6.3.2) or directly to the NF Service Consumer (as defined in clause 6.3.3). The load information reflects the operating status of the resources of the NF Service Producer.

Load control allows for better balancing of the load across NF Service Producers, so as to attempt to prevent their overload in first place (preventive action). Load control does not trigger overload mitigation actions, even if the NF Service Producer reports a high load.

NOTE: the load information can be used along similar principles as those described for node-level load control in clause 4A.2 in TS 29.303 [39], but with the priority and capacity parameters of candidate NFs obtained from the NRF.

6.3.2 Load Control based on load signalled via the NRF

This clause specifies details of the Load Control based on load signalled via the NRF solution.

During NF discovery procedures (see clause 4.17.4 and 4.17.5 of TS 23.502 [4]), the NRF may provide the NF instance and/or the NF service instance information with the NF capacity information advertised during NF Service Registration and/or NF Service Update procedures (see clause 4.17.1 and 4.17.2 of TS 23.502 [4] and clause 6.2.6 of TS 23.501 [3]). The NRF may also provide load information of the NF instance and/or the NF service instance in NF discovery response.

The NF service consumer that is discovering the NF service producer, may use the available information (e.g. NF capacity information, load information) to select the appropriate NF instance as specified in TS 29.510 [8].

6.3.3 Load Control based on LCI Header

6.3.3.1 General

This clause specifies details of the Load Control based on LCI Header solution (LC-H). The solution extends the Load Control based on load signalled via the NRF solution by enabling a direct exchange of the LCI between the NF Service Producer and NF Service Consumer.

Support for the LC-H solution is optional, but if the feature is supported, the requirements specified in the following clauses shall apply.

NOTE 1: Load control and overload control can be supported and activated independently in the network, based on operator policy.

An NF Service Producer that supports the LC-H feature shall signal its load information as further specified in this clause. An NF Service Consumer supporting the LC-H feature shall utilize the load information, for a given scope, that is received with the most recent timestamp from the NRF or from the NF Service Producer via direct signalling, to adaptively balance the load across the candidate NF Service Producers according to their effective load e.g. when creating a resource at an NF Service Producer.

NOTE 2: An NF Service Consumer supporting the LC-H feature can receive the load information without a timestamp from the NRF and an LCI (with a timestamp) from the NF Service Producer. It is an implementation matter how the NF Service Consumer determines which of these contains the most recent load information.

An SCP/SEPP that supports the LC-H feature may additionally piggyback its LCI with a scope set to the SCP FQDN/SEPP FQDN, in HTTP request or response messages, as further specified in this clause. An HTTP client supporting the LC-H feature shall utilize the load information of the SCP/SEPP, which is received with the most recent timestamp, to adaptively balance the load across the available SCPs/SEPPs to reach the HTTP server.

In scenarios with multiple SCPs or with SCP(s) and SEPPs between the NF service producer and the NF service consumer, an SCP or SEPP that receives in a service response or in a notification request an LCI with a scope set to an SCP or SEPP FQDN, shall remove this LCI header when forwarding the message to the next hop, and shall perform load control to adaptively balance the load across the available next hop SCPs/SEPPs for the subsequent service requests according to the received LCI information.

NOTE 3: An NF service consumer can only receive LCI with a scope set to an SCP or SEPP FQDN from its next hop SCP or SEPP.

The SCPs and SEPPs shall forward LCI headers with a scope set to NF Producer received in a service response or notification request when forwarding the message to the next hop. The NF consumer shall perform the load control to adaptively balance the load across the available NF Producers for the subsequent service requests according to the received LCI information.

A SEPP may advertise its own LCI information to a next hop NF or SCP in the same PLMN, and/or to the peer SEPP based on operator's policy. In the latter case, LCI may be advertised in N32-f signalling; when PRINS is used over N32-f, the LCI header for the SEPP FQDN shall be inserted in the outer N32-f message, i.e. not within the N32-f content.

6.3.3.2 Conveyance of Load Control Information

LCI shall be conveyed within the 3gpp-Sbi-Lci HTTP header. When conditions for generating an LCI are met, an NF Service Producer, SCP or SEPP shall include the 3gpp-Sbi-Lci header, or LCI header, see clause 5.2.3.2.10) to its peer entities (NF Service Consumers). The LCI header shall be piggybacked on a signalling message that is sent to the NF Service Consumer.

The NF Service Producer, SCP or SEPP shall send the 3gpp-Sbi-Lci header, regardless of whether the peer NF Service Consumer supports the feature (see clause 6.3.3.5). The header is ignored by the NF Service Consumer if the latter does not support the LC-H feature.

6.3.3.3 Frequency of Conveyance

How often or when the sender conveys the LCI is implementation specific. The sender shall ensure that new or updated Load Control Information is conveyed to the target receivers with an acceptable delay, such that the purpose of the information (i.e. the effective load balancing) is achieved.

Considering the processing requirement of the receiver of the LCI (e.g. handling of the new information), the sender should refrain from advertising every small variation (e.g. with the granularity of 1 or 2), in the Load Metric which does not result in useful improvement in NF service producer selection logic at the receiver. A larger variation in the Load Metric, e.g. 5 or more units, should be considered as reasonable enough for advertising the new Load Control Information.

6.3.3.4 Load Control Information

6.3.3.4.1 General Description

A NF Service Producer may include one or more LCI header(s) in a service response or in a notification/callback request message sent to a NF Service Consumer. An NF Service Producer may report LCI with different scopes, e.g.:

- to report LCIs for an NF service instance, an NF service set and/or an NF instance;
- to report LCIs at the level of an SMF (service) instance or SMF (service) set, and for specific S-NSSAI/DNNs;
- to report LCIs for different S-NSSAI/DNNs of an SMF (service) instance or SMF (service) set.

A NF Service Producer may also include LCI header(s) with different scopes in different messages, e.g. an SMF may report LCI for the SMF instance first, and then report LCI for both the SMF instance and for specific S-NSSAI/DNN(s), if S-NSSAI/DNN based load control is enabled.

An NF Service Consumer that receives LCI headers with different scopes, in the same message or in different messages, shall handle each LCI independently from each other. For instance, if an NF Service Consumer receives one LCI with the scope of an NF (Service) Set and then another LCI with the scope of an NF (Service) instance that pertains to the NF (Service) Set, the NF Service Consumer shall store the latter LCI and also consider that the former LCI is still valid for the NF (Service) Set.

For S-NSSAI/DNN based load control (see clause 6.3.3.4.2.2), when signalling LCI for an SMF (service) instance or an SMF (service) set in a message, the SMF shall always include the full set of load control information applicable to the SMF (service) instance or SMF (service) set, i.e. LCI for the SMF (service) instance or the SMF (service) set level and/or LCI for specific S-NSSAI/DNNs, even if only a subset of the LCI has changed; these LCIs shall contain the same Load Control Timestamp.

An SCP or SEPP may additionally include one LCI in a service request or response, or in a notification request or response, sent towards a NF Service Consumer or NF Service Producer.

Each LCI shall always include the Timestamp, Load Metric and Scope parameters (see clause 5.2.3.2.10 for the complete list of parameters).

6.3.3.4.2 Load Control Timestamp

The Timestamp parameter indicates the time when the LCI was generated. It shall be used by the receiver of the LCI to properly collate out-of-order LCI, e.g. due to HTTP/2 stream multiplexing, prioritization and flow control, and to determine whether the newly received load control information has changed compared to load control information previously received for the same scope.

The receiver shall overwrite any stored load control information of a peer NF, NF set, NF service, NF service set, SCP or SEPP (according to the scope of the new received LCI) with the newly received load control information, if the new load control information is more recent than the stored information. For instance, for S-NSSAI/DNN based load control, if the receiver had stored LCI for a peer SMF instance and LCI for a specific S-NSSAI/DNN of that SMF instance, it shall overwrite these LCIs with the new LCI received in a message carrying LCI for the same SMF instance.

If the newly received LCI has the same or an older Timestamp as the previously received LCI for the same scope (e.g. from the same NF, NF Set, NF Service, NF Service Set, SCP or SEPP), then the receiver shall discard the newly received LCI whilst continuing to apply the load control procedures according to the previously stored value.

NOTE: An NF Service Consumer can receive LCI for the same target scope from different NF service producers, when the scope of the LCI corresponds to an NF set or NF service set.

6.3.3.4.3 Load Metric

The Load Metric shall indicate the current load level for the scope of the LCI, e.g. current load level of the NF instance if the scope indicated in the LCI indicates an NF instance, as a percentage within the range of 0 to 100, where 0 means no or 0% load and 100 means maximum or 100% load reached (i.e. no further load is desirable). The computation of the load metric is implementation specific.

6.3.3.4.4 Scope of LCI

6.3.3.4.4.1 Introduction

The scope of LCI indicates the applicability of the LCI, i.e. it identifies the components of the LCI sender to which the LCI relates to.

The following clauses provide a detailed description of the parameters that define the scope of the LCI header.

6.3.3.4.4.2 Scope of LCI signalled by an NF Service Producer

6.3.3.4.4.2.1 General

The LCI sent by an NF Service Producer shall include one of the parameters defined in Table 6.3.3.4.4.2.1-1.

Table 6.3.3.4.4.2.1-1: Supported scopes for LCI signalled by an NF Service Producer

Parameter	Value	LCI scope (i.e. LCI applies to)	Examples
NF-Instance	NF Instance ID	All services of the NF instance identified by the NF Instance ID.	NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8
NF-Set	NF Set ID	All services of all NF instances of the NF set identified by the NF Set ID.	NF-Set: set1.udmset.5gc.mnc012.mcc345
NF-Service-Instance (and NF-Inst)	NF Service Instance ID (and NF Instance ID)	The service instance identified by the NF Service Instance ID and the NF instance Id (if available) or the last known NF instance ID.	NF-Service-Instance: serv1.smf1; NF-Inst: 54804518-4191-46b3-955c-ac631f953ed8
NF-Service-Set	NF Service Set ID	All service instances of the NF service set identified by the NF service set ID.	NF-Service-Set: setxyz.snsmf-pdusection.nfi54804518-4191-46b3-955c-ac631f953ed8.5gc.mnc012.mcc345

If an NF Service Consumer receives more than one LCI with overlapping scopes, i.e. one with NF (service) instance scope and another with NF (service) Set scope, the NF Service Consumer should perform load balancing considering the LCI received with the finer scope for each candidate NF instance or NF service instance (i.e. in this example the load of the NF (service) instance).

6.3.3.4.4.2.2 Additional scope parameters for S-NSSAI/DNN based load control by SMF

It is optional for the SMF to support S-NSSAI/DNN based load control. When supported, the following requirements shall apply.

S-NSSAI/DNN level load control refers to advertising of the load information at S-NSSAI and DNN level granularity and selection of the target SMF service instance based on this information. It helps to achieve an evenly load balanced network at S-NSSAI/DNN granularity by the use of the dynamic load information provided within the Load Control Information with the S-NSSAI/DNN scope. Only an SMF may advertise S-NSSAI/DNN level load information.

NOTE 1: When all the resources of an SMF (service) instance are available for all the S-NSSAI/DNNs served by that SMF (service) instance, load control at SMF (service) set or SMF (service) instance level is exactly the same as S-NSSAI/DNN level overload information of that SMF, for each of its S-NSSAIs/DNNs, and hence, performing load control at SMF (service) set or SMF (service) instance level is sufficient.

The "Load Metric" shall indicate the current resource utilization for the indicated S-NSSAI/DNN(s), as a percentage, as compared to the total resources configured for the indicated S-NSSAI/DNNs at the SMF.

When performing S-NSSAI/DNN based load control, the LCI scope shall indicate, in addition to either an NF-Instance, NF-Set, NF-Service-Instance or NF-Service-Set (see Table 6.3.3.4.2.1-1), the combinations of S-NSSAI and DNN for which the LCI sender wants to advertise the load information using the following parameters:

- the S-NSSAI parameter, indicating one or more S-NSSAI values; and
- the DNN parameter, indicating one or more DNN values from the indicated S-NSSAI(s).

NOTE 2: It is not allowed to report LCI for a DNN only or for an S-NSSAI only.

See Table 6.3.3.4.4.2.1-1.

Table 6.3.3.4.4.2.1-1: Additional scope parameters for S-NSSAI/DNN based load control by SMF

Parameter	Value	LCI scope (i.e. LCI applies to)	Examples
S-NSSAI	One or more S-NSSAI values	DNN(s) from indicated S-NSSAI(s), for the service(s) of NF instance(s) as defined in Table 6.3.3.4.4.2.1-1.	S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN: internet.mnc012.mcc345.gprs
DNN	One or more DNN values		
NOTE 1: Both the S-NSSAI and DNN parameters shall be present. The S-NSSAI and DNN parameters shall be provided with either the NF-Instance, NF-Set, NF-Service-Instance or NF-Service-Set parameter (see Table 6.3.3.4.4.2.1-1).			
NOTE 2: The S-NSSAI parameter in the Example corresponds to the JSON encoding: {"sst": 1, "sd": "A08923"} (see clause 5.2.3.1).			

An SMF shall advertise S-NSSAI/DNN based load control for at most 10 DNNs.

NOTE 3: Considering various aspects such as the processing and storage requirements at the overloaded SMF entity and the receiver, the number of important DNNs for which overload control advertisement could be necessary, interoperability between the NFs of different vendors, it was chosen to define a limit on the maximum number of DNNs for advertising the load control information.

The SMF may advertise load information for different DNNs of one or more S-NSSAIs in a single LCI header (if the same LCI information, e.g. load metric or relative capacity, applies to all the DNNs of the S-NSSAI(s)) or in up to 10 LCI headers (if different LCI information needs to be advertised for different DNNs).

6.3.3.4.4.3 Scope of LCI signalled by an SCP

The LCI sent by an SCP shall include one of the parameters defined in Table 6.3.3.4.4.3-1.

Table 6.3.3.4.4.3-1: Supported scopes for LCI signalled by an SCP

Parameter	Value	LCI scope (i.e. LCI applies to)	Examples
SCP-FQDN	SCP FQDN	SCP identified by the SCP FQDN	SCP-FQDN: scp1.example.com

6.3.3.4.4.4 Scope of LCI signalled by an SEPP

The LCI sent by an SEPP shall include one of the parameters defined in Table 6.3.3.4.4.4-1.

Table 6.3.3.4.4-1: Supported scopes for LCI signalled by an SEPP

Parameter	Value	LCI scope (i.e. LCI applies to)	Examples
SEPP-FQDN	SEPP FQDN	SEPP identified by the SEPP FQDN	SEPP-FQDN: sepp1.example.com

6.3.3.4.5 S-NSSAI/DNN Relative Capacity

When applying S-NSSAI/DNN based load control (see clause 6.3.3.4.2.2), the LCI shall include the S-NSSAI/DNN relative capacity indicating the resources configured for the combinations of S-NSSAIs and DNNs reported in the LCI, compared to the total resources configured at the SMF (service) instance or SMF (service) set, as a percentage.

This parameter enables the NF selecting an SMF service instance to determine the available resources for a given S-NSSAI/DNN for different candidate SMF service instances (considering the static capacity of the SMF service instance, the S-NSSAI/DNN relative capacity and the load of the S-NSSAI/DNN).

6.3.3.5 LC-H feature support

6.3.3.5.1 Indicating supports for the LC-H feature

When registering with the NRF (NFRegister) or updating the NRF (NFUpdate), an NF that supports the LC-H feature shall indicate the feature support (see clause 6.1.6.2.2 in TS 29.510 [8]).

When an NF Service Consumer queries an NRF (NFDiscover) to discover services offered by NF Service Producers, the NRF shall indicate to the NF Service Consumer, if the NF Service Producers support the LC-H feature (see clause 6.2.6.2.3 in TS 29.510 [8]).

6.3.3.5.2 Utilizing the LC-H feature support indication

When selecting an NF Service Producer that supports the LC-H feature, the NF Service Consumer should not subscribe at the NRF to notifications triggered by the changes in the load of the selected NF Service Producer (see clause 5.2.2.5 in TS 29.510 [8]).

6.4 Overload Control

6.4.1 General

Service Based Interfaces use HTTP/2 over TCP for communication between the NF Services. TCP provides transport level congestion control mechanisms as specified in IETF RFC 5681 [16], which may be used for congestion control between two TCP endpoints (i.e., hop by hop). HTTP/2 also provides flow control mechanisms and limitation of stream concurrency that may be configured for connection level congestion control, as specified in IETF RFC 9113 [7].

In addition to TCP and HTTP/2 congestion control mechanisms, the following end to end application-level overload control mechanisms are defined.

Overload control enables an NF Service Producer, an NF Service Consumer, an SCP or an SEPP becoming or being overloaded to gracefully reduce its incoming signalling load, by instructing NF Service Consumers to reduce sending service requests or by instructing NF Service Producers to reduce sending notification requests respectively, according to its available signalling capacity to successfully process the requests. An NF Service Producer, NF Service Consumer, SCP or an SEPP is in overload when it operates over its signalling capacity.

When being instructed by a NF Service Consumer to apply overload control, the NF Service Producer shall perform the signaling reduction towards the NF Service Consumer only for the notifications or callback requests according to the overload scope, and not for any NF services which may be produced by the same NF (for which separate OCI may be advertised by the NF when acting as NF producer), even when the overload scope is on NF Instance level or NF Set level.

Overload control aims at shedding the incoming traffic as close to the traffic source as possible generally when an overload has occurred (reactive action), so to avoid spreading the problem inside the network and to avoid using resources of intermediate entities in the network for signalling that cannot anyhow be served by the overloaded entity.

Overload control should continue to allow for preferential treatment of priority users (e.g. MPS) and emergency services.

Overload control may be performed based on HTTP status codes returned in HTTP responses (as defined in clause 6.4.2) or based on Overload Control Information (OCI) signalled in HTTP request or response (as defined in clause 6.4.3).

The NF that performs overload control enforcement may either reject a fraction of request messages, or redirect some request messages towards an alternative NF if possible, to reduce sending HTTP requests towards an overloaded NF. (see clause 6.4.3.5).

6.4.2 Overload Control based on HTTP status codes

6.4.2.1 General

Overload control based on HTTP status code shall be supported per NF service / API according to the principles defined in this clause.

An NF Service Producer may mitigate a potential overload status by sending the NF Service Consumer the following HTTP status codes as a response to requests received during, or close to reaching, an overload situation:

- 503 Service Unavailable;
- 429 Too Many Requests; or
- 307 Temporary Redirect

The first 2 status codes (503 and 429) are intended to inform the NF Service Consumer that the server cannot handle the current received traffic rate, so it shall abate the traffic sent to the NF Service Producer by throttling part of this traffic locally at the NF Service Consumer, or diverting it to an alternative destination (another NF Service Producer where an alternative resource exists) that is not overloaded. If possible, traffic diversion shall always be preferred to throttling; the result of the throttling is a permanent rejection of the transaction.

If the client needs to abate a certain part of the available traffic, it shall do it based on the determined priority of each message.

Depending on regional/national requirements and network operator policy, requests related to priority traffic (e.g. MPS) and emergency shall be the last to be throttled by the client, and shall be exempted from throttling due to overload control up to the point where the required traffic reduction cannot be achieved without throttling the priority requests.

When an NF Service Consumer (e.g. AMF) issues a service request to an NF Service Producer (e.g. SMF), but the NF Service Producer cannot fulfil the request due to receiving a 503 or 429 status code from another NF Service Producer (e.g. PCF), the former NF Service Producer shall reject the service request from the NF Service Consumer with a 502 Bad Gateway HTTP error response including the "problemDetails" with the "cause" attribute set to "INBOUND_SERVER_ERROR". As an exception, the 503 or 429 status code shall be relayed by a V-SMF/I-SMF between AMF and H-SMF/SMF with the remoteError attribute set to "true" as specified in clause 6.1.7.1 of TS 29.502 [28].

The last status code (307) is intended to inform the NF Service Consumer about the availability of other endpoints where the service offered by the NF Service Producer is available, so the NF Service Consumer does not need to discard traffic locally.

6.4.2.2 HTTP Status Code "503 Service Unavailable"

This status code should be sent when the NF Service Producer undergoes an overload situation, and it needs to reject HTTP requests. The NF Service Producer may include detailed information about its status in the ProblemDetails JSON element, in the HTTP response body. Also, the HTTP header field "Retry-After" may be added in the response to convey an estimated time (in number of seconds) for the recovery of the service.

As for all 5xx status codes, this indicates a server-related issue (not limited to a specific client, or HTTP method), and it indicates that the server is incapable of performing the request.

Upon receipt of a "503 Service Unavailable" status code, the NF Service Consumer shall monitor the amount of rejected and timed-out traffic, in comparison to the accepted traffic by the NF Service Producer, and it shall abate (by diversion or throttling) the traffic sent to the NF Service Producer in such a way that the rate between accepted and rejected traffic improves with time, and eventually reaches a situation where the server accepts all requests once the overload status ceases at the server. The mechanism to achieve this is implementation-specific; Annex A contains a description of an example algorithm based on "adaptive throttling" of the traffic sent by the NF Service Consumer towards an NF Service Producer.

6.4.2.3 HTTP Status Code "429 Too Many Requests"

This status code may be sent, if supported by the server, when the NF Service Producer detects that NF Service Consumers are sending excessive traffic which, if continued over time, may lead to (or may increase) an overload situation in the NF Service Producer.

The HTTP header field "Retry-After" may be added in the response to indicate how long the NF Service Consumer has to wait before making a new request.

As for all 4xx status codes, this indicates a client-related issue (not limited to a specific HTTP method), and it indicates that the client seems to be misbehaving.

6.4.2.4 HTTP Status Code "307 Temporary Redirect"

This status code should be sent when the NF Service Producer decides to redirect HTTP requests to another less loaded server, or HTTP/2 end point, to offload some part of the incoming traffic, with the goal to avoid entering (or to mitigate) an overload situation. The NF Service Producer shall not use it if it does not know the load status of the alternative server.

How the NF Service Producer becomes aware of the load levels of other servers or HTTP/2 end points is deployment-specific, and out of the scope of this specification. The URI for the temporary redirection shall be given by the Location header field of the response.

This status code should also be sent when the SCP or SEPP decides to redirect HTTP requests to another less loaded SCP or SEPP to offload some part of the incoming traffic if it knows the load status of the alternative SCP or SEPP.

As for all 3xx status codes (redirection), this indicates a client-related action; the client shall be responsible of the detection of infinite redirection loops.

6.4.3 Overload Control based on OCI Header

6.4.3.1 General

This clause specifies details of the Overload Control based on OCI Header (OLC-H) solution. The solution is independent from the Overload Control based on HTTP status codes solution.

Support of OLC-H is optional, but if the feature is supported, the requirements specified in the following clauses shall apply.

Overload conditions are detected by an NF Service Producer/Consumer when the number of incoming service requests exceeds the maximum number of messages supported by the receiving entity, e.g. when the internally available resources of the NF Service Producer/Consumer, such as processing power or memory, are not sufficient to serve the number of incoming requests. How an NF Service Producer/Consumer identifies that it is overloaded is implementation specific.

When an NF Service Producer/Consumer reaches an implementation dependent overload threshold, the NF Service Producer/Consumer shall convey the Overload Control Information (OCI, see clause 6.4.3.4) to its peer entity (Consumer or Producer, respectively). Based on the received OCI, the peer shall adjust the signaling it sends to the overloaded entity according to the OCI as specified in clause 6.4.3.5. The OCI is piggybacked in HTTP request or response messages such that the exchange of the OCI does not trigger extra signaling.

An SCP/SEPP experiencing an overload may additionally piggyback OCI with a scope set to the SCP FQDN/SEPP FQDN in HTTP request or response messages, to adapt the signaling traffic sent towards the SCP/SEPP.

NOTE 1: Overload control and load control can be supported and activated independently in the network, based on operator policy.

In scenarios with multiple SCPs or with SCP(s) and SEPPs between the NF service producer and the NF service consumer, an SCP or SEPP that receives in an HTTP request or response an OCI with a scope set to an SCP or SEPP FQDN, shall remove this OCI header when forwarding the message to the next hop, and shall perform overload control to reduce sending subsequent service requests or notification requests to the next hop overloaded SCP or SEPP according to the received OCI information.

NOTE 2: An NF service consumer can only receive OCI with a scope set to an SCP or SEPP FQDN from its next hop SCP or SEPP.

The SCPs and SEPPs shall forward OCI headers with a scope set to NF Service Producer or NF Service Consumer received in an HTTP request or response when forwarding the message to the next hop. The NF consumer shall perform overload control to reduce sending subsequent service requests to the overloaded NF Service Producer according to the received OCI information. The NF Service Producer shall perform overload control to reduce sending subsequent notification requests to the overloaded NF Service Consumer according to the received OCI information.

A SEPP may advertise its own OCI information to a next hop NF or SCP in the same PLMN, and/or to the peer SEPP based on operator's policy. In the latter case, OCI may be advertised in N32-f signalling; when PRINS is used over N32-f, the OCI header for the SEPP FQDN shall be inserted in the outer N32-f message, i.e. not within the N32-f content.

6.4.3.2 Conveyance of Overload Control Information

OCI shall be conveyed within the 3gpp-Sbi-Oci HTTP header. When an NF Service Producer/Consumer/SCP/SEPP detects overload conditions, it shall send OCI within the 3gpp-Sbi-Oci HTTP header (i.e. OCI header, see clause 5.2.3.2.9) to the peer entity (Consumer or Producer, respectively). The OCI header shall be piggybacked on a signalling message that is sent to the peer.

The NF Service Producer/Consumer/SCP/SEPP shall send the "3gpp-Sbi-Oci" header, regardless of whether the peer supports the feature (see clause 6.4.3.6). The header is ignored by the receiver if the latter does not support the OLC-H feature.

6.4.3.3 Frequency of Conveyance

How often or when the sender conveys the OCI is implementation specific. The sender shall ensure that new or updated OCI is conveyed to the target receivers with an acceptable delay, such that the purpose of the information (i.e. effective overload control protection) is achieved. The following are some of the potential approaches the sender may implement for conveying the OCI:

- the sender may convey the OCI towards a receiver only when the new/changed value has not already been conveyed to the given receiver;
- the sender may convey the OCI periodically;
- the sender may convey the OCI towards a receiver to restart the OCI period of validity.

The sender may also implement a combination of one or more of the above approaches.

6.4.3.4 Overload Control Information

6.4.3.4.1 General Description

A NF Service Producer may include one or more OCI header(s) in a service response with any HTTP status code (e.g. 2xx, 3xx, 4xx), or in a notification request message sent to a NF Service Consumer. An NF Service Producer may report OCI for different scopes, e.g.:

- to report OCIs for an NF service instance, an NF service set and/or an NF instance;

- to report OCIs at the level of an SMF (service) instance or SMF (service) set, and for specific S-NSSAI/DNNs;
- to report OCIs for different S-NSSAI/DNNs of an SMF (service) instance or SMF (service) set.

A NF Service Producer may also include OCI header(s) with different scopes in different messages, e.g. an SMF may report OCI for the entire SMF instance first, and then for a specific S-NSSAI/DNN only, if the overload conditions have changed and the SMF ends up with an overload only affecting a specific S-NSSAI/DNN.

An NF that receives OCI headers with different scopes, in the same message or in different messages, shall handle each OCI independently from each other. For instance, if an NF service consumer receives one OCI with the scope of an NF (Service) Set and then another OCI with the scope of an NF (Service) instance that pertains to the NF (Service) Set, the NF shall store the latter OCI and also consider that the former OCI is still valid for the NF (Service) Set until the related period of validity expires.

If an NF Service Consumer receives more than one OCI with overlapping scopes, e.g. one OCI with NF (service) instance scope and another OCI with NF (service) Set scope, the NF Service Consumer should perform overload control towards a target NF service instance considering the OCI received with the finer scope (i.e. in this example the overload of the NF (service) instance). For instance, if an AMF receives one OCI with an SMF instance scope and with an overload reduction metric of 20%, and one OCI with the scope of a specific SMF service set of the same SMF instance and with an overload reduction of 50%, the AMF should throttle 50% of the traffic targeting the specific SMF service set and 20% of the traffic targeting other SMF services instances of the SMF instance (if no valid OCI is available for the other SMF service instances).

For S-NSSAI/DNN based overload control (see clause 6.4.3.4.5.2.2), when signalling OCI for an SMF (service) instance or an SMF (service) set in a message, the SMF shall always include the full set of overload control information applicable to the SMF (service) instance or SMF (service) set, i.e. OCI for the SMF (service) instance or an SMF (service) set level and/or OCI for specific S-NSSAI/DNNs, even if only a subset of the OCI has changed; these OCIs shall contain the same Overload Control Timestamp. When including OCI for some S-NSSAI/DNN(s), the SMF should not provide any OCI for the SMF (service) instance or an SMF (service) set level unless OCI for such level is also applicable.

If an NF Service Consumer receives OCIs with overlapping scopes for an SMF (service) instance or an SMF (service) set level and for specific S-NSSAI/DNNs, the NF Service Consumer should perform overload control towards a target SMF service instance and S-NSSAI/DNN considering the OCI received with the finer scope. For instance, if an AMF receives an OCI for an SMF instance with an overload reduction metric of 20%, and one OCI for a specific S-NSSAI/DNN of the same SMF instance with an overload reduction of 50%, the AMF should throttle 50% of the traffic targeting the specific S-NSSAI/DNN and 20% of the traffic targeting other S-NSSAI/DNNs of the SMF instance (if no valid OCI is available for the other S-NSSAI/DNN).

A NF Service Consumer may include one OCI header in a notification response sent with any HTTP status code (e.g. 2xx, 3xx, 4xx), or in a service request sent to a NF Service Producer.

An SCP/SEPP may additionally include one OCI in any service request or response, or notification request or response, sent towards a NF Service Consumer or NF Service Producer.

The OCI shall always include the Overload Timestamp, Overload Reduction Metric, OCI Period of Validity and Scope parameters (see clause 6.4.3.4.2 for the complete list of parameters).

6.4.3.4.2 Overload Control Timestamp

The Timestamp parameter indicates the time when the OCI was generated. It shall be used by the receiver of the OCI to properly collate out-of-order OCI headers, e.g. due to HTTP/2 stream multiplexing, prioritization and flow control, and to determine whether the newly received OCI has changed compared to the OCI previously received for the same scope.

The receiver shall overwrite any stored OCI for a peer NF, NF set, NF service, NF service set, Callback URI, SCP or SEPP (according to the scope of the new received OCI) with the newly received OCI, if the new OCI is more recent than the stored information. For instance, for S-NSSAI/DNN based overload control, if the receiver had stored OCI for a peer SMF instance and OCI for a specific S-NSSAI/DNN of that SMF instance, it shall overwrite these OCIs with the new OCI received in a message carrying OCI for the same SMF instance.

If the newly received OCI has the same or an older Timestamp than the previously received OCI for the same scope (e.g. for the same NF, NF Set, NF Service, NF Service Set, Callback URI, SCP or SEPP), then the receiver shall discard the newly received OCI and continue to apply the overload control procedures based on the previously received OCI values with the most recent Timestamp value.

NOTE: An NF Service Producer/Consumer can receive OCI for the same target scope from different NF service Consumers/Producers, when the scope of the OCI corresponds to an NF set or NF service set.

An entity generating an OCI shall update the Overload Control Timestamp whenever it modifies some information in the OCI or whenever it wants to extend the period of validity of the OCI. The Overload Control Timestamp shall not be updated otherwise.

6.4.3.4.3 Overload Reduction Metric

The Overload Reduction Metric parameter shall have a value in the range from 0 to 100 and shall indicate the percentage of traffic reduction the OCI sender requests the receiver to apply. An Overload Reduction Metric of "0" indicates that the OCI sender is not overloaded (i.e. overload control enforcement procedures are not necessary). The computation of the overload metric is implementation specific.

Considering the processing requirement of the OCI receiver, e.g. to perform overload control enforcement based on the updated Overload Reduction Metric, the sender should refrain from advertising every small variation, e.g. with the granularity up to 5 percentage units. Larger variations should be considered as reasonable enough for advertising a new Overload Reduction Metric and thus justifying the processing requirement (to handle the new information) of the receiver. The exact granularity of the Overload Reduction Metric is an implementation matter.

The conveyance of the OCI signals that an overload situation is occurring, unless the Overload Reduction Metric is set to "0", which signals that the overload condition has ceased. Conversely, the absence of the OCI header in a message does not mean that the overload has abated.

6.4.3.4.4 Overload Control Period of Validity

The Period of Validity parameter is a timer, which shall indicate the length of time during which the overload condition specified by the OCI header shall be considered as valid (unless overridden by subsequent new OCI).

An overload condition shall be considered as valid from the time the OCI is received until the Overload Control Period of Validity expires or until another OCI with a new set of information (identified by a more recent Timestamp) is received for the same scope. The timer corresponding to the Period of Validity shall be restarted each time an OCI with a new set of information is received for the same scope. When this timer expires, the last received OCI shall be considered outdated and obsolete (i.e. any associated overload condition shall be considered to have ceased) and the overload control enforcement shall be stopped.

The Period of Validity parameter achieves the following:

- it avoids the need for the overloaded NF Service Producer/Consumer/SCP/SEPP to convey the OCI frequently to its peers when the overload state does not change. Therefore, this minimizes the processing required at the overloaded NF Service Producer/Consumer/SCP/SEPP and its peers upon sending/receiving HTTP/2 signalling;
- it allows to reset the overload condition after some time the NF Service Consumer/Producer having received an overload indication from the overloaded peer, e.g. if no signalling traffic takes place between these HTTP peers for some time due to overload mitigation actions. This also removes the need for the overloaded NF Service Producer/Consumer/SCP/SEPP to remember the list of its peers to which it has sent a non-null overload reduction percentage and to which it would subsequently need to convey when the overload condition ceases.

6.4.3.4.5 Scope of OCI

6.4.3.4.5.1 Introduction

The scope of OCI indicates the service requests or notification requests to which the OCI applies, i.e. it identifies the traffic that the OCI sender requests the receiver to process in accordance with the OCI.

The following clauses provide a detailed description of the parameters that define the scope of the OCI header.

6.4.3.4.5.2 Scope of OCI signalled by an NF Service Producer

6.4.3.4.5.2.1 General

The OCI sent by an NF Service Producer shall include one and only one of the parameters defined in Table 6.4.3.4.5.2-1.

Table 6.4.3.4.5.2-1: Supported scopes for OCI signalled by an NF Service Producer

Parameter	Value	OCI scope (i.e. OCI applies to)	Examples
NF-Instance	NF Instance ID	All services of the NF instance identified by the NF Instance ID.	NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8
NF-Set	NF Set ID	All services of all NF instances of the NF set identified by the NF Set ID.	NF-Set: set1.udmset.5gc.mnc012.mcc345
NF-Service-Instance (and NF-Inst)	NF Service Instance ID (and NF Instance ID)	The service instance identified by the NF Service Instance ID and the NF instance Id (if available) or the last known NF instance ID.	NF-Service-Instance: serv1.smf1; NF-Inst: 54804518-4191-46b3-955c-ac631f953ed8
NF-Service-Set	NF Service Set ID	All service instances of the NF service set identified by the NF service set ID.	NF-Service-Set: setxyz.snsmf-pdusession.nfi54804518-4191-46b3-955c-ac631f953ed8.5gc.mnc012.mcc345

6.4.3.4.5.2.2 Additional scope parameters for S-NSSAI/DNN based overload control by SMF

It is optional for the SMF to support S-NSSAI/DNN based overload control. When supported, the following requirements shall apply.

S-NSSAI/DNN level overload control refers to advertising of the overload information at S-NSSAI and DNN level granularity and hence applying the mitigation policies based on this information to the signalling traffic related to this S-NSSAI and DNN only. Only an SMF may advertise S-NSSAI/DNN level overload information when it detects overload for certain S-NSSAI/DNNs, e.g. based on shortage of internal or external resources for an S-NSSAI/DNN (e.g. IP address pool).

NOTE 1: When all the internal and external resources, applicable to the S-NSSAI/DNNs, are available for all the S-NSSAI/DNNs served by an SMF, overload control at SMF (service) set or SMF (service) instance level is exactly the same as S-NSSAI/DNN level overload information of that SMF, for each of its S-NSSAI/DNNs, and hence, performing overload control at SMF (service) set or SMF (service) instance level is sufficient.

When performing S-NSSAI/DNN based overload control, the OCI scope shall indicate, in addition to either an NF-Instance, NF-Set, NF-Service-Instance or NF-Service-Set (see Table 6.4.3.4.5.2-1), the combinations of S-NSSAI and DNN for which the OCI sender wants to advertise the overload information using the following parameters: - the S-NSSAI parameter, indicating one or more S-NSSAI values; and

- the DNN parameter, indicating one or more associated DNN values from the indicated S-NSSAI(s).

NOTE 2: It is not allowed to report OCI for a DNN only or for an S-NSSAI only.

See Table 6.4.3.4.5.2.2-1.

Table 6.4.3.4.5.2.2-1: Additional scope parameters for S-NSSAI/DNN based overload control by SMF

Parameter	Value	OCI scope (i.e. OCI applies to)	Examples
S-NSSAI	One or more S-NSSAI values	DNN(s) from indicated S-NSSAI(s), for the service(s) of NF instance(s) as defined in Table 6.4.3.4.5.2-1.	S-NSSAI: %7B%22sst%22%3A%201%2C%20%22sd%22%3A%20%22A08923%22%7D; DNN: internet.mnc012.mcc345.gprs
DNN	One or more DNN values		
NOTE 1: Both the S-NSSAI and DNN parameters shall be present. The S-NSSAI and DNN parameters shall be provided with either the NF-Instance, NF-Set, NF-Service-Instance or NF-Service-Set parameter (see Table 6.4.3.4.5.2-1).			
NOTE 2: The S-NSSAI parameter in the Example corresponds to the JSON encoding: {"sst": 1, "sd": "A08923"} (see clause 5.2.3.1).			

An SMF shall advertise S-NSSAI/DNN based overload control for at most 10 DNNs.

NOTE 3: Considering various aspects such as the processing and storage requirements at the overloaded SMF entity and the receiver, the number of important DNNs for which overload control advertisement could be necessary, interoperability between the NFs of different vendors, it was chosen to define a limit on the maximum number of DNNs for advertising the overload control information.

The SMF may advertise overload information for different DNNs of one or more S-NSSAIs in a single OCI header (if the same OCI information, e.g. overload reduction metric, applies to all the DNNs of the S-NSSAI(s)) or in up to 10 OCI headers (if different OCI information needs to be advertised for different DNNs).

An NF selecting an SMF service instance for a given S-NSSAI/DNN shall apply the S-NSSAI/DNN level overload information, if available for that S-NSSAI/DNN.

6.4.3.4.5.3 Scope of OCI signalled by an NF Service Consumer

The OCI sent by an NF Service Consumer shall include one and only one of the parameters defined in Table 6.4.3.4.5.3-1.

Table 6.4.3.4.5.3-1: Supported scopes for OCI signalled by an NF Service Consumer

Parameter	Value	OCI scope (i.e. OCI applies to)	Examples
Callback-Uri	One or more URI(s) including a scheme, authority and an optional path	All notifications (or callbacks) with a notification (or callback) URI matching the Callback-Uri parameter value. (NOTE 1)	Callback-Uri: "https://pcf12.operator.com" Callback-Uri: "https://pcf12.operator.com/serviceY" Callback-Uri: "https://pcf12.operator.com/serviceY/abc" & "https://pcf12.operator.com/serviceY/def"
NF-Instance	NF Instance ID	All notifications (or callbacks) bound to: - the NF Instance ID; or - an NF service instance or an NF service set of this NF Instance ID. (NOTE 2)	NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8
NF-Set	NF Set ID	All notifications (or callbacks) bound to: - the NF Set ID; - an NF instance of the NF Set ID; or - an NF service instance or an NF service set of an NF Instance of the NF Set ID. (NOTE 2)	NF-Set: set1.udmset.5gc.mnc012.mcc345
NF-Service-Instance (and NF-Inst)	NF Service Instance ID (and NF Instance ID)	All notifications (or callbacks) bound to: - the NF service instance identified by the NF Service Instance ID and the NF instance Id (if available) or the last known NF instance ID. (NOTE 2)	NF-Service-Instance: serv1.smf1; NF-Inst: 54804518-4191-46b3-955c-ac631f953ed8
NF-Service-Set	NF Service Set ID	All notifications (or callbacks) bound to: - the NF Service Set ID; or - an NF service instance of the NF Service Set ID. (NOTE 2)	NF-Service-Set: setxyz.snnsmf-pdusession.nfi54804518-4191-46b3-955c-ac631f953ed8.5gc.mnc012.mcc345
NF-Instance or NF-Set and Service-Name	As defined above and as defined for servname in clause 5.2.3.2.5	All notifications (or callbacks) bound to the service indicated in Service-Name within the NF instance ID or NF Set ID, as defined above (NOTE 2)	NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8; Service-Name: def
NOTE 1: A notification (or callback) URI matches the Callback-Uri parameter value if the former contains the same scheme, the same authority and has a path that encompasses the path of the latter. NOTE 2: Notification (or callbacks) may be bound to an NF instance, an NF set, an NF service instance or an NF service set by a request creating a subscription or a callback resource with a Binding Indication as specified in clauses 6.12.4 and 6.12.5.			

EXAMPLE 1: Assuming that a PCF has created the following subscriptions in an AMF:

- subscription 1: notification URI=https://pcf12.example.com/serviceX/1234
- subscription 2: notification URI=https://pcf12.example.com/serviceY/abc

- subscription 3: notification URI=https://pcf12.example.com/serviceY/def

When experiencing overload, if the PCF signals the following OCI scope:

- Callback-Uri: "https://pcf12.example.com", the OCI applies to notifications of all the subscriptions;
- Callback-Uri: "https://pcf12.example.com/serviceY", the OCI applies to notifications of subscriptions 2 and 3;
- Callback-Uri: "https://pcf12.example.com/serviceY/abc", the OCI applies to notifications of subscription 2.

EXAMPLE 2: Assuming that a PCF has created the following subscriptions in an AMF:

- subscription 1: notifications bound to PCF service set X, within PCF12 Instance ID, within PCF Set Z
- subscription 2: notifications bound to PCF service set Y, within PCF12 Instance ID, within PCF Set Z
- subscription 3: notifications bound to PCF12 Instance ID and service "def", within PCF Set Z

When experiencing overload, if the PCF signals the following OCI scope:

- NF-Instance=PCF12 Instance ID, the OCI applies to notifications of all the subscriptions;
- NF-Service-Set=Service Set Y of PCF12 Instance ID, the OCI applies to notifications of subscription 2;
- NF-Instance=PCF12 Instance ID and Service="def", the OCI applies to notifications of subscription 3.

6.4.3.4.5.4 Scope of OCI signalled by an SCP

The OCI sent by an SCP shall include one and only one of the parameters defined in Table 6.4.3.4.5.4-1.

Table 6.4.3.4.5.4-1: Supported scopes for OCI signalled by an SCP

Parameter	Value	OCI scope (i.e. OCI applies to)	Examples
SCP-FQDN	SCP FQDN	All requests towards the SCP identified by the SCP FQDN.	SCP-FQDN: scp1.example.com

6.4.3.4.5.5 Scope of OCI signalled by an SEPP

The OCI sent by an SEPP shall include one of the parameters defined in Table 6.4.3.4.5.5-1.

Table 6.4.3.4.5.5-1: Supported scopes for OCI signalled by an SEPP

Parameter	Value	LCI scope (i.e. LCI applies to)	Examples
SEPP-FQDN	SEPP FQDN	All requests towards the SEPP identified by the SEPP FQDN.	SEPP-FQDN: sepp1.example.com

6.4.3.4.6 Extension of registration timer in OCI

The "Extend-Registration-Timer" parameter is a boolean indication (true / false) applicable to the IMS network that may optionally be included by an NF Service Producer (e.g. HSS) to inform the NF Service Consumer (e.g. I-CSCF or S-CSCF) about the need to abate the traffic by extending the registration timers of the UEs trying to get access to the IMS network; this situation shall be enforced during the time period indicated in the "Period-of-Validity" parameter (see clause 6.4.3.4.4).

6.4.3.5 Overload Control Enforcement

6.4.3.5.1 Message Throttling

As part of the overload mitigation, the overload control enforcement NF, i.e. an entity that receives OCI (with a non-null overload reduction metric), shall reduce the total number of request messages, which would have been sent

otherwise, towards the overloaded peer(s) corresponding to the received scope, e.g. towards all the NF instances of the NF Set when the scope indicates an NF Set ID and shall not redirect its requests to another entity pertaining to the same scope.

This shall be achieved by discarding a fraction of the request messages in proportion to the overload level of the peer, which is called request message throttling. The message throttling shall be achieved either by rejecting the request messages, or by redirecting some of request messages to an alternative NF if possible, e.g. when a binding indication was received for the target resource/session context containing binding entities which have not been reported as overloaded.

When sending (i.e. redirecting) the request towards an alternative NF to address an existing resource/session context, the sending NF may include a 3gpp-Sbi-Request-Info header in the request to indicate whether the request is redirected to an alternative NF as result of overload enforcement (see clause 5.2.3.3.12). The alternative NF may use this header to determine whether to accept the request, e.g. when accepting the request will not further overload the overloaded NF.

Message throttling shall apply to HTTP requests only (any service request including notification request).

Network Functions shall support and use the "Loss" algorithm as specified in clause 6.4.3.5.2.

6.4.3.5.2 Loss Algorithm

An overloaded NF Service Producer/Consumer/SCP/SEPP shall ask its peers to reduce the number of HTTP requests they would otherwise send by conveying in the OCI header the requested traffic reduction percentage within the Overload Reduction Metric parameter, as specified in clause 6.4.3.4.3.

The recipients of the Overload Reduction Metric shall reduce the number of request messages by that percentage, either by redirecting them to an alternate destination if possible (e.g. an HTTP POST request for the Nsmf_PDUSession_CreateSMContext service operation can be sent to an alternate SMF in the same SMF set, if the olcScope is at the NF instance level and the binding indication of the service resource is for an SMF set), or by failing the request and treating it as if it was rejected by the destination entity.

NOTE: For example, if an NF Service Producer/Consumer/SCP/SEPP requests a peer to reduce the traffic by 10%, then that peer throttles 10% of the traffic that would have otherwise been sent to this NF Service Producer/Consumer/SCP/SEPP.

6.4.3.6 OLC-H feature support

6.4.3.6.1 Indicating supports for the OLC-H feature

When registering with the NRF (NFRegister) or updating the NRF (NFUpdate), an NF that supports the OLC-H feature shall indicate the feature support (see clause 6.1.6.2.2 in TS 29.510 [8]).

When an NF Service Consumer queries an NRF (NFDiscover) to discover services offered by NF Service Producers, the NRF shall indicate to the NF Service Consumer, if the NF Service Producers support the OLC-H feature (see clause 6.2.6.2.3 in TS 29.510 [8]).

6.5 Support of Stateless NFs

6.5.1 General

A NF may become stateless by decoupling the "compute" resource and "storage" resource as specified in clause 4.1 of TS 23.501 [3].

6.5.2 Stateless AMFs

6.5.2.1 General

AMF may become stateless by storing the UE related information in the UDSF. Procedures for AMF planned removal or the AMF auto-recovery are specified in clauses 5.21.2.2 and 5.21.2.3 of TS 23.501 [3].

6.5.2.2 AMF as service consumer

1. When the AMF subscribes to notifications from another NF Service Producer, the AMF shall provide its GUAMI to the NF Service Producer to enable the latter to discover AMFs within the AMF set, or information about a backup AMF, in addition to Callback URI in the subscription resource.

The AMF may also provide the name of the AMF service to which these notifications are to be sent (this service shall be one of the service produced by the AMF and registered in the NRF or a custom service registered in the NRF for the purpose of receiving these notifications);

NOTE 1: Providing an AMF service name allows the NF Service Producer to find the endpoint address and callback URI prefix (if any) to deliver the notifications (see bullet 2). The provided AMF service might not use itself the information received in these notifications.

2. A NF service producer may also use the Nnrf_NFDISCOVERY service to discover AMFs within an AMF set or backup AMF.

If the AMF provided the name of its service (see bullet 1), the NF Service Producer shall look up for the same AMF service from the AMFs within the AMF set or from backup AMF, and use endpoint addresses (i.e. IP addresses, transport and port information, or FQDN) and callback URI prefix (if any) of that service to send notifications (see bullet 6). Otherwise, the notifications shall be sent to an endpoint address registered in the NF Profile of the AMF.

NOTE 2: The AMF can register different endpoint addresses in the NRF for different services.

3. An NF service producer may subscribe to GUAMI changes using the AMFStatusChange service operation of the Namf_Communication service.
4. An NF service producer may become aware of AMF changes (at the time of the AMF change or subsequently when sending signalling to the AMF) via Namf_Communication service AMFStatusChange Notifications, via HTTP Error response from the old or a wrongly selected new AMF, via link level failures (e.g. no response from the AMF), or via a notification from the NRF that the AMF has deregistered. The HTTP error response may be a 3xx redirect response pointing to a new AMF.

NOTE 3: AMFs are identified by GUAMIs. A GUAMI can point to an individual AMF or to some or all AMFs within an AMF set. If a GUAMI points to several AMFs, and the UE is served by one of those, all those AMFs can immediately handle communication for that service, and the NF service producer does not need to be aware which of those AMFs is serving a UE.

5. When becoming aware of an AMF change, and the new AMF is not known, the NF service producer shall select an AMF within the AMF set or the possibly earlier received backup AMF.
6. When becoming aware of an AMF change, the NF service producer shall
 - use the default notification URI of the default notification subscription of the reselected AMF (as specified in clause 6.10.5.2), if this is a default notification from a default notification subscription; otherwise
 - replace the authority part and callback URI prefix (if any) of the Notification URI with new AMF information; when replacing the authority, if the port number or and callback URI prefix are not available for the new AMF, e.g. when the NF instance level information of the new AMF is to be used, the port number and callback URI prefix (if any) in old Notification URI should be reused in the alternative notification URI.

The NF service producer shall then use that URI in subsequent communication.

7. Each AMF within the AMF set shall be prepared to receive notifications from the NF service producer, by either handling the notification to the Notification URI constructed according to bullet 6 with the own address as authority part and callback URI prefix (if any), or by replying with an HTTP 3xx redirect pointing to a new AMF, or by replying with another HTTP error.
8. The NF service producer shall be prepared to receive updates to resources of the related service from any AMF within the set.
9. If the UE moves to an AMF from a different AMF Set, or to an AMF from the same AMF set that does not support handling the notification as specified in bullet 7, the new AMF should update peer NFs with the new callback URI for the notification.

6.5.2.3 AMF as service producer

1. When AMF receives request to establish a service, it may provide information about a backup AMF in a suitable resource.
2. NF service consumer may also use the Nnrf_NFDiscovery service to discover AMFs within an AMF set.
3. An NF service consumer may subscribe to GUAMI changes using the AMFStatusChange service operation of the Namf_Communication service.
4. An NF service consumer may become aware of AMF changes (at the time of the AMF change or subsequently when sending signalling to the AMF) via Namf_Communication service AMFStatusChange Notifications, via Error response from the old or a wrongly selected new AMF, via link level failures (e.g. no response from the AMF), or via a notification from the NRF that the AMF has deregistered. The HTTP error response may be a 3xx redirect response pointing to a new AMF.

NOTE. AMFs are identified by GUAMIs. A GUAMI can point to an individual AMF or to some or all AMFs within an AMF set. If a GUAMI points to several AMFs, and the UE is served by one of those, all those AMFs can immediately handle communication for that service, and the NF service consumer does not need to be aware which of those AMFs is serving a UE.

5. When becoming aware of an AMF change, and the new AMF is not known, the NF service consumer shall select an AMF within the AMF set or the possibly earlier received backup AMF.
6. When becoming aware of an AMF change, the NF service consumer shall exchange the apiRoot of resource URIs with new AMF's apiRoot and shall use that URI in subsequent communication.
7. Each AMF within the AMF set shall be prepared to receive updates for resources from the NF service consumer, by either handling the updates to the resource URIs constructed according to step 6 with its own apiRoot, or by replying with an HTTP 3xx redirect pointing to a new AMF, or by replying with another HTTP error.
8. For a service that includes notifications from the AMF, the NF service consumer shall be prepared to receive for the service notifications from any AMF within the set.

NOTE: If the UE moves to an AMF from a different AMF Set, or to an AMF from the same AMF set that does not support handling the updates as specified in bullet 7, but mechanisms exist to transfer information related to the resource to the AMF, service specific mechanism can exist to notify the NF service consumer about the resource at the AMF. For instance, for the Namf_EventExposure service, information and an event subscription is transferred to the new AMF in such a manner and the new AMF will then report an event-change event.

6.5.3 Stateless NFs (for any 5GC NF type)

6.5.3.1 General

An NF may become stateless by storing its contexts as unstructured data in the UDSF. An UDM, PCF and NEF may also store own structured data in the UDR. An UDR and UDSF cannot become stateless.

An NF may also be deployed such that several stateless network function instances are present within a set of NF instances. Additionally, within an NF, an NF service may have multiple instances grouped into a NF Service Set if they are interchangeable with each other because they share the same context data. See clause 5.21 of TS 23.501 [3].

A UDM / AUSF / UDR / PCF group may consist of one or multiple UDM / AUSF / UDR / PCF sets.

6.5.3.2 Stateless NF as service consumer

1. When the NF service consumer subscribes (explicitly or implicitly) to notifications from another NF service producer, the NF service consumer may provide a binding indication to the NF service producer as specified in clause 6.3.1.0 of TS 23.501 [3] and clause 4.17.12.4 of TS 23.502 [4], to enable the related notifications to be sent to an alternative NF service consumer within the NF (service) set, in addition to providing the Callback URI in the subscription resource. The NF service consumer may provide the NRF API URI(s) in 3gpp-Sbi-Nrf-Uri-Callback header which can be used for reselection of NF service consumer.

2. A NF service producer or SCP may use the Nnrf_NFDiscovery service to discover NF service consumers within an NF (service) set. If the NRF API URI(s) was received in the 3gpp-Sbi-Nrf-Uri-Callback header in bullet 1, the NRF NFDiscovery API URI should be used for the reselection.
3. An NF service producer may become aware of a NF service consumer change, via receiving an updated binding information (i.e. when the binding entity corresponding to the binding level is changed) in a HTTP request message, or via an Error response to a notification, via link level failures (e.g. no response from the NF), or via a notification from the NRF that the NF service consumer has deregistered. The HTTP error response may be a 3xx redirect response pointing to a new NF service consumer.

NOTE 1: When the binding entity other than the one corresponding to the binding level is changed, it indicates the resilience information of the session is changed, i.e. more or less consumer instances are able to handle the Notification/Callback request message; the NF service producer is not expected to change Notification/Callback URI.

NOTE 2: When a Binding Indication is included in an acceptance response message, the NF service producer stores the Binding Indication, but does not check it to determine whether there is a NF service consumer change. Accordingly, the NF service producer continues to use its current Notification/Callback URI for subsequent requests, until it becomes aware of an NF service consumer change, at which point in time it uses the last received binding information to reselect a different instance.

4. When becoming aware of an NF service consumer change, and if the new NF service consumer is not known, the NF service producer shall select a new NF service consumer as specified in clause 6.6 of TS 23.527 [38]. If binding information is available and the binding mechanism is supported by the NF service producer, the reselection should be based on the binding information, as specified in clause 6.6.2 of TS 23.527 [38], in clause 6.3.1.0 of TS 23.501 [3] and in clause 4.17.12.4 of TS 23.502 [4]. If binding information is not available or the binding mechanism is not supported by the NF service producer, the reselection is performed as specified in clause 6.6.3 of TS 23.527 [38].
5. When becoming aware of an NF service consumer change, the NF service producer or SCP shall
 - use the default notification URI of the default notification subscription of the reselected NF service consumer (as specified in clause 6.10.5.2), if this is a default notification from a default notification subscription; otherwise
 - replace the authority and callback URI Prefix (if any) part of the Notification/Callback URI with the new NF service consumer information; when replacing the authority, if the port number or and callback URI prefix are not available for the new NF service consumer, e.g. when the NF instance level information of the new NF service consumer is to be used, the port number and callback URI prefix (if any) in old Notification URI should be reused in the alternative notification URI.

The NF service producer or SCP shall then use that URI in subsequent communications, as specified in clause 6.6 of TS 23.527 [38].

6. When the NF service consumer is changed, and if the new NF service consumer does not support handling notifications as specified in the above bullet 5, the new NF service consumer should update the NF service producers with the new Notification URI. For explicit subscriptions, this is achieved by updating the existing subscription or creating a new subscription, depending on the NF service producer's API. For implicit subscriptions, this is carried out via a service update request message.
7. The new NF service consumer may include an updated binding indication in a service request or notification response message to the NF service producer.
8. Each NF service consumer within the NF (service) set shall be prepared to receive notifications from the NF service producer, either by handling the notifications to the Notification URI constructed according to the above bullet 5 with its own address as authority part and callback URI Prefix (if any), by handling the notifications to the Notification URI notified in the above bullet 6, or by replying with an HTTP 3xx redirect pointing to a new NF service consumer or with another HTTP error.
9. The NF service producer shall be prepared to receive updates to resources of the related service from any NF service consumer within the NF (service) set.
10. If an SCP detects that the target NF service consumer of a notification/callback request is not available, the SCP shall reselect a new NF service consumer based on the Routing Binding Indication and/or 3gpp-Sbi-Discovery

headers, if such information has been provided by the NF service producer in the notification/callback request. See clause 6.6 in TS 23.527 [38].

6.5.3.3 Stateless NF as service producer

1. When the NF service producer receives a request to establish a service, it may provide binding information as specified in clause 6.3.1.0 of TS 23.501 [3] and clause 4.17.12 of TS 23.502 [4] to establish a binding between the NF service consumer and the NF service producer for subsequent related requests.
2. The NF service consumer or SCP may use the Nnrf_NFDiscovery service to discover NF service producers within an NF (service) set.
3. An NF service consumer may become aware of a NF service producer change, by receiving an updated binding information (i.e. when the binding entity corresponding to the binding level is changed) in a HTTP request message, or via an Error response from the old or a selected new NF service producer, via link level failures (e.g. no response from the NF), or via a notification from the NRF that the NF has deregistered. The HTTP error response may be a 3xx redirect response pointing to a new NF.

NOTE 1: When the binding entity other than the one corresponding to the binding level is changed, it indicates the resilience information of the resource context is changed, i.e. more or less service instances are able to handle the service request message; the NF service consumer is not expected to change the Location of the resource context in the NF service producer.

NOTE 2: When a Binding Indication is included in an acceptance response message, the NF service consumer stores the Binding Indication, but does not check it to determine whether there is a NF service producer change. Accordingly, the NF service consumer continues to use its current Resource URI for subsequent requests, until it becomes aware of an NF service producer change, at which point in time it uses the last received binding information to reselect a different instance.

4. When becoming aware of an NF service producer change, and if the new NF service producer is not known, the NF service consumer shall select a new NF service producer, as specified in clause 6.5 of TS 23.527 [38]. If binding information is available and the binding mechanism is supported by the NF service consumer, the reselection should be based on the binding information, as specified in clause 6.12 of this specification (see also clause 6.5.2 of TS 23.527 [38]) and in clause 6.3.1.0 of TS 23.501 [3]. If binding information is not available or the binding mechanism is not supported by the NF service consumer, the reselection is performed as specified in clause 6.5.3 of TS 23.527 [38].
5. When becoming aware of an NF service producer change, the NF service consumer or SCP shall replace the apiRoot of the resource URI with the new NF service producer's apiRoot and shall use that URI in subsequent communications as specified in clause 6.5 of TS 23.527 [38].
6. When the NF service producer changes, the new NF service producer may include an updated binding indication in a notification/callback request sent to the NF service consumer if sending updated binding indication in notification requests is not restricted (i.e. if "3gpp-Sbi-Binding-Indication-Notify-Restricted" header was not previously received or the notification callback URI is not restricted by the received "3gpp-Sbi-Binding-Indication-Notify-Restricted" header). The new NF service producer may also generate a new resource URI and return it to the NF service consumer upon reception of a service request related to the resource from that NF service consumer, e.g. the new NF service producer may reply with an HTTP 3xx redirect status code pointing to the new location of the resource.
7. Each NF service producer within the NF (service) set shall be prepared to receive updates for resources from the NF service consumer, either by handling the updates to the resource URIs constructed according to the above bullet 5 with its own apiRoot, by handling the updates to the resource URIs notified in the above bullet 6, by replying with an HTTP 3xx redirect pointing to a new NF service producer, or by replying with another HTTP error.
8. For a service that includes notifications from the NF service producer, the NF service consumer shall be prepared to receive for that service notifications from any NF service producer within the NF (service) set.
9. If an SCP detects that the target NF service producer is not available, the SCP shall reselect a new NF service producer based on the Routing Binding Indication and/or 3gpp-Sbi-Discovery headers, if such information has been provided by the NF service consumer in the request. See clause 6.5 in TS 23.527 [38].

6.6 Extensibility Mechanisms

6.6.1 General

This clause describes the extensibility mechanisms supported in the Service-Based Architecture in 3GPP 5GC, such as feature negotiation, vendor-specific extensions, etc.

6.6.2 Feature negotiation

A versioning of services in the request URI shall be supported by 3GPP 5G APIs, but version upgrades shall only be applied for non-backward compatible changes or the introduction of new mandatory features.

The following mechanism to negotiate applicable features shall be used by 5G APIs. This supported feature mechanism shall be applied separately for each API.

For any API that defines resources, suitable resources associated to or representing the NF Service Consumer (e.g. a top-level resource or a sub-resource representing the NF Service Consumer) shall be identified in each API to support the negotiation of the applicable features between the NF Service Consumer and NF Service Producer for this resource. Each such resource for a 5G API shall contain an attribute (e.g. "supportedFeatures") of the SupportedFeatures data type defined in TS 29.571 [13] containing a bitmask to indicate supported features. The features and their positions in that bitmask are defined separately for each API.

The HTTP client acting as NF service consumer shall include the attribute of the SupportedFeatures data type defined in TS 29.571 [13] in the HTTP PUT or POST requests to create the resource associated to or representing the NF Service Consumer of 5G API. This attribute indicates which of the features defined for the corresponding service are supported by the HTTP client. The HTTP server shall determine the supported features for the corresponding resource by comparing the supported features indicated by the client with the supported features the HTTP server supports. Features that are supported both by the client and the server are supported for that resource. The HTTP server shall include the attribute of the SupportedFeatures data type defined in TS 29.571 [13] indicating those features in the representation of the resource it returns to the HTTP client in the HTTP response confirming the creation of the resource.

The HTTP client acting as NF service consumer may include a query parameter (e.g. "supported-features") of the SupportedFeatures data type defined in TS 29.571 [13] in HTTP GET requests to fetch resource(s) associated to the NF Service Consumer of 5G API. This query parameter indicates which of the features defined for the corresponding service are supported by the HTTP client. The HTTP server shall determine the supported features for the corresponding resource(s) by comparing the supported features indicated by the client with the supported features the HTTP server supports. Features that are supported both by the client and the server are supported for the resource(s); attributes or enumerated values that are only of relevance to a feature unsupported by the requested resource(s) should be omitted from the representation sent in the response. The HTTP Server shall include the attribute of the SupportedFeatures data type defined in TS 29.571 [13] indicating those features in the HTTP GET response, if supported by the API definition.

The supported features for a resource associated to or representing the NF Service Consumer shall also be applicable to all subordinate resources of that resource, and for all custom operations related to any of those resources. If any of those resources is used for the subscription to notifications (see clause 4.6.2 of TS 29.501 [5]), the supported features shall also apply to those notifications. For default notification subscriptions, the supported features shall be determined by comparing the supported features if registered by the NF Service Consumer for the corresponding default notification subscription in NRF with the features supported by the NF Service Producer. Features that are supported by both are supported for the default notification subscription, which shall also apply to related default notifications.

When an NF service consumer subscribes to notifications from an NF service producer via an intermediate NF (see Figure 7.1.2-3 of TS 23.501 [5]), the NF service consumer may indicate, to the intermediate NF and the NF service producer, the API version(s) and the features it supports for the corresponding NF service producer's service by including the 3gpp-Sbi-Consumer-Info header in its request as described in clauses 5.2.3.3.7 and 6.2.2. When required by a given API, the NF service producer may then indicate the features supported by both the NF service consumer and NF service producer for the subscription and its notifications by including an attribute of the SupportedFeatures data type in the notification request it sends towards the NF service consumer.

NOTE 1: The above principles can also apply in scenarios where:

- a) the intermediate NF creates the subscription at the NF service producer by other means than a subscribe service operation (e.g. AF using the PCC framework to indicate its subscriptions to SMF events), by passing the 3gpp-Sbi-Consumer-Info header value within an attribute to the NF service producer (e.g. in a PCC rule); or
- b) the NF service consumer's subscription is indicated to the NF service producer as a service parameter data stored in the UDR (acting as the intermediate NF) by passing the 3gpp-Sbi-Consumer-Info header value within an attribute to the UDR.

Details are to be specified by each such API.

Attributes used for the representation of a resource, particular values in enumerated data types, and/or procedural description can be marked to relate to a particular supported feature. Such attributes shall not be mandated in data structures. Such attributes or enumerated values shall only be sent and such procedures shall only be applied if the corresponding feature is supported.

Unknown attributes and values shall be ignored by the receiving entity. Unsupported query parameters shall be handled as specified in clause 5.2.9.

NOTE 2: The sender can send such information before the supported features for a resource have been determined.

For an API that does not define any resources, only custom operations without associated resources or notifications without subscription will be used. For such APIs, if a feature negotiation is desired, the request and response bodies of a suitable custom operation or notification need to be defined in such a manner that an attribute of the SupportedFeatures data type defined in TS 29.571 [13] is included. The client invoking that custom operation or notification shall indicate its supported features for that API within the corresponding HTTP request. The data structures to be included in the HTTP request as defined for that API, shall include the attribute of the SupportedFeatures data type defined in TS 29.571 [13], preferably in the outermost data structure for cases where the body contains a complex structure with several layers of JSON objects. The server shall determine the supported features by comparing the supported features indicated by the client with its own supported features. Features that are supported both by the client and the server are supported for subsequent custom operations and notifications of that API. The server shall include the attribute of the SupportedFeatures data type defined in TS 29.571 [13] indicating those features in the successful response to the custom operation or notification. The data structures to be included in the HTTP response as defined for that API, shall include the attribute of the SupportedFeatures data type defined in TS 29.571 [13], preferably in the outermost data structure for cases where the body contains a complex structure with several layers of JSON objects. The client and server shall only use those supported features in subsequent communication of that API between each other until the supported feature negotiation performed as part of that communication yields a new result.

Additionally, a NF instance should register all the features it supports to the NRF, to enable NF Service Consumers to discover NF Service Producers supporting specific features. A NF instance should register all the features it supports as NF Service Consumer in the corresponding default notification subscription in its NF profile to the NRF, to enable NF Service Producer to select NF Service Consumer supporting specific features.

NOTE 3: A feature that is supported (and hence is registered by the NF instance in the NRF) can appear not to be of any relevance in some communication cases between NF Service Consumer and NF Service Producer. In such cases the NF Service Consumer is not prevented from hiding support of the feature to the NF Service Producer. However, feature support hiding is not recommended as a required NF behaviour.

Specific requirements for support of Indirect Communication with Delegated Discovery are specified in clause 6.10.6.

6.6.3 Vendor-specific extensions

Information elements sent on the 3GPP 5GC APIs should be extensible with vendor-specific data. The definition of JSON data structures using OpenAPI as Interface Definition Language (see OpenAPI Specification [9]) allows to extend by default any JSON object with additional member elements, as long as no specific directives are included in the schema definition preventing such extension (e.g., by setting "additionalProperties" to false).

NOTE 1: The only JSON data types that can be extended, by defining additional members, are JSON objects; simple data types (and arrays of items of simple data types) cannot be extended in this way.

However, in order to avoid duplication of member names inside a same object (see TS 29.501 [5], clause 5.2.4.2, for the requirement of uniqueness of member names in JSON objects), it is necessary to comply with a naming scheme for vendor-specific data elements, to avoid clashing names between vendors.

Vendor-specific member names in JSON objects shall be named in the following manner:

```
"vendorSpecific-nnnnnn": {
  ...
}
```

where the value "nnnnn" is a fixed 6-digit string, using the IANA-assigned "SMI Network Management Private Enterprise Codes" [18] value associated to a given vendor, and padding with leading digits "0" to complete a 6-digit value.

NOTE 2: The content (value) of those vendor-specific member elements, and their usage, is not to be defined by any of the 3GPP Technical Specifications. Also, the type of value assigned to these members is not defined by 3GPP, and therefore, they can be any of the types allowed in the JSON specification: objects, arrays, or simple types (string, number, Boolean, etc.). However, to allow future extensibility of these values, it is recommended that they are defined as objects.

EXAMPLE: The vendor-specific member name for vendor "3GPP" would be:

```
"vendorSpecific-010415": {
  ...
}
```

6.6.4 Extensibility for Query parameters

New query parameters may be defined after the OpenAPI freeze of the first 3GPP release that contains an API.

A new feature should be defined in the API for any query parameter added in a new version of the API or for any new functionality resulting in defining new query parameter(s). A single feature may be defined, if appropriate, when adding multiple query parameters in a new version of the API.

Prior to using such a query parameter in an HTTP request, the NF Service Consumer should determine, if possible, whether the query parameter is supported by the NF Service Producer, using the feature negotiation mechanism specified in clause 6.6.2.

NOTE 1: Not doing so could result in the NF Service Producer rejecting the request if it does not support the query parameters, see clause 5.2.9.

NOTE 2: A NF Service Consumer can discover the features (and therefore the query parameters) supported by a NF Service Producer using the NRF, if the latter has registered the features it supports to the NRF.

If the NF Service Consumer includes the query parameter (e.g. "supported-features") of the SupportedFeatures data type defined in TS 29.571 [13] in an HTTP GET request (see clause 6.6.2), the NF Service Producer shall include the attribute (e.g. "supportedFeatures") of the SupportedFeatures data type defined in TS 29.571 [13] in the HTTP GET response, set as defined for HTTP responses in clause 6.6.2, if supported by the API definition.

NOTE 3: This allows a NF Service Consumer to discover the features (and therefore the query parameters) supported by the NF Service Producer when the first interaction with the NF Service Producer is an HTTP GET request and the service was not discovered via the NRF, e.g. for a NF Discovery Request sent to the NRF.

NOTE 4: Some APIs are designed to allow returning the attribute of the SupportedFeatures data type defined in TS 29.571 [13] in the HTTP GET response, regardless of whether the query parameter of the SupportedFeatures data type defined in TS 29.571 [13] is present in the request.

If a NF Service Consumer uses such a query parameter in an HTTP GET request without prior knowledge of whether it is supported by the NF Service Producer, the NF Service Consumer shall be prepared to receive a successful response that may not match all the query parameters sent in the request, and to act accordingly. The NF Service Consumer may use the attribute of the SupportedFeatures data type defined in TS 29.571 [13] returned by the NF Service Producer in the HTTP GET response, if available, to determine the features (and thus query parameters) not supported by the NF Service Producer.

When defining new query parameters in a new version of an API, it needs to be checked that the addition of the query parameter does not cause backward compatibility problems with NF Service Producers complying with an earlier version of the API, e.g. if the query parameter is ignored in a HTTP GET request. Otherwise, it needs to be ascertained that the NF Service Consumer does not use such a query parameter without prior knowledge that it is supported by the NF Service Producer.

6.7 Security Mechanisms

6.7.1 General

The security mechanisms for service based interfaces are specified in clause 13 of TS 33.501 [17].

Security Protection Edge Proxy (SEPP), as specified in TS 33.501 [17], shall be used between service based interfaces across PLMNs. The NFs in a PLMN shall use the SEPP as a HTTP/2 proxy for the HTTP/2 messages that carry ":authority" pseudo header with a uri-host formatted as specified in clause 6.1.4.3.

6.7.2 Transport layer security protection of messages

As specified in clause 13.1 of TS 33.501 [17], TLS shall be used for the security protection of messages at the transport layer for the service based interfaces if network security is not provided by other means.

6.7.3 Authorization of NF service access

Clause 13 in TS 33.501 [17] specifies two alternative authorization mechanisms:

- Static authorization, which is based on local authorization policy at the NRF and the NF Service Producer (see clause 13.3.0);
- Token based authorization, when NRF acts as the Authorization Server and provides access (see clause 13.3.1).

As specified in clause 13.4.1 of TS 33.501 [17] OAuth 2.0 (see IETF RFC 6749 [22]) may be used for authorization of NF service access. All NFs and the NRF shall support the OAuth 2.0 authorization framework with "Client Credentials" grant type as specified in clause 4.4 of IETF RFC 6749 [22], except that there is no "Authorization" HTTP request header in the access token request.

The NRF shall act as the Authorization Server providing "Bearer" access tokens (IETF RFC 6750 [23]) to the NF service consumers to access the services provided by the NF service providers.

If an NF service (i.e. API) receives an OAuth 2.0 access token in the "Authorization" HTTP request header field, the NF service shall validate the access token, its expiry and its access scope before allowing access to the requested resource, as specified in clause 7 of IETF RFC 6749 [22].

The access scope required to get access to a given resource may be, based on local configuration of the NF service producer, either:

- the service name of the NF Service; this scope grants generic access to a given API, for those operations on resources that don't require a specific authorization, or
- both, the service name of the NF Service, and a string that uniquely represents the type of operation (e.g. create/modify/read), the resource and the service; those two scopes, together, grant access to those operations on resources that require a specific authorization.

An NF service consumer shall support OAuth 2.0.

For request/response semantics service operations and for the subscribe and unsubscribe operations of subscribe/notify semantics service operations, an NF service consumer may use OAuth 2.0 for the authorization of the API access, based on local configuration. The NF service consumer discovers the additional scopes (resource/operation-level scopes) that might be required to invoke a certain service operation, based on the authorization information registered in NRF by the NF service producer in its NF profile.

When OAuth2 authorization is used, the NF service consumer shall use the token received from NRF as a "Bearer" token and include it in the Authorization header of the HTTP service requests, as described in IETF RFC 6750 [23] clause 2.1.

An NF service producer shall decide to accept or reject an API request without the OAuth2.0 access token in the "Authorization" HTTP request header field, based on local configuration.

If an NF service producer rejects an API request without the OAuth2.0 access token or an API request with an invalid OAuth2.0 access token, it shall return the HTTP "401 Unauthorized" status code together with the "WWW-Authenticate" header as specified in IETF RFC 9110 [11], where:

- the scheme for challenge in the "WWW-Authenticate" header shall be set to "Bearer" (IETF RFC 6750 [23]),
- the "realm" attribute shall be set to the URI of the service (i.e. API URI) for which the access failed, in the case of request / response service operations,
- the "error" attribute shall be set to "invalid_token", as described in IETF RFC 6750 [23], if the request contained a token which was deemed as invalid by the NF service producer (e.g. it is expired, malformed...); if the request did not contain a token, the "error" attribute shall not be included.

If the request contains an invalid OAuth2.0 access token due to missing claims, the NF service producer may additionally include a ProblemDetails object in the 401 Unauthorized response with the cause attribute set to "ACCESS_TOKEN_CLAIM_MISSING" and with the invalidParams attribute indicating the missing claim name(s) necessary to access the protected resource (the invalidParams attribute may additionally indicate the "WWW-Authenticate" header as erroneous).

If the request contains an invalid OAuth2.0 access token due to claims mismatch, the NF service producer may additionally include a ProblemDetails object in the "403 Forbidden" response with the cause attribute set to "ACCESS_TOKEN_CLAIM_MISMATCH" and with the invalidParams attribute indicating the mismatching claim name(s) necessary to access the protected resource (the invalidParams attribute may additionally indicate the "WWW-Authenticate" header as erroneous).

If an NF service producer rejects an API request with an OAuth2.0 access token not having the required scopes to invoke the service operation, it shall return the HTTP "403 Forbidden" status code together with the "WWW-Authenticate" header, where:

- the scheme for challenge in the "WWW-Authenticate" header shall be set to "Bearer",
- the "realm" attribute shall be set to the URI of the service (i.e API URI) for which the access failed, in the case of request / response service operations,
- the "error" attribute shall be set to "insufficient_scope" as described in IETF RFC 6750 [23],
- the "scope" attribute shall be set with the scope(s) necessary to access the protected resource.

For the notify operation of subscribe/notify semantics service operations, in this release of this specification OAuth 2.0 access token is not used.

When an NF service consumer receives a "401 Unauthorized" or a "403 Forbidden" status code with a "WWW-Authenticate" header containing "Bearer" as the scheme for challenge it shall not repeat the same request without an OAuth2.0 access token or with an access token that has been already used. The NF service consumer may repeat the same request with a new OAuth 2.0 access token.

NOTE: If a NF service producer accepts a request without the OAuth 2.0 access token, based on local policy, it is assumed that such accesses are allowed based on trust relationships and hence full access to the resource as it would have been otherwise allowed, is provided.

6.7.4 Application layer security across PLMN

6.7.4.1 General

HTTP/2 messages sent across the PLMN between the SEPPs shall follow the application layer security procedures specified in clause 13.2 of TS 33.501 [17].

6.7.4.2 N32 Procedures

As specified in clause 13.2 of TS 33.501 [17], the following procedures shall be supported across N32

- Capability Negotiation Request and Response;
- Parameter Exchange Request and Response;
- forwarding of the JOSE (see IETF RFC 7516 [25] and IETF RFC 7515 [26]) protected messages over N32.

Based on the capability negotiation and parameters exchanged between the SEPPs, the service based interface messages sent across N32 interface shall be subjected to JOSE based protection (see IETF RFC 7516 [25] and IETF RFC 7515 [26]) as specified in clause 13.2.4 of TS 33.501 [17].

TS 29.573 [27] specifies protocol for the exchange of the messages described above over N32, the format of the JOSE (see IETF RFC 7516 [25] and IETF RFC 7515 [26]) protected messages and the procedure for forwarding of the JOSE protected messages over N32.

6.7.5 Client credentials assertion and authentication

6.7.5.1 General

The Client credentials assertion (CCA) and authentication procedure specified in clause 13.3.8 of TS 33.501 [17] may be used to enable the NRF or the NF Service Producer to authenticate the NF Service Consumer, when using indirect communication.

If so, an HTTP request shall include the 3gpp-Sbi-Client-Credentials header (see clause 5.2.3.2.11) containing the client credentials assertion. The verification of the client credentials assertion shall be performed by the receiving entity as specified in clause 13.3.8.3 of TS 33.501 [17].

If the verification of the CCA fails at the receiving entity (e.g. NRF or NF service producer), a "403 Forbidden" response shall be returned with the cause attribute set to "CCA_VERIFICATION_FAILURE".

If the subject claim (i.e., the NF Instance Id of the NF Service Consumer) in the access token does not match the subject claim in the CCA at the receiving entity (e.g. NRF or NF service producer), a "403 Forbidden" response shall be returned with the cause attribute set to "TOKEN_CCA_MISMATCH".

6.7.5.2 Authorization of NF Service Consumers for data access via DCCF

Requirements for authentication and authorization of NF Service Consumers for data access via DCCF are specified in clause X.2 of TS 33.501 [17].

From the perspective of the NF Service Producer, the NF Service Consumer defined in clause X.2 of TS 33.501 [17] correspond to the Source NF Instance, and the DCCF corresponds to the NF Service Consumer, defined in TS 29.510 [8] and in this specification.

The following requirements shall apply when the source NF Instance and/or the DCCF need to signal their own respective CCAs towards the NF Service Producer:

- In the service request from the source NF instance to the DCCF:
 - the 3gpp-Sbi-Client-Credentials shall convey the client credentials assertion of the source NF Instance.
- In the service request from the DCCF to the NF Service Producer:
 - the 3gpp-Sbi-Client-Credentials shall convey the client credentials assertion of the DCCF; and
 - the 3gpp-Sbi-Source-NF-Client-Credentials shall convey the client credentials assertion of the source NF Instance.

If the verification of the 3gpp-Sbi-Client-Credentials fails at the receiving entity (e.g. NRF or NF service producer), a "403 Forbidden" response shall be returned with the cause attribute set to "CCA_VERIFICATION_FAILURE".

If the verification of the 3gpp-Sbi-Source-NF-Client-Credentials fails at the receiving entity (e.g. NRF or NF service producer), a "403 Forbidden" response shall be returned with the cause attribute set to "SOURCE_NF_CCA_VERIFICATION_FAILURE".

If the subject claim (i.e., the NF Instance Id of the NF Service Consumer) in the access token does not match the subject claim in the 3gpp-Sbi-Client-Credentials at the receiving entity (e.g. NRF or NF service producer), a "403 Forbidden" response shall be returned with the cause attribute set to "TOKEN_CCA_MISMATCH".

If the sourceNFInstanceId claim (i.e., the NF Instance Id of the Source NF Instance) in the access token does not match the subject claim in the 3gpp-Sbi-Source-NF-Client-Credentials at the receiving entity (e.g. NRF or NF service producer), a "403 Forbidden" response shall be returned with the cause attribute set to "TOKEN_SOURCE_NF_CCA_MISMATCH".

If a source NF Instance requests the DCCF to access data for which the DCCF already created a subscription to the NF Service Producer, the DCCF shall proceed as defined in Figure X.2-1 of TS 33.501 [17], where the NF Service Request in step 10 shall be a PATCH or PUT request (as supported by the corresponding NF service) targeting the existing subscription, including the 3gpp-Sbi-Client-Credentials and the 3gpp-Sbi-Source-NF-Client-Credentials as described above. The PATCH or PUT request may modify the subscription to satisfy the requirements of the new Source NF Instance; if the current subscription already satisfies the requirements of the new Source NF Instance, the PATCH or PUT request may extend the expiry time of the subscription (see clause 4.6.2.2 of TS 29.501 [5]), or may request to patch an attribute of the subscription to its current value (for a PATCH request), or may provide the same subscription information (for a PUT request). The DCCF shall consider that the source NF Instance is authorized to access the data if the request is accepted.

6.7.5.3 Authorization of requesting ML models on behalf of another ML model consumer

The requirements for security for AI/ML model storage and sharing are specified in clause X.10 of TS 33.501 [17].

From the perspective of the NF Service Producer, the ML model consumer (e.g. NWDAF containing AnLF) defined in clause X.10 of TS 33.501 [17] correspond to the Source NF Instance, and the NWDAF containing MTLF, or in short MTLF, corresponds to the NF Service Consumer, defined in TS 29.510 [8] and in this specification.

The following requirements apply when the source NF Instance and/or the MTLF need to signal their own respective CCAs towards the NF Service Producer:

- In the service request from the source NF instance to the MTLF:
 - the 3gpp-Sbi-Client-Credentials conveys the client credentials assertion of the source NF Instance.
- In the service request from the MTLF to the NF Service Producer:
 - the 3gpp-Sbi-Client-Credentials shall convey the client credentials assertion of the MTLF; and
 - if received from the source instance NF, the 3gpp-Sbi-Source-NF-Client-Credentials shall convey the client credentials assertion of the source NF Instance.

The "403 Forbidden" response with the cause attribute values described in clause 6.7.5.2 is applicable.

6.8 SBI Message Priority Mechanism

6.8.1 General

The primary usage of SBI Message Priority (SMP) is to provide guidance to 5GC NF acting as HTTP/2 clients or servers and HTTP/2 proxies when making throttling decisions related to overload control. The priority information may also be used for routing in proxies. Eventually a server may use the priority information to process higher-priority requests before lower-priority requests.

The SMP mechanism defined in this clause uses the "3gpp-Sbi-Message-Priority" custom HTTP header defined in clause 5.2.3.2.1 to set and carry the message priority between the client and the server.

NOTE: The custom HTTP header enforces the message priority end to end between the client and the server through one or more proxies.

The SMP mechanism should not use the stream priority mechanism, because that is deprecated in IETF RFC 9113 [7] clause 5.3.

HTTP/2 clients, servers and proxies implementing SBIs shall support the custom HTTP header and may support the stream priority.

6.8.2 Message level priority

A client, proxy and server shall use the "3gpp-Sbi-Message-Priority" value (see clause 5.2.3.2.1) when setting or evaluating the priority of a message.

The client shall assign the request priority by adding the "3gpp-Sbi-Message-Priority" custom HTTP header (see clause 5.2.3.2.1) to the message and setting its value.

If the "3gpp-Sbi-Message-Priority" custom HTTP header is not present in a response message then the HTTP nodes shall use the priority indicated in the "3gpp-Sbi-Message-Priority" of the associated request message.

If the server wants to assign a different priority to the response message than the request one then the server shall assign the response priority by adding the "3gpp-Sbi-Message-Priority" custom HTTP header to the message and setting its value.

6.8.3 Stream priority

The purpose of HTTP/2 stream priority is to allow an endpoint to prioritize streams for transmitting frames when there is limited capacity for sending and to express how it would prefer its peer to allocate resources when managing concurrent streams. Setting the stream priority indicates a priority treatment to a message between the two endpoints of an HTTP/2 connection.

Clause 5.3 of the IETF RFC 9113 [7] deprecates the HTTP/2 stream priority signaling.

For interoperability with implementations complying with earlier releases, 5GC NFs shall support receiving priority info in HEADER or PRIORITY frames to the extent that headers or frames containing stream priority information shall not be rejected, but processed as if they contained no stream priority information, i.e. the stream priority information is ignored.

NOTE: Server implementations can alternatively use the stream priority information as specified in clause 5.3.2 of IETF RFC 9113 [7], but it is recommended to use the SBI Message Priorities instead.

6.8.4 Recommendations when defining SBI Message Priorities

The recommendations provided in this clause are compliant with clause 10 of IETF RFC 7944 [19]. The priority value range defined over SBIs spans from 0 to 31 (decimal), where 0 indicates the highest priority and 31 indicates the lowest priority. The recommendations have been adapted to 5G services and Service Based Architecture.

The priorities defined for all messages across all SBIs used in an HTTP/2 administrative domain must be defined in a consistent and coordinated fashion, taking the default priority (see below for default priority values) into account.

The following are some guidelines to be considered when defining the SMPs to be used in SBA networks that support HTTP/2 nodes handling multiple services.

- As with any prioritization scheme, it is possible for higher-priority messages to block lower-priority messages from ever being handled. In 5GC, this will often result in the messages being retried. This may result in more traffic than the network would have handled without use of the SMP mechanism.

One potential guideline to prevent unwanted starving of lower-priority messages is to have higher-priority messages represent a relatively small portion of messages handled by the 5GC under normal scenarios. The Multimedia Priority Service (see TS 23.501 [5] clause 5.16.5) and the Mission Critical Service (see TS 23.501 [5] clause 5.16.6) typically generate little traffic compared to the total traffic of a 5GC.

The Multimedia Priority Service (MPS) and the Mission Critical Service (MCX) require the blocking of lower-priority services.

- A network entity (e.g., the AMF) that has received an RRC Establishment Cause associated with priority service (e.g., mps-PriorityAccess, mcs-PriorityAccess, or highPriorityAccess) or has determined that the UE has a priority subscription (e.g., MPS, MCX) in the UDM/UDR, shall select an SMP value appropriate for the priority service (e.g., MPS, MCX) for use in requests and responses.

NOTE 1: Other situations (e.g., certain ARP, 5QI Priority Level or 5QI values) and/or other network elements (e.g., UDM, PCF, etc.) can also set the SMP to an appropriate level for a priority service.

- When setting priorities for Multimedia Priority Services, Mission Critical Services or Emergency calls, it is important to use the same priority values across all APIs and services exposed by the 5GC. For instance, if the MPS priority levels of [1; n] are assigned the values of [k; k+n-1], then the same values shall be defined in the same order on all SBIs for the same service.
- Messages related to MPS, MCX and Emergency calls may be ranked according to their priority (e.g., based on ARP priority level, 5QI priority level, MPS Priority) based on regional/national regulatory and operator policies when it is known by the application sending the message. Otherwise MPS and MCX should have higher priorities than Emergency calls. Emergency call related messages should have higher priority than the rest of the messages. The ranking can be used to select SMP values.

NOTE 2: In some situations (e.g. REGISTRATION or SERVICE REQUEST procedure); it is possible to identify that the message belongs to a procedure of a high priority user without knowing the identity of the priority service. In such a case, all the messages sent over an SBI of these high priority procedures should be given the same SBI message priority.

The following are general requirements:

- Requests without the "3gpp-Sbi-Message-Priority" header shall be assigned the default priority value of "24".
- When defining priorities of the messages of a service, the same rules apply independently of the application, the SBI and the service:
 - When there is a series of request/response required to complete a procedure, it is appropriate to mark request/responses that occur later in the series at a higher priority than those that occur early in the series.
 - The requests that establish new sessions should have a lower priority than the requests that update or end a session.
 - The requests that will result in deregistering users if they failed (e.g., due to authentication vector retrieval, or update location) shall have a higher priority than the requests of a non-registered user.
 - Request/responses of optional procedures and of delay tolerant services should have lower priority than those of mandatory procedures.

6.8.5 HTTP/2 client behaviour

The client sending a request shall determine its required priority according to 6.8.4. It shall include a "3gpp-Sbi-Message-Priority" header (see clause 5.2.3.2.1) indicating the required priority level in the request and shall prioritise the requests according to the required priority level. If the client also uses the stream priority at the HTTP/2 connection level then it shall map the header value into a Weight and include it in the HEADERS of the request message.

When the client receives a response with the "3gpp-Sbi-Message-Priority" header, it shall prioritise the received response according to the priority level received, otherwise according to the priority level of the corresponding request. This includes determining the order in which responses are handled and resources that are applied to the handling of the responses.

6.8.6 HTTP/2 server behaviour

The server should use the "3gpp-Sbi-Message-Priority" header (see clause 5.2.3.2.1) to determine how to handle the request. This includes determining the order in which requests are handled and resources that are applied to the handling of the request.

Servers should use "3gpp-Sbi-Message-Priority" value when making overload throttling decisions.

Servers should not use stream priority information when making overload throttling decisions at the connection level.

When the priority of the response message needs to have a different value than the request, a server shall include a "3gpp-Sbi-Message-Priority" header in the response message which value is set to the response required priority level.

A server should not set the stream priority as described in IETF RFC 9113 [7], via priority information in the HEADERS frame or in a PRIORITY frame.

6.8.7 HTTP/2 proxy behaviour

A proxy should forward request and response without removing the "3gpp-Sbi-Message-Priority" header or changing its value.

While done only in exceptional circumstances, a proxy may modify priority information when relaying request and response by changing the "3gpp-Sbi-Message-Priority" value. For example, a SEPP may modify the priority set by a roaming partner.

Proxies should use the request priority information (respectively response priority information) according to the "3gpp-Sbi-Message-Priority" value when making overload throttling decisions to a request (respectively a response).

Proxies may use the priority information according to the "3gpp-Sbi-Message-Priority" value when relaying a request or a response messages. This includes the selection of routes (only for the requests) and the ordering of messages relayed.

6.8.8 DSCP marking of messages

A client, proxy or server may prioritize traffic at IP level by placing messages into different traffic classes and marking them with an appropriate Differentiated Services Code Point (DSCP).

Multiple HTTP/2 connections between two HTTP/2 end points are necessary: one per DSCP value. All messages sent over a connection are assigned the same traffic class and receive the same DSCP marking. The "3gpp-Sbi-Message-Priority" value shall be considered in the selection of the appropriate connection to send the message.

6.9 Discovering the supported communication options

6.9.1 General

The OPTIONS method, as described in clause 9.3.7 of IETF RFC 9110 [11], may be used by a NF Service Consumer to determine the communication options supported by a NF Service Producer for a target resource.

Clause 6.9.2.1 describes example communication options that may be discovered using the OPTIONS method.

The Accept-Encoding header, as described in clause 12.5.3 of IETF RFC 9110 [11], may be used by a NF Service Producer to determine the communication options supported by a NF Service Consumer.

Clause 6.9.2.2 describes example communication options that may be discovered using the Accept-Encoding header.

6.9.2 Discoverable communication options

6.9.2.1 Content-encodings supported in HTTP requests

Certain service operations may result in large HTTP request contents, e.g. to register NF profiles in the NRF (see TS 29.510 [8]) or to update the NSSF with the available S-NSSAIs supported by Tracking Areas (see TS 29.531 [32]). Gzip coding (see IETF RFC 1952 [34]) may be supported to optimally reduce the content size of HTTP requests in this case.

A NF Service Consumer may determine the content-encodings supported by the NF Service Producer in HTTP requests targeting a particular resource by:

- sending an HTTP OPTIONS request targeting the resource of the NF Service Producer; and/or

- receiving an "Accept-Encoding" header in HTTP responses from the NF Service Producer for requests targeting the resource.

A NF Service Producer that receives an HTTP OPTIONS request for a target resource shall include an "Accept-Encoding" header in the HTTP 200 OK response (see IETF RFC 9110 [11]), if specific content-encodings, e.g. Gzip coding (e.g. see IETF RFC 1952 [34]) are supported in HTTP requests targeting the resource.

A NF Service Producer that receives an HTTP request with a content-encoding that it does not support shall reject the request with the status code "415 Unsupported Media Type" and include an "Accept-Encoding" header in the response indicating the supported encodings in HTTP requests, as described in clause 12.5.3 of IETF RFC 9110 [11].

A NF Service Producer may include an "Accept-Encoding" header in the HTTP 2xx response for requests other than HTTP OPTIONS if specific content-encodings, e.g. Gzip coding (e.g. see IETF RFC 1952 [34]), are supported in HTTP requests targeting the resource, to optimize future interactions, e.g. when the request content was big enough to justify use of a compression coding but the client did not do so.

For notification requests, a NF Service Producer may determine the content-encodings supported by the NF Service Consumer from the 3gpp-Sbi-Notif-Accepted-Encoding HTTP header, defined in clause 5.2.3.3.6, included in the received subscription request.

6.9.2.2 Content-encodings supported in HTTP responses

Certain service operations may result in large HTTP response contents, e.g. to send NF profiles by the NRF (see TS 29.510 [8]) or to send the available S-NSSAIs supported by Tracking Areas by the NSSF (see TS 29.531 [32]). Gzip coding (see IETF RFC 1952 [34]) may be supported to optimally reduce the content size of HTTP responses in this case (see "Content-Encoding" header in Table 5.2.2.2-2).

A NF Service Consumer may include an "Accept-Encoding" header in HTTP requests to indicate the content-encodings, e.g. Gzip coding (e.g. see IETF RFC 1952 [34]), that are supported for the associated HTTP responses, as specified in Table 5.2.2.2-1 and in clause 12.5.3 of IETF RFC 9110 [11].

A NF Service Producer may determine the content-encodings supported by the NF Service Consumer in HTTP responses by receiving an "Accept-Encoding" header in the associated HTTP requests from the NF Service Consumer.

6.10 Support of Indirect Communication

6.10.1 General

NF Service Consumers and NF Service Producers may support or be configured to use Indirect Communication models via SCP as specified in clauses 6.3 and 7.1 of TS 23.501 [3]. This clause defines specific requirements to support Indirect Communication models. Specific requirements to support indirect communication between different PLMNs with or without delegated discovery with possible NF selection at target PLMN are defined in clause 6.10.12.

An SCP may be known to the NF (e.g. SCP based on independent deployment units) or not (e.g. SCP based on service mesh, with co-located NF and SCP within the same deployment unit). If the SCP is known to the NF, the NF shall be configured with a scheme, authority, and optionally a deployment-specific prefix of the SCP. The scheme may be "http" or "https". If the scheme is "https" then the authority shall contain an FQDN and not a literal IP address. If the scheme is "http" then the authority shall contain either an FQDN or a literal IP address. In either case, the authority may optionally contain a port number. If the SCP is known to the NF, but the NF is not configured with a deployment-specific prefix of the SCP, the NF shall consider the deployment-specific prefix of the SCP to be empty. If the SCP is unknown to the NF, the NF may still be configured to use delegated discovery through the unknown SCP as detailed in Clause 6.10.2A.

NOTE: See Annex G of TS 23.501 [3] for SCP deployment examples.

Indirect Communication models shall support the same level of security as Direct Communication ones. Security requirements for Indirect Communications are specified in clauses 5.9.2.4 and 13.3 of TS 33.501 [17]. TLS shall be used between the SCP and NFs, if network security is not provided by other means. When co-located, the NF and associated SCP may interact using HTTP. Clause 6.7.5 specifies how to support the client credentials assertion and authentication procedure specified in clause 13.3.8 of TS 33.501 [17].

More than one SCP may be present in the communication path between an NF Service Consumer and an NF Service Producer. Clauses 6.2.19 and 6.3.16 of TS 23.501 [3] describe how to route messages through SCPs.

6.10.2 Routing Mechanisms with SCP Known to the NF

6.10.2.1 General

The routing mechanisms specified in clause 6.1 shall apply for Indirect Communication models with the additions or modifications specified in this clause. This routing mechanism shall be used when TLS is used between the NF and the SCP, or between two SCPs. This routing mechanism may also be used when TLS is not used, i.e. when network security is provided by other means.

6.10.2.2 HTTP/2 connection management

Separate HTTP(S) connections shall be set up between the HTTP client and the SCP, between SCPs (if there is more than one SCP in the communication path between the HTTP client and the HTTP server), and between the SCP and the HTTP server. HTTP CONNECT requests shall not be used for Indirect Communication models.

The NFs and the SCP shall manage the HTTP/2 connections as defined in clause 5.2.6.

6.10.2.3 Connecting inbound

If the request is not satisfied by a local cache, the NF (acting as an HTTP/2 client) shall connect inbound by establishing (or reusing) a connection to an available SCP as defined in clause 7.3.2 of IETF RFC 9110 [11] when sending HTTP/2 request.

When forwarding a request to another SCP, an SCP shall connect inbound by establishing (or reusing) a connection to the next hop SCP.

When the SCP forwards the request to the HTTP server, the SCP (acting as an HTTP/2 client) shall connect inbound to an authority server for the target resource. For connecting inbound to an authority not in the same PLMN, the SCP shall connect to the Security Edge Protection Proxy.

6.10.2.4 Pseudo-header setting

For Indirect Communications with or without delegated discovery, when sending a request to the SCP, the HTTP client shall set the pseudo-headers as follows:

- ":scheme" set to "http" or "https";
- ":authority" set to the FQDN or IP address of the SCP (if the scheme is "http"), or to the FQDN of the SCP (if the scheme is "https");
- ":path" including the optional deployment-specific string of the SCP and the path and query components of the target URI excluding the optional deployment-specific string of the target URI.

An HTTP client which has not received information whether the callback URI contains any deployment specific string or not shall behave assuming that there is no deployment specific string in the callback (i.e. target) URI. If the HTTP client has previously received the prefix of the callback URI it shall include it, if available, in the 3gpp-Sbi-Target-apiRoot header (see clause 6.10. 2.5).

When an HTTP client sending a notification request corresponding to default notification subscription where the target URI is unknown (e.g. for Indirect Communication with Delegated Discovery, as specified in clause 6.10.3.3), it shall include the optional deployment-specific string of the SCP and the pseudo target URI for default subscription ("/scp-default-sub-notify-uri") in the ":path".

Additionally, for HTTP requests for which an HTTP client may cache responses (e.g. GET request), the HTTP client should include the cache key (ck) query parameter set to an implementation specific value that is bound to the target NF (see clause 6.10.2.6).

The HTTP client shall include the apiRoot of an authority server for the target resource (including the optional deployment-specific string of the target URI), if available, in the 3gpp-Sbi-Target-apiRoot header (see clause 6.10. 2.5).

When forwarding a request to another SCP, an SCP shall replace the apiRoot of the SCP received in the request URI of the incoming request by the apiRoot of the next hop SCP. The SCP shall include a 3gpp-Sbi-Target-apiRoot header set

to the apiRoot of an authority server for the target resource (including the optional deployment-specific string of the target URI), if available, e.g. if the 3gpp-Sbi-Target-apiRoot header was received in the request. The SCP shall set the pseudo-headers as specified in clause 6.1, with the following additions:

- the SCP shall modify the ":authority" HTTP/2 pseudo-header field to the FQDN or IP address of the next hop SCP (if the scheme is "http"), or to the FQDN of the SCP (if the scheme is "https").
- the SCP shall remove any optional deployment-specific string of the SCP in the ":path" HTTP/2 pseudo-header and add any optional deployment-specific string of the next hop SCP;
- the SCP shall remove the cache key query parameter, if this parameter was received in the request;
- if the pseudo target URI for default subscription ("/scp-default-sub-notify-uri") is present in the ":path", the SCP shall replace it with the real path of the target URI registered in the selected default subscription.

When forwarding a request to the HTTP server, the SCP shall replace the apiRoot of the SCP received in the request URI of the incoming request by the apiRoot of the target NF service instance. If the 3gpp-Sbi-Target-apiRoot header was received in the request, the SCP shall use it as the apiRoot of the target NF service instance, if the SCP does not (re)select a different HTTP server, and regardless shall remove it from the forwarded request. The SCP shall set the pseudo-headers as specified in clause 6.1, with the following additions:

- the SCP shall modify the ":authority" HTTP/2 pseudo-header field to the FQDN or IP address of the target NF service instance (if the scheme is "http"), or to the FQDN of the target NF service instance (if the scheme is "https").
- the SCP shall remove any optional deployment-specific string of the SCP in the ":path" HTTP/2 pseudo-header and add any optional deployment-specific string of the target URI;
- the SCP shall remove the cache key query parameter, if this parameter was received in the request;
- if the pseudo target URI for default subscription ("/scp-default-sub-notify-uri") is present in the ":path", the SCP shall replace it with the real path of the target URI registered in the selected default subscription.

EXAMPLE 1: For indirect communication without delegated discovery, if the NF Service Consumer needs to send the request "GET https://example.com/a/b/c/nudm-sdm/v1/{supi}/nssai" to the NF Service Producer (represented by the FQDN "example.com" and where "/a/b/c" is the "apiPrefix" of the NF service producer figured out from NRF discovery):

- the NF service consumer shall send the request "GET https://scp.com/1/2/3/nudm-sdm/v1/{supi}/nssai" to the SCP (where "/1/2/3" is the "apiPrefix" of the SCP), with the "3gpp-sbi-target-apiRoot" header set to "https://example.com/a/b/c".
- the SCP shall send the request "GET https://example.com/a/b/c/nudm-sdm/v1/{supi}/nssai" to the NF Service Producer, without any "3gpp-sbi-target-apiRoot" header.

EXAMPLE 2: For indirect communication, if the NF Service Producer needs to send a notification request "POST https://example.com/a/b/c/notification" to the NF Service Consumer (represented by the FQDN "example.com", i.e. the host part of the callback URI):

- the NF service producer shall send the request "POST https://scp.com/1/2/3/a/b/c/notification" to the SCP (where "/1/2/3" is the "apiPrefix" of the SCP), with the "3gpp-sbi-target-apiRoot" header set to "https://example.com".
- the SCP shall send the request "POST https://example.com/a/b/c/notification" to the NF Service Consumer, without any "3gpp-sbi-target-apiRoot" header.

EXAMPLE 3: For indirect communication with Delegated Discovery, if the NF Service Producer needs to send a notification request to a default subscription and SCP selects a target default notification subscription (with callback URI "https://example.com/a/b/c/notification" registered):

- the NF service producer shall send the request "POST https://scp.com/1/2/3/scp-default-sub-notify-uri" to the SCP (where "/1/2/3" is the "apiPrefix" of the SCP).
- the SCP shall send the request "POST https://example.com/a/b/c/notification" to the selected NF Service Consumer.

EXAMPLE 4: For indirect communication, if the NF Service Producer needs to send a notification request "POST https://example.com/prefix123/a/b/c/notification" to the NF Service Consumer with a callback URI Prefix "/prefix123":

- the NF service producer shall send the request "POST https://scp.com/1/2/3/a/b/c/notification" to the SCP (where "/1/2/3" is the Prefix of the SCP), with the "3gpp-sbi-target-apiRoot" header set to "https://example.com/prefix123".
- the SCP shall send the request "POST https://example.com/prefix123/a/b/c/notification" to the NF Service Consumer, without any "3gpp-sbi-target-apiRoot" header.

6.10.2.5 3gpp-Sbi-Target-apiRoot header setting

For Indirect Communications with or without delegated discovery, the HTTP client shall include a 3gpp-Sbi-Target-apiRoot header set to the apiRoot of an authority server for the target resource, if available, in requests it sends to the SCP. In particular:

- for Indirect Communication without Delegated Discovery, a service request sent to the SCP to create a resource shall include a 3gpp-Sbi-Target-apiRoot header set to the apiRoot of the selected NF service instance of the NF Service Producer, if the NF Service Consumer has indeed selected a specific NF service instance;
- after a resource has been created, subsequent service requests sent to the SCP and targeting the resource shall include a 3gpp-Sbi-Target-apiRoot header set to the apiRoot received earlier in Location header of service responses from the NF Service Producer;
- notifications or callbacks sent via the SCP shall include a 3gpp-Sbi-Target-apiRoot header set to the apiRoot of the notification or callback URI (i.e. "http" or "https" scheme, the fixed string "://" and authority (host and optional port) as defined in IETF RFC 3986 [14] followed by the Callback URI prefix when available).

An SCP shall include a 3gpp-Sbi-Target-apiRoot header set to the apiRoot of an authority server for the target resource, if available, in requests it sends to the next hop SCP. In particular:

- if the received request does not include a 3gpp-Sbi-Target-apiRoot header containing the apiRoot of a selected NF service instance, and NF service discovery is not delegated to a next hop SCP, then the SCP shall select a target NF service instance (performing an NF service discovery with the NRF or based on local configuration (i.e. without interacting with NRF) according to the received "3gpp-Sbi-Discovery-*" request header(s)) and insert a 3gpp-Sbi-Target-apiRoot header set to the apiRoot of the selected target NF service instance;
- if the received request includes a 3gpp-Sbi-Target-apiRoot header containing the apiRoot of a selected NF service instance, but the SCP needs to reselect a different NF service instance, the SCP shall modify and set the 3gpp-Sbi-Target-apiRoot header to the apiRoot of the newly selected target NF service instance;
- if the received request includes a 3gpp-Sbi-Target-apiRoot header containing the apiRoot of a selected NF service instance and the SCP does not reselect a different NF service instance, the SCP shall forward the received 3gpp-Sbi-Target-apiRoot header to the next hop SCP.

When forwarding the request to the HTTP server, the SCP shall set the pseudo-headers as specified in clause 6.10.2.4.

6.10.2.6 Cache key (ck) query parameter

The cache key (ck) query parameter may be used by cache systems in the NF service consumer and/or in the SCP in order to distinguish cache objects.

It shall be set to a string value that is linked to the apiRoot of the target NF, i.e. the same apiRoot shall always produce the same value for the content of the ck parameter. The ck parameter may contain e.g. a short compact hash value of the whole apiRoot of the target NF.

The ck parameter shall only be used in HTTP requests between the NF service consumer and the SCP, it shall not be sent to the actual NF service producer.

The ck parameter is not part of the actual service definition and therefore it is not documented in OpenAPI specification files. It shall comply with the following OpenAPI definition:

```
paths:
  /<resource>:
```

```

<method>:
  parameters:
    - name: ck
      in: query
      description: cache key parameter
      schema:
        type: string

```

6.10.2A Routing Mechanism with SCP Unknown to the NF

6.10.2A.1 Connecting inbound

When indirect communication models are used and a NF sends an HTTP/2 request, this NF (acting as an HTTP/2 client) shall connect directly to an authority for the target resource via an available SCP, which then acts as an "interception proxy" as defined in clause 2.5 of IETF RFC 3040 [36] and also referred to in clause 3.7 of IETF RFC 9110 [11]. This routing mechanism is incompatible with and shall not be used when TLS is used between the NF and the SCP.

6.10.2A.2 HTTP/2 connection management

The NF shall manage the HTTP/2 connections as defined in clause 5.2.6.

6.10.2A.3 Pseudo-header setting

The NF service consumer shall set the following pseudo-headers:

- ":scheme" pseudo-header shall be set to "http";

NOTE: When the SCP is implemented using service mesh technologies (e.g. as described in Annex G.2 in TS 23.501 [3]), the SCP needs to be able to read the start line and the header fields of HTTP/2 messages, and https cannot be used by NFs. In this case, mutual authentication and TLS between a NF and its associated SCP can be implicit by physical security; mutual authentication and TLS is then set up between SCP interfaces.

- ":authority" pseudo-header shall be set as follows:
 - if delegated discovery is used and has not yet been performed by the SCP, or if the NF Service Consumer only selects a target NF (service) set when in Indirect Communication without delegated discovery: set based on local configuration.
 - if delegated discovery is not used or has already been performed by the SCP: set as specified in clause 6.1.4.
- ":path" pseudo-header shall include the path and query components of the target URI as specified in clause 6.1.4.

6.10.3 NF Discovery and Selection for indirect communication with Delegated Discovery

6.10.3.1 General

This clause specifies the requirements that shall apply when the discovery and associated selection of NF instances or NF service instances is delegated to an SCP (see clause 6.3 and Model D in Annex E of TS 23.501 [3]).

6.10.3.2 Conveyance of NF Discovery Factors

When the NF service consumer is configured to use delegated service discovery, it shall include in the HTTP/2 request message the necessary NF service discovery factors (see clauses 6.2.3.2.3.1 and 6.3.5.2.2 of TS 29.510 [8]) to be used by the SCP to perform the NF service discovery procedures and the Service access authorization procedures (see clause 13.4.1.3.2 of TS 33.501 [17]) on behalf of the NF service consumer. The latter shall convey these NF service discovery factors using the "3gpp-Sbi-Discovery-*" request headers. How to set the values of these "3gpp-Sbi-Discovery-*" request headers is detailed in clause 5.2.3.2.7. The NF service consumer should also include at least the target NF type and service name in the corresponding "3gpp-Sbi-Discovery-*" request header(s) in its request to the

SCP. The NF service consumer may indicate the NRF to use, e.g. as a result of an NSSF query, by including the 3gpp-Sbi-Nrf-Uri header with the NRF API URIs.

If the NF service consumer delegates the reselection of a target NF service instance to the SCP (see clause 6.5 of TS 23.527 [38]), the NF service consumer shall also include "3gpp-Sbi-Discovery-*" headers in an HTTP/2 request targeting an existing resource context in the NF service producer, if the "3gpp-Sbi-Routing-Binding" header is not included in the HTTP/2 request message (e.g. when no binding information was received from the NF service producer during the resource creation, or if the NF service consumer does not support the binding procedures), to enable the SCP to reselect an NF service producer instance, e.g. if the NF service producer instance indicated in the "3gpp-Sbi-Target-apiRoot" header or target URI is not reachable. Additionally, regardless of whether a 3gpp-Sbi-Routing-Binding header is included or not in the HTTP/2 request message, the NF service consumer should include at least the target NF type and the service name in the corresponding "3gpp-Sbi-Discovery-*" request header(s) in its request to the SCP.

NOTE 1: Other 3gpp-Sbi-Discovery-*" request header(s) can also be included in any service request sent to an SCP, regardless of whether the 3gpp-Sbi-Routing-Binding" header is included or not in the HTTP/2 request message, to convey requester's information necessary for the NRF to validate whether the requester is allowed to discover and access a given NF (see NOTE 12 of Table 6.2.3.2.3.1-1 of TS 29.510 [8]).

NOTE 2: A request including a 3gpp-Sbi-Routing-Binding header needs not include the requested S-NSSAI in the corresponding 3gpp-Sbi-Discovery-*" request header, since if the NF service producer supports different sets of NF service instances serving different network slices, the NF Service Set ID in the binding indication is available for reselecting an NF service instance (see clauses 5.2.3.2.5 and 6.12.1).

If the NF service consumer includes more than one service name in the 3gpp-Sbi-Discovery-service-names header, the service name corresponding to the service request shall be listed as the first service name in the header.

NOTE 3: The SCP can assume that the service request corresponds to the first service name in the header.

An NF service consumer should also include "3gpp-Sbi-Discovery-*" headers in an HTTP/2 request targeting an existing resource context in the NF service producer to enable the SCP to perform the Service access authorization procedures (see clause 13.4.1.3.2 of TS 33.501 [17]).

Likewise, an NF service producer may also include 3gpp-Sbi-Discovery-*" headers in a notification or callback request, if the "3gpp-Sbi-Routing-Binding" header is not included in the HTTP/2 request message, to enable the SCP to reselect a different NF service consumer instance, e.g. if the NF service consumer instance indicated in the "3gpp-Sbi-Target-apiRoot" header or target URI is not reachable. Additionally, regardless of whether a 3gpp-Sbi-Routing-Binding header is included or not in the HTTP/2 request message, the NF service producer should include at least the target NF type (i.e. the type of the NF service consumer) in the corresponding "3gpp-Sbi-Discovery-*" request header(s) in its request to the SCP, if available. See clause 6.6 of TS 23.527 [38].

When the 3gpp-Sbi-Selection-Info header is included in a HTTP request message and if the SCP supports this header, the SCP shall use it together with 3gpp-Sbi-Routing-Binding or 3gpp-Sbi-Discovery-* heads whichever available.

Based on SCP configuration, an SCP deciding to address a next-hop SCP for a service request may delegate the NF instance and/or service instance discovery and selection to subsequent SCPs, in which case it shall forward the "3gpp-Sbi-Discovery-*" request headers to the next-hop SCP.

When receiving a request containing "3gpp-Sbi-Discovery-*" request headers and a selection/reselection of the target NF service instance is required, the SCP shall take into account all the NF service discovery factors contained in the "3gpp-Sbi-Discovery-*" request headers to perform the selection or reselection. The SCP should use the NRF indicated in the 3gpp-Sbi-Nrf-Uri header if this header is present in the request. It is also possible for the SCP to be internally configured to fulfil these service discovery tasks without interacting with the NRF.

If the service request contains "3gpp-Sbi-Discovery-*" request header(s) that are not supported by the SCP, the latter should include the corresponding query parameters in the discovery request to the NRF. Based on operator policy, the SCP may alternatively reject the request and return a response with the status code "400 Bad Request" to the NF service consumer with an "INVALID_DISCOVERY_PARAM" error.

If the service request does not contain the 3gpp-Sbi-Discovery-preferred-api-versions header, the SCP shall select an NF instance and/or service instance that supports the MAJOR version received in the request URI of the service request message. Otherwise, the preferred API MAJOR version included in the 3gpp-Sbi-Discovery-preferred-api-versions header shall be the same as the MAJOR version of the request URI of the service request message. The SCP shall reject the request and return a response with the status code "400 Bad Request" to the NF service consumer with an

"INVALID_API" error if no NF profile is found matching the MAJOR version; in this case, the SCP may indicate in the problem details the MAJOR API version(s) known to be supported by NF service producers for the corresponding service.

NOTE 4: When delegated discovery and/or reselection is enabled, the NF consumer does not include the "3gpp-Sbi-Discovery-*" headers which indicate constraints on the NRF discovery result such as content encoding or size which are specific to the NRF consumer (e.g. 3gpp-Sbi-Discovery-limit, 3gpp-Sbi-Discovery-max-payload-size, 3gpp-Sbi-Discovery-requester-features, etc.) in the request sent to the SCP. If such headers are received in the request, the SCP can either discard them or adjust the value in the corresponding query parameters (according to the SCP capabilities) in the discovery request sent to the NRF for delegated discovery and/or reselection.

6.10.3.3 Notifications corresponding to default notification subscriptions

An NF may register default notification subscriptions in its NF profile or NF services in the NRF for notifications the NF is prepared to consume, including for each type of notification the corresponding notification endpoint (i.e. callback URI).

NOTE: This can be used e.g. by an AMF to discover the notification endpoint of other AMFs to forward N1 or N2 messages, or by an AMF to notify location information to a GMLC, or by an UDR to notify data change or removal to an UDM.

A NF producer may be configured with the types of notifications corresponding to default notification subscriptions it needs to generate, and send such notifications using indirect communication with delegated discovery, i.e. with an SCP discovering and selecting an NF service consumer with a corresponding default notification subscription. In this case, the NF producer shall include in the notification request:

- the 3gpp-Sbi-Callback header including the name of the notify or callback service operation and the API major version if higher than 1, (see also clause 6.10.7);
- the 3gpp-Sbi-Discovery-notification-type header set to the type of notification being set;
- the 3gpp-Sbi-Discovery-n1-msg-class header set to the N1 Message Class of the target default subscription if notification type is "N1_MESSAGE", or the 3gpp-Sbi-Discovery-n2-info-class header set to the N2 Information Class of the target default subscription if the notification type is "N2_INFORMATION";
- the 3gpp-Sbi-Discovery-target-nf-type header indicating the type of the consumer NF; and
- optionally, additional NF service discovery factors header to be used by the SCP to discover and select the consumer NF.

The SCP shall use these 3gpp-Sbi-Discovery* headers to select/reselect a notification endpoint.

6.10.3.4 Returning the Producer's NF Instance ID and NF Group ID to the NF Service Consumer

The following requirements shall apply when using indirect communication with delegated discovery, or indirect communication without delegated discovery when the NF service consumer only selects an NF set and delegates the selection of the NF service instance to the SCP (see clause 6.10.5.1):

- an SCP that (re)selected the target NF service instance shall include the 3gpp-Sbi-Producer-Id header and, for indirect communication with delegated discovery, if the target NF service instance pertains to an NF Group, the 3gpp-Sbi-Target-Nf-Group-Id header, in the 2xx HTTP response it forwards towards the NF Service Consumer.

The 3gpp-Sbi-Producer-Id header shall contain the NF Instance ID and it should contain the NF Service Instance ID and, if the NF service instance pertains to an NF set and/or an NF service set, the NF set ID and/or NF service set ID of the NF Service Producer selected by the SCP.

The 3gpp-Sbi-Target-Nf-Group-Id shall contain the NF Group ID of the NF Service Producer selected by the SCP (see clause 6.10.3.2); if the SCP received a 4xx/5xx HTTP response including a 3gpp-Sbi-Response-Info header with "context-transferred" parameter set to value "true" from the reselected target NF service instance, which indicates the corresponding resource or context has been transferred to the reselected target NF service instance, the SCP shall also insert a 3gpp-Sbi-Producer-Id header and conditionally a 3gpp-Sbi-Target-Nf-

Group-Id header in the HTTP response it forwards to the NF Service Consumer.

If the SCP receives a service request including the 3gpp-Sbi-Retry-Info header set to "no-retries", and no successful response is received by the SCP after forwarding the request once, the SCP should include the 3gpp-Sbi-Producer-Id header, indicating the NF (service) instance ID that the SCP selected, in a 4xx/5xx HTTP response it sends towards the NF Service Consumer.

- If the 3gpp-Sbi-Producer-Id header or the 3gpp-Sbi-Target-Nf-Group-Id header is already present in an HTTP response (e.g. in scenarios with multiple SCPs between the NF service consumer and NF service producer), the SCP shall forward the respective header unmodified in the response towards the HTTP client (without adding any new such respective header).

NOTE 1: This allows to support deployments where not all NF Service Producers or NF Service Consumers have been upgraded to support the binding procedures.

NOTE 2: In scenarios where the same NF Service Producer needs to be selected when creating new resources, e.g. when the AMF needs to establish a new PDU session towards the same SMF as the one selected for a previous PDU session, the NF Service Consumer can include the 3gpp-Sbi-Discovery-target-nf-instance-id header set to the NF Instance ID of the NF Service Producer in the request creating the new resource.

NOTE 3: An SCP needs not insert a 3gpp-Sbi-Producer-Id header nor a 3gpp-Sbi-Target-Nf-Group-Id header in an HTTP response if it received a 3gpp-Sbi-Target-apiRoot header in the related HTTP request and it did not reselect a different NF Service Producer.

NOTE 4: Inserting the NF Service Instance ID, NF Set ID and/or NF Service Set ID in the 3gpp-Sbi-Producer-Id header enables NF service consumers to perform overload control towards a specific NF producer service instance, NF set or NF service set when the NF service producer advertises overload control with a scope set to a specific NF service instance, NF set or NF service set (see clause 6.4.3). It also enables NF service consumers to reselect another NF service instance in the same NF instance, NF set or NF service set when so required (see e.g. clause 6.12.1 and clause 6.5.3 of TS 23.527 [38]).

NOTE 5: If the NF Service Consumer sends a service request including the 3gpp-Sbi-Retry-Info header set to "no-retries" and receives a 4xx/5xx HTTP response from the SCP including the 3gpp-Sbi-Producer-Id header, the NF Service Consumer can retransmit the request to the SCP requesting the SCP to reselect an NF service producer possibly excluding the NF (service) instance ID that was indicated in the 3gpp-Sbi-Producer-Id header, by including corresponding instructions in the 3gpp-Sbi-Selection-Info (see clause 5.2.3.3.10).

6.10.3.5 Returning an Alternate CHF instance ID to the NF Service Consumer

The CHF may include the 3gpp-Sbi-Alternate-Chf-Id header in an HTTP response towards its NF Service Consumer, containing an alternate charging server identity (i.e. secondary CHF Instance ID of a primary CHF instance, or primary CHF Instance ID of a secondary CHF instance).

The following requirements apply when using indirect communication with delegated discovery, or indirect communication without delegated discovery when the NF service consumer only selects an NF set and delegates the selection of the NF service instance to the SCP (see clause 6.10.5.1):

- an SCP that selected a target CHF service instance may include the 3gpp-Sbi-Alternate-Chf-Id header in the HTTP response it forwards towards the NF Service Consumer, containing either the secondary CHF Instance ID of the primary CHF instance selected by the SCP, or containing the primary CHF Instance ID of the secondary CHF instance selected by the SCP;
- If the 3gpp-Sbi-Alternate-Chf-Id header is already present in an HTTP response (e.g. in scenarios with multiple SCPs between the NF service consumer and CHF service producer, or in scenarios where the header is already included by the CHF producer), the SCP shall forward the header unmodified in the response towards the HTTP client (without adding any new such header).

NOTE 1: Subsequently, if the CHF service consumer needs to reselect the alternate CHF instance, it can send its request with the 3gpp-Sbi-Discovery-target-nf-instance-id set to the alternate CHF instance ID and with no 3gpp-Sbi-Target-apiRoot header. This leads the SCP to route the request towards the secondary CHF instance, and the SCP includes in the response the 3gpp-Sbi-Target-apiRoot header set to the apiRoot of the alternate CHF instance as specified in clause 6.10.4.

NOTE 2: The SCP remains agnostic of applicative requirements on failure handling and retry handling. Accordingly, failure handling and retry handling is controlled by CHF's consumers.

6.10.4 Authority and/or deployment-specific string in apiRoot of resource URI

For Indirect Communications with or without delegated discovery, the SCP may select or reselect the specific NF (service) instance towards which to send a request.

NOTE 1: For Indirect Communications without delegated discovery, the SCP selects for instance a specific (service) instance within a NF (Service) Set that was selected by the NF Service Consumer.

Consequently, NF as HTTP client shall be capable to receive and process an authority and/or deployment-specific string in the apiRoot of the created or updated resource URI that differs from the authority and/or deployment-specific string of the apiRoot of the Request URI.

If the NF Service Producer includes a relative URI (see IETF RFC 3986 [14]) in the "Location" header of an HTTP response creating a resource, the SCP shall resolve the URI reference using the target URI included in the HTTP POST request sent to the NF Service Producer as base URI, and return an absolute URI in the "Location" header in the HTTP response sent to the NF Service consumer, unless the SCP did not change the target URI when forwarding the HTTP POST request from the NF Service Consumer to the NF Service Producer.

NOTE 2: The target URI can remain unchanged when forwarding an HTTP POST request from the NF Service Consumer to the NF Service Producer if indirect communication without delegated discovery and without TLS is used between the NF Service Consumer and the SCP, and the SCP uses the NF (service) instance of the NF Service Producer that is selected by the NF Service Consumer.

If the SCP changed the target URI when forwarding the request from the HTTP client to HTTP server (including when the SCP does so to re-send the request upon receiving a 3xx redirect message, see clause 6.10.9.1) and no "Location" header is included in the HTTP response (e.g. subsequent service request towards a created resource), the SCP shall include a "3gpp-Sbi-Target-apiRoot" header with value set to the apiRoot of the target HTTP server when forwarding the 2xx HTTP response, or an 4xx/5xx HTTP response including a 3gpp-Sbi-Response-Info header with "context-transferred" parameter set to value "true", to the NF as HTTP client.

NOTE 3: To avoid further reselection of HTTP server by SCP, the NF as HTTP client updates the locally stored URI (e.g. resource URI or notification callback URI) used in the request with the target apiRoot received in the HTTP response, and thus send subsequent request to the updated target URI.

6.10.5 NF / NF service instance selection for Indirect Communication without Delegated Discovery

6.10.5.1 General

For Indirect Communication without Delegated Discovery, the NF Service Consumer performs the discovery procedure by querying the NRF and the selection of a NF (Service) Set or a specific NF (service) instance. The selection of the target NF service instance may hence be done either by the NF Service Consumer or the SCP (e.g. based on NF (Service) Set information received from the NF Service Consumer).

The NF Service Consumer shall send its request to the SCP containing:

- the 3gpp-Sbi-Target-apiRoot header set to the apiRoot of the selected NF service instance, if the SCP is known to the NF Service Consumer and if the NF Service Consumer has selected a specific NF service instance;
- the identity of the selected NF (Service) Set in the associated "3gpp-Sbi-Discovery-*" request header(s) (see clause 6.10.3.2), if the NF Service Consumer has selected a target NF (Service) Set ID.

If the NF Service Consumer only selected an NF (service) Set, it should also include at least the following information in its request to the SCP:

- the target NF type, the service name, and the requested S-NSSAI in the corresponding 3gpp-Sbi-Discovery-* request header(s) (see clause 6.10.3.2).

NOTE 1: This is to allow the SCP to discover and select a target NF service instance from the target NF (service) set for the corresponding service request and supporting the requested S-NSSAI, e.g. when the NF service producer supports different NF service instances serving different network slices. Likewise, other "3gpp-Sbi-Discovery-*" request header(s), e.g. target-plmn-list, requester-plmn-list, can also be included for the same purpose.

The NF service consumer may indicate the NRF to use, e.g. as a result of an NSSF query, by including the 3gpp-Sbi-Nrf-Uri header with the NRF API URIs.

If the NF service consumer includes more than one service name in the 3gpp-Sbi-Discovery-service-names header, the service name corresponding to the service request shall be listed as the first service name in the header.

NOTE 2: The SCP can assume that the service request corresponds to the first service name in the header.

SCPs shall support Indirect Communication without Delegated Discovery, which requires support for the following:

- discovering and selecting a target NF service instance from the target NF (service) set identified in the 3gpp-Sbi-Discovery-target-nf-set-id, 3gpp-Sbi-Discovery-target-nf-service-set-id, 3gpp-Sbi-Discovery-amf-region-id and/or 3gpp-Sbi-Discovery-amf-set-id; and
- at least the following additional discovery headers: 3gpp-Sbi-Discovery-target-nf-type, 3gpp-Sbi-Discovery-service-names, 3gpp-Sbi-Discovery-snsais, 3gpp-Sbi-Discovery-target-plmn-list, 3gpp-Sbi-Discovery-requester-plmn-list.

NOTE 3: The SCP can derive the requester NF type from the User-Agent header.

SCPs shall additionally support reselecting an alternative target NF service instance when a (Routing) Binding Indication is not available, as specified in clauses 6.5.3 and 6.6.3 of TS 23.527 [38] and shall also support the 3gpp-Sbi-Discovery-target-nf-instance-id.

NOTE 4: The inclusion of the 3gpp-Sbi-Discovery-target-nf-instance-id in an HTTP request enables the SCP to discover the profile of the target NF instance and to possibly reselect a different target NF service instance from the same NF instance or from a different NF instance in the same set, e.g. when the target NF instance is not reachable, as specified in TS 23.527 [38].

If the request does not include the apiRoot of a selected NF service instance, or if the SCP needs to reselect a different NF service instance, the SCP shall select an NF service instance using the NF (Service) Set ID and any additional information (e.g. S-NSSAI, service name, target NF type) received in the corresponding "3gpp-Sbi-Discovery-*" request header(s), if available. If the SCP is to invoke NF service discovery towards the NRF to fulfil this task, the SCP should use the NRF indicated in the 3gpp-Sbi-Nrf-Uri header, if this header is present in the request. The SCP that reselected the target NF service instance shall include the 3gpp-Sbi-Producer-Id header in the 2xx HTTP response it forwards towards the NF Service Consumer, containing the NF Instance ID and the NF Service Instance ID of the NF Service Producer selected by the SCP, as specified in clause 6.10.3.4; if the SCP received a 4xx/5xx HTTP response including a 3gpp-Sbi-Response-Info header with "context-transferred" parameter set to value "true" from the reselected target NF service instance, which indicates the corresponding resource or context has been transferred to the reselected target NF service instance, the SCP shall also insert a 3gpp-Sbi-Producer-Id header in the HTTP response it forwards to the NF Service Consumer.

The SCP shall then route the request to the selected NF service instance of the target NF service producer.

NOTE 5: For Indirect Communication without Delegated Discovery, the NF Service Consumer decides if it will perform the reselection or delegate the SCP to perform the reselection as specified in clause 6.5 of TS 23.527 [38].

When the 3gpp-Sbi-Selection-Info header is included in a HTTP request message and if the SCP supports this header, the SCP shall use it together with 3gpp-Sbi-Routing-Binding or 3gpp-Sbi-Discovery-* heads whichever available.

6.10.5.2 Notifications corresponding to default notification subscriptions

An NF may register default notification subscriptions in its NF profile or NF services in the NRF for notifications the NF is prepared to consume, including for each type of notification the corresponding notification endpoint (i.e. callback URI).

NOTE: This can be used e.g. by an AMF to discover the notification endpoint of other AMFs to forward N1 or N2 messages, or by an AMF to notify location information to a GMLC, or by an UDR to notify data change or removal to an UDM.

The following procedures may be used to support notifications corresponding to default notification subscriptions with indirect communication without delegated discovery.

An NF producer may perform a discovery request towards the NRF (possibly through an SCP) to discover default notification subscriptions of an NF consumer, and if so, send notifications to the corresponding notification endpoints, using routing mechanisms specified in clause 6.10.2 / 6.10.2A. The NF producer shall include in the notification request:

- the 3gpp-Sbi-Callback header including the name of the notify or callback service operation and the API major version if higher than 1, (see also clause 6.10.7);
- the 3gpp-Sbi-Target-apiRoot which is set to the notification uri, or the target URI is set to the notification uri as specified in clause 6.10.2 or 6.10.2A respectively;

If the NF producer does not perform reselection, i.e. the reselection is to be done by SCP, the NF producer shall further include in the notification request:

- the 3gpp-Sbi-Discovery-notification-type header set to the type of notification being set; and
- the 3gpp-Sbi-Discovery-n1-msg-class header set to the N1 Message Class of the target default subscription if notification type is "N1_MESSAGE", or the 3gpp-Sbi-Discovery-n2-info-class header set to the N2 Information Class of the target default subscription if the notification type is "N2_INFORMATION"; and
- the 3gpp-Sbi-Routing-Binding header for the default notification based on the Binding Indication value in the NF profile of the NF Service Consumer if available (see also clause 6.12.4); or when the 3gpp-Sbi-Routing-Binding header is not available, the 3gpp-sbi-discovery* headers containing the NF service discovery factors header to be used by the SCP to reselect a consumer NF (to receive the notification request) and the Callback URI Prefix (if any) included in the 3gpp-Sbi-Request-Info header.

The NF producer or SCP may perform a reselection if it cannot reach the target NF as indicated in the 3gpp-Sbi-Target-apiRoot or the target URI, and if a reselection is performed, the entity responsible for reselection (either SCP or NF producer) shall perform reselection as below:

- the NF producer may use the Binding Information that is associated to the default notification;
- The SCP shall use, if available, the Routing Binding Indication (that is associated to the default notification) or alternatively 3gpp-Sbi-discovery* and the 3gpp-Sbi-Request-Info headers to reselect an alternative NF Service Consumer.

After reselection is performed, the NF producer or the SCP shall fetch the alternative notification endpoint from the corresponding default notification subscription registered by the alternative NF Service Consumer. The SCP shall use the 3gpp-Sbi-Discovery-notification-type header and additionally the 3gpp-Sbi-Discovery-n1-msg-class header (for "N1_MESSAGE" notification type) or the 3gpp-Sbi-Discovery-n2-info-class header (for "N2_INFORMATION" notification type) to locate the corresponding default notification subscription of the alternative NF Service Consumer.

6.10.6 Feature negotiation for Indirect Communication with Delegated Discovery

The requirements specified in clause 6.6.2 for feature negotiation shall apply with the following additions.

With Indirect Communications with Delegated Discovery, the NF Service Consumer cannot discover the features supported by the NF Service Producer via the NRF.

The NF Service Consumer shall include in HTTP PUT or POST requests to create a resource that requires specific features to be supported by the NF Service Producer, the 3gpp-Sbi-Discovery-required-features header set to the required features to be supported.

The SCP shall reject the request with the HTTP status code "400 Bad Request" and the protocol error "NF_DISCOVERY_FAILURE" if no NF Service Producer supporting the required features can be discovered. In such a case the SCP may also include a noProfileMatchInfo attribute in the ProblemDetails body in order to provide more

information about the reason the discovery failed and allow the consumer to undertake appropriate action (e.g. downgrade API version, remove required features, etc).

6.10.7 Notification and callback requests sent with Indirect Communication

Notification and callback requests that are sent using indirect communication shall include a 3gpp-Sbi-Callback header including the name of the notification or callback service operation (see Annex B) and the API major version if higher than 1.

The SCP shall derive from the presence of this header that a service request is a notification or callback request.

NOTE: This can be used by the SCP to apply differentiated treatments for notification and callback requests compared to other service requests, e.g. for authorization (access token is not used in notification / callback, see clause 6.7.3).

The NF service producer should include the NRF API URI(s) for NF service consumer reselection in the 3gpp-Sbi-Nrf-Uri header, if this information was previously received from the NF service consumer in the 3gpp-Sbi-Nrf-Uri-Callback header (see clause 6.5.3.2) and the NF service producer delegates the NF service consumer reselection to the SCP.

If the Callback URI included a prefix and binding procedures are not supported, the NF service producer should include the callback URI Prefix in the 3gpp-Sbi-Request-Info header.

6.10.8 Error Handling

6.10.8.1 General

A request from an HTTP client (i.e. a service request from an NF service consumer, or a notification request from an NF service producer) may traverse one or more SCPs and/or SEPPs and may fail at an SCP, SEPP or at the HTTP server.

The HTTP client should be able to figure out whether the request failed at its next hop SCP or SEPP, or at the HTTP server, e.g. to be able to adapt its behaviour for the on-going request or subsequent request accordingly. For instance, the HTTP client may retry the request or send subsequent requests towards the same HTTP server via a different SCP or SEPP if an SCP or SEPP rejected a request due to insufficient resources, or towards a different HTTP server (via the same or a different SCP or SEPP) if the HTTP server rejected the request due to insufficient resources.

NOTE: An SCP or SEPP can also retry a request towards a different SCP or SEPP, or towards a different HTTP server, instead of relaying the response back to the originator, if a next hop SCP or SEPP or if the HTTP server rejected a request e.g. due to insufficient resources.

If the SCP or HTTP client receives an error response including the 3gpp-Sbi-Response-Info header with the "no-retry" parameter set to "true", the SCP or HTTP client shall consider that the request cannot be served anywhere and should not retry the request at the original HTTP server instance or at any other alternative HTTP server instance; the SCP shall forward the error response to the HTTP client including the 3gpp-Sbi-Response-Info header.

When receiving an error response, the HTTP client should be able to figure out whether the SCP attempted to retransmit the request to an alternative HTTP server instance. To enable so, if the SCP attempted to retransmit the request to an alternative HTTP server instance, it shall indicate so in the error response by including the 3gpp-Sbi-Response-Info header with the "request-retransmitted" parameter set to "true" and by optionally including the list of NF instances, NF sets, NF service instances or NF service sets that it attempted. The SCP may indicate in the error response that it did not attempt to retransmit the request to an alternative HTTP server instance by including the 3gpp-Sbi-Response-Info header with the "request-retransmitted" parameter set to "false". The HTTP client may use this information to determine whether it may retransmit the request to an alternative HTTP server instance.

If an SCP or SEPP receives an error response including the 3gpp-Sbi-Response-Info header with the "request-retransmitted" parameter set to "true" (e.g. in a scenario with two SCPs between the HTTP client and HTTP server), the SCP (if it does not reselect a target NF) or SEPP shall forward the error response with the the 3gpp-Sbi-Response-Info header unmodified towards the HTTP client; alternatively, the SCP may reselect a target NF and, if the NF reselection fails, the SCP may add the list of of NF instances, NF sets, NF service instances or NF service sets that it attempted in the 3gpp-Sbi-Response-Info header returned in the error response towards the HTTP client.

NOTE 1: This can correspond to errors originated by the SCP or by an HTTP server.

NOTE 2: Rel-17 onwards compliant SCPs support and can be configured (or not) to reselect a different NF service producer or consumer, e.g. when the target URI of a service request (or notification request) is not reachable, as specified in TS 23.527 [38].

6.10.8.2 Requirements for the originator of an HTTP error response

To enable an HTTP client to determine the originator of an HTTP error response, the originator of an error (e.g. HTTP server, SCP or SEPP) should include a Server header in the HTTP error response with the following information:

- the type of the NF or network entity generating the error, set to the NFType value as defined in clause 6.1.6.3.3 of TS 29.510 [8], e.g. "SCP", "SEPP", "SMF";
- the identity of the NF or network entity generating the error, set to the FQDN of the SCP or SEPP, or to the NF Instance ID of the HTTP server.

NOTE: The information carried in the Server header can also be useful for trouble-shooting.

EXAMPLE 1: Error generated by an SCP: Server: SCP-scp1.operator.com

EXAMPLE 2: Error generated by a SEPP: Server: SEPP-sepp1.operator.com

EXAMPLE 3: Error generated by an SMF: Server: SMF-54804518-4191-46b3-955c-ac631f953ed8

The presence of a Server header set to the next hop SCP or SEPP or to the HTTP server in an HTTP error response shall be an indication for the HTTP client that the next hop SCP or SEPP or the HTTP server is the originator of the error.

If neither the target NF nor alternative NFs that the SCP tries to (re)select based on the Routing Binding Indication or Discovery headers are reachable, the SCP shall return a HTTP 504 Gateway Timeout response including the "problemDetails" with the "cause" attribute set to "TARGET_NF_NOT_REACHABLE" and the Server header which is set to the FQDN of the SCP.

If the cSEPP receives the HTTP request from the NF with "encBlockIndex" included as specified in clause 5.9.3.2 of TS 33.501 [17], the cSEPP shall return a HTTP 400 Bad Request response including the "problemDetails" with the "cause" attribute set to "MANDATORY_IE_INCORRECT" or "OPTIONAL_IE_INCORRECT" and the "invalidParams" attribute indicates the incorrect IE. The Server header which is set to the FQDN of the cSEPP shall also be returned.

If the SCP cannot fulfill a service request due to NRF related errors, the SCP shall originate an error towards the HTTP client as follows:

- If the NRF is not reachable, the SCP shall reject the request with a 504 Gateway Timeout including the "problemDetails" with the "cause" attribute set to "NRF_NOT_REACHABLE";
- If the NRF rejected an NF discovery request with a 5xx or 429 response, the SCP shall reject the request with a 502 Bad Gateway including the "problemDetails" with the "cause" attribute set to "NF_DISCOVERY_ERROR";
- If the NRF rejected an NF discovery request with a 4xx response, the SCP shall reject the request with a 4xx response including the "problemDetails" with an appropriate "cause" attribute (e.g. same response code and cause as received from the NRF).
- If the NRF returned a NF Discovery 200 OK response without any NF service producer matching the query parameters, the SCP shall reject the request with a "400 Bad Request" and the protocol error "NF_DISCOVERY_FAILURE" as specified in clause 6.10.6;
- see also clause 6.10.11.2 for SCP error handling requirements for errors due to NF service access authorization.

In either case, the SCP shall include the Server header in the error response set with its own identity (i.e. SCP FQDN) as specified in this clause.

6.10.8.3 Requirements for an SCP or SEPP relaying an HTTP error response

To enable an HTTP client to determine the originator of an HTTP error response, e.g. when an HTTP server does not include a Server header in an HTTP error response, the SCP or SEPP that forwards the HTTP error response towards the HTTP client shall include a Via header in the HTTP error response with the following information:

- the received-protocol portion of the Via header as defined in clause 7.6.3 of IETF RFC 9110 [11];
- the type of the network entity forwarding the error, in the received-by portion formatted according to Table 5.2.2.2-2, set to the NFType value as defined in clause 6.1.6.3.3 of TS 29.510 [8], i.e. "SCP" or "SEPP";
- the identity of the network entity forwarding the error, in the received-by portion formatted according to Table 5.2.2.2-2, set to the FQDN of the SCP or SEPP.

NOTE: The information carried in the Via header can also be useful for trouble-shooting.

EXAMPLE 1: Error forwarded by an SCP: Via: HTTP/2.0 SCP-scp1.operator.com or Via: 2.0 SCP-scp1.operator.com

EXAMPLE 2: Error forwarded by a SEPP: Via: HTTP/2.0 SEPP-sepp1.operator.com or Via: 2.0 SEPP-sepp1.operator.com

The presence of a Via header set to the next hop SCP or SEPP in an HTTP error response shall be an indication for the HTTP client that the next hop SCP or SEPP is not the originator of the error.

A SEPP shall forward unmodified HTTP status codes and application errors that it receives.

NOTE: When PRINS in N32-f communication is used, an error might also be received from Roaming Intermediary with a HTTP status code identified in Table 5.2.7.1-1, along with the application errors. These errors are handled by the receiving SEPP like any errors from remote SEPP or target NF.

6.10.9 HTTP redirection

6.10.9.1 General

An HTTP request sent using indirect communication may be redirected either to a different target NF service instance (from a same NF service set or NF set) or to a different SCP.

When an HTTP server or SCP redirects an HTTP request (i.e. service request or notification/callback request) to a different target NF service instance, the URI of the target NF service instance towards which the request is redirected shall be given by the Location header field of the 307 Temporary Redirect or 308 Permanent Redirect response. When redirecting a request to a different NF instance (e.g. in a same NF set), the NF (service) instance ID of the target NF (service) instance towards which the request is redirected should be indicated in the 3gpp-Sbi-Target-Nf-Id header of the 307 Temporary Redirect or 308 Permanent Redirect response; it may be indicated otherwise (e.g. when redirecting a request to a different NF service instance of the same NF instance and overload control is to be performed per NF service instance). The HTTP client should then send the HTTP request towards the new target NF service instance using the same or a different SCP.

The SCP shall forward a 3xx redirect response towards the HTTP client, if the URI in the Location header of the 3xx redirect response contains a different URI path than the URI path that was sent in the request to the HTTP server. In other cases, based on local policies, when appropriate (e.g. HTTP request creating a resource), the SCP may send the HTTP request towards the new target NF service instance instead of forwarding the 307/308 response to the HTTP client; in such a case, the SCP shall include the 3gpp-Sbi-Target-apiRoot header set to the apiRoot of the target URI in the response message sent to the HTTP client, if no Location header is included in the response received from the new target NF service instance, according to the requirements specified in clause 6.10.4.

NOTE 1: As specified in step 6 of clause 6.5.3.3, upon a change of NF service producer, the new NF service producer can redirect a service request from the NF service consumer with a 3xx redirect message including a new URI (including a different apiRoot and possibly URI path) for the existing resource. In scenarios with an SCP between the NF service consumer and the NF service producer, the SCP needs to pass the new URI of the existing resource to the NF service consumer.

An SCP may redirect an HTTP request towards a different SCP by sending a 307 Temporary Redirect or 308 Permanent Redirect response to the HTTP client including a RedirectResponse data structure (see TS 29.571 [13]) with the cause attribute set to "SCP_REDIRECTION" and with the targetScp attribute indicating the apiRoot of the SCP towards which the request is redirected. In this scenario, the 307 Temporary Redirect or 308 Permanent Redirect response shall include a Location header field where the content of the Location header field is implementation specific. The HTTP client should then send the HTTP request towards the target NF service instance using the SCP indicated in the 307 or 308 response; when doing so, the HTTP client shall ignore the information received in the Location header field if it

receives a 307 Temporary Redirect or 308 Permanent Redirect response with the cause attribute set to "SCP_REDIRECTION" and including a Location header field, and it shall use the apiRoot included in targetScp IE as the apiRoot of the request URI to retransmit the HTTP request message via the alternative SCP.

NOTE 2: The SCP can alternatively forward the request message to another SCP when there is a failure between the SCP and the target NF, and if the SCP knows that another SCP can reach the target NF and the 3gpp-Sbi-Max-Rsp-Time included the request message has not expired.

NOTE 3: An SCP implementation can set the content of the Location header field e.g. to the request URI received in the service request but with the apiRoot of the SCP to which the request is redirected.

A SEPP may redirect an HTTP request towards a different SEPP over a non-N32 interface by sending a 307 Temporary Redirect or 308 Permanent Redirect response to the HTTP client including a RedirectResponse data structure (see TS 29.571 [13]) with the cause attribute set to "SEPP_REDIRECTION" and with the targetSepp attribute indicating the apiRoot of the SEPP towards which the request is redirected. In this scenario, the 307 Temporary Redirect or 308 Permanent Redirect response shall include a Location header field where the content of the Location header field is implementation specific. The HTTP client should then send the HTTP request towards the target NF service instance using the SEPP indicated in the 307 or 308 response; when doing so, the HTTP client shall ignore the information received in the Location header field if it receives a 307 Temporary Redirect or 308 Permanent Redirect response with the cause attribute set to "SEPP_REDIRECTION" and including a Location header field, and it shall use the apiRoot included in targetSepp IE as the apiRoot of the request URI to retransmit the HTTP request message via the alternative SEPP.

NOTE 4: A SEPP implementation can set the content of the Location header field e.g. to the request URI received in the service request but with the apiRoot of the SEPP to which the request is redirected.

NOTE 5: See clause 6.1.8 of TS 29.573 [27] for the redirection of an N32 HTTP request from a SEPP to a different SEPP.

6.10.10 Detection and handling of loop path when relaying message with indirect communication

6.10.10.1 General

For indirect communications, request messages may be forwarded through multiple SCPs. In case of misconfiguration or error processing on intermediate SCPs, request messages may be relayed via unexpected paths or trapped in loops.

The following two optional solutions may be used to enable the SCPs to detect and handle dead looping when relaying request messages in the network with indirect communication.

6.10.10.2 Message Forwarding Depth Control

If Message Forwarding Depth Control is enabled, an HTTP client, or an SCP if the 3gpp-Sbi-Max-Forward-Hops header is not received in an incoming request, shall include in the request the 3gpp-Sbi-Max-Forward-Hops header with the node type "scp" indicating the maximum number of allowed intermediate SCPs to relay the message, before reaching the target HTTP server.

When forwarding a request that includes the 3gpp-Sbi-Max-Forward-Hops header with node type "scp" to a next hop SCP, the SCP shall check whether the value of the header is zero or not, then

- if the value of 3gpp-Sbi-Max-Forward-Hops header with node type "scp" is zero, the SCP shall reject the request with the HTTP status code "502 Bad Gateway" and the protocol error "MAX_SCP_HOPS_REACHED";
- otherwise, the SCP shall decrement the value of the 3gpp-Sbi-Max-Forward-Hops header with node type "scp" by 1 before forwarding the request.

6.10.10.3 Loop Detection with Via header

The Via header shall be inserted by HTTP proxies, SCPs and SEPPs when relaying an HTTP message (see clause 5.2.2.2).

Upon receiving a request message, if Loop Detection through Via header is enabled, the SCP shall check the presence of itself, i.e. whether an "SCP-<SCP FQDN>" with its own FQDN is included in the Via headers received. If present, the SCP shall reject the request with the HTTP status code "400 Bad Request" and the protocol error "MSG_LOOP_DETECTED".

NOTE: If topology hiding is applied within the network, entities in Via header may be replaced at domain borders.

6.10.11 Authorization of NF service access

6.10.11.1 General

Service access authorization for indirect communication shall be supported as specified in clause 13.4.1.3 of TS 33.501 [17].

6.10.11.2 Authorization for indirect communication with delegated discovery

6.10.11.2.1 General

When the NF service consumer is configured to use delegated service discovery, requirements in clause 13.4.1.3.2 of TS 33.501 [17] shall apply with the following additions.

If the NF service consumer received an access token in a previous service response that is valid for the new service request, the NF service consumer should include the access token in the Authorization header in the service request. An access token received in a previous service response is valid for the new service request if:

- it has a matching scope, or it has a matching service-level scope only (i.e. the new service request also requires a resource/operation-level scope that is not present in the scope of the access token received in the previous service response);
- it has a matching audience (i.e. matching producer's NF type or NF instance ID);
- it has a matching producer's NF set ID, S-NSSAI, NSI and PLMN ID, if the access token contains a producer NF set ID, S-NSSAI, NSI and PLMN ID respectively; and
- the access token has not expired.

NOTE 1: If the NF service consumer has multiple cached access tokens that are valid for a service request, it is left for implementation how to select the access token to include in the request. Access tokens with a matching scope, if any, are to be used in preference to access tokens with a matching service-level scope only.

NOTE 2: Including an access token that has a matching service-level scope only but not a matching resource/operation-level scope enables the reuse of the access token when the NF service producer is not configured to require the resource/operation-level scope.

If the NF service consumer does not include an access token in the service request, or if it does but the access token has a matching service-level scope only but not a matching resource/operation-level scope, or if it does but the access token is NF instance specific and reselection of a different producer instance may apply at the SCP (e.g. a routing binding header or a discovery header provides the producer's NF set ID), the NF service consumer shall include in the service request:

- the necessary NF service discovery factors to be used by the SCP for the Service access authorization procedures, as specified in clause 6.10.3.2; and
- the 3gpp-Sbi-Access-Scope header indicating the access scope of the service request for access authorization, as defined for the specific resource/operation in the corresponding API (see clause 5.2.3.2.16).

For service requests targeting a different PLMN (or SNPN), if the NF service consumer serves a PLMN consisting of more than one PLMN ID (or if it supports multiple SNPN IDs), the NF service consumer shall include, in the 3gpp-Sbi-Discovery-requester-plmn-list header (or in the 3gpp-Sbi-Discovery-requester-snpn-list header), the PLMN ID (or SNPN ID) on behalf of which the service requests are sent.

For service requests within the same PLMN (or SNPN), if the NF service consumer serves a PLMN consisting of more than one PLMN ID (or if it supports multiple SNPN IDs) and if it includes the 3gpp-Sbi-Discovery-requester-plmn-list header (or the 3gpp-Sbi-Discovery-requester-snpn-list header) in its service requests, the NF service consumer shall include in this header the PLMN ID (or SNPN ID) on behalf of which the service requests are sent.

In service requests including the 3gpp-Sbi-Access-Scope header, the NF service consumer may also include the 3gpp-Sbi-Other-Access-Scopes header indicating other access scopes that are desired to be obtained for the access token, in addition to the scopes indicated in the 3gpp-Sbi-Access-Scope, that are not required for the service request itself but that may be required for further service requests, when requesting an access token to the NRF.

NOTE 3: Indicating other access scopes in the service request can allow obtaining an access token that can be reused in subsequent service requests requiring different scopes than those required for the current service request.

The NF service consumer may also include its Client Credentials Assertion as specified in clause 6.7.5.

The SCP should use the access scope information received in the 3gpp-Sbi-Access-Scope header to determine the access scope required for access authorization for an incoming service request. The SCP may also use the scopes required by the NF service producer (as registered in its NF profile) for this determination and, if a new access token is required, request an access token to the NRF for a list of scopes that is the intersection of the scopes indicated in the 3gpp-Sbi-Access-Scope header and the scopes expected by the NF Service producer. If the 3gpp-Sbi-Other-Access-Scopes header is received in the incoming service request, the SCP may also include the other access scopes received in this header to the scopes requested to the NRF for the access token.

If the NF service consumer has included an access token in the service request without including the 3gpp-Sbi-Access-Scope header, or if the SCP has a cached granted access token that matches the service request, the SCP should reuse the available access token. If the NF service consumer has included an access token in the service request and the 3gpp-Sbi-Access-Scope header, the 3gpp-Sbi-Access-Scope header contains multiple scopes, and the access token has a matching scope only for a subset of the scopes present in the 3gpp-Sbi-Access-Scope header, the SCP should consider that the access token has a valid scope for the service request if the NF service producer does not require any scope not granted in the Access Token (as determined from its NF profile); otherwise, the SCP shall request a new access token for the service request.

NOTE 4: This allows the SCP to consider that an access token has a valid scope if the 3gpp-Sbi-Access-Scope header contains a service-level scope and a resource/operation-level scope, the access token has a scope matching only the service-level scope, and the NF service producer is not configured to require the resource/operation-specific scope.

When the NRF receives a request to obtain an access token for a list of scopes, but the NF service producer's profile does not allow to grant a token for all the requested scopes, the NRF should grant an access token but restricted to the allowed scope, unless the request cannot be authorized for other reasons. If the NRF grants an access token restricted to the allowed scope, the NRF shall indicate the allowed scope in the scope parameter of the Access Token Response as defined in clause 6.3.5.2.3 of TS 29.510 [8].

NOTE 5: This allows the NRF to grant an access token for a service-level scope, in response to an access token request for a list of scopes including a service-level scope and a resource/operation-level scope, when the NF service producer's profile is not configured to require the resource/operation-level scope.

A failure to obtain an access scope received in the 3gpp-Sbi-Other-Access-Scopes header in the granted token shall not result in the SCP failing the service request, as long as all the scopes required for access authorization for the incoming service request have been authorized by the NRF.

When the SCP requests an access token for a service request, the SCP may include the access token it has obtained from the NRF in the service response it forwards to the NF service consumer, by including the 3gpp-Sbi-Access-Token header, for possible re-use in subsequent service requests by the NF service consumer. The NF service consumer should store the access token received in a service response and use it in subsequent service requests as defined above.

6.10.11.2.2 Error handling when the SCP fails to obtain an access token

If the SCP cannot issue an Access Token Request towards the NRF due to missing information in the incoming service request, e.g. if the 3gpp-Sbi-Discovery-requester-nf-instance header is missing, or if the SCP can issue an Access Token Request towards the NRF but the NRF rejects the Access Token Request with a 403 Forbidden response with the MISSING_PARAMETER cause, the SCP shall reject the service request with a 400 Bad Request response including a ProblemDetails IE with:

- the cause attribute set to MISSING_ACCESS_TOKEN_INFO;
- the invalidParams attribute indicating the missing parameters (e.g. missing discovery header).

If the SCP can issue an Access Token Request towards the NRF, but the NRF rejects the request for a cause other than MISSING_PARAMETER (e.g. because the validation of the Client Credentials Assertion fails at the NRF or because the NF service consumer is not authorized to access the requested service), the SCP shall reject the service request towards the NF service consumer with a 403 Forbidden response including a ProblemDetails IE with the cause attribute set to ACCESS_TOKEN_DENIED. The ProblemDetails IE should also contain:

- the accessTokenError attribute set to the accessTokenErr content received from the NRF;

and it may contain:

- the accessTokenRequest attribute set to the Access Token Request content sent to the NRF;
- the nrfId attribute set to the FQDN of the NRF that rejected the access token request.

In either case, the SCP shall include the Server header in the error response set with its own identity (i.e. SCP FQDN) as specified in clause 6.10.8.2.

6.10.11.2.2A Error handling when the SCP obtains an access token without all the scopes required for access authorization of the incoming service request

If the SCP issues an Access Token Request towards the NRF and the NRF returns an access token not granting authorization for all the scopes required for access authorization of the incoming service request, the SCP shall reject the service request towards the NF service consumer with a 403 Forbidden response including a ProblemDetails IE with the cause attribute set to ACCESS_TOKEN_DENIED. The ProblemDetails IE may contain:

- the accessTokenRequest attribute set to the Access Token Request payload sent to the NRF;
- the nrfId attribute set to the FQDN of the NRF that rejected the access token request.

The SCP shall include the Server header in the error response set with its own identity (i.e. SCP FQDN) as specified in clause 6.10.8.2.

The SCP may include the access token it has obtained from the NRF (e.g. granting authorization for the other access scopes indicated in the service request) in the service response to the NF service consumer, by including the 3gpp-Sbi-Access-Token header, for possible re-use in subsequent service requests by the NF service consumer. The NF service consumer should store the access token received in a service response and use it in subsequent service requests as defined above.

NOTE 1: This error scenario can happen e.g. when the incoming service request includes a first service name in the 3gpp-Sbi-Access-Scope header, a second service name in 3gpp-Sbi-Other-Access-Scopes header, and the NRF grants an access token with a scope of the second service name only.

NOTE 2: The SCP does not reject the service request (i.e. it is not an error) if the 3gpp-Sbi-Access-Scope header contains a service-level scope and a resource/operation-level scope, the access token granted by the NRF has a scope matching only the service-level scope and the NF service producer is not configured to require the resource/operation-specific scope. See NOTE 3 of clause 6.10.11.2.1.

6.10.11.2.3 Error handling when SCP receives a "401 Unauthorized" response or a "403 Forbidden" response with a "WWW-Authenticate" header

If the NF service producer rejects the service request with a "401 Unauthorized" response or with a "403 Forbidden" response with a "WWW-Authenticate" header containing "Bearer" as the scheme for challenge:

- if the SCP had included an access token received from the NF service consumer in the service request to the NF service producer, the SCP shall forward the response to the NF service consumer; the NF service consumer shall then delete the corresponding access token and may repeat the request without an access token or with a different access token;

- if the SCP had included an access token it had cached or obtained from the NRF, the SCP shall not repeat the request with the access token that was used; the SCP may repeat the request with a new access token; otherwise, or if the repeated request fails, the SCP shall forward the response to the NF service consumer;
- if the SCP had not included an access token in the service request to the NF service producer, the SCP should request an access token to the NRF and repeat the request; otherwise, the SCP shall forward the response to the NF service consumer.

When requesting a new access token, the NF service consumer or the SCP should consider the error information received in the error response, e.g. to retrieve a new access token with the scopes or claims required by the NF service producer (see clause 6.7.3).

6.10.11.3 Authorization for indirect communication without delegated discovery

Requirements in clause 13.4.1.3.1 of TS 33.501 [17] shall apply with the following additions.

If selection or reselection of a producer's NF instance may apply at the SCP (e.g. initial service request containing the target NF Set ID, or service request containing a routing binding header or a discovery header with the producer's NF set ID), the NF service consumer shall include in the service request an access token that is valid for any producer's NF instance that the SCP may select or reselect, i.e. an access token that is not specific to a particular producer's NF instance. This shall be an access token valid for the target NF type and producer's NF set.

6.10.12 Indirect communication with or without delegated discovery between different PLMNs with possible NF selection at target PLMN

NOTE: Although the requirements in this clause refer to indirect communication between different PLMNs and possible NF selection at the target PLMN, they can also be used for indirect communication between different source and target PLMNs or SNPNs and thus for selecting the target NF at the target PLMN or SNPN.

6.10.12.1 General

This clause specifies specific requirements to support indirect communication with or without delegated discovery between different PLMNs with possible NF selection at target PLMN, as specified in clause 6.3.1 of TS 23.501 [3] and clause 4.17.10a of TS 23.502 [4].

The terms "source PLMN" and "target PLMN" throughout this clause refer respectively to the PLMN originating the request (i.e. the PLMN of the NF service consumer for a service request, or the PLMN of the NF service producer for a notification request) and the PLMN receiving/serving the request (i.e. the PLMN of the NF service producer for a service request, or the PLMN of the NF service consumer for a notification request).

6.10.12.2 NF Discovery and Selection

6.10.12.2.1 NF Discovery and Selection for indirect communication with delegated discovery

When using indirect communication with delegated discovery between different PLMNs, the NF service consumer (for a service request) or the NF service producer (for a notification request), and the SCP in the source PLMN shall include the necessary NF service discovery factors (as specified in clause 6.10.3.2) in the request sent towards the target PLMN. The request shall also include the 3gpp-Sbi-Discovery-target-plmn-list (or 3gpp-Sbi-Discovery-target-snpn) header identifying the target PLMN (or SNPN) and the 3gpp-Sbi-Discovery-requester-plmn-list (or 3gpp-Sbi-Discovery-requester-snpn-list) header identifying the source PLMN (or SNPN) ID. If the NF service consumer serves a PLMN consisting of more than one PLMN ID, the NF service consumer shall include, in the 3gpp-Sbi-Discovery-requester-plmn-list header, the PLMN ID on behalf of which the service request is sent.

When the SCP in the target domain receives a request containing "3gpp-Sbi-Discovery-*" request headers and a selection/reselection of the target NF service instance is required, the SCP shall take into account all the NF service discovery factors contained in the "3gpp-Sbi-Discovery-*" request headers to perform the selection or reselection, as specified in clause 6.10.3.2.

6.10.12.2.2 NF Discovery and Selection for indirect communication without delegated discovery

When using indirect communication without delegated discovery between different PLMNs, the NF service consumer or the SCP of the source PLMN shall select the NF (Service) Set or a specific NF (service) instance.

If the source PLMN only selects the NF (Service) Set, the selection of the target NF service instance shall be done by an SCP in the target PLMN. If the source PLMN selects a specific NF (service) instance, an SCP in the target PLMN may reselect the target NF service instance within the same NF (Service) Set.

When using indirect communication without delegated discovery between different PLMNs, the NF service producer or the SCP of the source PLMN shall select the NF service consumer (e.g. for sending a notification corresponding to a default notification subscription as defined in clause 6.10.5.2).

The requirements specified in clause 6.10.5 shall apply. Additionally, the request sent from the source PLMN to the target PLMN shall also include the 3gpp-Sbi-Discovery-target-plmn-list (or 3gpp-Sbi-Discovery-target-snpn) header identifying the target PLMN (or SNPN) and the 3gpp-Sbi-Discovery-requester-plmn-list (or 3gpp-Sbi-Discovery-requester-snpn-list) header identifying the source PLMN (or SNPN) ID. If the NF service consumer serves a PLMN consisting of more than one PLMN ID, the NF service consumer shall include, in the 3gpp-Sbi-Discovery-requester-plmn-list header, the PLMN ID on behalf of which the service request is sent.

6.10.12.3 Routing mechanisms

6.10.12.3.1 General

This clause specifies how to route a request from an NF service consumer (for a service request) or an NF service producer (for a notification request) in a source PLMN towards an NF service producer or an NF service consumer (respectively) in a different PLMN, with possible NF selection at the target PLMN.

6.10.12.3.2 Routing within the source PLMN

Requests shall be routed between the NF service consumer (or NF service producer) and the SCP within the source PLMN as specified for indirect communications in clauses 6.10.2 and 6.10.2A.

Requests shall be routed between the SCP and the SEPP within the source PLMN:

- as specified in clause 6.1.4.3, if a target NF instance has been selected by the source PLMN; additionally, if the address of an SCP in the target domain is also available (i.e. was received in the NF Discovery response from the NRF), the SCP in the source PLMN shall include the 3gpp-Sbi-Scp-apiRoot header in the request it forwards to the SEPP set to the apiRoot of the SCP in the target domain;
- as specified for indirect communications from one SCP to a next hop SCP in clause 6.10.2, with the SEPP taking the role of the next hop SCP, if no target NF instance has been selected by the source PLMN; additionally, if the address of an SCP in the target domain is available (i.e. was received in the NF Discovery response from the NRF), the SCP in the source PLMN shall include the 3gpp-Sbi-Scp-apiRoot header in the request it forwards to the SEPP set to the apiRoot of the SCP in the target domain. The SCP in the source PLMN shall determine the SEPP towards which to send the request taking into account the target PLMN (or SNPN) ID indicated in the 3gpp-Sbi-Discovery-target-plmn-list (or 3gpp-Sbi-Discovery-target-snpn) header. In this case, no 3gpp-Sbi-Target-apiRoot header shall be included in the request sent towards the SEPP and the apiRoot of the request URI shall be set to the apiRoot of the SEPP.

6.10.12.3.3 Routing between SEPPs

A request shall be routed between the cSEPP and the pSEPP as specified in clause 6.1.4.3.4 if it contains the address of the target NF instance, i.e. it contains the 3gpp-Sbi-Target-apiRoot header or a telescopic FQDN identifying the target address.

A request shall be routed between the cSEPP and the pSEPP as follows if it does not contain the address of the target NF instance, i.e. no 3gpp-Sbi-Target-apiRoot header is present and no telescopic FQDN was used:

- if PRINS security is negotiated between the SEPPs, the request shall be routed as specified in clause 6.1.4.3.4. Additionally:

- the cSEPP shall set the apiRoot of the Request URI of the encapsulated protected message to the apiRoot of the pSEPP; and
- the cSEPP shall forward the 3gpp-Sbi-Scp-apiRoot header in the encapsulated protected message, if this header was received in the incoming request.

or

- if TLS security is negotiated between the SEPPs, the cSEPP shall set the apiRoot of the request URI it forwards on the N32-f interface to the apiRoot of the pSEPP. The cSEPP shall then send the request towards the pSEPP, without including the 3gpp-Sbi-Target-apiRoot header. The cSEPP shall forward the 3gpp-Sbi-Scp-apiRoot header in the request, if this header was received in the incoming request.

NOTE: In either case above, the pSEPP receives the request from the cSEPP with the apiRoot of the Request URI set to the apiRoot of the pSEPP.

6.10.12.3.4 Routing within the target PLMN

If the request contains the 3gpp-Sbi-Scp-apiRoot header, the pSEPP shall route the request to the target SCP in the target domain indicated in the 3gpp-Sbi-Scp-apiRoot header, if possible. If the SCP indicated in the 3gpp-Sbi-Scp-apiRoot header is not reachable, the pSEPP shall select an alternative target SCP in the target domain (e.g. an alternative SCP if the authority of the target SCP apiRoot in the 3gpp-Sbi-Scp-apiRoot header can resolve into several SCPs, or e.g. an SCP pre-configured in the pSEPP or discovered from the NRF). When forwarding the request towards the target SCP, the pSEPP shall remove the 3gpp-Sbi-Scp-apiRoot header and it shall set the apiRoot of the request URI to the apiRoot of the target SCP; the pSEPP shall forward the 3gpp-Sbi-Target-apiRoot header, if this header was received in the incoming request.

If the request does not contain the 3gpp-Sbi-Scp-apiRoot header and does not contain the target apiRoot (i.e. if the apiRoot of the Request URI corresponds to the apiRoot of the pSEPP), the pSEPP shall route the request to an SCP in the target domain (e.g. pre-configured in the pSEPP or discovered from the NRF) as specified for indirect communication from one SCP to a next hop SCP in clause 6.10.2, with the pSEPP taking the role of the SCP and the SCP in the target domain taking the role of the next hop SCP. The apiRoot of the Request URI shall be set to the apiRoot of the SCP in the target domain. The request shall not contain the 3gpp-Sbi-Target-apiRoot header.

If the request does not contain the 3gpp-Sbi-Scp-apiRoot header but contains the target apiRoot and at least one 3gpp-Sbi-Discovery header or the 3gpp-Sbi-Routing-Binding header, the pSEPP should route the request to an SCP in the target domain (e.g. pre-configured in the pSEPP or discovered from the NRF) to enable the SCP to reselect the target NF if necessary. The pSEPP may also do so if the request does not contain the 3gpp-Sbi-Scp-apiRoot header and neither the 3gpp-Sbi-Routing-Binding header nor any 3gpp-Sbi-Discovery header but contains the target apiRoot. When forwarding the request towards the target SCP, the pSEPP shall set the apiRoot of the Request URI to the apiRoot of the SCP in the target domain and it shall include the 3gpp-Sbi-Target-apiRoot header.

NOTE: This use case occurs e.g. for an initial service request when using Indirect communication without delegated discovery when the source NF has selected the target NF instance, or e.g. for a subsequent service request (targeting an existing resource) when using Indirect communication with or without delegated discovery when the source NF delegates the reselection of the target NF to the SCP (of the target PLMN), when the source PLMN does not know the address of an SCP in the target PLMN.

The request shall then be routed from the SCP in the target domain to the target NF instance (possibly (re)selected by the target NF instance) as specified in clauses 6.1 or 6.10.2 (if the request is sent directly, or via another SCP towards the target NF instance, respectively).

6.10.12.4 Authorization of NF service access

6.10.12.4.1 Authorization for indirect communication with delegated discovery

Stage 2 requirements for authorization for indirect communication with delegated discovery procedure are specified in clause 13.4.1.5.3.3 of TS 33.501 [17].

When using indirect communication with delegated discovery between different PLMNs, the requirements specified in clause 6.10.11.2 shall apply with the following additions:

- the SCP at the source PLMN shall forward, in the service request towards the target PLMN, the NF service discovery factors necessary for the service access authorization procedure;
- the SCP at the source PLMN shall forward, in the service request towards the target PLMN, the Authorization header, 3gpp-Sbi-Access-Scope header and/or 3gpp-Sbi-Other-Access-Scopes header, if these headers are received in the incoming service request; and
- the SCP at the target PLMN shall be responsible for requesting an access token for authorizing a service request received from the source PLMN, if the service request does not include any access token (obtained in an earlier service response), i.e. the requirements specified for the SCP in clause 6.10.11.2 shall apply to the SCP at the target PLMN.

6.10.12.4.2 Authorization for indirect communication without delegated discovery

Stage 2 requirements for authorization for indirect communication without delegated discovery procedure are specified in clause 13.4.1.5.3.2 of TS 33.501 [17].

When using indirect communication without delegated discovery between different PLMNs, if the selection or reselection of a producer's NF instance may apply at the SCP in the target PLMN (e.g. initial service request containing the target NF Set ID, or service request containing a routing binding header or a discovery header with the producer's NF set ID), the NF service consumer or the SCP at the source PLMN (if the NF service consumer did not include an access token request in its service request) shall include in the service request an access token that is valid for any producer's NF instance that the SCP in the target PLMN may select or reselect, i.e. an access token that is not specific to a particular producer's NF instance. This shall be an access token valid for the target NF type and producer's NF set. NOTE: As specified in step 7 of clause 4.17.10a of TS 23.502 [4], if an indication that "indirect communication without delegated discovery with NF selection at target domain is requested" was received from the target PLMN but an NF instance or NF set was not yet selected (because the NF service consumer applied indirect communication with delegated discovery), the SCP of the source PLMN selects an NF service producer instance or a NF set (where the SCP in target PLMN needs to do the NF service producer instance selection). In this case, the SCP of the source PLMN is responsible for requesting an access token to the target PLMN, and the NF service consumer includes in its service request the discovery factors necessary for the service access authorization procedure as described in clause 6.10.3.2.

6.11 Detection and handling of late arriving requests

6.11.1 General

The procedures specified in this clause aim at handling more efficiently requests which may arrive late at upstream entities, e.g. in networks experiencing processing or transport delays.

These procedures are optional to support. When supported, the use of these procedures is dependent on operator policy.

6.11.2 Detection and handling of requests which have timed out at the HTTP client

6.11.2.1 General

This procedure enables an HTTP server which receives a request to know when the request times out at the HTTP client, and to stop further processing, at the receiver and further upstream NFs, a request which has timed out at the HTTP client.

The HTTP client and HTTP server shall be NTP synchronized. This procedure may be used between NFs pertaining to the same PLMN, and if allowed by operator policy, between NFs pertaining to different PLMNs.

6.11.2.2 Principles

An HTTP client originating a request may include in the request the 3gpp-Sbi-Sender-Timestamp and the 3gpp-Sbi-Max-Rsp-Time headers indicating respectively the absolute time at which the request is originated and the maximum

time period to complete the processing of the request; both headers together indicate the absolute time at which the request times out at the HTTP client. When the original request has timed out, the HTTP client may decide to retry the same request message to the same NF service producer or an alternative NF service producer (e.g., from the same NF set) as specified in clause 5.2.8, if so, the HTTP client shall set the 3gpp-Sbi-Sender-Timestamp to a new value, i.e. to the absolute time at which the request is resent, and the 3gpp-Sbi-Max-Rsp-Time header shall indicate the maximum time period to complete the processing of the resent request.

When forwarding a request that includes the 3gpp-Sbi-Sender-Timestamp and the 3gpp-Sbi-Max-Rsp-Time headers, the SCP or SEPP may forward these headers unmodified; if the SCP or SEPP modifies and sets the 3gpp-Sbi-Sender-Timestamp to the time when it forwards the request, it shall adjust the 3gpp-Sbi-Max-Rsp-Time accordingly such as to properly reflect the time until which the HTTP client waits for a response.

Upon receipt of a request which contains the 3gpp-Sbi-Sender-Timestamp and the 3gpp-Sbi-Max-Rsp-Time headers, the HTTP server should check that the request has not already timed out at the originating HTTP client. If the request triggers the HTTP server to invoke an upstream NF service and to get the corresponding response, before further proceeding the incoming request, the HTTP server may include the 3gpp-Sbi-Sender-Timestamp and the 3gpp-Sbi-Max-Rsp-Time headers in the request towards the upstream NF service producer, in which case the HTTP server shall set the 3gpp-Sbi-Max-Rsp-Time (if included) to a value smaller than the one it received in the original request message. The HTTP server may perform additional similar checks during the processing of the request, e.g. upon receipt of a response from the next upstream NF service.

Based on local configuration, the HTTP server may reject a request that is known to have timed out with the HTTP status code "504 Gateway Timeout" and the protocol error "TIMED_OUT_REQUEST"; it may alternatively drop the request. If so, the HTTP server should initiate the release of any resource it may have successfully created towards an upstream entity, to avoid hanging resources in the network.

6.12 Binding between an NF Service Consumer and an NF Service Resource

6.12.1 General

A Binding Indication for an NF Service Resource may be provided to an NF Service Consumer of the resource as part of the Direct or Indirect Communication procedures, to be used in subsequent related service requests. This allows the NF Service Resource owner to indicate that the NF Service Consumer, for a particular resource, should be bound to an NF service instance, NF instance, NF service set or NF set. See clause 6.3.1.0 of TS 23.501 [3] and clause 4.17.12 of TS 23.502 [4].

A binding may be established or updated as part of a:

- 1) service response creating or modifying a resource, to be used for subsequent requests targeting this resource (see clause 4.17.12.2 of TS 23.502 [4]), for any API that defines resources;
- 2) service request, if the NF Service Consumer can also act as an NF Service Producer for later communication from the contacted NF Service Producer, to be used for subsequent service requests initiated by the contacted NF Service Producer (see clause 4.17.12.3 of TS 23.502 [4]);
- 3) service request creating or modifying an explicit or an implicit subscription, or as part of a notification response, to be used for subsequent notification requests initiated by the NF Service Producer (see clause 4.17.12.3 of TS 23.502 [4]);
- 4) service response creating an implicit or explicit subscription or updating a subscription, or as part of a notification request, to be used for subsequent operations on the subscription (see clause 4.17.12.4 of TS 23.502 [4]);
- 5) service request creating a callback (other than notification) resource (e.g. V-SMF or I-SMF callback URI sent to the H-SMF or SMF), or as part of a callback response, to be used for subsequent callback requests initiated by the NF Service Producer (e.g. H-SMF or SMF initiated PDU session modification);
- 6) callback request sent from a NF Service Producer to update the binding for the resource context, to be used by the NF Service Consumer for subsequent service requests addressing the resource context.

Two types of binding information are defined to manage the binding between an NF Service Consumer and an NF Service Resource:

- 1) A Binding Indication conveys binding information for a resource which must be stored by the consumer (client) of that resource and used by the client to direct future requests to the resource. When contained in a service request, the binding information is associated with a resource owned by the NF Service Consumer for the current transaction. When contained in a service response, the binding information is associated with a resource owned by the NF Service Producer for the current transaction.
- 2) A Routing Binding Indication conveys binding information to direct a request from a client to a server which has the context. A Routing Binding Indication shall only be contained in an HTTP request.

A same service request may convey more than one Binding Indication, e.g.:

- to provide bindings for notification or callback (i.e. bullets 3 or 5) and for other services that the NF service consumer can provide later as a NF Service Producer (i.e. bullet 2); or
- to provide binding information for different event notifications, when creating a subscription on behalf of another NF (see clause 6.12.4).

The scope parameter in a Binding Indication in a service request (or notification or callback response) identifies the applicability of (i.e. scenario associated with) the binding information.

A service request may convey one or more Binding Indications as described above using a 3gpp-Sbi-Binding header and/or include a Binding Routing Indication to influence routing of the request e.g. to an appropriate set of NF Service Producers or to an appropriate service set of the NF Service Producer using a 3gpp-Sbi-Routing-Binding header. A service response may convey a Binding Indication for a resource using a 3gpp-Sbi-Binding header.

NOTE 1: An HTTP request can contain for instance one 3gpp-Sbi-Binding header containing two Binding Indications for other services and for callbacks, and one 3gpp-Sbi-Routing-Binding header conveying a Routing Binding Indication.

Once a binding indication has been received for a particular resource or scope, the absence of a binding indication for the same resource or scope in a subsequent request/response message shall be interpreted as meaning that the earlier received binding indication for that resource or scope has not changed, unless specified otherwise in the rest of the specification (see scenarios with NF service producer or consumer change further down, and clause 6.12.4 for inter-AMF mobility scenarios).

In scenarios with NF service producer change (e.g. V-SMF or I-SMF change), the NF service consumer (e.g. AMF) shall delete any earlier binding indication received from the old NF service producer (e.g. old V-SMF/I-SMF) for the producer's resource (e.g. SM context resource) and replace it by any new binding indication possibly received from the new NF service producer (e.g. new V-SMF/I-SMF).

In scenarios with NF service consumer change (e.g. inter-AMF mobility), the NF service producer (e.g. SMF) shall delete any earlier binding indication received from the old NF service consumer (e.g. binding indication for callback request received from the old AMF) and replace it by any new binding indication possibly received from the new NF service consumer (e.g. new AMF).

If an SCP receives a Routing Binding Indication within a service or notification request and decides to forward that request to a next-hop SCP, it shall include the Routing Binding Indication in the forwarded request. The SCP shall remove the Routing Binding Indication if it forwards the request to the target NF.

Binding Indications and Routing Binding Indications shall include the Binding level and one or more Binding entity IDs representing all NF service instances that are capable to serve service requests targeting the resource, i.e. that share the same resource contexts.

The Binding Level indicates a preferred binding to either a NF Instance, a NF set, a NF Service Instance or a NF Service Set.

When sending a request targeting the resource context in a NF Service Producer or the session context in a NF Service Consumer, the resource URI received in the Location header or the Notification/Callback URI shall be used first if available to set the "3gpp-Sbi-Target-apiRoot" header or target URI; as an exception, if the binding indication earlier received for the target resource context or session context indicates a binding level of "NF service set", "NF Instance" or "NF Set" and alternative NF service instances within the preferred binding entity corresponding to the binding level are available, the request may alternatively be sent to one of these alternative NF service instances. When the URI received

in the Location header or the Notification/Callback URI is not reachable or when becoming aware of a NF Service Producer or Consumer change as specified in bullet 3 of clauses 6.5.3.2 and 6.5.3.3, the binding entity corresponding to the binding level shall be selected whenever possible. If this is not possible, e.g. because the preferred binding entity is not reachable, the request should be sent to any other Binding entity signalled in the Binding Indication or Routing Binding Indication, in the following decreasing order of priority:

- select an NF service instance if available in the backup NF instance, if a backup NF service instance and/or backup NF instance was signalled in the Binding Indication or Routing Binding Indication;
- select an NF service instance in the same NF service set, if a NF service Set ID was signalled in the Binding Indication or Routing Binding Indication;
- select an equivalent NF service instance in the same NF instance, if an NF instance ID was signalled in the Binding Indication or Routing Binding Indication;
- select an NF service instance in an equivalent NF service set of the backup AMF instance, if a NF service Set ID and backup AMF Instance ID was signalled in the Binding Indication or Routing Binding Indication;
- select an equivalent NF service instance in the backup AMF instance, if backup AMF Instance ID was signalled in the Binding Indication or Routing Binding Indication;
- select an NF service instance in an equivalent NF service set of another NF instance of the NF set, if an NF Service Set ID and an NF Set ID were signalled in the Binding Indication or Routing Binding Indication;
- select an equivalent NF service instance in another NF instance of the NF Set, if an NF Set ID was signalled in the Binding Indication or Routing Binding Indication.

NOTE 2: NF service instances from different NF instances are equivalent NF service instances if they share the same MCC, MNC, NID (for SNPN), ServiceName, API version, and, if applicable, NF Service Set ID (see clause 28.13 of TS 23.003 [15]).

Binding Indications shall not be used if a particular resource can only be served by a specific NF service instance of an NF instance, i.e. if NF service instances of a same NF service are not capable to share resource inside the NF Instance, unless the receiver of the Binding Indication has indicated its support of the no-redundancy indication in the Binding Indication in the SupportFeatures attribute for a specific API (see clause 5.2.3.2.6). A resource for which no Binding Indication or Routing Binding Indication is signalled shall be considered to be bound exclusively to one NF service instance, unless the NF Service resource owner instance is part of an NF set (or AMF set) or an NF service set, or unless its NF profile in the NRF indicates that it supports NF service persistence within the NF instance (see clause 6.5 of TS 23.527 [38]).

An NF service producer supporting different sets of NF service instances, e.g. serving different network slices, shall include the NF Service Set ID in the Binding Indication to enable the reselection (when required) of an alternative NF service instance from the same or an equivalent NF service set. See also clause 6.10.3.2 for requirements on the inclusion of "3gpp-Sbi-Discovery-*" headers in service requests targeting an existing resource context in the NF service producer.

A Binding Indication may be shared by multiple resource/session contexts, i.e. these resource contexts (in the NF Service Producer) or session contexts (in the NF Service Consumer) are sharing the same resilience information. The Binding Indication for multiple contexts has the same semantics as the one for a single resource/session context but with the following additions. When it is supported as indicated in the Supported Features for a specific service API:

- both NF Service Consumer and NF Service Producer can indicate if the Binding Indication for multiple contexts; and if the Binding Indication is for multiple contexts, the "group" parameter in the Binding Indication shall be set to "true";
- a group id may be included in the Binding Indication to indicate the group to which resource/session contexts pertain are sharing the same Binding Indication, when the resource/session context is created;
- the Binding Indication for a group of contexts may be updated towards each Resource URI with different apiRoot part (representing different peer NF (service) instances) or towards each Notification URI with different authority part, or with the same authority part but different callback-uri-prefix (see clause 5.2.3.3.7) if it is provided in 3gpp-Sbi-Consumer-Info header when the NF service consumer provides the Callback URI, e.g. when the NF is changed, by including an oldgroupid, oldnfnst, oldsersvset, oldsersvinst or uribase to address applicable contexts for the update of the Binding Indication. When the oldgroupid is present, the groupid shall also be present to indicate the new group id which is newly allocated. Additionally, the Binding Indication may

be updated for a group of UE contexts by including the gumai to address applicable UE contexts for the update of the Binding Indication.

6.12.2 Binding created as part of a service response

An NF Service Producer may provide a Binding Indication in a service response by including a 3gpp-Sbi-Binding header (see clause 5.2.3.2.5) in the HTTP response with:

- the binding level (bl) parameter indicating a preferred binding to either a NF Service Instance, a NF Service Set, a NF Instance or a NF set;
- at least one of the NF Service Instance (nfservinst), NF Service Set (nfserviceset), NF instance (nfinst) and NF Set (nfset) parameters, set to a NF Service Instance ID, NF Service Set ID, NF Instance ID and NF Set ID respectively, as described in Table 6.3.1.0-1 of TS 23.501 [3].

The NF Service Consumer shall store the Binding Indication received from the NF Service Producer and include it in a 3gpp-Sbi-Routing-Binding header in subsequent related service requests targeting the NF Service Resource. The NF Service Consumer or the SCP shall use this information for selecting or reselecting an NF Service Producer which has access to the NF Service Resource context, for direct or indirect communication respectively, as specified in clause 6.3.1.0 of TS 23.501 [3].

NOTE: The Binding Indication can be part of an HTTP response with or without content, e.g. in a 204 No Content. The Routing Binding Indication can be part of an HTTP request with or without content, e.g. in a DELETE request.

6.12.3 Binding created as part of a service request

As specified in clause 4.17.12.3 of TS 23.502 [4], when an AMF, V-SMF or I-SMF as NF Service Consumer sends a service request to an SMF as NF Service Producer, or when an AMF as NF Service Consumer sends a service request to a PCF or an SMSF as NF Service Producer or when an AMF as NF Service Consumer sends a service request to an I-SMF or V-SMF, or when a SMF as NF Service Consumer sends a service request to a NEF as NF Service Producer, the NF Service Consumer may provide a Binding Indication in a service request by including a 3gpp-Sbi- Binding header (see clause 5.2.3.2.6) in an HTTP request with:

- the binding level (bl) parameter indicating a preferred binding to either a NF Service Instance, a NF Service Set, a NF Instance or a NF set;
- at least one of the NF Service Instance (nfservinst), NF Service Set (nfserviceset), NF instance (nfinst) and NF Set (nfset) parameters, set to a NF Service Instance ID, NF Service Set ID, NF Instance ID and NF Set ID respectively, as described in Table 6.3.1.0-1 of TS 23.501 [3];
- the scope parameter indicating "other-service";
- optionally the service name parameter indicating the service(s) for which the binding information applies. If no service name is indicated in the Binding Indication, the binding information applies to any service that the NF Service Consumer can provide as an NF Service Producer.

When receiving a service request from an NF Service Consumer with a Binding Indication with the scope set to "other-service", the V-SMF, the I-SMF, the (Home) SMF, the SMSF, the PCF or the NEF acting as the NF Service Producer shall use this binding information when sending later on service requests for the "other-service" for existing or new resource context(s) in the original NF service consumer that are related to:

- the PDU session for which the service request is received, when the other service corresponds to an SMF service, e.g. SMF event exposure service or SMF NIDD service; or
- the UE for which the service request is received, when the other service corresponds to an AMF service, e.g. AMF event exposure service or AMF Communication Service.

The NF Service Producer shall store the Binding Indication received from the NF Service Consumer and include it in a 3gpp-Sbi-Routing-Binding header in subsequent service requests it sends, where the NF Service Consumer acts as an NF Service Producer. The NF Service Producer (when acting as a NF service consumer) or the SCP shall use this information for selecting or reselecting an NF Service Producer which has access to the original consumer's NF Service Resource context, for direct or indirect communication respectively, as specified in clause 6.3.1.0 of TS 23.501 [3].

6.12.4 Binding for explicit or implicit subscription requests

A NF Service Consumer may provide a Binding Indication:

- in a service request creating an explicit or an implicit subscription, or in a notification response, by including a 3gpp-Sbi-Binding header (see clause 5.2.3.2.6) in an HTTP request or response respectively; or
- for a default notification subscription in its NF profile in NRF (see clause 6.1.6.2.4 of TS 29.510 [8]).

The Binding Indication shall contain:

- the binding level (bl) parameter indicating a preferred binding to either a NF Service Instance, a NF Service Set, a NF Instance or a NF set;
- at least one of the NF Service Instance (nfservinst), NF Service Set (nfserviceset), NF instance (nfinst) and NF Set (nfset) parameters, set to a NF Service Instance ID, NF Service Set ID, NF Instance ID and NF Set ID respectively, as described in Table 6.3.1.0-1 of TS 23.501 [3];
- the scope parameter indicating "subscription-events" if the binding information is applicable to subscription change event notification (see clause 4.17.12.4 of TS 23.502 [4]);
- optionally, the scope parameter indicating "callback" if the binding information is applicable to notification and callback requests; the absence of this parameter shall also be interpreted as binding information is applicable to callback (i.e. notification) requests;
- optionally the service name parameter indicating the service that will handle the notification.
- optionally the prefix of the Callback URI.

When binding information is applicable to notification/callback requests, corresponding notifications are bound to:

- the NF instance or NF set (according to the binding level), if no service name was received;
- the specific service (indicated by the service name parameter) of the NF instance or NF set (according to the binding level), if a service name was received; or
- the NF service instance or NF service set (according to the binding level).

NOTE 1: The NF Service Consumer in a NF Instance or NF Set can be identified by the NF Instance Id or NF Set Id, with or without a service name parameter, or a NF Service Instance Id (together with the NF Instance Id or the NF Service Set Id) or a NF Service Set Id, where the service can be either a standardised service or a custom service.

NOTE 2: A notification can be sent to a service instance of any binding entity included in the Binding Indication, i.e. the binding entity may be other than the one(s) indicated by the binding level, if the latter(s) are not reachable. For instance, if the Binding Indication contains an NF Set ID, an NF Instance ID and a binding level is set to NF Instance, the notification can be sent to any NF instance of the NF set if the NF instance identified by the NF Instance ID is not reachable. See clause 6.3.1.0 of TS 23.501 [3].

The NF Service Producer shall store the Binding Indication received from the NF Service Consumer and include it in a 3gpp-Sbi-Routing-Binding header in subsequent notification requests it sends to the NF Service Consumer (that acts as an HTTP server) related to this subscription. See also clause 6.10.3.2 for requirements on the inclusion of "3gpp-Sbi-Discovery-*" headers in notification requests. For a default notification subscription, the NF Service Producer shall fetch the Binding Indication value (if available) from the NF profile of the NF Service Consumer and include it in a 3gpp-Sbi-Routing-Binding header in related notification requests. For notifications corresponding to default notification subscriptions using Indirect Communication with Delegated Discovery (see clause 6.10.3.3), when the notification is targeting a specific NF instance/NF service instance, the SCP shall fetch the Binding Indication value (if available) for the default notification subscription from the NF profile of the NF Service Consumer. The NF Service Producer or the SCP shall use this information for selecting or reselecting an NF Service Consumer (HTTP server) which has access to the original consumer's NF Service Resource context, for direct or indirect communication respectively, as specified in clause 6.3.1.0 of TS 23.501 [3]. If the notification endpoint provided in the subscription is not reachable, the NF Service Producer or SCP shall look up for an alternative notification endpoint address at the service level (i.e. NF Service registered in NRF) if the Binding Indication contains a service name or a binding to an NF Service Instance or NF Service Set, or at the NF instance level (i.e. NF Profile registered in NRF) otherwise. The NF Service Producer or SCP

shall derive the alternative notification URI (or callback URI) as described in clauses 6.5.2.2 and 6.5.3.2 and shall use that URI in subsequent notifications.

The NF Service Consumer may provide an updated Binding Indication to the NF Service Producer in a service request modifying the subscription or in a notification response.

A subscription request by an intermediate NF may also contain a 3gpp-Sbi-Binding-Indication-Notify-Restricted header that can be used to indicate the NF producer to restrict sending updated binding indication in all notification requests or in notification requests towards a specific callback URI (see clause 5.2.2.3.15).

The NF Service Producer may also provide a Binding Indication in a service response creating or modifying an explicit or an implicit subscription, by including a 3gpp-Sbi-Binding header (see clause 5.2.3.2.5) in the HTTP response, as specified in clause 6.12.2. If sending updated binding indication in notification requests is not restricted (i.e. if "3gpp-Sbi-Binding-Indication-Notify-Restricted" header was not previously received or the notification callback URI is not restricted by the received "3gpp-Sbi-Binding-Indication-Notify-Restricted" header) the Service Producer may also provide updated Binding Indication in a notification request generated for this subscription respectively. If the service request creates a resource and a subscription, the Binding Indication returned in the HTTP response shall apply to both the NF Service Resource and the subscription, i.e. the created resource and subscription shall be bound to the same (service) set of producers or producer instance. The NF Service Consumer shall store the Binding Indication received from the NF Service Producer and include it in a 3gpp-Sbi-Routing-Binding header in subsequent related service requests as specified in clause 6.12.2.

NOTE 3: If the NF producer can determine that a notification is sent to the NF consumer which is not the owner of the subscription, e.g. the AMF can determine that an event subscription is created by an intermediate NF on behalf of the NF consumer when different callback URIs were received for event notifications and subscription-event notifications, the NF producer does not include updated binding indication in such a notification.

For a default notification subscription, a NF Service Consumer shall update the Binding Indication value in NF profile when binding information of the default notification subscription has changed.

A subscription request may also contain a Routing Binding Indication that can be used in case of indirect communication by the SCP to route the message to the NF Service Producer.

A service request may create an explicit subscription on behalf of another NF when a "source" NF issues a service request to an "intermediate" NF who sends a subsequent request to a "target" NF. The following requirements apply in this case:

- 1) If the "target" NF is an AMF, e.g. when a UDM ("intermediate" NF) subscribes to an AMF ("target" NF) event on behalf of the NEF ("source" NF), the "intermediate" NF may include two Binding Indications:
 - a first Binding Indication for subscription change event notification sent from the "target" NF to the "intermediate" NF (e.g. notifications to UDM upon AMF change) and
 - a second Binding Indication for the event notifications sent from the "target" NF to the "source" NF (e.g. AMF notification to the NEF).

In the first Binding Indication, the scope parameter shall be set to "subscription-events".

In the second Binding Indication, the scope parameter shall be set to "callback" or be absent, and the other binding parameters ("bl", "nfset", etc.) shall be taken from the original service request from the "source" to the "intermediate" NF (e.g. binding parameters in the service request from NEF to UDM). The "source" NF (e.g. NEF) or "intermediate" NF (e.g. UDM) may also include an "nr" (notification receiver) parameter in its Binding Indication conveying the notification URI used by the "target" NF (e.g. AMF) in subsequent event notifications. This "nr" parameter allows the "target" NF to match binding information with different types of notification events in scenarios in which the "intermediate" NF combines multiple subscriptions to the "target" NF, in a single subscription request.

Upon receipt of a subscription change event notification, the "intermediate" NF may include in the notification response an (updated) Binding Indication for subscription change event notification with the scope parameter set to "subscription-events".

Upon receipt of an event notification from the "target" NF, the "source" NF may include in the notification response an (updated) Binding Indication for event notifications sent from the "target" NF to the "source" NF with the scope parameter set to "callback" or absent.

NOTE 4: Binding indications for subscription change event notification and for event notifications sent from the "target" NF to the "source" NF are transferred by the source AMF to the target AMF during inter-AMF mobility procedures, if the source AMF supports the binding procedures. Accordingly, the "intermediate" NF only needs to include a Binding Indication for subscription change event notification in the notification response if the Binding Indication is updated.

NOTE 5: Upon receipt of a subscription change event notification, the "intermediate NF" needs not include a Binding Indication for event notifications sent from the "target" NF to the "source" NF. Doing so could conflict with binding updates sent by the "source" NF to the "target" NF, if the "intermediate" NF has not been updated (yet) by the "source" NF with the binding updates.

During an inter-AMF UE mobility, if the target AMF notifies an NF service consumer of an AMF event subscription that the subscription Id has changed (see clause 5.3.2.4.1 of TS 29.518 [31]), the NF service consumer shall delete any earlier binding indication received from the source AMF for the AMF event subscription resource and replace it by any new binding indication possibly received from the target NF in the notification request.

- 2) If the "target" NF is an NF other than AMF, e.g. when a PCF/SMF ("intermediate" NF) subscribes to an UPF ("target" NF) for QoS monitoring event on behalf of an AF/NEF ("source" NF), with direct reporting of UPF events to the AF/NEF, the "intermediate" NF should forward the Binding Indication received from the "source" NF if any for the event notifications sent from the "target" NF to the "source" NF (e.g. UPF event notifications to the AF/NEF), to the "target" NF. If so, the requirements specified in 1) for the second Binding Indication shall apply for this Binding Indication. The Binding Indication may include the "nr" (notification receiver) parameter conveying the notification URI used by the "target" NF (e.g. UPF) in subsequent event notifications, to indicate that the Binding Indication corresponds to the Notification URI.

6.12.5 Binding for service requests creating a callback resource

A NF Service Consumer may provide a Binding Indication in a service request creating a callback (other than notification) resource (e.g. V-SMF or I-SMF callback URI sent to the H-SMF or SMF), by including a 3gpp-Sbi-Binding header (see clause 5.2.3.2.6) in an HTTP request as specified in clause 6.12.4, with the scope parameter being absent or indicating "callback".

The NF Service Producer shall behave as specified in clause 6.12.4, with the "notification endpoint and callback URI prefix" being replaced by the callback endpoint and callback URI prefix.

The NF Service Consumer may provide an updated Binding Indication as part of a callback response, to be used for subsequent callback requests initiated by the NF Service Producer, by including a 3gpp-Sbi-Binding header (see clause 5.2.3.2.6) in an HTTP response as specified in clause 6.12.4, with the scope parameter being absent or indicating "callback".

6.13 SBI messages correlation for network troubleshooting

6.13.1 General

The procedures defined in this clause provide means for correlating 5GC internal SBI messages (request or response) with a UE identity, by network management tools or probes that are used for network performance analysis and troubleshooting.

The procedures are optional to support. When supported, the use of these procedures is dependent on operator's policy, regulatory guidelines and security considerations.

6.13.2 SBI messages correlation using UE identifier

6.13.2.1 General

The procedure enables network analytics tools or probes, to easily identify messages that were exchanged for a given UE. When supported and configured to be used by operator's policy, an NF service consumer or NF service producer may include the UE identity in 3gpp-Sbi-Correlation-Info header, to identify the UE related to the HTTP request or response, as further defined in clause 6.13.2.2.

When included in an HTTP request or response, the 3gpp-Sbi-Correlation-Info header should contain at least one UE identifier, and no more than one of each type of UE identifier (ctype).

The NF should comply with 5GC SBI interface specific and security requirements while selecting a UE identifier to be included in the 3gpp-Sbi-Correlation-Info header. Additionally, based on operator's policy and regulatory requirements some UE identifiers may be not be allowed in the 3gpp-Sbi-Correlation-Info header for certain HTTP request or response messages.

6.13.2.2 Principles

An HTTP client originating a request may include in the request the 3gpp-Sbi-Correlation-Info header containing the UE identity that is related to the request. The HTTP client should include the SUPI in the 3gpp-Sbi-Correlation-Info header when it is available. If the SUPI is not available, the header should contain a UE identity that is known to the NF and is the most appropriate for the message context.

Upon receipt of a request that includes the 3gpp-Sbi-Correlation-Info header, the HTTP server, if it supports this header, may include the header in the response sent to the HTTP client, with the same UE identity that was contained in the 3gpp-Sbi-Correlation-Info header of the received HTTP request. If the HTTP request creates a subscription to notifications and the HTTP server supports this header, it should store the UE identifier received in the header and should include the header containing the stored UE identifier in subsequent callback notification requests.

The HTTP server may include the 3gpp-Sbi-Correlation-Info header in a successful response even when the header is not included in the request received from the HTTP client.

In an HTTP error response, the HTTP server may include the same UE identifier that was received in the 3gpp-Sbi-Correlation-Info header of the HTTP request or should not include the 3gpp-Sbi-Correlation-Info header if the header was not included in the HTTP request.

When forwarding a request or response that includes the 3gpp-Sbi-Correlation-Info header, the SCP should forward this header unmodified. For NF Discovery and (re)selection in indirect communication with or without Delegated Discovery, if the service request is received including the 3gpp-Sbi-Correlation-Info header, the SCP should include this header unmodified when it initiates a NF Discovery Request to the NRF. In indirect communication with or without Delegated Discovery, if the SCP reselects an alternative NF, the SCP should also include this header unmodified when it sends the HTTP request to the alternative NF service instance. In an inter-PLMN scenario, the SEPP may remove the header based on operator policies and regulatory requirements.

6.13.3 SBI messages correlation using NF Peer Information

6.13.3.1 General

The procedure enables network elements (such as NFs, SCPs, SEPPs, network analytics tools or probes, etc.), to obtain source and destination information of messages that were exchanged between a specified pair of NF (Service) instances. An NF as HTTP client or NF as HTTP server should include the NF (Service) instance IDs in 3gpp-Sbi-NF-Peer-Info header or the FQDN of the destination NF (if the NF (Service) instance ID of the destination NF is unknown), to identify the HTTP requests or responses between the given pair of NF (Service) instances, as further defined in clause 6.13.3.2.

6.13.3.2 Principles

An HTTP client originating a request should include in the request the 3gpp-Sbi-NF-Peer-Info header containing the Source NF (Service) instance ID and if available the Destination NF (Service) instance ID.

Upon receipt of a request that includes the 3gpp-Sbi-NF-Peer-Info, the HTTP server should insert the header in the response sent to the HTTP client, with source and destination peer info corresponding to the destination and source peer info in the request respectively (i.e. swap the received source and destination peer info in the response). When the destination NF FQDN was included in the request, the HTTP server should include the FQDN as the source NF FQDN in the response. The HTTP server should include the 3gpp-Sbi-NF-Peer-Info header in a response even when the header is not included in the request received from the HTTP client.

If the destination peer information provided by HTTP client in the request does not match the information of the HTTP server (e.g. due to the HTTP server having updated its NF (Service) instance ID), the HTTP server should include the updated NF (Service) instance ID values in the response header sent to HTTP client.

When forwarding a request or response that includes the 3gpp-Sbi-NF-Peer-Info header, the SCP should forward this header and may update the destination peer info if the receiver NF is (re)selected; the SCP shall also update the srcscp/dstscp components, based on the source and destination SCP of the forwarded HTTP request or response, as described in clause 5.2.3.2.21; the SEPP shall also update the dstscp component (if SEPP relays the request towards an SCP).

In an inter-PLMN scenario, the SEPP may remove the header based on operator policies. If an SCP or SEPP generates an error response to a request including this header, the SCP and SEPP should insert the header in the response with source peer info containing the information of the SCP or SEPP, and with destination peer info containing the source peer info in the request respectively.

6.14 Indicating the purpose of Inter-PLMN signaling

6.14.1 General

The procedures defined in this clause provide means for two PLMNs to send/receive the purpose of inter-PLMN signaling. This can be used to help operators to avoid receiving any signaling from different operator without any relevant contract.

SEPP shall be preconfigured to understand the relationship with other PLMNs, e.g. roaming. The means on how to configure the relationship is outside the scope of the Release.

6.14.2 Inclusion of the intended purpose

An NF Consumer or SCP in case of model D may include in the HTTP request the intended purpose of the request that is targeted to PLMN different from the source PLMN, using 3gpp-Sbi-Interplmn-Purpose as defined in clause 5.2.3.3.11. The purposes shall be selected from the values specified in TS 29.573 [27] clause 6.1.5.3.9. In addition, the any additional information may be associated for indicating the exact purpose.

6.14.3 Evaluating the intended purpose

When the SEPP receives request from NF consumer or SCP of the same network bound to another network (in case of cSEPP), or from the peer SEPP (in case of pSEPP), the SEPP shall evaluate the intended purpose of the signaling from the following information:

- Source PLMN;
- Target PLMN; and
- intended purpose in the received in the 3gpp-Sbi-Interplmn-Purpose header, if available

If the SEPP (i.e. cSEPP) receives request from NF consumer or SCP of the same network bound to another network including 3gpp-Sbi-Interplmn-Purpose header, the receiving SEPP shall compare the value received in the header with the preconfigured value of allowed intended purpose between the source and target PLMN.

If the SEPP (i.e. pSEPP) receives from the peer SEPP including 3gpp-Sbi-Interplmn-Purpose header, the receiving SEPP shall compare the value received in the header with the pre-negotiated value of allowed intended purpose between the source and target PLMN during Security Capability Negotiation procedure specified in TS 29.573 [27].

The receiving SEPP shall:

- If the purpose in the 3gpp-Sbi-Interplmn-Purpose header matches with any one of the preconfigured purposes (for cSEPP) or pre-negotiated purposes (for pSEPP) as allowed by the receiving SEPP, then the receiving SEPP shall continue processing the request.
- Else, the receiving SEPP shall reject the message with 403 Forbidden with ProblemDetails REQUESTED_PURPOSE_NOT_ALLOWED as defined in Table 5.2.7.4-1.

EXAMPLE The following example describes how cSEPP and pSEPP evaluates and process with regards to the intended purposes.

- a) cSEPP and pSEPP are configured with the allowed purpose =X, Y

- Case 1:
NFc/SCP sends the first message to cSEPP with purpose = X. In this case, cSEPP validates the message against the configured purpose and allows it. Using the N32 connection established between cSEPP and pSEPP for purpose = X, cSEPP delivers the message to pSEPP. Then only pSEPP checks the purpose=X over N32f with the pre-negotiated purpose.
 - Case 2:
NFc/SCP sends a second message to cSEPP with the purpose=Z. Here, cSEPP rejects it on its own because it is not allowed purpose for cSEPP (configured).
- b) cSEPP is configured with allowed purpose X, Y and pSEPP is configured with X, K. The negotiated/allowed purpose between cSEPP and pSEPP is X.
- Case 3:
NFc/SCP sends a message to cSEPP with purpose =Y. In this case, cSEPP autonomously rejects the message based on the negotiated purposes (i.e. purpose X) with pSEPP.

If the SEPP receives request from NF consumer or SCP of the same network bound to another network (in case of cSEPP), or from the peer SEPP (in case of pSEPP) that does not include the 3gpp-Sbi-Interplmn-Purpose header, the receiving SEPP shall by default consider this as roaming and inter-PLMN mobility in order to allow backward compatibility for NF consumers which do not support the 3gpp-Sbi-Interplmn-Purpose header, and apply the policy accordingly. The purpose of transactions of AMFs or SMFs between two different VPLMNs, i.e. inter AMF or inter V-SMF signalling from VPLMN1 to VPLMN2 shall be considered as inter PLMN mobility.

Annex A (informative): Client-side Adaptive Throttling for Overload Control

This clause contains an example algorithm to make an NF Service Consumer adjust the traffic rate sent to an NF Service Producer based on the number of received "rejects" of HTTP requests with a status code "503 Service Unavailable", or requests that have timed-out and the response was never received. This algorithm is described in the book "Betsy Beyer, et al; Google: Site Reliability Engineering" (<https://landing.google.com/sre/book.html>), clause 21, "Handling Overload".

NOTE: The reference link provided to the book can change and hence the name of the book is expected to be used for referring to the latest edition.

Each client (NF Service Consumer) keeps track of the following counters during a certain time window:

- Requests: The number of requests that the client (NF Service Consumer) needs to handle. Under normal operation (no overload), all these requests are sent to the server (NF Service Producer). Under an overload situation, part of these requests are locally rejected by the client (and not sent to the server), and the rest of the requests are sent to the server.
- Accepts: The number of requests accepted by the server (i.e., requests for which a response has been effectively received at the client, with a status code other than "503 Service Unavailable").

When there is no server overload, these values are equal.

When there is an overload status in the server, the rate between "Accepts" and "Requests" decreases progressively. When this rate falls below a certain point (given by an algorithm parameter named "K"), the client shall start dropping some requests locally and not send them to the server.

The local rejection of requests can be done by calculating a "Client request rejection probability", as:

$$\max\left(0, \frac{\text{requests} - K \times \text{accepts}}{\text{requests} + 1}\right)$$

So, for example, assuming that the K parameter is set at 1.5:

- if the server accepts >67% of the traffic, and rejects <33% of the traffic, the client does not take any throttling action, and keeps sending to the server all the traffic it has available for processing
- if, during a first time-window, the server accepts, e.g., only 60% of the requests, and rejects 40% due to overload, the application of this algorithm implies that the client must drop locally 10% of the requests (probabilistically), and only send to the server the remainder 90% of its traffic.
- if, during a second time-window, the client keeps the same amount of available traffic to handle, but the server continues rejecting requests with same rate as before (40%) of the received requests, the application of the algorithm again, results in increasing the drop rate to 14.5%, and sending to the server only 85.5% of the available traffic.

The value of the parameter K, along with the size of the time window during which the total number of "requests" and "accepts" is accounted for, has a fundamental role on how the algorithm behaves. If K is higher, the algorithm is more "permissive", and the client does not start dropping requests locally until the rejection rate is higher (e.g., >50%, for K = 2); if K is lower, the algorithm is more "aggressive", and the client starts dropping requests sooner (e.g., K = 1.1 implies to start dropping requests as soon as the server rejects >10% of the requests).

Annex B (normative): 3gpp-Sbi-Callback Types

This annex specifies the allowed 3GPP SBI callback type values for the "3gpp-Sbi-Callback" HTTP custom header specified in clause 5.2.3.2.3.

Table B-1 contains a non-exhaustive list of callbacks that are invoked in the 5GS.

Table B-1: Non-exhaustive list of values for the "3gpp-Sbi-Callback" Custom HTTP Header

Value for "3gpp-Sbi-Callback" Custom HTTP Header	Reference
"Nsmf_PDUSession_Update"	TS 29.502 [28], clause 5.2.2.8.3.
"Nsmf_PDUSession_StatusNotify"	TS 29.502 [28], clause 5.2.2.10.
"Nudm_SDM_Notification"	TS 29.503 [29], clause 6.1.5.2.
"Nudm_UECM_DeregistrationNotification"	TS 29.503 [29], clause 6.2.5.2.
"Nudm_UECM_PCSCFRestorationNotification"	TS 29.503 [29], clause 6.2.5.3.
"Nnrf_NFManagement_NFStatusNotify"	TS 29.510 [8], clause 6.1.5.2.
"Namf_EventExposure_Notify"	TS 29.518 [31], clause 6.2.5.2.
"Npcf_UEPolicyControl_UpdateNotify"	TS 29.525 [35], clauses 4.2.4, 5.5.2 and 5.5.3.
"Nnssf_NSSAIAvailability_Notification"	TS 29.531 [32], clause 6.2.5.2.
"Namf_Communication_AMFStatusChangeNotify"	TS 29.518 [31], clause 6.1.5.2.
"Ngmlc_Location_EventNotify"	TS 29.515 [40], clause 6.1.4.2.
"Nchf_ConvergedCharging_Notify"	TS 32.291 [42], clause 6.1.5.2.
"Nnssaaf_NSSAA_ReAuthentication"	TS 29.526 [44], clause 6.1.5.2.
"Nnssaaf_NSSAA_Revocation"	TS 29.526 [44], clause 6.1.5.3.
"N5g-ddnmf_Discovery_MonitorUpdateResult"	TS 29.555 [46], clause 6.1.5.2.
"N5g-ddnmf_Discovery_MatchInformation"	TS 29.555 [46], clause 6.1.5.3.
NOTE:	The clauses referenced in the second column of the table identify the specific callback procedures for which the value defined in the first column shall apply.

For notification and callback service operations (used across PLMNs or within a PLMN) that are not part of Table B.1, the value of the header shall be constructed as follows:

"<API name taken from the heading of the relevant annex A.x as defined in the corresponding TS of that API>_<name of the callback service operation in the corresponding OpenAPI specification file>"

EXAMPLE: 3gpp-Sbi-Callback: Nsmf_PDUSession_smContextStatusNotification

for the Notify SM Context Status service operation where "smContextStatusNotification" corresponds to the name of the callback service operation defined in the OpenAPI specification file of the Nsmf_PDUSession API (as shown in the following excerpt from Annex A.2 of TS 29.502 [28]):

```

callbacks:
  smContextStatusNotification:
    '{$request.body#/smContextStatusUri}':

```

NOTE: Several values in Table B-1 do not comply with the construction rule for the "3gpp-Sbi-Callback" HTTP header described in this clause; in those cases, the values explicitly included in Table B-1 take precedence over the construction rule.

Values for the "3gpp-Sbi-Callback" Custom HTTP Header shall be case-insensitive.

Annex C (informative): Internal NF Routing of HTTP Requests

The internal details of the architecture of a Network Function instance is out of the scope of 3GPP and are entirely implementation-specific. This annex describes how an instance of an NF Service Producer can route internally HTTP requests received on a given Service-Based Interface.

Figure C-1 illustrates an example component architecture where incoming HTTP requests are received and processed in a component named as "Ingress Proxy" module and route them to the appropriate computing resource in the NF.

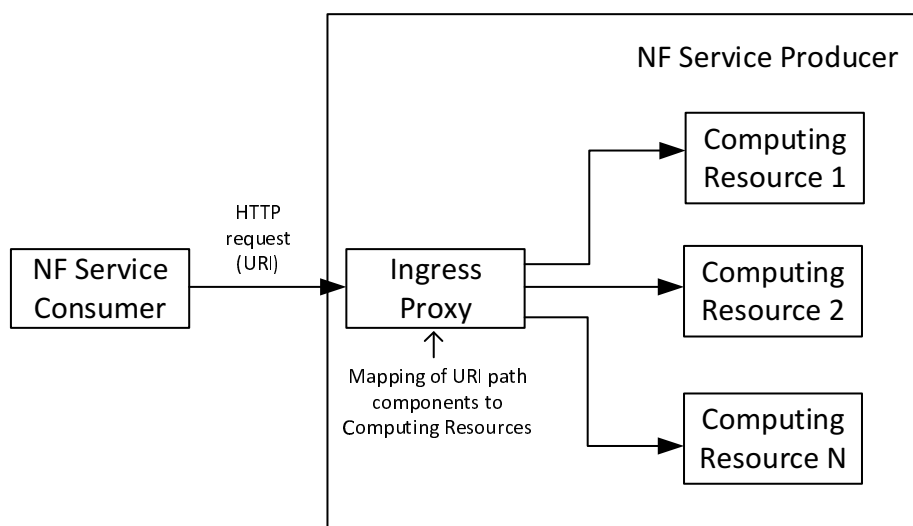


Figure C-1: Internal message routing inside NF Service Producer

The Ingress Proxy may parse any of the different components in the HTTP request, but typically it may parse the path of the URI (i.e. the `:path` pseudo-header in the HTTP/2 request). Parsing of other component in the request message, such as the HTTP body, is also possible but it is not desirable as it requires the parsing of the entire body (i.e. a JSON document) which is a much more computing-intensive task.

The path component of the URI contains the service name of the requested SBA service, so frequently the routing is done based on this component.

It is also frequent to inspect other components of the path (i.e. path segments), to do a more fine-grained routing and direct requests done on a specific HTTP resource(s) towards a given computing resource(s).

It can be noted that the path components used to determine the target computing resource typically do not need to be statically defined but are frequently defined in terms of "variables", or placeholders, similarly to how they are defined in the OpenAPI specification language (a mechanism usually known as "path templating"). See: <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.0.md#path-templating>

Annex D (Normative): ABNF grammar for 3GPP SBI HTTP custom headers

D.1 General

This Annex contains a self-contained set of ABNF rules, comprising the re-used rules from IETF RFCs, and the rules defined by the 3GPP custom headers defined in this specification (see clause 5.2.3).

This grammar may be used as input to existing tools to help implementations to parse 3GPP custom headers.

Given that this Annex is included in relation to ABNF tooling, the following aspects should be observed:

- The "list extension" rule defined in IETF RFC 9110 [11], section 5.6.1, is typically not supported by ABNF tooling, so rules included this syntax have been re-written with their equivalent syntax:

```
Rule1 = 1#element
```

```
Rule2 = #element
```

is re-written as:

```
Rule1 = element *( OWS "," OWS element )
```

```
Rule2 = [ element *( OWS "," OWS element ) ]
```

- The ABNF specification defined in IETF RFC 5234 [43] does not describe how to evaluate the "alternative" operator (i.e., "Rule1 / Rule2"), when both rules match, but one of them have a "longer" match than the other. Based on existing ABNF rules in different RFCs, it seems that the approach is to apply a "longest" match (see, for example, the "dec-octet" rule in IETF RFC 3986 [14]); however, most existing ABNF tools apply a "left-most" match. When this occurs, the rules in this Annex have been re-written by placing the rule with a longest match on the left side, and the shortest match on the right side of the alternative operator.

NOTE: This ambiguity in the ABNF language has been identified in other IETF specifications, such as IETF RFC 3501; in that case, a requirement was added to the RFC (see section 9): *"In the case of alternative or optional rules in which a later rule overlaps an earlier rule, the rule which is listed earlier MUST take priority"*.

For example:

```
Rule1 = "imei" / "imeisv"
```

```
Rule2 = DIGIT / ( "1" DIGIT ) ; allows an integer in the range 0-19
```

is re-written as:

```
Rule1 = "imeisv" / "imei"
```

```
Rule2 = ( "1" DIGIT ) / DIGIT ; allows an integer in the range 0-19
```

- The rules containing prose text ("*<...>*") are typically not supported by ABNF tooling, so the rule in IETF RFC 3986 [14]:

```
path-empty = 0<pchar>
```

is re-written as:

```
path-empty = 0pchar
```

- Some rules in IETF RFC 9110 [11] ("day", "month", "year") contain the same rule names as in IETF RFC 5322 [37], so these clashing rule names in IETF RFC 9110 [11] have been renamed as "day-rfc9110", "month-rfc9110" and "year-rfc9110" respectively.

D.2 ABNF definitions (Filename: "TS29500_CustomHeaders.abnf")

```

; -----
; RFC 5234
; -----

HTAB  = %x09 ; horizontal tab

LF    = %x0A ; linefeed

CR    = %x0D ; carriage return

SP    = %x20

DQUOTE = %x22 ; " (Double Quote)

DIGIT = %x30-39 ; 0-9

ALPHA = %x41-5A / %x61-7A ; A-Z / a-z

VCHAR = %x21-7E ; visible (printing) characters

WSP   = SP / HTAB ; white space

CRLF  = CR LF ; Internet standard newline

HEXDIG = DIGIT / "A" / "B" / "C" / "D" / "E" / "F"

; -----
; RFC 3986
; -----

unreserved  = ALPHA / DIGIT / "-" / "." / "_" / "~"

pct-encoded = "%" HEXDIG HEXDIG

sub-delims  = "!" / "$" / "&" / "'" / "(" / ")" / "*" / "+" / "," / ";" / "="

pchar       = unreserved / pct-encoded / sub-delims / ":" / "@"

segment     = *pchar

segment-nz  = 1*pchar

path-abempty = *( "/" segment )

path-absolute = "/" [ segment-nz *( "/" segment ) ]

path-rootless = segment-nz *( "/" segment )

path-empty  = 0pchar

IPvFuture   = "v" 1*HEXDIG "." 1*( unreserved / sub-delims / ":" )

dec-octet   = "25" %x30-35 / "2" %x30-34 DIGIT / "1" 2DIGIT / %x31-39 DIGIT / DIGIT

h16         = 1*4HEXDIG

ls32        = ( h16 ":" h16 ) / Ipv4address

Ipv4address = dec-octet "." dec-octet "." dec-octet "." dec-octet

Ipv6address = 6( h16 ":" ) ls32
             / "::" 5( h16 ":" ) ls32
             / [ h16 ] "::" 4( h16 ":" ) ls32
             / [ *1( h16 ":" ) h16 ] "::" 3( h16 ":" ) ls32
             / [ *2( h16 ":" ) h16 ] "::" 2( h16 ":" ) ls32
             / [ *3( h16 ":" ) h16 ] "::" h16 ":" ls32
             / [ *4( h16 ":" ) h16 ] "::" ls32
             / [ *5( h16 ":" ) h16 ] "::" h16

```

```

        / [ *6( h16 ":" ) h16 ] "::"
IP-literal   = "[ ( Ipv6address / IpvFuture ) ]"
reg-name     = *( unreserved / pct-encoded / sub-delims )
host         = IP-literal / Ipv4address / reg-name
port         = *DIGIT
scheme       = ALPHA *( ALPHA / DIGIT / "+" / "-" / "." )
userinfo     = *( unreserved / pct-encoded / sub-delims / ":" )
authority    = [ userinfo "@" ] host [ ":" port ]
hier-part    = "/" authority path-abempty / path-absolute / path-rootless / path-empty
query        = *( pchar / "/" / "?" )
fragment     = *( pchar / "/" / "?" )
URI          = scheme ":" hier-part [ "?" query ] [ "#" fragment ]

; -----
; RFC 5322
; -----

obs-FWS      = 1*WSP *( CRLF 1*WSP )
FWS          = ( [ *WSP CRLF ] 1*WSP ) / obs-FWS
obs-NO-WS-CTL = %d1-8 / %d11 / %d12 / %d14-31 / %d127
obs-ctext    = obs-NO-WS-CTL
ctext        = %d33-39 / %d42-91 / %d93-126 / obs-ctext
obs-qp       = "\" ( %d0 / obs-NO-WS-CTL / LF / CR )
quoted-pair  = ( "\" ( VCHAR / WSP ) ) / obs-qp
ccontent     = ctext / quoted-pair / comment
comment      = "( " *( [ FWS ] ccontent ) [ FWS ] )"
CFWS         = ( 1*( [ FWS ] comment ) [ FWS ] ) / FWS
day-name     = "Mon" / "Tue" / "Wed" / "Thu" / "Fri" / "Sat" / "Sun"
obs-day-of-week = [ CFWS ] day-name [ CFWS ]
day-of-week  = ( [ FWS ] day-name ) / obs-day-of-week
obs-day      = [ CFWS ] 1*2DIGIT [ CFWS ]
day          = ( [ FWS ] 1*2DIGIT FWS ) / obs-day
month        = "Jan" / "Feb" / "Mar" / "Apr" / "May" / "Jun"
             / "Jul" / "Aug" / "Sep" / "Oct" / "Nov" / "Dec"
obs-year     = [ CFWS ] 2*DIGIT [ CFWS ]
year         = ( FWS 4*DIGIT FWS ) / obs-year
date         = day month year
obs-hour     = [ CFWS ] 2DIGIT [ CFWS ]
hour         = obs-hour / 2DIGIT
obs-minute   = [ CFWS ] 2DIGIT [ CFWS ]
minute       = obs-minute / 2DIGIT
obs-second   = [ CFWS ] 2DIGIT [ CFWS ]

```

```

second          = obs-second / 2DIGIT
time-of-day     = hour ":" minute [ ":" second ]
obs-zone        = "UT" / "GMT" / "EST" / "EDT" / "CST" / "CDT" / "MST" / "MDT"
                 / "PST" / "PDT" / %d65-73 / %d75-90 / %d97-105 / %d107-122
zone            = ( FWS ( "+" / "-" ) 4DIGIT ) / obs-zone
time            = time-of-day zone
date-time       = [ day-of-week "," ] date time [ CFWS ]

; -----
;   RFC 6749
; -----

NQCHAR = %x21 / %x23-5B / %x5D-7E

; -----
;   RFC 9110
; -----

OWS          = *( SP / HTAB )
RWS          = 1*( SP / HTAB )
tchar        = "!" / "#" / "$" / "%" / "&" / "'" / "*" / "+" / "-"
               / "." / "^" / "_" / "`" / "|" / "~" / DIGIT / ALPHA
token        = 1*tchar
BWS          = OWS
quoted-string = DQUOTE *( qdtext / quoted-pair ) DQUOTE
qdtext       = HTAB / SP / %x21 / %x23-5B / %x5D-7E / obs-text
obs-text     = %x80-FF
date1        = day-rfc9110 SP month-rfc9110 SP year-rfc9110 ; e.g., 02 Jun 1982
day-rfc9110  = 2DIGIT
month-rfc9110 = %x4A.61.6E ; "Jan", case-sensitive
               / %x46.65.62 ; "Feb", case-sensitive
               / %x4D.61.72 ; "Mar", case-sensitive
               / %x41.70.72 ; "Apr", case-sensitive
               / %x4D.61.79 ; "May", case-sensitive
               / %x4A.75.6E ; "Jun", case-sensitive
               / %x4A.75.6C ; "Jul", case-sensitive
               / %x41.75.67 ; "Aug", case-sensitive
               / %x53.65.70 ; "Sep", case-sensitive
               / %x4F.63.74 ; "Oct", case-sensitive
               / %x4E.6F.76 ; "Nov", case-sensitive
               / %x44.65.63 ; "Dec", case-sensitive
year-rfc9110 = 4DIGIT
codings       = content-coding / "identity" / "*"
content-coding = token
weight        = OWS ";" OWS "q=" qvalue
qvalue        = ( "0" [ "." *3DIGIT ] ) / ( "1" [ "." *3"0" ] )
credentials   = auth-scheme [ 1*SP ( token68
                 / [ ( "," / auth-param ) *( OWS "," [ OWS auth-param ] ) ] ) ]
auth-scheme   = token
auth-param    = token BWS "=" BWS ( token / quoted-string )

```

```

token68          = 1*( ALPHA / DIGIT / "-" / "." / "_" / "~" / "+" / "/" ) "*" =

; -----
; 3GPP TS 29.500
;
; Version: 19.5.0 (December 2025)
;
; (c) 2025, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).
; -----

;
; Header: 3gpp-Sbi-Message-Priority
;

Sbi-Message-Priority-Header = "3gpp-Sbi-Message-Priority:" OWS
                               ( "3" %x30-31 / %x31-32 DIGIT / DIGIT ) OWS

;
; Header: 3gpp-Sbi-Callback
;

Sbi-Callback-Header = "3gpp-Sbi-Callback:" OWS cbtype *( ";" OWS "apiversion=" majorversion ) OWS
cbchar                = "-" / "_" / DIGIT / ALPHA
cbtype                = 1*cbchar
majorversion          = *DIGIT

;
; Header: 3gpp-Sbi-Target-apiRoot
;

Sbi-Target-ApiRoot-Header = "3gpp-Sbi-Target-apiRoot:" OWS sbi-scheme "://" sbi-authority
                               [ prefix ] OWS

sbi-scheme              = "https" / "http"
sbi-authority            = host [ ":" port ]
prefix                  = path-absolute

;
; Header: 3gpp-Sbi-Scp-apiRoot
;

Sbi-Scp-ApiRoot-Header = "3gpp-Sbi-Scp-apiRoot:" OWS sbi-scheme "://" sbi-authority
                               [ prefix ] OWS

;
; Header: 3gpp-Sbi-Routing-Binding
;

Sbi-Routing-Binding-Header = "3gpp-Sbi-Routing-Binding:" OWS "bl=" blvalue
                               1*( ";" OWS parameter )
                               [ ";" OWS callback-uri-prefix ] OWS

blvalue                 = "nf-instance"
                               / "nf-set"
                               / "nfservice-instance"
                               / "nfservice-set"

parametername           = "nfinst"
                               / "nfset"
                               / "nfservinst"
                               / "nfserviceset"
                               / "servname"

```

```

        / "backupamfinst"
        / "backupnf"

parameter          = parametername "=" token

;
; Header: 3gpp-Sbi-Binding
;

Sbi-Binding-Header = "3gpp-Sbi-Binding:" OWS binding-element *( OWS "," OWS binding-element ) OWS

binding-element   = "bl=" blvalue 1*( ";" OWS bh-parameter )
                   [ ";" OWS recoverytime ]
                   [ ";" OWS notif-receiver ]
                   [ ";" OWS "group=" groupvalue ]
                   [ 1*( ";" OWS groupparameter ) ]
                   [ ";" OWS "no-redundancy=" no-red-value ]
                   [ ";" OWS callback-uri-prefix ] OWS

bh-parametername  = parametername / "scope"

bh-parameter      = bh-parametername "=" token

recoverytime      = "recoverytime=" OWS DQUOTE date-time DQUOTE

notif-receiver     = "nr=" URI

groupvalue         = "true" / "false"

groupparametername = "oldgroupid"
                   / "groupid"
                   / "uribase"
                   / "oldnfinst"
                   / "oldservset"
                   / "oldservinst"
                   / "guami"

groupparameter     = groupparametername "=" token

no-red-value       = "true"

;
; Header: 3gpp-Sbi-Producer-Id
;

Sbi-Producer-Id-Header = "3gpp-Sbi-Producer-Id:" OWS "nfinst=" nfinst
                        [ OWS ";" OWS "nfservinst=" nfservinst ]
                        [ OWS ";" OWS "nfset=" nfset ]
                        [ OWS ";" OWS "nfserviceset=" nfserviceset ] OWS

nfinst             = 8HEXDIG "-" 4HEXDIG "-" 4HEXDIG "-" 4HEXDIG "-" 12HEXDIG

nfservinst         = token

nfset              = token

nfserviceset       = token

;
; Header: 3gpp-Sbi-Oci
;

Sbi-Oci-Header     = "3gpp-Sbi-Oci:" OWS oci-element *( OWS "," OWS oci-element ) OWS

oci-element         = timestamp ";" RWS validityPeriod ";" RWS olcMetric ";" RWS olcScope
                   [ ";" RWS olcExtRegTimer ]

timestamp           = "Timestamp:" RWS DQUOTE date-time DQUOTE

validityPeriod      = "Period-of-Validity:" RWS 1*DIGIT "s"

olcMetric           = "Overload-Reduction-Metric:" RWS ( "100" / %x31-39 DIGIT / DIGIT ) "%"

```

```

olcScope          = nfProducerScope / nfConsumerScope / scpScope / seppScope
nfProducerScope = ( ( "NF-Instance:" RWS nfirst )
                    / ( "NF-Set:" RWS nfset )
                    / ( "NF-Service-Instance:" RWS nfirst [ ";" RWS "NF-Inst:" RWS nfirst ] )
                    / ( "NF-Service-Set:" RWS nfserviceset )
                    ) [ ";" RWS sNssaiList ";" RWS dnnList ]
nfConsumerScope = ( "NFC-Instance:" RWS nfirst [ ";" RWS "Service-Name:" RWS servname ] )
                  / ( "NFC-Set:" RWS nfset [ ";" RWS "Service-Name:" RWS servname ] )
                  / ( "NFC-Service-Instance:" RWS nfirst [ ";" RWS "NF-Inst:" RWS nfirst ] )
                  / ( "NFC-Service-Set:" RWS nfserviceset )
                  / ( "Callback-Uri:" RWS DQUOTE URI DQUOTE *( RWS "&" RWS DQUOTE URI DQUOTE ) )

servname          = token
scpScope          = "SCP-FQDN:" RWS fqdn
seppScope         = "SEPP-FQDN:" RWS fqdn
fqdn              = token
dnnList           = "DNN:" RWS 1*tchar *( RWS "&" RWS 1*tchar )
sNssaiList        = "S-NSSAI:" RWS snssai *( RWS "&" RWS snssai )
snssai            = 1*tchar
olcExtRegTimer    = "Extend-Registration-Timer:" RWS ert-value
ert-value         = "true" / "false"

;
; Header: 3gpp-Sbi-Lci
;
Sbi-Lci-Header    = "3gpp-Sbi-Lci:" OWS lc-element *( OWS "," OWS lc-element ) OWS
lc-element        = timestamp ";" RWS lcMetric ";" RWS lcScope
lcMetric          = "Load-Metric:" RWS ( "100" / %x31-39 DIGIT / DIGIT ) "%"
lcScope           = lcNfProducerScope / scpScope / seppScope
lcNfProducerScope = ( ( "NF-Instance:" RWS nfirst )
                    / ( "NF-Set:" RWS nfset )
                    / ( "NF-Service-Instance:" RWS nfirst [ ";" RWS "NF-Inst:" RWS nfirst ] )
                    / ( "NF-Service-Set:" RWS nfserviceset )
                    ) [ ";" RWS sNssaiList ";" RWS dnnList ";" RWS relativeCapacity ]
relativeCapacity = "Relative-Capacity:" RWS ( "100" / 1*2DIGIT ) "%"

;
; Header: 3gpp-Sbi-Client-Credentials
;
Sbi-Client-Credentials-Header = "3gpp-Sbi-Client-Credentials:" OWS jwt OWS
jwt                            = 1*b64urlchar "." 1*b64urlchar "." 1*b64urlchar
b64urlchar                     = ALPHA / DIGIT / "-" / "_"

;
; Header: 3gpp-Sbi-Source-NF-Client-Credentials
;
Sbi-Source-NF-Client-Credentials-Header = "3gpp-Sbi-Source-NF-Client-Credentials:" OWS jwt OWS

;

```

```

; Header: 3gpp-Sbi-Nrf-Uri
;

Sbi-Nrf-Uri-Header = "3gpp-Sbi-Nrf-Uri:" OWS nrfUriParam *( OWS ";" OWS nrfUriParam ) OWS

nrfUriParam      = nrfUriParamName ":" RWS ( nrfUriParamValue1 / nrfUriParamValue2 )

nrfUriParamName  = "nnrf-disc"
                  / "nnrf-nfm"
                  / "nnrf-oauth2"
                  / "oauth2-requested-services"
                  / token

nrfUriParamValue1 = DQUOTE URI DQUOTE

nrfUriParamValue2 = ( nrfServiceName *( RWS "&" RWS nrfServiceName ) )

nrfServiceName   = "nnrf-disc" / "nnrf-nfm"

;
; Header: 3gpp-Sbi-Target-Nf-Id
;

Sbi-Target-Nf-Id-Header = "3gpp-Sbi-Target-Nf-Id:" OWS "nfinst=" nfinst
                          [ ";" OWS "nfservinst=" nfservinst ] OWS

;
; Header: 3gpp-Sbi-Max-Forward-Hops
;

Sbi-Max-Forward-Hops-Header = "3gpp-Sbi-Max-Forward-Hops:" OWS ( %x31-39 DIGIT / DIGIT )
                              ";" OWS "nodetype=" nodetypevalue OWS

nodetypevalue          = "scp"

;
; Header: 3gpp-Sbi-Originating-Network-Id
;

Sbi-Originating-Network-Id-Header = "3gpp-Sbi-Originating-Network-Id:" OWS 3DIGIT "-" 2*3DIGIT
                                     [ "-" 11HEXDIG ] [ ";" OWS srcinfo ] OWS

srcinfo                 = "src" ":" RWS srctype "-" srcfqdn

srctype                 = "SCP" / "SEPP"

srcfqdn                 = 4*( ALPHA / DIGIT / "-" / "." )

;
; Header: 3gpp-Sbi-Access-Scope
;

Sbi-Access-Scope-Header = "3gpp-Sbi-Access-Scope:" OWS scope-token *( SP scope-token ) OWS

scope-token             = 1*NQCHAR

;
; Header: 3gpp-Sbi-Other-Access-Scopes
;

Sbi-Other-Access-Scopes-Header = "3gpp-Sbi-Other-Access-Scopes:" OWS scope-token
                                 *( SP scope-token ) OWS

;
; Header: 3gpp-Sbi-Access-Token
;

```

```

Sbi-Access-Token-Header = "3gpp-Sbi-Access-Token:" OWS credentials OWS

;
; Header: 3gpp-Sbi-Target-Nf-Group-Id
;

Sbi-Target-Nf-Group-Id-Header = "3gpp-Sbi-Target-Nf-Group-Id:" OWS "nfgid=" nfGroupIdValue OWS
nfGroupIdValue                = DQUOTE token DQUOTE

;
; Header: 3gpp-Sbi-Nrf-Uri-Callback
;

Sbi-Nrf-Uri-Callback-Header = "3gpp-Sbi-Nrf-Uri-Callback:" OWS
                             nrfUriCallbackParam *( OWS ";" OWS nrfUriCallbackParam ) OWS
nrfUriCallbackParam          = nrfUriCallbackParamName ":" RWS nrfUriCallbackParamValue
nrfUriCallbackParamName     = "nnrf-disc" / "nnrf-nfm" / token
nrfUriCallbackParamValue    = DQUOTE URI DQUOTE

;
; Header: 3gpp-Sbi-NF-Peer-Info
;

Sbi-NF-Peer-Info-Header = "3gpp-Sbi-NF-Peer-Info:" OWS peerinfo *( ";" OWS peerinfo ) OWS
peerinfo                 = peertype "=" token
peertype                  = "srcinst"
                           / "srcservinst"
                           / "srcscp"
                           / "srcsepp"
                           / "dstinst"
                           / "dstservinst"
                           / "dstscp"
                           / "dstsepp"
                           / "dstfqdn"
                           / "srcfqdn"

;
; Header: 3gpp-Sbi-Sender-Timestamp
;

Sbi-Sender-Timestamp-Header = "3gpp-Sbi-Sender-Timestamp:" OWS
                             day-name "," SP datel SP time-of-day "." milliseconds SP "GMT" OWS
milliseconds                = 3DIGIT

;
; Header: 3gpp-Sbi-Max-Rsp-Time
;

Sbi-Max-Rsp-Time-Header = "3gpp-Sbi-Max-Rsp-Time:" OWS 1*5DIGIT OWS

;
; Header: 3gpp-Sbi-Correlation-Info
;

Sbi-Correlation-Info-Header = "3gpp-Sbi-Correlation-Info:" OWS
                              correlationinfo *( ";" OWS correlationinfo ) OWS
correlationinfo             = ctype "-" cvalue

```

```

ctype                = extension-token / "imsi" / "impi" / "suci" / "nai" / "gci" / "gli"
                    / "impu" / "msisdn" / "extid" / "imeisv" / "imei" / "mac" / "eui"

extension-token      = 1*( "!" / "#" / "$" / "%" / "&" / "'" / "*" / "+" / "." / "^" / "_"
                    / "~" / "|" / "~" / DIGIT / ALPHA )

cvalue               = 1*( tchar / "@" )

;
; Header: 3gpp-Sbi-Alternate-Chf-Id
;

Sbi-Alternate-Chf-Id-Header = "3gpp-Sbi-Alternate-Chf-Id:" OWS
                            "nfinst=" nfinst ";" OWS ( "primary" / "secondary" ) OWS

;
; Header: 3gpp-Sbi-Notif-Accepted-Encoding
;

Sbi-Notif-Accepted-Encoding-Header = "3gpp-Sbi-Notif-Accepted-Encoding:" OWS
                                    encoding-element *( OWS "," OWS encoding-element ) OWS

encoding-element     = codings [ weight ]

;
; Header: 3gpp-Sbi-Consumer-Info
;

Sbi-Consumer-Info-Header = "3gpp-Sbi-Consumer-Info:" OWS
                           consumer-info-element *( OWS "," OWS consumer-info-element ) OWS

consumer-info-element = ( supportedService ";" OWS supportedVersions
                        [ ";" OWS supportedFeatures ]
                        [ ";" OWS acceptEncoding ]
                        [ ";" OWS callback-uri-prefix ]
                        [ ";" OWS intraPlmnCallbackRoot ";" OWS interPlmnCallbackRoot ]
                        [ ";" OWS intermediateNfIndication ]
                        )

supportedService      = "service=" servicename

servicename           = 1*( "-" / %x30-39 / %x41-5A / "_" / %x61-7A )

supportedVersions     = "apiversion=" "(" OWS
                        [ apimajorversion *( RWS apimajorversion ) OWS ] ")"

apimajorversion       = %x31-39 [ *DIGIT ]

supportedFeatures     = "supportedfeatures=" features

features              = *HEXDIG

acceptEncoding        = "acceptencoding=" %x22 encodingList %x22

encodingList          = [ encoding-element *( OWS "," OWS encoding-element ) ]

intraPlmnCallbackRoot = "intraPlmnCallbackRoot="
                        DQUOTE sbi-scheme "://" sbi-authority [ prefix ] DQUOTE

interPlmnCallbackRoot = "interPlmnCallbackRoot="
                        DQUOTE sbi-scheme "://" sbi-authority [ prefix ] DQUOTE

callback-uri-prefix   = "callback-uri-prefix=" DQUOTE prefix DQUOTE

intermediateNfIndication = "intermediate-nf=" intermediateNfValue

intermediateNfValue   = "true"

;

```

```

; Header: 3gpp-Sbi-Response-Info
;

Sbi-Response-Info-Header = "3gpp-Sbi-Response-Info:" OWS
    resp-info-param *( OWS ";" OWS resp-info-param ) OWS

resp-info-param          = resp-info-param-name "=" OWS resp-info-param-value

resp-info-param-name     = "request-retransmitted"
    / "nfinst"
    / "nfset"
    / "nfservinst"
    / "nfserviceset"
    / "context-transferred"
    / "no-retry"
    / token

resp-info-param-value    = token

;
; Header: 3gpp-Sbi-Selection-Info
;

Sbi-Selection-Info-Header = "3gpp-Sbi-Selection-Info:" OWS
    selection-info-element *( OWS "," OWS selection-info-element ) OWS

selection-info-element   = ( "reselection=" reselectionvalue *( ";" OWS selection-criteria ) )
    / ( selection-criteria *( ";" OWS selection-criteria ) )

reselectionvalue         = "true" / "false"

selection-criteria       = selection-action "=" token

selection-action         = "not-select-nfservinst"
    / "not-select-nfserviceset"
    / "not-select-nfinst"
    / "not-select-nfset"

;
; Header: 3gpp-Sbi-Interplmn-Purpose
;

Sbi-Interplmn-Purpose-Header = "3gpp-Sbi-Interplmn-Purpose:" OWS N32Purpose ":" OWS
    additional-info OWS

N32Purpose                = "ROAMING"
    / "INTER_PLMN_MOBILITY"
    / "SMS_INTERCONNECT"
    / "ROAMING_TEST"
    / "INTER_PLMN_MOBILITY_TEST"
    / "SMS_INTERCONNECT_TEST"
    / "SNPN_INTERCONNECT"
    / "SNPN_INTERCONNECT_TEST"
    / "DISASTER_ROAMING"
    / "DISASTER_ROAMING_TEST"
    / "DATA_ANALYTICS_EXCHANGE"
    / "DATA_ANALYTICS_EXCHANGE_TEST"
    / "INDIRECT_NETWORK_SHARING"
    / token

additional-info           = token

;
; Header: 3gpp-Sbi-Request-Info
;

Sbi-Request-Info-Header = "3gpp-Sbi-Request-Info:" OWS req-param *( ";" OWS req-param ) OWS

req-param                = req-param-name "=" OWS req-param-value

req-param-name           = "retrans"
    / "redirect"

```

```

        / "reason"
        / "idempotency-key"
        / "receivedrejectioncause"
        / "callback-uri-prefix"
        / "nfirst"
        / "nfservinst"
        / "redirection-cause"
        / token

req-param-value      = token

;
; Header: 3gpp-Sbi-Retry-Info
;

Sbi-Retry-Info-Header = "3gpp-Sbi-Retry-Info:" OWS retriesindication OWS

retriesindication    = "no-retries"

;
; Header: 3gpp-Sbi-Binding-Indication-Notify-Restricted
;

Sbi-Binding-Indication-Notify-Restricted-Header = "3gpp-Sbi-Binding-Indication-Notify-Restricted:"
                                                    OWS restrict-value OWS
                                                    [ ";" OWS callback-root ]
                                                    [ ";" OWS callback-uri-prefix ] OWS

restrict-value       = "true"

callback-root        = "callback-root=" DQUOTE sbi-scheme "://"
                        sbi-authority [ prefix ] DQUOTE

```

Annex E (informative): Change history

Change history							
Date	Meeting	TDoc	CR	R ev	Cat	Subject/Comment	New version
2017-10	CT4#80	C4-175246				TR skeleton	0.1.0
2017-10	CT4#80	C4-175390				Implementation of pCRs agreed at CT4#80.	0.2.0
2017-12	CT4#81	C4-176433				Implementation of pCRs agreed at CT4#81.	0.3.0
2018-01	CT4#82	C4-181387				Implementation of pCRs agreed at CT4#82.	0.4.0
2018-03	CT4#83	C4-182430				Implementation of pCRs agreed at CT4#83.	0.5.0
2018-03	CT#79	CP-180028				Presented for information	1.0.0
2018-04	CT4#84	C4-183512				Implementation of pCRs agreed at CT4#84.	1.1.0
2018-05	CT4#85	C4-184617				Implementation of pCRs agreed at CT4#85. The following pCRs are implemented. C4-184589, C4-184580, C4-184347, C4-184590, C4-184338, C4-184591, C4-184349, C4-184490, C4-184350, C4-184579, C4-184577 and C4-184498.	1.2.0
2018-06	CT#80	CP-181098				Presented for approval	2.0.0
2018-06	CT#80					Approved in CT#80	15.0.0
2018-09	CT#81	CP-182053	0001	4	F	OAuth2.0 Clarifications	15.1.0
2018-09	CT#81	CP-182053	0002	2	B	Specifying N32 Aspects	15.1.0
2018-09	CT#81	CP-182053	0003	1	F	Determination of SBI message priorities	15.1.0
2018-09	CT#81	CP-182053	0004	5	F	Stateless AMF support	15.1.0
2018-09	CT#81	CP-182053	0005		F	Support of status code "501 Not implemented"	15.1.0
2018-09	CT#81	CP-182053	0006	2	B	Default port number	15.1.0
2018-12	CT#82	CP-183011	0009	3	F	Keep-alive on idle HTTP connections	15.2.0

2018-12	CT#82	CP-183011	0010	1	F	Stream Concurrency for overload control	15.2.0
2018-12	CT#82	CP-183011	0011	1	F	Update of missing status code 429	15.2.0
2018-12	CT#82	CP-183011	0012	1	F	Correction of the entity upon which content encoding is performed	15.2.0
2018-12	CT#82	CP-183011	0013	2	F	Custom header for notifications	15.2.0
2018-12	CT#82	CP-183011	0014	3	F	Routing across PLMN	15.2.0
2018-12	CT#82	CP-183011	0015		F	HTTP status code "406 Not Acceptable"	15.2.0
2018-12	CT#82	CP-183011	0016	1	F	Support of HTTP status code "414 URI Too Long"	15.2.0
2018-12	CT#82	CP-183011	0017	1	F	Error indicating "Unspecified resource URI structure"	15.2.0
2018-12	CT#82	CP-183011	0018		F	HTTP status code "414 URI Too Long" on PUT method	15.2.0
2018-12	CT#82	CP-183011	0020	1	F	Correction of Stream Priority in HTTP/2 Server Behaviour	15.2.0
2018-12	CT#82	CP-183194	0022	2	F	Change 403 to mandatory and clarify conditional headers	15.2.0
2018-12						Change history annex number corrected	15.2.1
2019-03	CT#83	CP-190016	0023	1	F	Extensibility mechanism for Query parameters	15.3.0
2019-03	CT#83	CP-190016	0024	1	F	Bearer Tokens	15.3.0
2019-03	CT#83	CP-190016	0025	1	F	Handling of Incorrect IEs	15.3.0
2019-03	CT#83	CP-190016	0026	2	F	Clarification on Handling of Incorrect Optional IEs	15.3.0
2019-03	CT#83	CP-190016	0027		F	Status Codes	15.3.0
2019-06	CT#84	CP-191027	0030	1	F	Content-encodings supported in HTTP requests	15.4.0
2019-06	CT#84	CP-191027	0031	3	F	Missing Application Error Codes	15.4.0
2019-06	CT#84	CP-191027	0032	2	F	Correction on Feature Negotiation	15.4.0
2019-06	CT#84	CP-191027	0037	1	F	Allowed values of 3gpp-Sbi-Callback header field	15.4.0
2019-06	CT#84	CP-191027	0038	1	F	Adding the Control Plane interfaces that support service based interface	15.4.0
2019-06	CT#84	CP-191055	0033	1	B	Support of Indirect Communication (General)	16.0.0
2019-06	CT#84	CP-191055	0034	2	B	Support of Indirect Communication (Routing to SCP)	16.0.0
2019-06	CT#84	CP-191055	0035	1	B	Support of Indirect Communication (NF discovery and selection)	16.0.0
2019-06	CT#84	CP-191057	0036	2	B	Partially implemented PATCH	16.0.0
2019-09	CT#85	CP-192194	0040	2	B	Support of stateless NFs	16.1.0
2019-09	CT#85	CP-192194	0041	1	B	Routing mechanisms for Indirect Communication	16.1.0
2019-09	CT#85	CP-192194	0042	1	B	Routing extensions for Indirect Communication	16.1.0
2019-09	CT#85	CP-192194	0043	-	B	Authority and/or deployment-specific string in apiRoot of resource URI for Indirect Communication	16.1.0
2019-09	CT#85	CP-192194	0044	1	B	NF / NF service instance selection for Indirect Communication without Delegated Discovery	16.1.0
2019-09	CT#85	CP-192194	0045	-	B	Feature negotiation for Indirect Communication with Delegated Discovery	16.1.0
2019-09	CT#85	CP-192194	0053	2	B	Routing for indirect Communication with HTTP between NFs and SCP	16.1.0
2019-09	CT#85	CP-192123	0046	-	B	Timestamp in HTTP messages	16.1.0
2019-09	CT#85	CP-19212	0047	1	B	Handling of timed out requests	16.1.0
2019-09	CT#85	CP-19212	0049	1	B	Indicating partially implemented PATCH	16.1.0
2019-09	CT#85	CP-19212	0052	2	F	Adding the Control Plane interfaces that support service based interface	16.1.0
2019-12	CT#86	CP-193036	0059	1	F	Load Info used for Load Control	16.2.0
2019-12	CT#86	CP-193036	0062	-	F	Informative description of internal NF routing of HTTP messages	16.2.0
2019-12	CT#86	CP-193057	0039	5	B	Binding between NF Service Consumer and NF Service Producer	16.2.0
2019-12	CT#86	CP-193057	0056	2	B	Routing of Indirect Communication with TLS between NFs and SCP	16.2.0
2019-12	CT#86	CP-193057	0057	2	B	Routing of Indirect Communication without TLS between NFs and SCP	16.2.0
2019-12	CT#86	CP-193057	0060	3	B	Conveyance of Delegated Discovery Parameters in HTTP/2 Headers	16.2.0
2019-12	CT#86	CP-193057	0064	1	B	Binding indication for subscribe/notify	16.2.0
2019-12	CT#86	CP-193057	0065	-	B	General Introduction for Delegated Discovery	16.2.0
2019-12	CT#86	CP-193057	0067	1	B	Handling of relative URIs with indirect communication	16.2.0

2019-12	CT#86	CP-193057	0068	2	B	Use of 3gpp-Sbi-Target-apiRoot header in HTTP requests from NFs to SEPP	16.2.0
2019-12	CT#86	CP-193057	0069	2	B	Returning NF Producer ID to NF Consumer when using Delegated Discovery	16.2.0
2019-12	CT#86	CP-193057	0066	1	B	Handling of default notification subscriptions with Delegated Discovery	16.2.0
2019-12	CT#86	CP-193063	0063	1	F	Clarification of Cause MANDATORY_IE_INCORRECT	16.2.0
2020-01						6.10.7 was removed (same as 6.10.2A0)	16.2.1
2020-03	CT#87e	CP-200025	0074	6	B	Description of the 3GPP Rel-16 OLC	16.3.0
2020-03	CT#87e	CP-200025	0081	6	B	Dynamic Load Control	16.3.0
2020-03	CT#87e	CP-200016	0085	2	F	Security requirements for Indirect Communication	16.3.0
2020-03	CT#87e	CP-200016	0086	3	F	Corrections to routing mechanism with TLS between NF and SCP	16.3.0
2020-03	CT#87e	CP-200016	0087	4	F	Binding procedures	16.3.0
2020-03	CT#87e	CP-200016	0088	3	F	Notifications sent with indirect communication	16.3.0
2020-03	CT#87e	CP-200016	0089	4	F	Handling of Discovery headers not supported by the SCP	16.3.0
2020-03	CT#87e	CP-200016	0091	2	F	Clarification to the SBI priority range	16.3.0
2020-03	CT#87e	CP-200016	0095	2	F	Indirect Communication Configuration Fixes With or Without TLS	16.3.0
2020-03	CT#87e	CP-200016	0096	1	B	Stateless Network Functions	16.3.0
2020-03	CT#87e	CP-200016	0097	2	F	NF set / NF service set usage in Indirect Communication models	16.3.0
2020-03	CT#87e	CP-200016	0100	-	F	Complement to 3gpp-Sbi-Callback Types in Annex B	16.3.0
2020-03	CT#87e	CP-200020	0090	2	B	Failover cause	16.3.0
2020-03	CT#87e	CP-200020	0098	1	B	Usage of compression for HTTP responses	16.3.0
2020-03	CT#87e	CP-200039	0092	2	D	Editorial fixes	16.3.0
2020-06	CT#88e	CP-201030	0104	2	F	Essential definitions for the binding concept	16.4.0
2020-06	CT#88e	CP-201030	0106	1	F	Correction of references	16.4.0
2020-06	CT#88e	CP-201030	0107	-	F	ABNF definition of 3gpp-Sbi-Target-apiRoot header	16.4.0
2020-06	CT#88e	CP-201030	0108	1	F	Error handling for indirect communications	16.4.0
2020-06	CT#88e	CP-201030	0113	1	B	Populating Recovery Information in the Binding Indication	16.4.0
2020-06	CT#88e	CP-201030	0114	1	B	Binding Indication sent from a Service Consumer	16.4.0
2020-06	CT#88e	CP-201030	0118	-	F	Binding indications / headers	16.4.0
2020-06	CT#88e	CP-201030	0119	1	F	HTTP redirection for indirect communication	16.4.0
2020-06	CT#88e	CP-201030	0121	3	F	Clarifications for scenarios with more than one SCP	16.4.0
2020-06	CT#88e	CP-201030	0124	2	F	Service instance binding level	16.4.0
2020-06	CT#88e	CP-201030	0127	2	F	Binding header clarification	16.4.0
2020-06	CT#88e	CP-201030	0128	-	F	Correcting parameter names in the binding headers	16.4.0
2020-06	CT#88e	CP-201030	0134	-	F	Clarifications of Binding concepts	16.4.0
2020-06	CT#88e	CP-201030	0135	1	B	Client credentials assertion and authentication	16.4.0
2020-06	CT#88e	CP-201030	0136	-	F	URI used in communications after an NF Service Producer change	16.4.0
2020-06	CT#88e	CP-201030	0137	-	F	Discovery parameters	16.4.0
2020-06	CT#88e	CP-201030	0138	-	F	Support of Nchf_ConvergedCharging_Notify in 3gpp-Sbi-Callback	16.4.0
2020-06	CT#88e	CP-201030	0139	1	F	Complete the description of custom headers	16.4.0
2020-06	CT#88e	CP-201039	0109	1	B	Scope of OCI signalled by an NF service consumer	16.4.0
2020-06	CT#88e	CP-201039	0110	1	B	Load and Overload Control for Indirect Communications	16.4.0
2020-06	CT#88e	CP-201039	0111	1	B	S-NSSAI/DNN based Load/Overload Control	16.4.0
2020-06	CT#88e	CP-201039	0112	1		Handling of multiple LCI/OCIs with different scopes	16.4.0
2020-06	CT#88e	CP-201039	0116	1	F	The Overload Control clarification when OCI sent from a Service Consumer	16.4.0
2020-06	CT#88e	CP-201071	0122	2	F	Failure detection on idle HTTP connections by NFs acting as HTTP/2 server	16.4.0
2020-06	CT#88e	CP-201071	0142	1	F	Missing abbreviations	16.4.0
2020-06	CT#88e	CP-201034	0120	1	B	Resource-Level Authorization	16.4.0
2020-06	CT#88e	CP-201034	0123	2	F	Delimiters - ABNF specific	16.4.0
2020-06	CT#88e	CP-201034	0140	1	F	HTTP Connection management and HTTP Retry	16.4.0

2020-06	CT#88e	CP-201034	0141	1	B	Definition of error "RESOURCE_CONTEXT_NOT_FOUND"	16.4.0
2020-06	CT#88e	CP-201047	0132	-	B	NSSAA Callback	16.4.0
2020-09	CT#89e	CP-202119	0148	1	F	Custom headers related to indirect communication	16.5.0
2020-09	CT#89e	CP-202119	0150	1	F	TLS security with the 3gpp-Sbi-Target-apiRoot header on N32f	16.5.0
2020-09	CT#89e	CP-202119	0151	1	F	Determining the NF Service Producer Identity without support of binding procedures	16.5.0
2020-09	CT#89e	CP-202119	0152	1	F	Clarifications for Indirect Communications	16.5.0
2020-09	CT#89e	CP-202119	0153	-	F	Incorrect references and editorial errors	16.5.0
2020-09	CT#89e	CP-202119	0154	-	F	Sending the NRF API URIs to the SCP for indirect communication	16.5.0
2020-09	CT#89e	CP-202119	0157	-	F	Discovery Headers	16.5.0
2020-09	CT#89e	CP-202119	0158	1	F	Reselection with indirect communication	16.5.0
2020-09	CT#89e	CP-202119	0159	2	F	API Root Change Handling	16.5.0
2020-09	CT#89e	CP-202119	0160	2	F	Notification Binding for Default Subscription	16.5.0
2020-09	CT#89e	CP-202119	0161	1	F	Notification for Default Subscriptions	16.5.0
2020-09	CT#89e	CP-202117	0156	-	F	3gpp-Sbi-Oci and 3gpp-Sbi-Lci headers	16.5.0
2020-09	CT#89e	CP-202110	0163	2	F	Corrections on expressions based on ABNF	16.5.0
2020-09	CT#89e	CP-202121	0144	1	D	Editorial fixes	17.0.0
2020-09	CT#89e	CP-202121	0145	-	D	Fixing proofing language	17.0.0
2020-12	CT#90e	CP-203054	0171	-	A	Service access authorization for indirect communication with delegated discovery	17.1.0
2020-12	CT#90e	CP-203054	0173	1	A	NF instance selection in SCP for Indirect Communication mode C	17.1.0
2020-12	CT#90e	CP-203054	0175	2	A	Essential Clarification on the Default Notification with indirect communication	17.1.0
2020-12	CT#90e	CP-203054	0177	1	A	Reporting NF is not reachable	17.1.0
2020-12	CT#90e	CP-203054	0181	1	A	Update a binding indication	17.1.0
2020-12	CT#90e	CP-203060	0168	2	F	Clarifications to stateless NF definition	17.1.0
2020-12	CT#90e	CP-203054	0183	1	A	Binding indication with the scope set to other service	17.1.0
2020-12	CT#90e	CP-203054	0185	1	A	NF instance reselection and use of location	17.1.0
2020-12	CT#90e	CP-203054	0191	1	A	Handling of binding on behalf of another NF	17.1.0
2020-12	CT#90e	CP-203054	0193	1	A	Binding Indication for Backup AMF	17.1.0
2020-12	CT#90e	CP-203054	0197	-	A	Target NF (service) instance ID in HTTP 307/308 response	17.1.0
2020-12	CT#90e	CP-203054	0198	1	A	Clarifications to Stateless NF scenarios	17.1.0
2020-12	CT#90e	CP-203048	0195	2	B	Max SCP Hops	17.1.0
2020-12	CT#90e	CP-203061	0167	1	F	Incorrect reference	17.1.0
2020-12	CT#90e	CP-203061	0179	1	F	Direct retrying with another SCP	17.1.0
2020-12	CT#90e	CP-203064	0169	-	B	SBI correlation info header	17.1.0
2020-12	CT#90e	CP-203064	0186	1	C	Too Many Requests	17.1.0
2020-12	CT#90e	CP-203064	0187	-	F	Resolution of FQDN in Callback URIs	17.1.0
2020-12	CT#90e	CP-203064	0189	4	B	Asserted PLMN-ID Header	17.1.0
2021-03	CT#91e	CP-210037	0202	-	A	Authorization of NF service access for Indirect Communication with Delegated Discovery	17.2.0
2021-03	CT#91e	CP-210037	0204	-	A	Authorization of NF service access for Indirect Communication without Delegated Discovery	17.2.0
2021-03	CT#91e	CP-210037	0206	-	A	Error handling when SCP receives "401 Unauthorized" or "403 Forbidden" from NFp	17.2.0
2021-03	CT#91e	CP-210037	0208	1	A	Error handling when the SCP fails to obtain an access token	17.2.0
2021-03	CT#91e	CP-210037	0210	1	A	Rejection response upon Client credentials assertion verification failure	17.2.0
2021-03	CT#91e	CP-210037	0212	1	A	Scope parameter in notification and callback responses	17.2.0
2021-03	CT#91e	CP-210037	0214	1	A	User-Agent header	17.2.0
2021-03	CT#91e	CP-210037	0218	1	A	Essential Correction for interpretation of NF change	17.2.0
2021-03	CT#91e	CP-210037	0228	1	A	Signaling Primary or Secondary CHF instance ID in service response with Model D	17.2.0
2021-03	CT#91e	CP-210037	0233	1	A	HTTP status code used by SCP/SEPP	17.2.0
2021-03	CT#91e	CP-210043	0222	-	A	Essential Correction to Causes NF_FAILOVER and NF_SERVICE_FAILOVER	17.2.0

2021-03	CT#91e	CP-210043	0223	1	A	ABNF Definition of 3GPP Custom Headers	17.2.0
2021-03	CT#91e	CP-210058	0225	-	A	Correction to API Prefix	17.2.0
2021-03	CT#91e	CP-210021	0215	2	B	Update Binding Indication for multiple resource contexts	17.2.0
2021-03	CT#91e	CP-210021	0226	-	F	Corrections to the description of the 3gpp-Sbi-discovery custom headers	17.2.0
2021-03	CT#91e	CP-210021	0235	1	F	Usage of Via header	17.2.0
2021-03	CT#91e	CP-210021	0236	1	F	Overload Control based on HTTP status codes	17.2.0
2021-03	CT#91e	CP-210021	0237	-	D	Editorial Corrections	17.2.0
2021-03	CT#91e	CP-210034	0229	-	F	Text style correction	17.2.0
2021-06	CT#92e	CP-211059	0242	-	A	Correction to binding procedures	17.3.0
2021-06	CT#92e	CP-211059	0257	-	A	SCP Redirect	17.3.0
2021-06	CT#92e	CP-211059	0263	1	A	Requirements supported for Indirect Communication without Delegated Discovery	17.3.0
2021-06	CT#92e	CP-211059	0264	1	A	Sending requests taking the binding level into account	17.3.0
2021-06	CT#92e	CP-211065	0255	-	A	Match check between Token and CCA	17.3.0
2021-06	CT#92e	CP-211028	0238	1	F	New header for requests sent to an (alternate) NF and overload control	17.3.0
2021-06	CT#92e	CP-211028	0239	2	B	Enhancement of the supported communication options discovery	17.3.0
2021-06	CT#92e	CP-211028	0245	1	F	3GPP SBI Callback in Notification	17.3.0
2021-06	CT#92e	CP-211028	0247	1	F	Clarification on NF Service Advertisement URI	17.3.0
2021-06	CT#92e	CP-211028	0248	1	F	Detection and handling of message looping	17.3.0
2021-06	CT#92e	CP-211028	0249	1	B	Load control and overload control for SEPP	17.3.0
2021-06	CT#92e	CP-211028	0253	1		Passing the NF Group ID to the NF service consumer for Indirect communication	17.3.0
2021-06	CT#92e	CP-211028	0258	-	F	Incorrect CR implementation	17.3.0
2021-06	CT#92e	CP-211028	0260	-	F	Implementation errors	17.3.0
2021-06	CT#92e	CP-211028	0261	-	F	Service specific support of 409 status code for HTTP GET method	17.3.0
2021-06	CT#92e	CP-211028	0266	2	F	Consumer Information for Direct Notification Report	17.3.0
2021-06	CT#92e	CP-211039	0267	-	B	N5g-ddnmf Discovery Notifications	17.3.0
2021-06	CT#92e	CP-211046	0252	1	F	Error handling for encBlockIndex	17.3.0
2021-06	CT#92e	CP-211059	0250	-	F	OpenAPI Reference	17.3.0
2021-09	CT#93e	CP-212060	0272	1	A	Discovery headers in service request with binding information	17.4.0
2021-09	CT#93e	CP-212026	0268	-	F	Correct the examples of 3gpp-Sbi-Routing-Binding and 3gpp-Sbi-Binding	17.4.0
2021-09	CT#93e	CP-212026	0269	1	F	SCP response in NRF error scenarios	17.4.0
2021-09	CT#93e	CP-212026	0270	-	F	Via header encoding	17.4.0
2021-09	CT#93e	CP-212026	0274	-	B	Feature Negotiation for Default Subscription	17.4.0
2021-09	CT#93e	CP-212026	0275	1	F	NRF Uri for Callbacks	17.4.0
2021-09	CT#93e	CP-212055	0273	-	F	Correction on 3gpp-Sbi-Callback	17.4.0
2021-12	CT#94e	CP-213088	0293	-	A	Handling of 3gpp-Sbi-Sender-Timestamp and 3gpp-Sbi-Max-Rsp-Time in SEPP	17.5.0
2021-12	CT#94e	CP-213088	0295	1	A	Correction to HTTP status code 307 and 308	17.5.0
2021-12	CT#94e	CP-213086	0240	4	B	Indicating in error responses whether a request was retransmitted by SCP	17.5.0
2021-12	CT#94e	CP-213086	0283	1	B	3GPP SBI Response Information	17.5.0
2021-12	CT#94e	CP-213086	0284	1	B	NF Peer Informaiton	17.5.0
2021-12	CT#94e	CP-213086	0285	1	F	Multiple UE identities in 3gpp-Sbi-Correlation-Info HTTP header	17.5.0
2021-12	CT#94e	CP-213086	0286	1	F	Invalid characters in headers using tchar common component in ABNF syntax	17.5.0
2021-12	CT#94e	CP-213086	0288	-	F	NF Producer information sent to NF Consumer when using Delegated Discovery	17.5.0
2021-12	CT#94e	CP-213085	0276	1	F	Mandating SCPs to support NF reselection for Model C/D delegated discovery	17.5.0
2021-12	CT#94e	CP-213085	0277	1	B	Indicating possible use of OAuth2 authorization in 3gpp-Sbi-Nrf-Uri header	17.5.0

2021-12	CT#94e	CP-213085	0280	1	F	3gpp-Sbi-Correlation-Info in Indirect Communication with or without Delegated Discovery	17.5.0
2021-12	CT#94e	CP-213085	0281	1	F	Correction on 3gpp-Sbi-Callback	17.5.0
2021-12	CT#94e	CP-213105	0278	1	B	SBI message priority for MPS	17.5.0
2021-12	CT#94e	CP-213093	0287	1	F	Notifying different NF Service consumer instance sharing same authority	17.5.0
2021-12	CT#94e	CP-213087	0289	1	B	New application errors to indicate Congestion per service	17.5.0
2021-12	CT#94e	CP-213087	0290	1	F	NF_FAILOVER and NF_SERVICE_FAILOVER from NF service consumer	17.5.0
2021-12	CT#94e	CP-213087	0297	2	F	Event subscriptions with Inter PLMN and Intra PLMN callback URIs	17.5.0
2022-03	CT#95e	CP-220069	0300	2	A	3gpp-Sbi-Discovery-service-names header	17.6.0
2022-03	CT#95e	CP-220069	0309	-	A	ABNF correction for binding header	17.6.0
2022-03	CT#95e	CP-220069	0315	-	A	ABNF definition of the 3gpp-Sbi-Target-Nf-Id header	17.6.0
2022-03	CT#95e	CP-220069	0322	-	A	Correction on Pseudo-header Setting	17.6.0
2022-03	CT#95e	CP-220023	0298	1	F	Static Authorization	17.6.0
2022-03	CT#95e	CP-220023	0302	1	B	3gpp-Sbi-NF-Peer-Info header field	17.6.0
2022-03	CT#95e	CP-220023	0304	-	F	HTTP Client Behaviour	17.6.0
2022-03	CT#95e	CP-220023	0305	1	B	Group Binding for NF (Service) Levels	17.6.0
2022-03	CT#95e	CP-220023	0306	1	B	Other-service Scope support to PDU Session resource for NIDD	17.6.0
2022-03	CT#95e	CP-220023	0307	-	F	Producer ID for Indirect Communication without Delegated Discovery	17.6.0
2022-03	CT#95e	CP-220024	0303	1	B	Avoidance of (re)selection of the target NF for Indirect Communication	17.6.0
2022-03	CT#95e	CP-220024	0316	-	F	NF Type in User-Agent	17.6.0
2022-03	CT#95e	CP-220024	0323	-	F	Corrections to the examples on API URI	17.6.0
2022-03	CT#95e	CP-220024	0325	1	B	Retry with an Idempotency key for non-idempotent methods	17.6.0
2022-03	CT#95e	CP-220025	0312	1	F	S-NSSAI encoding in the 3gpp-Sbi-Lci and 3gpp-Sbi-Oci headers' examples	17.6.0
2022-03	CT#95e	CP-220025	0317	1	B	URL Encoding	17.6.0
2022-03	CT#95e	CP-220025	0326	3	B	Usage indication over N32-f	17.6.0
2022-03	CT#95e	CP-220090	0301	-	F	Editorial corrections	17.6.0
2022-03	CT#95e	CP-220092	0299	1	F	OpenAPI definition of 3gpp-Sbi-Discovery headers	17.6.0
2022-03	CT#95e	CP-220092	0310	-	F	ABNF correction for max forwarding header	17.6.0
2022-03	CT#95e	CP-220092	0311	-	F	SEPP Error Correction	17.6.0
2022-06	CT#96	CP-221027	0327	-	F	Corrections to Parameter Name in 3gpp-Sbi-Routing-Binding	17.7.0
2022-06	CT#96	CP-221027	0329	-	F	3gpp-Sbi-Request-Info	17.7.0
2022-06	CT#96	CP-221028	0335	-	F	Access Token reuse in indirect communication with delegated discovery	17.7.0
2022-06	CT#96	CP-221028	0330	2	F	New values for the 3gpp-Sbi-Callback HTTP custom header	17.7.0
2022-06	CT#96	CP-221045	0328	-	F	Style correction	17.7.0
2022-06	CT#96	CP-221045	0333	2	F	Removing misplaced string	17.7.0
2022-06	CT#96	CP-221045	0334	-	F	Target NF type in notification request for NF service consumer reselection by SCP	17.7.0
2022-09	CT#97e	CP-222026	0337	2	F	URL Encoding of data	17.8.0
2022-09	CT#97e	CP-222026	0344	1	F	Binding Indication for the resource bound to a specific NF service instance	17.8.0
2022-09	CT#97e	CP-222026	0345	1	F	Originating network ID	17.8.0
2022-09	CT#97e	CP-222026	0352	1	F	Removal of Editor's NOTE	17.8.0
2022-09	CT#97e	CP-222026	0353	1	F	Usage of 3gpp-Sbi-NF-Peer-Info HTTP header	17.8.0
2022-09	CT#97e	CP-222045	0340	1	F	Load and Overload Control Information of an NF service instance	17.8.0
2022-09	CT#97e	CP-222045	0346	-	F	Correct the order of parameters in 3gpp-Sbi-Producer-Id	17.8.0
2022-09	CT#97e	CP-222046	0341	1	A	Redirection to an SCP or SEPP	17.8.0
2022-09	CT#97e	CP-222053	0349	1	F	NF Producer Id for Resource Creation	17.8.0
2022-09	CT#97e	CP-222059	0351	1	A	API version in URI setting in indirect communication	17.8.0
2022-09	CT#97e	CP-222061	0348	1	A	Essential Correction on Headers in Indirect Communication	17.8.0
2022-12	CT#98e	CP-223041	0358	-	F	Incomplete implementation of CR 29.500 #0353	17.9.0

2022-12	CT#98e	CP-223041	0363	3	F	Identifying the entity inserting 3gpp-Sbi-Originating-Network-Id	17.9.0
2022-12	CT#98e	CP-223027	0355	1	F	Notification error handling	18.0.0
2022-12	CT#98e	CP-223027	0359	1	F	Extending the problem details with supported API versions	18.0.0
2022-12	CT#98e	CP-223027	0362	-	F	Standalone Notifications	18.0.0
2022-12	CT#98e	CP-223028	0357	1	F	Feature negotiation for subscriptions to notifications occurring via an indirect path	18.0.0
2022-12	CT#98e	CP-223029	0356	2	F	Restricting SCP to perform retries	18.0.0
2022-12	CT#98e	CP-223029	0360	2	F	Extending the 3gpp-Sbi-Nrf-Callback URI with the NRF NFManagement API URI	18.0.0
2022-12	CT#98e	CP-223029	0365	3	F	Clarification on error handling at SEPP	18.0.0
2023-03	CT#99	CP-230091	0372	-	A	NF Set ID encoding for an AMF set	18.1.0
2023-03	CT#99	CP-230029	0375	-	F	Binding created as part of a service request	18.1.0
2023-03	CT#99	CP-230051	0367	1	D	Correcting reference to RFC 1738	18.1.0
2023-03	CT#99	CP-230051	0368	1	D	Clarification to URL Encoding of URI components	18.1.0
2023-06	CT#100	CP-231086	0378	-	A	Redirection to an SCP or SEPP	18.2.0
2023-06	CT#100	CP-231029	0382	4	F	Correction of ABNF rules	18.2.0
2023-09	CT#101	CP-232072	0389	2	A	ABNF rule of 3gpp-Sbi-Oci header	18.3.0
2023-09	CT#101	CP-232061	0395	-	A	Correlation-Info Header	18.3.0
2023-09	CT#101	CP-232068	0393	1	A	Client Credentials Assertion of the source NF service consumer	18.3.0
2023-09	CT#101	CP-232073	0391	-	A	S-NSSAI encoding in 3gpp-Sbi-Oci and 3gpp-Sbi-Lci headers	18.3.0
2023-09	CT#101	CP-232033	0384	-	B	HTTP RFCs obsoleted by IETF RFC 9110, 9111 and 9113	18.3.0
2023-09	CT#101	CP-232033	0385	-	B	Binding indication with other-service scope from AMF to SMSF and PCF	18.3.0
2023-09	CT#101	CP-232033	0386	1	B	Access token with additional scopes for indirect com. with delegated discovery	18.3.0
2023-09	CT#101	CP-232033	0387	1	B	Prefix in Callback URIs	18.3.0
2023-09	CT#101	CP-232033	0398	-	F	Correction on NRF API URI	18.3.0
2023-09	CT#101	CP-232033	0400	2	F	504 Gateway Timeout response including the problemDetails	18.3.0
2023-12	CT#102	CP-233027	0403	-	B	ABNF definition and example of callback-uri-prefix in 3gpp-Sbi-Request-Info header	18.4.0
2023-12	CT#102	CP-233027	0404	-	F	HTTP requests redirected by SEPP	18.4.0
2023-12	CT#102	CP-233028	0402	1	F	Incorrect references to HTTP RFC 7230 obsoleted by IETF RFC 9110	18.4.0
2023-12	CT#102	CP-233028	0407	1	F	ABNF Corrections	18.4.0
2023-12	CT#102	CP-233030	0409	-	F	Correction on the general description for overload	18.4.0
2023-12	CT#102	CP-233031	0408	1	F	Reserved characters in JSON attributes defined as URI and in HTTP header parameters defined as URI or path-absolute	18.4.0
2023-12	CT#102	CP-233031	0412	-	F	Encoding for String Type JSON Attributes	18.4.0
2023-12	CT#102	CP-233056	0410	2	F	Essential Correction on Notification URIs of Reselected NF Consumers	18.4.0
2023-12	CT#102	CP-233056	0411	1	B	New protocol error for the rejection due to target PLMN not reachable	18.4.0
2023-12	CT#102	CP-233062	0406	1	A	3gpp-Sbi-Consumer-Info Header	18.4.0
2024-03	CT#103	CP-240066	0414	-	A	Authorization of NF Service Consumers for data access via DCCF	18.5.0
2024-03	CT#103	CP-240028	0415	1	F	Note referring to an incorrect header name	18.5.0
2024-03	CT#103	CP-240028	0416	-	F	SCP handling of 3xx redirect messages	18.5.0
2024-03	CT#103	CP-240028	0417	-	F	Extensions to the 3gpp-Sbi-Request-Info header	18.5.0
2024-03	CT#103	CP-240028	0418	-	F	3GPP SBI callback type values	18.5.0
2024-03	CT#103	CP-240028	0420	2	B	New application error for the HTTP status code 403	18.5.0
2024-03	CT#103	CP-240028	0421	3	F	Error description in 401 Unauthorized response	18.5.0
2024-03	CT#103	CP-240028	0424	1	F	Redundant clause removal	18.5.0
2024-03	CT#103	CP-240028	0425	-	F	NoProfileMatchInfo caching in SCPs	18.5.0
2024-03	CT#103	CP-240053	0422	-	F	ABNF correction for originating network id header	18.5.0
2024-06	CT#104	CP-241028	0426	2	F	Header case	18.6.0
2024-06	CT#104	CP-241028	0428	4	F	Indication of Intermediate NF	18.6.0

2024-06	CT#104	CP-241041	0433	2	B	Exchanging data or analytics between PLMNs	18.6.0
2024-06	CT#104	CP-241050	0434	-	F	Error code on 3gpp-Sbi-Originating-Network-Id mismatch	18.6.0
2024-06	CT#104	CP-241050	0435	-	F	Align the default N32 purpose	18.6.0
2024-06	CT#104	CP-241051	0431	1	F	Replacing IPX with Roaming Intermediary	18.6.0
2024-08						TS29500_CustomHeaders.abnf updated	18.6.1
2024-09	CT#105	CP-242039	0442		F	RWS replaced by OWS in HTTP custom headers	18.7.0
2024-09	CT#105	CP-242039	0443	1	F	HTTP redirection for multiple SEPPs per PLMN	18.7.0
2024-09	CT#105	CP-242046	0438	2	F	Delegated authorization to support AI/ML model sharing	18.7.0
2024-09	CT#105	CP-242053	0447	1	F	Incorrect example for evaluating the intended purpose	18.7.0
2024-09	CT#105	CP-242030	0437	1	F	Application Error for Reaching Maximum JSON Payload Size	19.0.0
2024-09	CT#105	CP-242030	0445	2	F	Feature clarification	19.0.0
2024-09	CT#105	CP-242033	0439	-	F	Correction of cause attribute value	19.0.0
2024-09	CT#105	CP-242033	0440	-	F	Correction of feature negotiation	19.0.0
2024-09	CT#105	CP-242033	0441	-	B	Indirect communication with or without delegated discovery between different PLMNs with possible NF selection at target PLMN	19.0.0
2024-12	CT#106	CP-243032	0455	1	F	Clarification on missing abbreviations	19.1.0
2024-12	CT#106	CP-243032	0450	2	F	Access Token Claim Mismatch	19.1.0
2024-12	CT#106	CP-243035	0448	1	F	Clarification on the Sender Timestamp and the Max-Rsp-Time	19.1.0
2024-12	CT#106	CP-243039	0453	1	B	Routing requests with Target-apiRoot and w/o Scp-apiRoot in target PLMN	19.1.0
2024-12	CT#106	CP-243039	0454	2	B	Routing requests via alternative SCPs in target PLMN	19.1.0
2025-03	CT#107	CP-250022	0462	-	A	Encoding of curly brackets in URI	19.2.0
2025-03	CT#107	CP-250032	0458	1	F	RESOURCE_URI_STRUCTURE_NOT_FOUND	19.2.0
2025-03	CT#107	CP-250036	0463	-	B	NF service consumer (re)selection by the target PLMN	19.2.0
2025-03	CT#107	CP-250046	0457	-	F	Change History	19.2.0
2025-06	CT#108	CP-251051	0469	2	A	New Peer Types	19.3.0
2025-06	CT#108	CP-251055	0467	1	F	Service request rejected by SCP due to missing parameter in Access Token Request	19.3.0
2025-06	CT#108	CP-251055	0471	1	F	Discovery Factors Handling for Indirect Communication	19.3.0
2025-06	CT#108	CP-251057	0466	2	B	Extend the N32Purpose for indirect network sharing	19.3.0
2025-06	CT#108	CP-251058	0470	-	B	Authorization of NF service access with NF selection at target PLMN	19.3.0
2025-06	CT#108	CP-251067	0472	1	F	Update and replace obsoleted HTTP RFC 4122	19.3.0
2025-09	CT#109	CP-252037	0474	1	F	Requester's PLMN IDs in service request to SCP	19.4.0
2025-09	CT#109	CP-252037	0475	-	F	Editor's note on NRF API URI usage for NFc reselection in inter-PLMN scenarios	19.4.0
2025-09	CT#109	CP-252037	0478	1	F	Binding Indication in Notification	19.4.0
2025-09	CT#109	CP-252037	0476	-	B	HTTP status code 422	19.4.0
2025-09	CT#109	CP-252041	0473	-	F	Fixing references for Authorization of NF service access	19.4.0
2025-09	CT#109	CP-252049	0477	-	F	Correction of misused reference to LCI and typo	19.4.0
2025-12	CT#110	CP-253158	0479	1	B	HTTP header parameter to extend registration timer	19.5.0
2025-12	CT#110	CP-253136	0482	-	A	Implementation error of CR#0236	19.5.0
2025-12	CT#110	CP-253143	0483	1	F	3gpp-Sbi-Callback type values	19.5.0
2025-12	CT#110	CP-253143	0484	1	F	Clarifications to Authorization for indirect communication with delegated discovery	19.5.0
2026-03	CT#111	CP-260026	0486	-	F	HTTP Headers	19.6.0
2026-03	CT#111	CP-260031	0487	4	F	Forwarding the Binding Indication corresponding to a Notification URI	19.6.0

History

Version	Date	Status
V19.4.0	January 2026	Publication
V19.5.0	February 2026	Publication
V19.6.0	March 2026	Publication