

# ETSI TS 129 155 V14.2.0 (2017-07)



**LTE;  
Traffic steering control;  
Representational State Transfer (REST) over  
St reference point  
(3GPP TS 29.155 version 14.2.0 Release 14)**



---

Reference

RTS/TSGC-0329155ve20

---

Keywords

LTE

**ETSI**

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

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

Siret N° 348 623 562 00017 - NAF 742 C  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° 7803/88

---

**Important notice**

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSI/DeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

---

**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 2017.

All rights reserved.

DECT™, PLUGTESTS™, UMTS™ and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.

3GPP™ and LTE™ are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

oneM2M logo is protected for the benefit of its Members.

GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

---

## Intellectual Property Rights

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

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

---

## Foreword

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, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under <http://webapp.etsi.org/key/queryform.asp>.

---

## 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
Foreword.....	2
Modal verbs terminology.....	2
Foreword.....	5
1 Scope .....	6
2 References .....	6
3 Definitions and abbreviations.....	7
3.1 Definitions .....	7
3.2 Abbreviations .....	7
4 St reference point .....	7
4.1 Overview .....	7
4.2 Reference model.....	7
4.3 Functional elements.....	8
4.3.1 TSSF.....	8
4.3.2 PCRF .....	8
4.4 Procedures over St reference point.....	8
4.4.1 General.....	8
4.4.2 Traffic Steering Policy Provisioning over St .....	8
4.4.3 Traffic Steering Rule Error Handling .....	9
4.4.4 UE IPv4 address provisioning .....	9
5 St protocol .....	10
5.1 Introduction .....	10
5.2 Transport layer .....	10
5.3 Application delivery layer .....	10
5.3.1 Methods .....	11
5.3.2 Resources and URI design.....	11
5.3.3 HTTP request/response formats.....	12
5.3.3.1 General .....	12
5.3.3.2 POST /stapplication/sessions .....	13
5.3.3.3 PUT /stapplication/sessions/{stsessionid} .....	15
5.3.3.4 PATCH /stapplication/sessions/{stsessionid} .....	16
5.3.3.5 DELETE /stapplication/sessions/{stsessionid} .....	17
5.3.3.6 GET /stapplication/sessions/{stsessionid} .....	18
5.3.3.7 POST /{notificationbaseurl}/{stsessionid} .....	19
5.3.4 St session ID .....	20
5.3.5 HTTP status codes .....	21
5.3.6 Feature negotiation .....	22
5.3.6.1 General .....	22
5.3.7 HTTP custom headers.....	22
5.3.7.1 3gpp-Optional-Features.....	22
5.3.7.2 3gpp-Required-Features .....	22
5.3.7.3 3gpp-Accepted-Features .....	23
5.3.7.4 3gpp-Notification-Base-URL.....	23
5.4 Specific application communication .....	23
5.4.1 General.....	23
5.4.2 Content type.....	23
5.4.3 JSON session fields .....	23
5.4.3.1 General.....	23
5.4.3.2 session-id .....	24
5.4.3.3 ue-ipv4 .....	24
5.4.3.4 ue-ipv6-prefix .....	24
5.4.3.5 tsrules.....	24
5.4.3.6 ts-rule-name .....	25

5.4.3.7	precedence .....	25
5.4.3.8	tdf-application-identifier .....	25
5.4.3.9	flow-information .....	25
5.4.3.10	flow-description .....	26
5.4.3.11	tos-traffic-class .....	26
5.4.3.12	security-parameter-index .....	26
5.4.3.13	flow-label .....	26
5.4.3.14	flow-direction .....	26
5.4.3.15	ts-policy-identifier-dl .....	26
5.4.3.16	ts-policy-identifier-ul .....	26
5.4.3.17	predefined-tsrules .....	26
5.4.3.18	predefined-group-of-tsrules .....	27
5.4.3.19	ts-rule-base-name .....	27
5.4.3.20	called-station-id .....	27
5.4.4	JSON errors and informational response fields .....	27
5.4.4.1	General .....	27
5.4.4.2	errors .....	28
5.4.4.3	error-type .....	28
5.4.4.4	error-message .....	28
5.4.4.5	error-tag .....	28
5.4.4.6	error-path .....	28
5.4.4.7	error-info .....	28
5.4.4.6	success-message .....	29
5.4.4.7	success-path .....	29
5.4.4.8	success-info .....	29
5.4.5	JSON report fields .....	29
5.4.5.1	General .....	29
5.4.5.2	ts-rule-reports .....	29
5.4.5.3	resource-paths .....	30
5.4.5.4	rule-status .....	30
5.4.5.5	rule-failure-code .....	30
5.4.6	JSON notification fields .....	31
5.4.6.1	General .....	31
5.4.6.2	notifications .....	31
5.4.6.3	notification-type .....	31
5.4.6.4	notification-message .....	32
5.4.6.5	notification-tag .....	32
5.4.6.6	notification-info .....	32
5.5	TSSF discovery .....	32
6	Secure communication .....	32
<b>Annex A (informative): Call Flows .....</b>		<b>33</b>
A.1	General .....	33
A.2	Traffic Steering Control Information Provisioning .....	33
A.3	Traffic Steering Control Information Update .....	33
A.4	Traffic Steering Control Information Removal .....	34
A.5	Traffic Steering Control Notification .....	35
<b>Annex B (normative): JSON Schema .....</b>		<b>36</b>
B.1	Session body schema .....	36
B.2	Error and Informational response schema .....	38
B.3	Report schema .....	40
B.4	Notification schema .....	41
<b>Annex C (informative): Change history .....</b>		<b>43</b>
History .....		44

---

# Foreword

This Technical Specification has been produced by the 3<sup>rd</sup> 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.

---

# 1 Scope

The present document describes the Representational State Transfer (REST) protocol-based St reference point, which is used to provision the traffic steering control information to the TSSF from the PCRF.

---

# 2 References

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

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

- [1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".
- [2] 3GPP TS 23.203: "Policy and charging control architecture".
- [3] IETF RFC 793: "Transmission Control Protocol".
- [4] IETF RFC 2616: "Hypertext Transfer Protocol – HTTP/1.1".
- [5] 3GPP TS 29.201: "Representational State Transfer (REST) reference point between Application Function (AF) and Protocol Converter (PC)".
- [6] IETF RFC 1786: "Uniform Resource Locators (URL)".
- [7] 3GPP TS 33.210: "3G security; Network Domain Security (NDS); IP network layer security".
- [8] IETF RFC 1983: "Internet Users' Glossary".
- [9] Void.
- [10] IETF RFC 7159: "The JavaScript Object Notation (JSON) Data Interchange Format".
- [11] Void.
- [12] IETF RFC 5789: "PATCH Method for HTTP".
- [13] IETF RFC 3986: "Uniform Resource Identifier (URI): Generic Syntax".
- [14] IETF RFC 6902: "JavaScript Object Notation (JSON) Patch".
- [15] IETF RFC 6901: "JavaScript Object Notation (JSON) Pointer".
- [16] IETF draft-newton-json-content-rules-08: "A Language for Rules Describing JSON Content".

**Editor's note:** The above document cannot be formally referenced until it is published as an RFC.

- [17] 3GPP TS 29.212: "Policy and Charging Control (PCC); Reference points".
- [18] 3GPP TS 29.213: "Policy and charging control signalling flows and Quality of Service (QoS) parameter mapping".
- [19] IETF RFC 2818: "HTTP Over TLS".
- [20] 3GPP TS 23.003: "Numbering, addressing and identification".
- [21] IETF RFC 6733: "Diameter Base Protocol".

---

## 3 Definitions and abbreviations

### 3.1 Definitions

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

(S)Gi-LAN: The network infrastructure connected to the 3GPP network over the SGi or Gi reference point that provides various IP-based services.

(S)Gi-LAN service function: A function located in the (S)Gi-LAN that provides value-added IP-based services e.g. NAT, anti-malware, parental control, DDoS protection.

JSON Content Rules: JSON Content Rules (JCR), as defined in IETF draft-newton-json-content-rules [16] is a language specifying the interchange of data in JSON as defined in RFC 7159 [10].

### 3.2 Abbreviations

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

JCR	JSON Content Rules
JSON	JavaScript Object Notation
PCRF	Policy and Charging Rules Function
TSSF	Traffic Steering Support Function

---

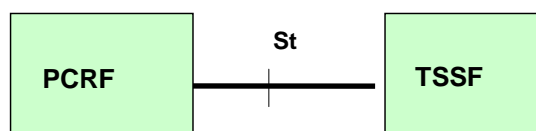
## 4 St reference point

### 4.1 Overview

The St reference point resides between the PCRF and TSSF. St reference point is used to provision the traffic steering control information from the PCRF to the TSSF.

### 4.2 Reference model

The St reference point resides between the PCRF and TSSF as depicted in figure 4.2.1. The overall PCC architecture is depicted in subclause 3a of 3GPP TS 29.213 [18].



**Figure 4.2.1: St reference model**



## 4.3 Functional elements

### 4.3.1 TSSF

The TSSF is a function that receives traffic steering control information from the PCRF and ensures that the related traffic steering policy is enforced in the (S)Gi-LAN.

A traffic steering policy is locally configured in TSSF and can be used for uplink, downlink or for both directions. To ensure that the traffic steering policy is enforced, the TSSF performs deployment specific actions as configured for that traffic steering policy.

### 4.3.2 PCRF

The PCRF functionality defined in 3GPP TS 23.203 [2] is applicable. In addition, the PCRF shall be able to make a decision of traffic steering policies used to control the steering of the subscriber's traffic to appropriate (S)Gi-LAN service functions.

NOTE: In order to allow the PCRF to select and provision an application based traffic steering policy, the reporting of detected applications to the PCRF or any other information can be used.

## 4.4 Procedures over St reference point

### 4.4.1 General

The procedures which can be operated at the St interface are described in the following subclauses.

### 4.4.2 Traffic Steering Policy Provisioning over St

When the PCRF determines the traffic steering control information needed for the IP-CAN session; the PCRF shall send an HTTP POST message to the TSSF to create a new session resource. The PCRF shall provide the representation of the session resource within the body of the HTTP POST. Within the body of the HTTP POST, the PCRF shall provide the St Session ID, the PDN information if required by the operator policies, the UE IPv4 address and/or UE IPv6 prefix and one or more sets of traffic steering control information to the TSSF.

The PCRF may modify or remove traffic steering control information at any time (e.g. due to subscription change or network status change) by sending either an HTTP PUT or HTTP PATCH message to the TSSF including the St session ID within the request URL. When using an HTTP PUT to modify the session resource, the PCRF shall provide the entire state of the session resource within the body of the HTTP PUT. In this case, the TSSF shall replace the existing session information associated with this resource with the information provided in the body of the HTTP PUT. When using an HTTP PATCH to modify the session resource, the PCRF shall provide the modifications within the body of the HTTP PATCH as defined in subclause 5.3.3.3. In this case, the TSSF shall update the existing session resource based on the content of the body of the HTTP PATCH.

In order to remove all of the traffic steering control information associated with an IP-CAN session when the IP-CAN session is terminated, the PCRF shall send to the TSSF an HTTP DELETE message including the St Session ID within the request URL.

Once the PCRF has requested the creation of a session resource, the PCRF may request the state of the session at any time by sending an HTTP GET request to the TSSF including the Session ID within the request URL. Upon receipt of an HTTP GET from the PCRF, the TSSF shall provide the session representation within the body of the response. Based on the received information, the PCRF may decide whether re-installation, modification, removal of the traffic steering control information or any other action applies.

When a combination of PCEF/TDF with traffic steering control feature and TSSF is deployed, the TSSF shall behave as specified in subclause 6.1.17 of 3GPP TS 23.203 [2]. In this case, the PCRF shall provide the traffic detection information as part of the service data flow information included in the tos-traffic-class within the flow-information field or within the tdf-application-identifier field. If traffic detection is performed using the flow-information field, the PCRF shall ensure that the relevant value of the tos-traffic-class field included within the flow-information field used for traffic detection is the same as the value provided in the traffic steering policy identifier(s) provided over Gx or Sd

reference point. If traffic detection is performed using the `tdf-application-identifier` field, the PCRF shall ensure that the identifier included within the `tdf-application-identifier` refers to the configured value(s) in the TSSF that correspond to the value provided in the traffic steering policy identifier(s) over Gx or Sd reference point. See 3GPP TS 29.212 [17].

NOTE 1: The PCRF and TSSF are configured with the traffic detection mechanism to be applied when the `tdf-application-identifier` field is used for traffic detection.

NOTE 2: The `tdf-application-identifier` can refer to the configured traffic detection information for uplink and/or downlink traffic.

### 4.4.3 Traffic Steering Rule Error Handling

If the installation/activation of one or more traffic steering rules fails, the TSSF shall inform the PCRF of the failed traffic steering rules in the corresponding JSON body of the HTTP response, by including a `ts-rule-reports` field, and by setting the `error-tag` to `TS_RULE_EVENT`. The `ts-rule-reports` field is an array of rule failure reports. Within each rule failure report, the TSSF shall identify the failed traffic steering rule(s) by including the `resource-paths` field containing a list of JSON pointer(s) to the relevant traffic steering rule resource(s), shall specify the failed reason code by including a `rule-failure-code` field and a `rule-status` as described below:

- If the installation/activation of one or more new traffic steering rules (i.e., rules which were not previously successfully installed) fails, the TSSF shall set the `rule-status` field to `INACTIVE`.
- If the modification of a currently active traffic steering rule fails, the TSSF shall retain the existing traffic steering rule as active without modification unless the reason for the failure also has an impact on the existing traffic steering rule.

Depending on the value of the `rule-failure-code` field, the PCRF may decide whether retaining of the old traffic steering rule, re-installation, modification, removal of the traffic steering rule or any other action applies.

If a traffic steering rule was successfully installed/activated, but can no longer be enforced by the TSSF, and if the notification supported feature is supported by the TSSF and PCRF, the TSSF shall notify the PCRF of the rule enforcement failure by sending an HTTP POST request to the notification URL provided by the PCRF as defined in subclause 5.3.3.2. The JSON body of this HTTP POST shall include a `notification-info` field containing a `ts-rule-reports` field and a `notification-tag` field set to a value of `TS_RULE_EVENT` filled. Each item of the `ts-rule-reports` shall include the corresponding `rule-failure-code` field and `rule-status` field to `INACTIVE`.

Note that the PCRF may poll the session state from the TSSF at any time via the HTTP GET method as described in subclause 4.4.2; and take further decisions based on the status of the traffic steering rules reported in the session state.

### 4.4.4 UE IPv4 address provisioning

When the PCRF is notified by the PCEF that either a `UE_IP_ADDRESS_ALLOCATE` or a `UE_IP_ADDRESS_RELEASE` event of the IP-CAN session occurred in the PCEF, the PCRF shall send an HTTP PATCH or PUT message to the TSSF including the St session ID within the request URL.

If the PCRF uses HTTP PUT to update the St session, the PCRF shall provide the entire state of the session resource including the new IPv4 in case a new IPv4 was allocated or omitting the IPv4 in case the IPv4 was released. If the PCRF uses the HTTP PATCH method to update the St session, and if the IPv4 was allocated, the PCRF shall use the “add” operation within the body of the HTTP PATCH to include the new IPv4 as specified in subclause 5.3.3.4; otherwise if the IPv4 was released, the PCRF shall use the “remove” operation within the body of the HTTP PATCH to remove the IPv4 as specified in subclause 5.3.3.4. If the PCRF provides the new UE IPv4 address to the TSSF, the TSSF shall additionally apply the traffic steering rules to the user plane traffic with the IP address matching the new UE IPv4 address. If the PCRF notifies to the TSSF that the UE IPv4 address has been released, the TSSF shall stop applying the traffic steering rule to the user plane traffic with the IP address matching the released UE IPv4 address.

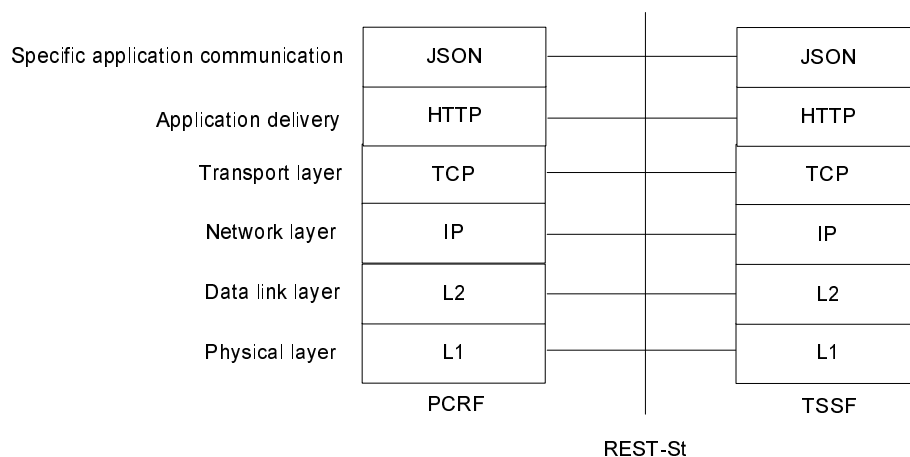
## 5 St protocol

### 5.1 Introduction

The following layers of the protocol stack for the St reference point between PCRF and TSSF are described in subclauses:

- TCP [3] provides the communication service at the transport layer.
- An optional communication security layer can be added between the transport and the application delivery layer (see subclause 6).
- The application delivery layer provides the transport of the specific application communication data using HTTP [4].
- The specific application communication layer constitutes the transport of the JSON content type.

Figure 5.1.1 illustrates the protocol stack of the RESTful St reference point.



**Figure 5.1.1: Protocol stack of the RESTful St reference point**

### 5.2 Transport layer

HTTP is layered over TCP, which provides a reliable transport.

For traffic steering policy provisioning and querying, the PCRF acts as an HTTP client and the TSSF acts as an HTTP server. As a result, the PCRF shall initiate a TCP connection with the TSSF.

If the notification feature is supported by TSSF and PCRF, then the TSSF and PCRF shall also respectively act as HTTP client and server. In this case, the TSSF shall initiate a TCP connection with the PCRF.

### 5.3 Application delivery layer

The application delivery layer shall use RESTful HTTP.

The application delivery layer provides the following services:

- session establishment including the creation of the session resource.

- session modification and termination procedures.
- session notification procedures.

In order to establish an St session, the PCRF shall send an HTTP POST message to the TSSF including all session information required. The PCRF shall assign a globally and eternally unique St Session ID and shall include it as part of the session resource representation within the body of the HTTP POST. The St Session ID is defined in subclause 5.3.4. If the session resource is successfully created at the TSSF, the TSSF shall provide the session resource URI within the Location header of the HTTP response.

In order to modify an St session, the PCRF shall send an HTTP PUT (full replacement of the St session resource) or an HTTP PATCH message, which includes the St session ID as a path element of the URI.

In order to query the state of an St session, the PCRF shall send an HTTP GET message including the St Session ID as a path element of the request URI.

In order to delete an St session, the PCRF shall send an HTTP DELETE message including the St session ID as a path element of the request URI.

In order to report traffic steering rule enforcement failures for an St session, if the notification feature is supported by both the PCRF and TSSF, the TSSF shall send an HTTP POST message including the St session ID as a path element of the notification URI.

Every HTTP message contains the specific communication information required for this case in its body.

### 5.3.1 Methods

Methods indicate to the server what action has to be performed. Every HTTP request message has a method. The following HTTP methods as defined in IETF RFC 2616 [4] and IETF RFC 5789 [12] can be used:

- POST:
  - Used to create a resource state. The request URI defines the address responsible for the creation of the resource.
  - Used to report traffic steering rule enforcement failures from the TSSF to the PCRF. The notification report is provided in the body of the message. The request URI defines the address of the impacted resource.
- PUT: Used to replace a resource state. The full state of the resource is provided in the body of the message. The request URI defines the resource which will be replaced.
- PATCH: The PATCH method applies partial modifications to a resource. The request URI defines the resource which will be modified.
- GET: Used to retrieve a resource state. The request URI defines the resource which is queried. The server returns the resource state representation within the body of the response.
- DELETE: Used to delete a resource state. The request URI defines the resource which will be deleted.

Every HTTP request results in a response message that comes back with a status code and further information in its body, if required. The PCRF waits for this response before initiating a further request for the same resource.

### 5.3.2 Resources and URI design

The St session is a RESTful resource that shall be identified by a URI.

The URI design shall be based on the structure defined in IETF RFC 3986 [13]:

```

scheme ":" hier-part [ "?" query ] [ "#" fragment ]
  hier-part = "://" authority path-abempty
    / path-absolute
    / path-rootless
    / path-empty

```

The scheme may be HTTP or HTTPS for the REST St interface. Within a scheme the definition of names shall follow the rules of HTTP URIs. Host and port are the main parts of the authority. The path element identifies the resources.

For the REST St interface, the following required parts of the URI shall be used as follows:

- scheme: The application delivery layer protocol "http" or "https".
- authority: It includes the server address and optionally a port as follows: host [":" port]
- path-absolute: The path-absolute should have the following ABNF: "/" mainapp "/" mainresource ["/" resourcepath], where for this release, "mainapp" is "stapplication", "mainresource" is "sessions" and "resourcepath" contains the path to identify a session resource, which in this release is the St Session ID as defined in subclause 5.3.4.

NOTE: A different path can be used when the Resource URI is preconfigured in the PCRF.

An example of the URI to identify the "sessions" main resource is: <http://tssfserver.example.com/stapplication/sessions>.

## 5.3.3 HTTP request/response formats

### 5.3.3.1 General

Session establishment, modification, termination and query procedures are performed through HTTP transactions consisting of a request initiated by the PCRF and its corresponding response provided by the TSSF.

Table 5.3.3.1-1 summarizes the content of the requests and responses. More detailed information is specified in the corresponding subclauses as indicated in the table.

Table 5.3.3.1-1: St requests/response summary table

Method	Resource URI's path (NOTE 1)	Clause Defined	Request body	Response body
POST	/stapplication/sessions	5.3.3.2	Content-Type: application/json The PCRF shall include the state of the session resource using the schema defined in Annex B.1.	Successful response: The TSSF may include informational data in the body of the response as per Annex B.2 Error response: The TSSF should include error data in the body of the response as per Annex B.2
PUT	/stapplication/sessions/ {stsessionid}	5.3.3.3	Content-Type: application/json The PCRF shall include the state of the session resource using the schema defined in Annex B.1.	Successful response: The TSSF may include informational data in the body of the response as per Annex B.2 Error response: The TSSF should include error data in the body of the response as per Annex B.2
PATCH	/stapplication/sessions/ {stsessionid}	5.3.3.4	Content-Type: application/json-patch+json The PCRF shall include the partial modifications to the state of the session resource using the schema defined in Annex B.1.	Successful response: The TSSF may include informational data in the body of the response as per Annex B.2 Error response: The TSSF should include error data in the body of the response as per Annex B.2
DELETE	/stapplication/sessions/ {stsessionid}	5.3.3.5	None	Successful response: The TSSF may include informational data in the body of the response as per Annex B.2 Error response: The TSSF should include error data in the body of the response as per Annex B.2
GET	/stapplication/sessions/ {stsessionid}	5.3.3.6	None	Successful response: The TSSF shall include the representation of the corresponding session resource in the body of the response as per Annex B.2 Error response: The TSSF should include error data in the body of the response as per Annex B.2
POST	{notificationbaseurl}/{s tsessionid}	5.3.3.7	Content-Type: application/json The TSSF shall report traffic steering rule enforcement failures to the PCRF using the JSON format definition in Annex B.4	Successful response: The PCRF may include informational data in the body of the response as per Annex B.2 Error response: The PCRF should include error data in the body of the response as per Annex B.2
NOTE: A different path from /stapplication/sessions/ may be used when it is configured in the PCRF. In that case the "path" part set in the different methods should use the configured one.				

### 5.3.3.2 POST /stapplication/sessions

The establishment of a session shall be performed by the PCRF by using the POST method as follows:

- the request URI formatted as defined in subclause 5.3.2 with the "path" part set to: /stapplication/sessions.
- the Content-Type header field set to "application/json"

- the body of the message encoded in JSON format as defined in Annex B.1, including the "session-id" field allocated by the PCRF.

The PCRF sets the URI based on the pre-configuration at the PCRF.

If the PCRF supports the Notification feature, it shall follow the procedures defined in subclause 5.3.6.1 to advertise support of the feature. In addition, the PCRF shall include the 3gpp-Notification-Base-URL header as defined in subclause 5.3.7.4.

Upon receipt of the HTTP POST, the TSSF shall respond to the PCRF indicating whether the creation of the resource was successful or not using one of the HTTP status codes as defined in subclause 5.3.5. If the resource creation is successful, the TSSF shall respond with an HTTP 201 (Created) status code, including a Location header field containing the URI for the created session resource based on the St Session ID as defined in subclause 5.3.4. If the creation of the session resource was not successful, the TSSF shall indicate the reason using an appropriate HTTP status code for the St interface as defined in subclause 5.3.5 and optionally additional information in the body of the response as defined in Annex B.2.

If both the PCRF and TSSF support the Notification feature, the TSSF shall remember the value of the 3gpp-Notification-Base-URL for the lifetime of the corresponding St session and shall initiate a TCP connection towards the PCRF (if it is not already established) based on the hostname and optional port elements of the 3gpp-Notification-Base-URL.

NOTE: Network deployment scenario with overlapping IP address assignment where the PCRF connects to multiple PCEFs connecting to the same PDN and where these multiple PCEFs connect to the same TSSF is not supported in this release.

Below is an example of an HTTP POST and a corresponding successful response:

POST /stapplication/sessions HTTP/1.1

Host: tssfserver.example.com

3gpp-Notification-Base-URL: <http://pcrfserver.example.com:8080/stapplication/notification>

3gpp-Optional-Features: Notification

Content-Type: application/json

Content-Length: ...

```
{
  "session-id": "pcrf.example.com;378388838383;123232",
  "ue-ipv4": "10.0.0.2",
  "called-station-id": "apncompany.com",
  "tsrules": {
    "ts-rule-3": {
      "ts-rule-name": "ts-rule-3",
      "tdf-application-identifier": "ftp-download",
      "precedence": 1,
      "ts-policy-identifier-dl": "firewall"
    }
  }
}
```

Here is an example of a successful response:

HTTP/1.1 201 Created

Location: http://tssfserver.example.com/stapplication/sessions/pcrf.example.com;378388838383;123232

Date: Mon, 23 Apr 2012 17:10:00 GMT

Server: tssf.example.com

3gpp-Accepted-Features: Notification

Content-Type: application/json

```
{
  "success-message": "Session was created successfully.",
}
```

### 5.3.3.3 PUT /stapplication/sessions/{stsessionid}

The full modification of a session resource may be performed by the PCRF using the PUT method. In this case, the PCRF shall send an HTTP PUT request as follows:

- the request URI formatted as defined in subclause 5.3.2 with the "path" part set to: /stapplication/sessions/{stsessionid}, where the stsessionid is the St Session ID as defined in subclause 5.3.4.
- the Content-Type header field set to "application/json"
- the body of the message encoded in JSON format as defined in Annex B.1 specifying the full representation of the session resource

Upon receipt of the HTTP PUT, the TSSF shall respond to the PCRF indicating whether the modification of the resource was successful or not using one of the HTTP status codes as defined in subclause 5.3.5. If the resource modification is successful, the TSSF shall respond with an HTTP 200 OK status code if it includes optional informational data in the body of the response as per Annex B.2 or HTTP 204 No Content status code otherwise. If the modification of the session resource was not successful, the TSSF shall indicate the reason using an appropriate HTTP status code for the St interface as defined in subclause 5.3.5 and optionally additional information in the body of the response as defined in Annex B.2.

Below is an example of an HTTP PUT:

PUT /stapplication/sessions/pcrf.example.com;378388838383;123232 HTTP/1.1

Host: tssfserver.example.com

Content-Type: application/json

Content-Length: ...

```
{
  "session-id": "pcrf.example.com;378388838383;123232",
  "ue-ipv4": "10.0.0.2",
  "tsrules": {
    "ts-rule-1": {
      "ts-rule-name": "ts-rule-1",
      "tdf-application-identifier": "ftp-download",

```



```

    "precedence": 1,
    "ts-policy-identifier-dl": "firewall"
  },
  "ts-rule-2": {
    "ts-rule-name": "ts-rule-2",
    "tdf-application-identifier": "application-x",
    "precedence": 2,
    "ts-policy-identifier-dl": "firewall"
  }
}
}

```

Here is an example of a successful response:

HTTP/1.1 200 OK

Date: Mon, 23 Apr 2012 17:11:00 GMT

Server: tssf.example.com

Content-Type: application/json

```

{
  "success-message": "Session was updated successfully.",
}

```

#### 5.3.3.4 PATCH /stapplication/sessions/{stsessionid}

The partial modification of a session resource may be performed by the PCRF using the PATCH method as defined in IETF RFC 5789 [12]. In this case, the PCRF shall send an HTTP PATCH request as follows:

- the request URI formatted as defined in subclause 5.3.2 with the "path" part set to: /stapplication/sessions/{stsessionid}, where the stsessionid is the St Session ID as defined in subclause 5.3.4.
- the Content-Type header field set to "application/json-patch+json"
- the body of the message shall be encoded using the JSON Patch format as defined in IETF RFC 6902 [14]. The following operations from [14] may be used:
  - add: to create a new traffic steering rule.
  - remove: to delete a traffic steering rule.
  - replace: to replace an existing rule with a new definition.

Upon receipt of the HTTP PATCH, the TSSF shall respond to the PCRF indicating whether the modification of the resource was successful or not using one of the HTTP status codes as defined in subclause 5.3.5. If the resource modification is successful, the TSSF shall respond with an HTTP 200 OK status code if it includes optional informational data in the body of the response as per Annex B.2 or HTTP 204 No Content status code otherwise. If the modification of the session resource was not successful, the TSSF shall indicate the reason using an appropriate HTTP

status code for the St interface as defined in subclause 5.3.5 and optionally additional information in the body of the response as defined in Annex B.2.

Below is an example of an HTTP PATCH that modifies a traffic steering rule and deletes another one:

```
PATCH /stapplication/sessions/pcrf.example.com;378388838383;123232 HTTP/1.1
```

```
Host: tssfserver.example.com
```

```
Content-Type: application/json-patch+json
```

```
Content-Length: ...
```

```
[
  {
    "op": "replace",
    "path": "/tsrules/ts-rule-1",
    "value": {
      "ts-rule-name": "ts-rule-1",
      "tdf-application-identifier": "ftp-download",
      "precedence": 1,
      "ts-policy-identifier-dl": "firewall2"
    }
  },
  {
    "op": "remove",
    "path": "/tsrules/ts-rule-2",
  }
]
```

Here is an example of a successful response:HTTP/1.1 200 OK

```
Date: Mon, 23 Apr 2012 17:12:00 GMT
```

```
Server: tssf.example.com
```

```
Content-Type: application/json
```

```
{
  "success-message": "Session was patched successfully.",
}
```

### 5.3.3.5 DELETE /stapplication/sessions/{stsessionid}

To delete an St session, the PCRF shall send an HTTP DELETE request to the TSSF as follows:

- the request URI formatted as defined in subclause 5.3.2 with the "path" part set to: /stapplication/sessions/{stsessionid}, where the stsessionid is the St Session ID as defined in subclause 5.3.4.

Upon receipt of the HTTP DELETE, the TSSF shall respond to the PCRF indicating whether the deletion of the resource was successful or not using one of the HTTP status codes as defined in subclause 5.3.5. If the resource deletion is successful, the TSSF shall respond with an HTTP 200 OK status code if it includes optional informational data in the body of the response as per Annex B.2 or HTTP 204 No Content status code otherwise. If the deletion of the session resource was not successful, the TSSF shall indicate the reason using an appropriate HTTP status code for the St interface as defined in subclause 5.3.5 and optionally additional information in the body of the response as defined in Annex B.2.

Below is an example of an HTTP DELETE:

```
DELETE /stapplication/sessions/pcrf.example.com;378388838383;123232 HTTP/1.1
```

```
Host: tssfserver.example.com
```

Here is an example of a successful response:

```
HTTP/1.1 204 No Content
```

```
Date: Mon, 23 Apr 2012 17:15:00 GMT
```

```
Server: tssf.example.com
```

### 5.3.3.6 GET /stapplication/sessions/{stsessionid}

To query the state of an St session resource, the PCRF shall send an HTTP GET request to the TSSF as follows:

- the request URI formatted as defined in subclause 5.3.2 with the "path" part set to: /stapplication/sessions/{stsessionid}, where the stsessionid is the St Session ID as defined in subclause 5.3.4.

Upon receipt of the HTTP GET, the TSSF shall respond to the PCRF indicating whether the querying of the resource was successful or not using one of the HTTP status codes as defined in subclause 5.3.3. If the resource exists, the TSSF shall respond with an HTTP 200 OK status code and include the session representation state within the body of the response as defined in Annex B.2. The TSSF shall also include the Content-Type header field set to "application/json".

The TSSF shall also include the set of features it supports in common with the PCRF within the "3gpp-Accepted-Features" HTTP header in the response to the HTTP GET.

Below is an example of an HTTP GET:

```
GET /stapplication/sessions/pcrf.example.com;378388838383;123232 HTTP/1.1
```

```
Host: tssfserver.example.com
```

Here is an example of a successful response :

```
HTTP/1.1 200 OK
```

```
Server: tssfserver.example.com
```

```
Content-Type: application/json
```

```
Content-Length: ...
```

```
{  
  "session-id": "pcrf.example.com;378388838383;123232",
```

```

"ue-ipv4": "10.0.0.2",
"called-station-id": "apncompany.com",
"tsrules": {
  "ts-rule-1": {
    "ts-rule-name": "ts-rule-1",
    "tdf-application-identifier": "ftp-download",
    "precedence": 1,
    "ts-policy-identifier-dl": "firewall"
  },
  "ts-rule-2": {
    "ts-rule-name": "ts-rule-2",
    "tdf-application-identifier": "application-x",
    "precedence": 2,
    "ts-policy-identifier-dl": "firewall"
  }
}
}
}

```

### 5.3.3.7 POST /{notificationbaseurl}/{stsessionid}

If the notification feature is supported, the TSSF shall inform the PCRF of state changes related to a session resource or sub resources (e.g. traffic steering rules). The TSSF does so by sending an HTTP POST request as follows:

- the request URI formatted as defined in subclause 5.3.2 with the "path" part set to: /{notificationbaseurl}/{stsessionid}, where the "stsessionid" is the St Session ID as defined in subclause 5.3.4, and the "notificationbaseurl" is set to the value of the 3gpp-Notification-Base-URL provided by the PCRF.
- the Content-Type header field set to "application/json"
- the body of the message encoded in JSON format as defined in Annex B.4 specifying the representation of the notification report.

Upon receipt of the notification HTTP POST, the PCRF shall respond to the TSSF with one of the HTTP status codes as defined in subclause 5.3.5. If the PCRF is able to successfully process the request, the PCRF shall answer with an HTTP 200 OK status code if it includes optional informational data in the body of the response as per Annex B.2 or HTTP 204 No Content status code otherwise. If the PCRF cannot successfully process the request, the PCRF shall indicate the reason using an appropriate HTTP status code for the St interface as defined in subclause 5.3.5 and optionally include additional information in the body of the response as defined in Annex B.2.

Below is an example of an HTTP POST used for notification of traffic steering rule enforcement failure.

```
POST /stapplication/notification/pcrf.example.com;378388838383;123232 HTTP/1.1
```

```
Host: pcrfserver.example.com
```

```
Content-Type: application/json
```

```
Content-Length: ...
```

```
{
  "notifications": [
    {
      "notification-message": "Traffic steering rules became inactive.",
      "notification-type": "application",
      "notification-tag": "TS_RULE_EVENT",
      "notification-info": {
        "ts-rule-reports": [
          {
            "resource-paths": [
              "/tsrules/ts-rule-1",
              "/predefined-tsrules/ts-rule-2",
              "/predefined-group-of-tsrules/group-rules-1"
            ],
            "rule-status": "INACTIVE",
            "rule-failure-code": "RESOURCES_LIMITATION"
          }
        ]
      }
    }
  ]
}
```

Here is an example of a successful response:

HTTP/1.1 200 OK

Date: Mon, 23 Apr 2012 17:11:00 GMT

Server: pcfrserver.example.com

Content-Type: application/json

```
{
  "success-message": "Notification was processed successfully.",
}
```

### 5.3.4 St session ID

The PCRF shall allocate an St Session ID that is used to uniquely identify the St session resource between the PCRF and TSSF. The St Session ID is provided as part of the session resource representation within the body of the HTTP message and shall remain unmodified during the lifetime of the St session resource. If the session resource is successfully created at the TSSF, the TSSF shall include the St session ID in the path element of an HTTP request URI to address the corresponding session resource.

NOTE: The PCRF allocates the St Session ID as opposed to the TSSF to avoid creating duplicate sessions at the TSSF in case an HTTP POST message is received twice by the TSSF (e.g. due to a retry by the PCRF). The PCRF ensures that the St Session ID is globally unique as described in this subclause to avoid conflicts at the TSSF.

The Session ID shall be globally and eternally unique. The Session ID includes a mandatory portion and an implementation specific portion.

The mandatory portion shall start with the PCRF's FQDN [8]. The remainder of the Session ID is delimited by a ';' character and may be any sequence that the PCRF can guarantee to be eternally unique as follows:

<PCRF FQDN>;<Implementation Specific Sequence>

The implementation specific sequence should follow the recommended implementation specific portion of the Session-Id AVP as defined in section 8.8 of IETF RFC 6733 [21].

Here is an example of the Session ID: pcrf123.acme.com;1876543210;523. Here is an example of a corresponding session URI: http://tssfserver.example.com/stapplication/sessions/pcrf123.acme.com;1876543210;523

Because the Session ID is included as a path element of the request URI, it shall follow the implementation-specific syntax standardized for path elements of URLs as defined in [6].

NOTE: Characters in the reserved set are protected from normalization and are therefore safe to be used by scheme-specific and producer-specific algorithms for delimiting data subcomponents within a URI [6].

### 5.3.5 HTTP status codes

A subset of the HTTP status codes as defined in IETF RFC 2616 [4] are applicable to the REST based St interface. Table 5.3.5-1 lists the applicable status codes for the St interface for the current release. Note that if the PCRF or TSSF receive a response with a status code not listed in the table below, they shall treat it based on the class of the status code as defined in [4].

**Table 5.3.5-1: St HTTP status codes**

Status Code	Description
200 OK	The HTTP request has succeeded. The information returned with the response is dependent on the method used in the request.
201 Created	The HTTP POST request has been fulfilled and resulted in a new resource being created at the TSSF.
204 No Content	The recipient has fulfilled the request but does not need to return an entity-body.
400 Bad Request	The request could not be understood by the recipient due to malformed syntax. The sender should not repeat the request without modifications.
403 Forbidden	The recipient understood the request, but is refusing to fulfill it.
404 Not Found	The recipient has not found any resources matching the Request-URI.
405 Method Not Allowed	The method specified in the Request-Line is not allowed for the resource identified by the Request-URI.
408 Request Timeout	The recipient did not respond to the request within the time that the sender is prepared to wait. The sender may repeat the request without modifications at any later time.
412 Precondition Failed	The precondition given in one or more of the request-header fields evaluated to false when it was tested on the recipient.
413 Request Entity Too Large	The recipient is refusing to process a request because the request entity is larger than the recipient is willing or able to process.
414 Request-URI Too Large	The recipient is refusing to service the request because the Request-URI is longer than the recipient is willing to interpret.
500 Internal Server Error	The recipient encountered an unexpected condition which prevented it from fulfilling the request.
501 Not Implemented	The recipient does not support the functionality required to fulfill the request.
503 Service Unavailable	The recipient is currently out of service.

## 5.3.6 Feature negotiation

### 5.3.6.1 General

The REST based St interface needs to provide a mechanism to advertise required and optional features supported by both the PCRF and TSSF for interoperability reasons as the functionality of the REST St based interface is augmented.

The PCRF shall include in the HTTP POST the set of supported features as follows:

- if a feature is required for the proper operation of the St session, it shall be included within the 3gpp-Required-Features header;
- if a feature is optional for the proper operation of the St session, it shall be included within the 3gpp-Optional-Features header.

The TSSF shall include, within the 3gpp-Accepted-Features header in the response to the HTTP POST, the set of features it supports in common with the PCRF.

If the TSSF does not support any of the required features advertised by the PCRF within the 3gpp-Required-Features header, the TSSF shall reject the HTTP POST with an HTTP 412 Precondition Failed status code and shall include the commonly supported features with the PCRF within the 3gpp-Accepted-Features.

If the TSSF requires certain features to be supported that aren't advertised by the PCRF, the TSSF shall reject the HTTP POST with an HTTP 412 Precondition Failed status code and shall include the commonly supported features with the PCRF within the 3gpp-Accepted-Features and the required features in the 3gpp-required-features.

If the PCRF and TSSF successfully negotiate supported features, the list of commonly supported features shall be applicable for the lifetime of the St session. Features that are not advertised as supported shall not be used within the lifetime of the St session.

The table below defines the features applicable to the REST St interface.

**Table 5.3.6.1-1: Features used in REST St Protocol**

Feature	M/O	Description
Notification	O	This feature indicates the support of the HTTP notification procedure from the TSSF to the PCRF.
Feature: A short name that can be used to refer to the bit and to the feature, e.g. "Notification". M/O: Defines if the implementation of the feature is mandatory ("M") or optional ("O") in this 3GPP Release. Description: A clear textual description of the feature.		

## 5.3.7 HTTP custom headers

This subclause defines any new HTTP custom headers introduced by this specification.

### 5.3.7.1 3gpp-Optional-Features

This header is used by the PCRF to advertise the optional features that are supported by the PCRF.

The encoding of the header follows the ABNF as defined in [4].

3gpp-Optional-Features = "3gpp-Optional-Features" ":" 1#token

An example is: 3gpp-Optional-Features: feature1, feature2

### 5.3.7.2 3gpp-Required-Features

This header is used by the PCRF to announce the mandatory features that must be supported in TSSF.

This header is also used by the TSSF to indicate the missing features that must be supported in PCRF.

The encoding of the header follows the ABNF as defined in [4].

`3gpp-Required-Features = "3gpp-Required-Features" ":" 1#token`

An example is: `3gpp-Required-Features: feature1, feature2`

### 5.3.7.3 3gpp-Accepted-Features

The header is used by the TSSF to confirm the commonly supported set of features with the PCRF.

The encoding of the header follows the ABNF as defined in [4].

`3gpp-Accepted-Features = "3gpp-Accepted-Features" ":" 1#token`

An example is: `3gpp-Accepted-Features: feature1, feature2`

### 5.3.7.4 3gpp-Notification-Base-URL

This header is used by the PCRF to provide the notification base URL.

The encoding of the header follows the ABNF as defined in [4].

`3gpp-Notification-Base-URL = "3gpp-Notification-Base-URL " ":" 1#token`

An example is: `3gpp-Notification-Base-URL: http://pcrfserver.example.com:8080/stapplication/notification`

## 5.4 Specific application communication

### 5.4.1 General

Specific application communication represents the presentation of application data structures by transforming data into the form that the application accepts. It establishes the context between application-layer entities.

Note: This release only supports the content type JSON

### 5.4.2 Content type

The body of HTTP messages shall be in JSON format. The content of the JSON text is defined in subclause 5.4.3 and Annex B.

The MIME media type that shall be used within the Content-Type header field is “application/json” as defined in IETF RFC 7159 [10].

### 5.4.3 JSON session fields

#### 5.4.3.1 General

Table 5.4.3.1.1 describes the JSON fields used within the body of the HTTP messages representing the session information. The table includes the information about the name of the field, the type of the fields and the applicability of the JSON field.



Table 5.4.3.1.1: St Specific JSON fields

Field Name	Subclause defined	JSON Value Type (NOTE 1)	JCR Type (NOTE 2)	Applicability (NOTE 3)
called-station-id	5.4.3.20	string	string	
session-id	5.4.3.2	string	string	
ue-ipv4	5.4.3.3	string	ipv4	
ue-ipv6-prefix	5.4.3.4	string	lIpv6	
tsrules	5.4.3.5	object	object	
ts-rule-name	5.4.3.6	string	string	
Precedence	5.4.3.7	number	number: 0..4294967295	
tdf-application-identifier	5.4.3.8	string	string	
flow-information	5.4.3.9	array	array	
flow-description	5.4.3.10	string	string	
tos-traffic-class	5.4.3.11	string	/^[0-9A-Fa-f]{4}\$/ (NOTE 4)	
security-parameter-index	5.4.3.12	string	/^[0-9A-Fa-f]{8}\$/ (NOTE 4)	
flow-label	5.4.3.13	string	/^[0-9A-Fa-f]{6}\$/ (NOTE 4)	
flow-direction	5.4.3.14	string	"BIDIRECTIONAL" "UPLINK" "DOWNLINK" (NOTE 5)	
ts-policy-identifier-dl	5.4.3.15	string	string	
ts-policy-identifier-ul	5.4.3.16	string	string	
predefined-tsrules	5.4.3.17	object	object	
predefined-group-of-tsrules	5.4.3.18	object	object	
ts-rule-base-name	5.4.3.19	string	string	
NOTE 1: The basic JSON value types are defined in IETF RFC 7159 [15].				
NOTE 2: The JCR types are defined in IETF draft-newton-json-content-rules [16].				
NOTE 3: Fields marked with a supported feature are applicable as described in subclause 5.3.6.				
NOTE 4: The regular expression for a string type.				
NOTE 5: The quoted strings for a string type.				

### 5.4.3.2 session-id

The session-id field is of type string, and contains the session id allocated by the PCRF. The session id format is defined in subclause 5.3.4.

### 5.4.3.3 ue-ipv4

The ue-ipv4 field is of type string and ip4 literal as defined in IETF draft-newton-json-content-rules [16].

### 5.4.3.4 ue-ipv6-prefix

The ue-ipv6-prefix field is of type string and ip6 literal as defined in IETF draft-newton-json-content-rules [16]. The PCRF uses it to indicate the IPv6 prefix allocated for the user.

### 5.4.3.5 tsrules

The tsrules field is of type object. It contains all of the dynamic traffic steering rules defined in the JSON body.

The following defines the content of the dynamic traffic steering rule:

The ts-rule-name field shall be included within the dynamic traffic steering rule.

A precedence value may be included.

Either flow-information(s) or the tdf-application-identifier field shall be included within the dynamic traffic steering rule.

The ts-policy-identifier-ul field and/or the ts-policy-identifier-dl field shall be included. If the traffic steering policies are identical in both downlink and uplink directions, the values of the ts-policy-identifier-ul field and the ts-policy-identifier-dl field shall be identical.

Below are the JCR for the dynamic traffic steering rule:

```
$tsrule = // : {
    $ts-rule-name,
    $precedence ?,
    ( $flow-information | $tdf-application-identifier ),
    ( $ts-policy-identifier-ul | $ts-policy-identifier-dl | ( $ts-policy-identifier-ul, $ts-policy-identifier-dl ) )
}
```

#### 5.4.3.6 ts-rule-name

The ts-rule-name is of type string, and it defines a name for traffic steering rule. For traffic steering rules provisioned by the PCRF, it uniquely identifies a traffic steering rule within one St session. For traffic steering rules pre-defined at the TSSF, it uniquely identifies a traffic steering rule within the TSSF.

#### 5.4.3.7 precedence

The precedence is of type number. Its value ranges from 0 to 4294967295.

Within the tsrule JSON content, the precedence field determines the order for the service data flow detection/the application detection at the TSSF to perform traffic steering. A traffic steering rule with a lower precedence value shall be applied prior to a traffic steering rule with a higher precedence value.

NOTE: The precedence value range defined within the traffic steering rule is operator configurable.

#### 5.4.3.8 tdf-application-identifier

The tdf-application-identifier is of type string. It references the application detection filter defined at the TSSF.

#### 5.4.3.9 flow-information

The flow-information field is of type array. It is sent from the PCRF to the TSSF and contains the information for a single IP flow packet filter.

The flow-description, a tos-traffic-class, security-parameter-index and flow-label specify the parameters to be used for matching payload packets. One of these fields shall be present in the object. The flow-direction field shall also be included.

The JCR for the flow-information is as follows:

```
$flow-information = "flow-information" : [
{
    $flow-description ?,
    $tos-traffic-class ?,
    $security-parameter-index ?,
    $flow-label ?,
    $flow-direction
```

```

    } +
  ]

```

#### 5.4.3.10 flow-description

The flow-description field is of type string, it defines a packet filter for an IP flow.

The content has the same encoding as the IPFilterRule AVP value as defined in IETF RFC 6733 [21]. All limitations applicable to the Flow-Description AVP in subclause 5.4.2 of 3GPP TS 29.212 [17] apply to this field.

#### 5.4.3.11 tos-traffic-class

The tos-traffic-class field is of type string. It is encoded on two octets, and represented as a hex string. The first octet contains the IPv4 Type-of-Service or the IPv6 Traffic-Class field and the second octet contains the ToS/Traffic Class mask field. When used for traffic detection, it defines the packet marking information used for traffic detection in the TSSF.

#### 5.4.3.12 security-parameter-index

The security-parameter-index field is of type string, and it contains the security parameter index of the IPSec packet.

#### 5.4.3.13 flow-label

The flow-label field is of type string, and it contains the IPv6 flow label header field.

#### 5.4.3.14 flow-direction

The flow-direction field is of type string with three enumerations. It indicates the direction(s) that a filter is applicable, downlink only, uplink only or both down- and uplink (bidirectional).

DOWNLINK

The corresponding filter applies for traffic to the UE.

UPLINK

The corresponding filter applies for traffic from the UE.

BIDIRECTIONAL

The corresponding filter applies for traffic both to and from the UE.

#### 5.4.3.15 ts-policy-identifier-dl

The ts-policy-identifier-dl is of type string and contains a reference to a pre-configured traffic steering policy at the TSSF in the downlink direction.

#### 5.4.3.16 ts-policy-identifier-ul

The ts-policy-identifier-ul is of type string and contains a reference to a pre-configured traffic steering policy at the TSSF in the uplink direction.

#### 5.4.3.17 predefined-tsrules

The predefined-tsrules contains all of the predefined traffic steering rules in the session.

The ts-rule-name field uniquely identifies the traffic steering rule defined at the TSSF.

The JCR definition for the predefined traffic steering rule is as follows:

```
$predefined-tsrule = // : {
```

```

    ts-rule-name
  }

```

### 5.4.3.18 predefined-group-of-tsrules

The predefined-group-of-tsrules field is of type object and contains the groups of the predefined traffic steering rules in the session.

The ts-rule-base-name field uniquely identifies the group of predefined traffic steering rules at the TSSF.

The JCR definition for the group of the dynamic traffic steering rules:

```

$predefined-group-of-tsrule = // : {
    ts-rule-base-name
}

```

### 5.4.3.19 ts-rule-base-name

The ts-rule-base-name field is of type string, and it indicates the name of a predefined group of traffic steering rules residing at the TSSF.

### 5.4.3.20 called-station-id

The called-station-id is of type string, and it defines the address the user is connected to (i.e. the PDN identifier). For GPRS and EPS the APN. When used to contain the APN, the APN is composed of the APN Network Identifier only, or the APN Network Identifier and the APN Operator Identifier as specified in 3GPP TS 23.003 [20], subclause 9.1. The inclusion of the APN Operator Identifier can be configurable.

## 5.4.4 JSON errors and informational response fields

### 5.4.4.1 General

Table 5.4.4.1.1 describes the JSON fields defined for the errors and informational responses including their types and the field names.

**Table 5.4.4.1.1: JSON fields for errors and informational response**

Field Name	Subclause defined	JSON Value Type (NOTE 1)	JCR Type (NOTE 2)
errors	5.4.4.2	array	array
error-type	5.4.4.3	string	"application" "interface" "server" "other" (NOTE 3)
error-message	5.4.4.4	string	string
error-tag	5.4.4.5	string	string
error-path	5.4.4.6	string	string
error-info	5.4.4.7	object	object
success-message	5.4.4.8	string	string
success-path	5.4.4.9	string	string
success-info	5.4.3.10	object	object
NOTE 1: The basic JSON value types are defined in IETF RFC 7159 [10].			
NOTE 2: The JCR types are defined in IETF draft-newton-json-content-rules [16].			
NOTE 3: The quoted strings for a string type.			

#### 5.4.4.2 errors

The errors field is of type array, and it contains a list of errors. Each error is defined as an object in the array. This field shall be present when the TSSF responds with error(s) back to an HTTP request initiated by the PCRF.

An error object shall contain the error-type field and the error-message.

Optionally, it may include the error-tag field, error-path field and error-info field for further details. The error tag is used to classify a special error from other errors. The error-path is used to identify which resource has the error. The error-info is provided for additional information.

The JCR format:

```
e
$errors = "errors" : [
  {
    $error-type,
    $error-message,
    $error-tag ?,
    $error-path ?,
    $error-info ?
  } +
]
```

#### 5.4.4.3 error-type

The error-type field is of type string. It can be one of 'application', 'interface', 'server' and 'other'.

When an error is related to the application processing in the TSSF, the error type shall be specified as 'application'. When the error is a protocol compliance error, 'interface' shall be used. A server failure is represented with the 'server' error type. For all other errors, the 'other' error type shall be used.

#### 5.4.4.4 error-message

The error-message field is of type string. It is a textual description of the error.

#### 5.4.4.5 error-tag

The error-tag field is of type string. It defines a tag for a particular error.

##### TS\_RULE\_EVENT

This error-tag shall be used when the traffic steering rules cannot be installed/activated at the TSSF. Affected traffic steering rules shall be provided in the error-info with a sub field named ts-rule-reports including the reason and status of the rules as described in subclause 5.4.5.

#### 5.4.4.6 error-path

The error-path field is of type string. It shall contain a JSON pointer as defined in IETF RFC 6901 [15] pointing to a resource where the error occurs.

#### 5.4.4.7 error-info

The error-info field is defined as an object type. It represents informational details regarding the error.

#### 5.4.4.6 success-message

The success-message is of type string. When included in the JSON body of an HTTP response, it shall be included under the root level of the JSON content.

#### 5.4.4.7 success-path

The success-path field is of type string. It is a JSON pointer as defined in IETF RFC 6901 [15] pointing to the resource that was successfully operated on. It is an optional root level field provided along with a success-message field.

#### 5.4.4.8 success-info

The success-info field is defined as an object type. It may include any additional information under the root level of the JSON content for an informational response.

### 5.4.5 JSON report fields

#### 5.4.5.1 General

Table 5.4.5.1.1 describes the JSON fields defined for the report information objects which are included in the error-info field or the notification-info field.

**Table 5.4.5.1.1: JSON fields for Report from the TSSF**

Field Name	Subclause defined	JSON Value Type (NOTE 1)	JCR Type (NOTE 2)
ts-rule-reports	5.4.5.2	array	array
resource-path	5.4.5.3	string	string
rule-status	5.4.5.4	string	string
rule-failure-code	5.4.5.5	string	string
NOTE 1: The basic JSON value types are defined in IETF RFC 7159 [10].			
NOTE 2: The JCR types are defined in IETF draft-newton-json-content-rules [16].			

#### 5.4.5.2 ts-rule-reports

The ts-rule-reports field is of type array and it contains a list of traffic steering rule reports.

The ts-rule-reports field can be used in an HTTP response to report failures in the installation/activation of traffic steering rules. In this case, it shall be provided as a sub field of the error-info field.

The ts-rule-reports field can also be used in a HTTP request from the TSSF to notify the PCRF about enforcement failures of traffic steering rules. In this case, it shall be provided as a sub field of the notification-info field.

Multiple traffic steering rule report instances shall be provided within the ts-rule-reports field if different rule-status or rule-failure-code values are applicable within the same HTTP response or notification.

A report instance shall contain the resource-paths, rule-status and rule-failure-code fields.

The JCR format for the ts-rule-report is:

```
$ts-rule-reports = "ts-rule-reports" : [
{
  $resource-paths,
  $rule-status,
  $rule-failure-code
```

```

} +
]

```

### 5.4.5.3 resource-paths

The resource-paths field is of type array of strings. Each string shall be a JSON pointer as defined in IETF RFC 6901 [15].

Each JSON pointer within the resource-paths field shall be a reference to a specific traffic steering rule or to a group of traffic steering rules which cannot be installed/activated or enforced at the TSSF.

### 5.4.5.4 rule-status

The rule-status is of type string and shall contain one of the string representations of the PCC-Rule-Status AVP values as defined in subclause 5.3.19 of 3GPP TS 29.212 [17].

NOTE: For the current release, only the INACTIVE value is used.

### 5.4.5.5 rule-failure-code

The rule-failure-code is of type string.

The following values defined in the Rule-Failure-Code AVP in subclause 5.3.38 of 3GPP TS 29.212 [17] are applicable for traffic steering rule errors.

UNKNOWN\_RULE\_NAME  
 GW/PCEF\_MALFUNCTION  
 RESOURCES\_LIMITATION  
 MISSING\_FLOW\_INFORMATION  
 INCORRECT\_FLOW\_INFORMATION  
 TDF\_APPLICATION\_IDENTIFIER\_ERROR  
 FILTER\_RESTRICTIONS

The following value defined in the Rule-Failure-Code AVP is reused for traffic steering rule errors, however with a different explanation:

RESOURCE\_ALLOCATION\_FAILURE

This value is used to indicate that the traffic steering rule could not be successfully installed or maintained since the resources or services of the traffic steering control specified by the rule cannot be reserved in the TSSF and/or the (S)Gi-LAN; or those resources or services were released in the TSSF and/or the (S)Gi-LAN, for example, due to the switch eviction mechanism.

In addition, the following values are added:

RESOURCE\_TIMEOUT

This value is used to indicate that the traffic steering rule could not be successfully installed or maintained since the booked resources or services of the traffic steering control specified by the rule have expired in the TSSF and/or the (S)Gi-LAN; for example, due to the flow expiry mechanism in the switches.

TS\_POLICY\_IDENTIFIER\_ERROR

This value is used to indicate that the rule could not be successfully installed or enforced because both the ts-policy-identifier-dl and ts-policy-identifier-ul are invalid, unknown, or not applicable to the downlink and uplink traffic flows for the traffic steering control.

**TS\_POLICY\_IDENTIFIER\_DL\_ERROR**

This value is used to indicate that the rule could not be successfully installed or enforced because the ts-policy-identifier-dl is invalid, unknown, or not applicable to the downlink traffic flows for the traffic steering control.

**TS\_POLICY\_IDENTIFIER\_UL\_ERROR**

This value is used to indicate that the rule could not be successfully installed or enforced because the ts-policy-identifier-ul is invalid, unknown, or not applicable to the uplink traffic flows for the traffic steering control.

## 5.4.6 JSON notification fields

### 5.4.6.1 General

Table 5.4.6.1.1 describes the JSON fields defined for the notification HTTP messages from the TSSF to the PCRF, including their JSON value types, and JCR types as well.

**Table 5.4.6.1.1: JSON fields for notifications from the TSSF**

Field Name	Subclause defined	JSON Value Type (NOTE 1)	JCR Type (NOTE 2)
notifications	5.4.6.2	array	array
notification-type	5.4.6.3	string	"application" "other" (NOTE 3)
notification-message	5.4.6.4	string	String
notification-tag	5.4.6.5	string	String
notification-info	5.4.6.6	object	Object
NOTE 1: The basic JSON value types are defined in IETF RFC 7159 [10].			
NOTE 2: The JCR types are defined in IETF draft-newton-json-content-rules [16].			
NOTE 3: The quoted strings for a string type.			

### 5.4.6.2 notifications

The notifications field is of type array, and it contains a list of notifications provided by the TSSF. Each notification is defined as an object in the array.

A notification object shall contain the notification-type and the notification-message fields.

Optionally, it may include the notification-tag and notification-info fields. The notification-tag is used to distinguish a specific notification from others. The notification-info is provided for additional information.

The JCR format of the notifications is as follows:

```
$notifications = "notifications" : [
{
  $notification-type,
  $notification-message,
  $notification-tag ?,
  $notification-info ?
} +
```

### 5.4.6.3 notification-type

The notification-type field is of type string. It can be one of 'application' or 'other'.



When a notification is due to a change to one of the St session resources/sub-resources, the type shall be specified as 'application'. For all other notifications, the 'other' type shall be used.

#### 5.4.6.4 notification-message

The notification-message field is of type string. It is a textual description of the notification.

#### 5.4.6.5 notification-tag

The notification-tag field is of type string. It defines a tag for a particular notification.

TS\_RULE\_EVENT

This notification-tag value shall be used when the traffic steering rules cannot be enforced at the TSSF. Affected traffic steering rules shall be provided by the TSSF in the notification-info with a sub field named ts-rule-reports including the report reason and status of the rules.

#### 5.4.6.6 notification-info

The notification-info field is defined as an object type. It represents informational details regarding the notification.

### 5.5 TSSF discovery

The TSSF URI is pre-configured on the PCRF, e.g. per PCEF.

---

## 6 Secure communication

Either the NDS/IP network layer security defined in 3GPP TS 33.210 [7] or HTTP over TLS as defined in [19] should be used to secure communication over the REST based St interface.

---

## Annex A (informative): Call Flows

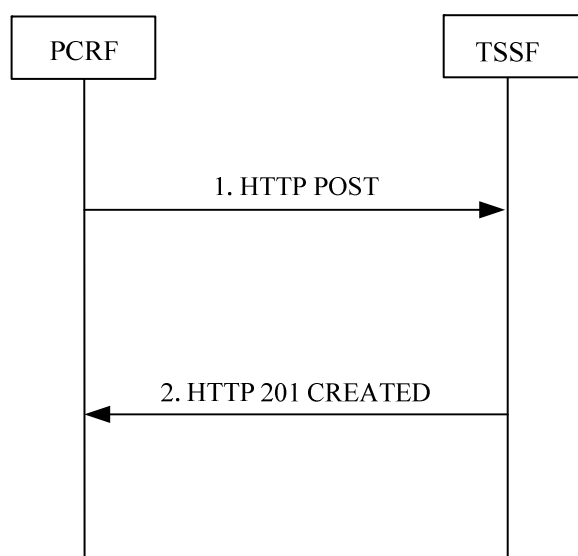
### A.1 General

This annex describes the procedures for the interactions between the PCRF and the TSSF

---

### A.2 Traffic Steering Control Information Provisioning

This subclause describes the signalling flow for the traffic steering control information provisioning.



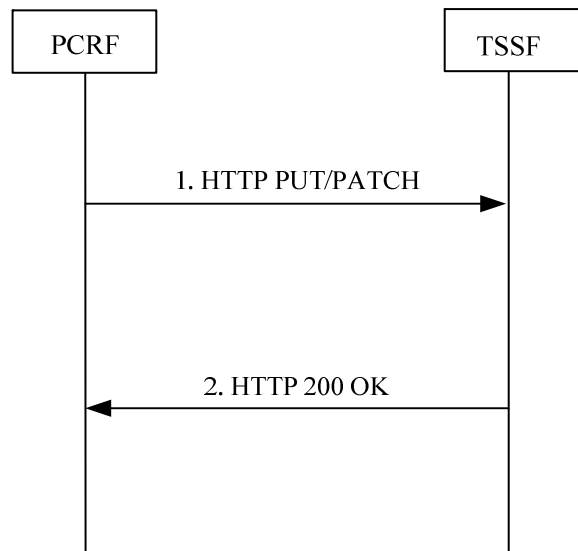
**Figure A.2.1: Traffic Steering Control Information Provisioning**

1. The PCRF sends the HTTP POST to the TSSF to initiate the creation of a resource state to provision the traffic steering control information including the parameters defined in subclause 4.4.2.
2. The TSSF sends the HTTP 201 CREATED response to the PCRF including the parameters defined in subclause 4.4.2.

---

### A.3 Traffic Steering Control Information Update

This subclause describes the signalling flow for the traffic steering control information update or the allocation/release notification of the UE IPv4 address.



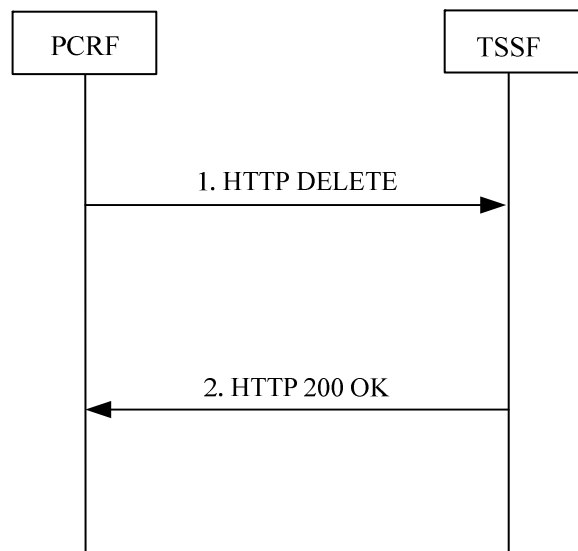
**Figure A.3.1: Traffic Steering Control Information Update**

1. The PCRF sends the HTTP PUT/PATCH to the TSSF to initiate the traffic steering control information update including the parameters defined in subclause 4.4.2. The PCRF also sends the HTTP PUT/PATCH to the TSSF to notify the UE IPv4 address allocation or release including the parameters defined in subclause 4.4.4.
2. The TSSF sends the HTTP 200 OK to the PCRF.

---

## A.4 Traffic Steering Control Information Removal

This subclause describes the signalling flow for the traffic steering control information removal.

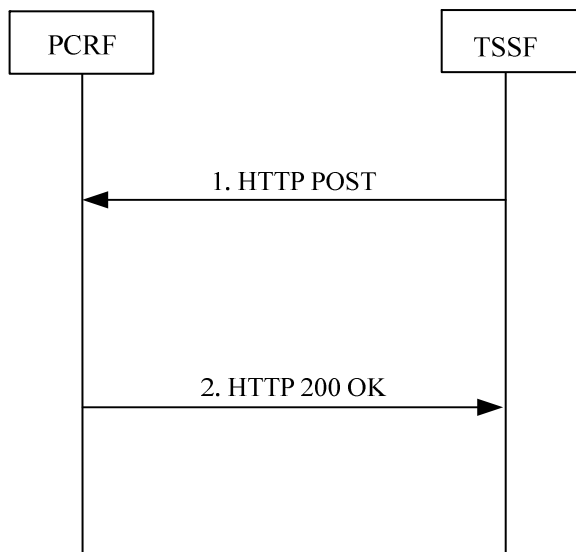


**Figure A.4.1: Traffic Steering Control Information Removal**

1. The PCRF sends the HTTP DELETE to the TSSF to initiate the traffic steering control information removal including the parameters defined in subclause 4.4.2.
2. The TSSF removes/deactivates all the ADC rules and sends the HTTP 200 OK to the PCRF including the parameters defined in subclause 4.4.2.

## A.5 Traffic Steering Control Notification

This subclause describes the signalling flow for the traffic steering control notification.



**Figure A.5.1: Traffic Steering Control Notification**

1. The TSSF sends an HTTP POST to the PCRF to inform it of changes to the state of traffic steering control rules as defined in subclause 4.4.3.
2. The PCRF acknowledges the notification and sends an HTTP 200 OK to the TSSF.

---

# Annex B (normative): JSON Schema

## B.1 Session body schema

This subclause defines the JSON schema for the body of HTTP requests and responses providing the state of a session resource. The schema is based on IETF draft-newton-json-content-rules [16] and is defined below:

```
# jcr-version 0.7
```

```
# ruleset-id 3gpp.stapplication.session
```

```
; A JCR representing the session resource
```

```
$session-root = @{root}{  
  $session-id,  
  ( $ue-ipv4 | $ue-ipv6-prefix | ( $ue-ipv4 , $ue-ipv6-prefix ) ),  
  $called-station-id ?,  
  $tsrules ?,  
  $predefined-tsrules ?,  
  $predefined-group-of-tsrules ?  
}
```

```
; Resource fields definitions
```

```
; Session ID allocated by the PCRF
```

```
$session-id = "session-id" : string
```

```
; The UE IPv4 address
```

```
$ue-ipv4 = "ue-ipv4" : ipv4
```

```
; The UE IPv6 prefix
```

```
$ue-ipv6-prefix = "ue-ipv6-prefix" : ipv6
```

```
; The PDN the UE accesses
```

```
$called-station-id = "called-station-id" : string
```

```
; The dynamic traffic steering rules
```

```
$tsrules = "tsrules" : { $tsrule + }
```

; The traffic steering rule

```
$tsrule = // : {
    $ts-rule-name,
    $precedence ?,
    ( $flow-information | $tdf-application-identifier ),
    ( $ts-policy-identifier-ul | $ts-policy-identifier-dl | ( $ts-policy-identifier-ul, $ts-policy-identifier-dl ) )
}
```

; The name of the traffic steering rule in the TSSF

```
$ts-rule-name = "ts-rule-name" : string
```

; The precedence of the traffic steering rule. From 0 to 4294967295.

```
$precedence = "precedence" : 0..4294967295
```

; The flow traffic information

```
$flow-information = "flow-information" : [
    {
        $flow-description ?,
        $tos-traffic-class ?,
        $security-parameter-index ?,
        $flow-label ?,
        $flow-direction
    }
]
```

; The IP filter rule for a service data flow

```
$flow-description = "flow-description" : string
```

; The ToS traffic class is encoded on two octets, and represented in hex string display.

```
$tos-traffic-class = "tos-traffic-class" : /^[0-9A-Fa-f]{4}$/
```

; The security parameter index is encoded on four octets, and represented in hex string display.

```
$security-parameter-index = "security-parameter-index" : /^[0-9A-Fa-f]{8}$/
```

; The flow label is encoded on three octets, and represented in hex string display.

\$flow-label = "flow-label" : /^[0-9A-Fa-f]{6}\$/

; The direction of the flow traffic

\$flow-direction = "flow-direction" : ( "BIDIRECTIONAL" | "UPLINK" | "DOWNLINK" )

; The detected application traffic id in the TSSF

\$tdf-application-identifier = "tdf-application-identifier" : string

; The traffic steering policy id for uplink data

\$ts-policy-identifier-ul = "ts-policy-identifier-ul" : string

; The traffic steering policy id for downlink data

\$ts-policy-identifier-dl = "ts-policy-identifier-dl" : string

; The predefined traffic steering rules

\$predefined-tsrules = "predefined-tsrules" : { \$predefined-tsrule + }

; The predefined traffic steering rule

\$predefined-tsrule = // : { \$ts-rule-name }

; The predefined traffic steering group rules

\$predefined-group-of-tsrules = "predefined-group-of-tsrules" : { \$predefined-group-of-tsrule + }

; The predefined traffic steering group rule

\$predefined-group-of-tsrule = // : { \$ts-rule-base-name }

; The name of the traffic steering group rule in the TSSF

\$ts-rule-base-name = "ts-rule-base-name" : string

---

## B.2 Error and Informational response schema

This subclause defines the JSON schema for the body of HTTP responses in case of errors or success. The schema is based on IETF draft-newton-json-content-rules [16] and is defined below:

# jcr-version 0.7

```
# ruleset-id 3gpp.stapplication.info
# import 3gpp.stapplication.report as report
; A JCR for the error/successful response body
; Errors information
$errors-root = @ {root} { $errors }

; Success information
$success-root = @ {root} {
    $success-message,
    $success-path ?,
    $success-info ?
}

; Resource fields definitions
; The list of errors returned in responses sent by the PCRF or TSSF
$errors = "errors" : [
    {
        $error-type,
        $error-message,
        $error-tag ?,
        $error-path ?,
        $error-info ?
    } +
]

; The error type for an error. It can be one of 'application', 'interface', 'server' and 'other'.
$error-type = "error-type" : ( "application" | "interface" | "server" | "other" )

; The error text message
$error-message = "error-message" : string

; The error tag for a specific error
$error-tag = "error-tag" : string
```



; A JSON pointer path to the error resource

```
$error-path = "error-path" : string
```

; Any additional information for the error

```
$error-info = "error-info" : {
```

```
  $report.ts-rule-reports ?,
```

```
  // : any *
```

```
}
```

; The successful text message

```
$success-message = "success-message" : string
```

; A JSON pointer path to the success resource

```
$success-path = "success-path" : string
```

; Any additional information for the success.

```
$success-info = "success-info" : { // : any * }
```

---

## B.3 Report schema

This clause defines the JSON schema for reports provided by the TSSF to the PCRF. The schema is based on IETF draft-newton-json-content-rules [16] and is defined below:

```
# jcr-version 0.7
```

```
# ruleset-id 3gpp.stapplication.report
```

; A JCR for the notification body

; Report fields definitions

; The list of ts rule reports sent to the PCRF

```
$ts-rule-reports = "ts-rule-reports" : [
```

```
{
```

```
  $resource-paths,
```

```
  $rule-status,
```

```
  $rule-failure-code
```

```
} +
```

```
]
```

; The list of JSON pointers to the resources

```
$resource-paths = "resource-paths" : [ string + ]
```

```
; The status for a traffic steering rule. It can be 'INACTIVE'.
```

```
$rule-status = "rule-status" : $rule-status-type
```

```
; The string format for the traffic steering rule status
```

```
$rule-status-type = ("INACTIVE")
```

```
; The failure code for a traffic steering rule
```

```
$rule-failure-code = "rule-failure-code" : $rule-failure-code-type
```

```
; The string format for the traffic steering rule failure code
```

```
$rule-failure-code-type = (
  "UNKNOWN_RULE_NAME" |
  "GW/PCEF_MALFUNCTION" |
  "RESOURCES_LIMITATION" |
  "MISSING_FLOW_INFORMATION" |
  "INCORRECT_FLOW_INFORMATION" |
  "TDF_APPLICATION_IDENTIFIER_ERROR" |
  "FILTER_RESTRICTIONS" |
  "RESOURCE_ALLOCATION_FAILURE" |
  "RESOURCE_TIMEOUT" |
  "TS_POLICY_IDENTIFIER_ERROR" |
  "TS_POLICY_IDENTIFIER_DL_ERROR" |
  "TS_POLICY_IDENTIFIER_UL_ERROR"
)
```

---

## B.4 Notification schema

This clause defines the JSON schema for the body of the notification from the TSSF to the PCRF via the HTTP POST request. The schema is based on IETF draft-newton-json-content-rules [16] and is defined below:

```
# jcr-version 0.7
# ruleset-id 3gpp.stapplication.notification
# import 3gpp.stapplication.report as report
; A JCR for the notification body
$notification-root = @{root} { $notifications }
```

; Resource fields definitions

; The list of notifications notified to the PCRF

```
$notifications = "notifications" : [  
  {  
    $notification-type,  
    $notification-message,  
    $notification-tag ?,  
    $notification-info ?  
  } +  
]
```

; The notification type for a notification. It can be one of 'application' or 'other'.

```
$notification-type = "notification-type" : ( "application" | "other" )
```

; The notification text message

```
$notification-message = "notification-message" : string
```

; The notification tag for a specific notification

```
$notification-tag = "notification-tag" : string
```

; Any additional information for the notification

```
$notification-info = "notification-info" : { $report.ts-rule-reports ?, // : any * }
```

## Annex C (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
2015-08	CT3#82				TS skeleton of St reference point stage 3.		0.0.0
2015-08	CT3#82	C3-153467			Inclusion of documents agreed in CT3#82. C3-153291, C3-153416, C3-153444, C3-153455, C3-153445, C3-153420, C3-153439, C3-153290. Added reference to IETF RFC 1786: "Uniform Resource Locators (URL)."	0.0.0	0.1.0
2015-10	CT3#82-Bis	C3-154345			Inclusion of documents agreed in CT3#82bis. C3-154231, C3-154049, C3-154231, C3-154217, C3-154218, C3-154319, C3-154139	0.1.0	0.2.0
2015-11	CT3#83	C3-155420			Inclusion of documents agreed in CT3#83. C3-155227, C3-155229, C3-155365, C3-155235, C3-155284	0.2.0	1.0.0
2015-11	CT#70	CP-150824			TS presented for presentation and approval	1.0.0	1.0.1
2015-12	CT#70	CP-150824			TS 29.155 upgraded to 13.0.0 after approval	1.0.1	13.0.0
Change history							
Date	TSG #	TSG Doc.	CR	Rev	Cat	Subject/Comment	New
2016-03	CT#71	CP-160091	0001	2	B	Traffic Steering Rule Error handling over the REST based St interface	13.1.0
2016-03	CT#71	CP-160091	0002	1	F	Miscellaneous corrections related to the REST based St	13.1.0
2016-03	CT#71	CP-160091	0003	3	B	Updates to JSON content rules for the notification procedures from the TSSF to the PCRF	13.1.0
2016-03	CT#71	CP-160091	0004	1	B	JSON content rules updates for error responses with traffic steering rule reports over the REST based St interface	13.1.0
2016-03	CT#71	CP-160091	0005	3	B	Traffic steering rule report JSON Content Rules over St	13.1.0
2016-12	CT#74	CP-160619	0008	4	F	Overlapping IP Addresses with FMSS	13.2.0
2016-12	CT#74	CP-160619	0009	3	F	Support of deployments with both PCEF/TDF and TSSF	13.2.0
2016-12	CT#74	CP-160619	0010	1	F	Correction of IETF newton json content rule draft version and relevant JCR schemas	13.2.0
2016-12	CT#74	CP-160619	0013	2	F	UE IPv4 address provisioning	13.2.0
2016-12	CT#74	CP-160616	0012	1	F	Diameter base protocol specification update	14.0.0
2017-03	CT#75	CP-170076	0014	1	F	Correction of Diameter base protocol specification update	14.1.0
2017-06	CT#76	CP-171133	0015	1	D	Remove the Void reference	14.2.0
2017-06	CT#76	CP-171120	0017	1	A	Correction of IETF newton json content rule draft version	14.2.0

---

# History

<b>Document history</b>		
V14.1.0	April 2017	Publication
V14.2.0	July 2017	Publication