**3GPP TSG-CT WG4 Meeting #98eC4-203314**

**E-Meeting, 02nd – 12th June 2020**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.0* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **29.510** | **CR** | **0361** | **rev** | **-** | **Current version:** | **16.3.0** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | NRF Notification Data | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Ericsson | | | | | | | | | |
| ***Source to TSG:*** | CT4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | SBIProtoc16 | | | | |  | ***Date:*** | | | 2020-05-16 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **B** |  | | | | | ***Release:*** | | | Rel-16 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) Rel-12 (Release 12)* *Rel-13 (Release 13) Rel-14 (Release 14) Rel-15 (Release 15) Rel-16 (Release 16)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | When the NF Profile of an NF Instance changes, it is not clear in the specification what is the expected behaviour of NRF when the NF status changes between the different states (REGISTERED - SUSPENDED - UNDISCOVERABLE). One possible interpretation could be that a change e.g. from SUSPEDED to REGISTERED results in NRF sending event type "NF\_REGISTERED", and another interpretation could be to send "NF\_PROFILE\_CHANGED".  It is proposed to keep an homogeneous approach across all different conditions, and notify "NF\_REGISTERED"/"NF\_DEREGISTERED" only when an NF is newly registered/deregistered into/from the NRF; then, in all cases of a profile change (including a change of status), the NRF notifies with "NF\_PROFILE\_CHANGED".  In addition, when an NF Instance starts or stops being part of a subscription condition to a set of NFs, the spec is already clear in that the event type should be NF\_PROFILE\_CHANGED, but it should be further clarified that:  - when an NF starts being part of the subscription condition, the full NF profile must be notified because the subscribing NF might not have any prior data for such NF Instance (from a potential previous discovery response)  - it is helpful for the consumer to get a clear indication of whetehr the notification if received as a result of the NF instance to start or stop being part of a subscription condition (otherwise, it must keep track of the subscription condition used, and analyze the received profile data in the notification to determine what did really happen, as the real reason for the notification). | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | - Clarify that changes of NF status always result in sending NF\_PROFILE\_CHANGED as event type.  - Add a new attribute in NotificationData, so the NRF can indicate explicitly whether the notification is due to the new NF Profile to start or stop being part of an existing subscription condition to a set of NFs. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Unclear and non-optimal NRF behaviour. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 5.2.2.6.2, 6.1.6.2.17, 6.1.6.3.xx (new), A.2 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | | This CR introduces backwards compatible corrections, with impacts on the following APIs:  - TS29510\_Nnrf\_NFManagement.yaml | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

\* \* \* First Change \* \* \* \*

##### 5.2.2.6.2 Notification from NRF in the same PLMN

The operation is invoked by issuing a POST request to each callback URI of the different subscribed NF Instances.



Figure 5.2.2.6.2-1: Notification from NRF in the same PLMN

1. The NRF shall send a POST request to the callback URI.

For notifications of newly registered NF Instances, the request body shall include the data associated to the newly registered NF, and its services, according to the criteria indicated by the NF Service Consumer during the subscription operation. These data shall contain the NFInstanceID of the NF Instance, an indication of the event being notified ("registration"), and the new profile data (including, among others, the services offered by the NF Instance).

For notifications of changes of the profile of a NF Instance, the request body shall include the NFInstancceID of the NF Instance whose profile was changed, an indication of the event being notified ("profile change"), and the new profile data.

For notifications of deregistration of the NF Instance from NRF, the request body shall include the NFInstanceID of the deregistered NF Instance, and an indication of the event being notified ("deregistration").

When an NF Service Consumer subscribes to a set of NFs (using the different subscription conditions specified in clause 6.1.6.2.35), a change in the profile of the monitored NF Instance may result in such NF becoming a part of the NF set, or stops becoming a part of it (e.g., an NF Service Consumer subscribing to all NFs offering a given NF Service, and then, a certain NF Instance changes its profile by adding or removing an NF Service of its NF Profile); in such case, the NRF shall use the "NF\_PROFILE\_CHANGED" event type in the notification. Similarly, a change of the status (i.e. the "nfStatus" attribute of the NF Profile) shall result into the NRF to send notifications to subscribing NFs with event type set to "NF\_PROFILE\_CHANGED".

When an NF Service Consumer subscribes to a set of NFs, using the subscription conditions specified in clause 6.1.6.2.35, in case of a change of profile(s) of NFs potentially related to those subscription conditions, the NRF shall send notification to subscribing NF Service Consumer(s) to those NFs no longer matching the subscription conditions, and to subscribing NF Service Consumer(s) to NFs that start matching the subscription conditions. In that case, the NRF indicates in the notification data whether the notification is due to the NF Instance to newly start or stop matching the subscription condition (i.e. based on the presence of the "conditionEvent" attribute of the NotificationData).

2a. On success, "204 No content" shall be returned by the NF Service Consumer.

2b. If the NF Service Consumer does not consider the "nfStatusNotificationUri" as a valid notification URI (e.g., because the URI does not belong to any of the existing subscriptions created by the NF Service Consumer in the NRF), the NF Service Consumer shall return "404 Not Found" status code with the ProblemDetails IE providing details of the error.

\* \* \* Next Change \* \* \* \*

##### 6.1.6.2.17 Type: NotificationData

Table 6.1.6.2.17-1: Definition of type NotificationData

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute name | Data type | P | Cardinality | Description |
| event | NotificationEventType | M | 1 | Notification type. It shall take the values "NF\_REGISTERED", "NF\_DEREGISTERED" or "NF\_PROFILE\_CHANGED". |
| nfInstanceUri | Uri | M | 1 | Uri of the NF Instance (see clause 6.1.3.3.2) associated to the notification event. |
| nfProfile | NFProfile | C | 0..1 | New NF Profile or Updated NF Profile; it shall be present when the notification type is "NF\_REGISTERED" and it may be present when the notificaiton type is "NF\_PROFILE\_CHANGED".  (NOTE X) |
| profileChanges | array(ChangeItem) | C | 1..N | List of changes on the profile of the NF Instance associated to the notification event; it may be present when the notification type is "NF\_PROFILE\_CHANGED" (see NOTE 1). |
| conditionEvent | ConditionEventType | C | 0..1 | Type of event indicating wether a change of NF Profile results in that the NF Instance starts or stops being part of a given set of NF Instances, as indicated in the subscription condition (see attribute "subscrCond" in clause 6.1.6.2.16).  It can take the value "NF\_ADDED" (if the NF Instance starts being part of a given set) or "NF\_REMOVED" (if the NF Instance stops being part of a given set).  (NOTE X) |
| NOTE 1: If "event" attribute takes the value "NF\_PROFILE\_CHANGED", then either "nfProfile" or "profileChanges" attributes shall be present, but not both.  NOTE X: When a change in an NF Profile results in an NF to start being part of a given set, the NRF shall indicate such condition by including the "conditionEvent" attribute with value "NF\_ADDED", and it shall include in the notification the "nfProfile" attribute with the full NF Profile of the NF Instance; the "profileChanges" attribute shall not be included. When a change in an NFProfile results in an NF to stop being part of a given set, the NRF shall indicate such condition by including the "conditionEvent" attribute with value "NF\_REMOVED", and both attributes "nfProfile" and "profileChanges" shall be absent. | | | | |

EXAMPLE: Notification payload sent from NRF when an NF Instance has changed its profile by updating the value of the "recoveryTime" attribute of its NF Profile, and updated the TCP port of the first NF Service Instance:

{

"event": "NF\_PROFILE\_CHANGED",

"nfInstanceUri": ".../nf-instances/4947a69a-f61b-4bc1-b9da-47c9c5d14b64",

"profileChanges": [

{

"op": "REPLACE",

"path": "/recoveryTime",

"newValue": "2018-12-30T23:20:50Z"

},

{

"op": "REPLACE",

"path": "/nfServices/0/ipEndPoints/0/port",

"newValue": 8080

}

]

}

\* \* \* Next Change \* \* \* \*

##### 6.1.6.3.xx Enumeration: ConditionEventType

Table 6.1.6.3.xx-1: Enumeration ConditionEventType

|  |  |
| --- | --- |
| Enumeration value | Description |
| "NF\_ADDED" | The NF Instance notified by NRF starts being part of a condition for a subscription on a set of NFs. |
| "NF\_REMOVED" | The NF Instance notified by NRF stops being part of a condition for a subscription on a set of NFs. |

\* \* \* Next Change \* \* \* \*

## A.2 Nnrf\_NFManagement API

***(… text not shown for clarity …)***

NotificationData:

description: Data sent in notifications from NRF to subscribed NF Instances

type: object

required:

- event

- nfInstanceUri

allOf:

#

# Condition: If 'event' takes value 'NF\_PROFILE\_CHANGED',

# then either 'nfProfile' or 'profileChanges' (but not both) must be present

#

- anyOf:

- not:

properties:

event:

type: string

enum:

- NF\_PROFILE\_CHANGED

- oneOf:

- required: [ nfProfile ]

- required: [ profileChanges ]

#

# Condition: If 'event' takes value 'NF\_REGISTERED',

# then 'nfProfile' must be present

#

- anyOf:

- not:

properties:

event:

type: string

enum:

- NF\_REGISTERED

- required: [ nfProfile ]

properties:

event:

$ref: '#/components/schemas/NotificationEventType'

nfInstanceUri:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/Uri'

nfProfile:

allOf:

- $ref: '#/components/schemas/NFProfile'

- not:

required: [ interPlmnFqdn ]

- not:

required: [ allowedPlmns ]

- not:

required: [ allowedNfTypes ]

- not:

required: [ allowedNfDomains ]

- not:

required: [ allowedNssais ]

- properties:

nfServices:

type: array

items:

allOf:

- $ref: '#/components/schemas/NFService'

- not:

required: [ interPlmnFqdn ]

- not:

required: [ allowedPlmns ]

- not:

required: [ allowedNfTypes ]

- not:

required: [ allowedNfDomains ]

- not:

required: [ allowedNssais ]

profileChanges:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/ChangeItem'

minItems: 1

conditionEvent:

$ref: '#/components/schemas/ConditionEventType'

***(… text not shown for clarity …)***

UdsfInfo:

type: object

properties:

groupId:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/NfGroupId'

supiRanges:

type: array

items:

$ref: '#/components/schemas/SupiRange'

minItems: 1

ConditionEventType:

description: Indicates whether a notification is due to the NF Instance to start or stop being part of a condition for a subscription to a set of NFs

anyOf:

- type: string

enum:

- NF\_ADDED

- NF\_REMOVED

- type: string

\* \* \* End of Changes \* \* \* \*