**3GPP TSG-CT WG4 Meeting #111-eC4-224**

**E-Meeting, 18th – 26th August 2022 revision of C4-224229**

|  |
| --- |
| *CR-Form-v12.2* |
| **CHANGE REQUEST** |
|  |
|  | **29.510** | **CR** | **0746** | **rev** | **1** | **Current version:** | **17.6.0** |  |
|  |
| *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:***  | No Profile Match information |
|  |  |
| ***Source to WG:*** | Nokia, Nokia Shanghai Bell, Verizon |
| ***Source to TSG:*** | CT4 |
|  |  |
| ***Work item code:*** | SBIProtoc17 |  | ***Date:*** | 2022-08-21 |
|  |  |  |  |  |
| ***Category:*** | F |  | ***Release:*** | Rel-17 |
|  | *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 canbe 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-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19)* |
|  |  |
| ***Reason for change:*** | 1) Consumers of the NRF may cache NF discovery results. However, in cases where various query parameters have been sent in the discovery request and an empty array of nfInstances is received in the response (indicating that no NF instance matched all the query parameters), the cache may be of limited use. E.g. the discovery request contains queryParam1=x and queryParam2=y and searchResult is empty (because no match for queryParam1=x was found), a subsequent request with queryParam1=x and queryParam2=z cannot make use of the cache. The cache can only indicate that for the combination of queryParam1=x and queryParam2=y no match was found.2) Furthermore, a 200 OK with an empty payload does not convey any specific details for not finding any NF profile matching the search filter criteria. For instance, when the UE attempts to authenticate to 5GC through the AMF, the AUSF requests to discover a UDM serving the subscriber. When receiving a discovery response with an empty payload, the AUSF cannot know whether this is due to e.g. the subscriber having no 5GS subscription (i.e. not found in any UDM), to the requester NF not being allowed to discover the UDM (e.g. requester PLMN without roaming agreement with the HPLMN), some temporary failure of the UDM (in SUSPENDED or NOT\_DISCOVERABLE state), etc. This prevents the AUSF to return a suitable error to the AMF (e.g. 404 User-Not-Found if the subscriber was not found) and the AMF to return the proper NAS cause code to the UE, e.g. to force the UE to LTE.  |
|  |  |
| ***Summary of change:*** | The SearchResult is extended with a new noProfileMatchInfo attribute that enables to provide reason and details for not finding an NF profile matching the search criteria. This includes the option to indicate query parameter combinations for which no nfInstance of the target-nf-type has registered. This information may be cached by the NRF consumer for subsequent use. |
|  |  |
| ***Consequences if not approved:*** | The requester of an NF Discovery request cannot determine the reason why no NF profile could be found matching the search filter criteria. This prevents the network to adapt its behavior (e.g. setting NAS cause code to the UE) accordingly. Unnecessary discovery requests (with empty response) are conveyed. |
|  |  |
| ***Clauses affected:*** | 5.3.2.2.2, 6.2.6.1, 6.2.6.2.2, 6.2.6.2.ww (new), 6.2.6.2.xx (new), 6.2.6.2.yy (new), 6.2.6.3.zz (new), A.3 |
|  |  |
|  | **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 impact to the following APIs:TS29510\_Nnrf\_NFDiscovery.yaml |
|  |  |
| ***This CR's revision history:*** |  |

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

##### 5.3.2.2.2 Service Discovery in the same PLMN

This service operation is executed by querying the "nf-instances" resource. The request is sent to an NRF in the same PLMN of the NF Service Consumer.



Figure 5.3.2.2.2-1: Service Discovery Request in the same PLMN

1. The NF Service Consumer shall send an HTTP GET request to the resource URI "nf-instances" collection resource. The input filter criteria for the discovery request shall be included in query parameters.

2a. On success, "200 OK" shall be returned. The response body shall contain a validity period, during which the search result can be cached by the NF Service Consumer, and an array of NF Profile objects, and/or a map of NFInstanceInfo objects of NF instances (if the NF service consumer indicated support of the Enh-NF-Discovery feature in the request) that satisfy the search filter criteria (e.g., all NF Instances offering a certain NF Service name in REGISTERED status, or empty array in case search filter criteria do not match a NF Instance in REGISTERED status). In the latter case, the response may include the noProfileMatchInfo attribute to provide the specific reason for not finding any NF instance that can match the search filter criteria.

2b. On failure or redirection:

- If the NF Service Consumer is not allowed to discover the NF services for the requested NF type provided in the query parameters, the NRF shall return "403 Forbidden" response.

- If the discovery request fails at the NRF due to errors in the input data in the URI query parameters, the NRF shall return "400 Bad Request" status code with the ProblemDetails IE providing details of the error.

- If the discovery request fails at the NRF due to NRF internal errors, the NRF shall return "500 Internal Server Error" status code with the ProblemDetails IE providing details of the error.

- In the case of redirection, the NRF shall return 3xx status code, which shall contain a Location header with an URI pointing to the endpoint of another NRF service instance.

The NF Profile objects returned in a successful result shall contain generic data of each NF Instance, applicable to any NF type, and it may also contain NF-specific data, for those NF Instances belonging to a specific type (e.g., the attribute "udrInfo" is typically present in the NF Profile when the type of the NF Instance takes the value "UDR"). In addition, the attribute "customInfo", may be present in the NF Profile for those NF Instances with custom NF types.

For those NF Instances, the "customInfo" attribute shall be returned by NRF, if available, as part of the NF Profiles returned in the discovery response.

The NRF shall also include, in the returned NF Profile objects, the Vendor-Specific attributes (see 3GPP TS 29.500 [4], clause 6.6.3) that may have been provided by the registered NF Instances.

If the response includes a map of NFInstanceInfo objects of NF instances, the NF Service Consumer may retrieve the NF profiles by issuing a service discovery request with the target-nf-instance-id parameter identifying the target NF Instance ID; the service discovery request shall also include the nrf-disc-uri parameter set to the API URI of the Nnrf\_NFDiscovery service of the NRF holding the NF profile, if the nrfDiscApiUri attribute was received in the NFInstanceInfo object and if the service discovery request is addressed to a different NRF than the NRF holding the NF profile.

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

#### 6.2.6.1 General

This clause specifies the application data model supported by the API.

Table 6.2.6.1-1 specifies the data types defined for the Nnrf service based interface protocol.

Table 6.2.6.1-1: Nnrf\_NFDiscovery specific Data Types

|  |  |  |
| --- | --- | --- |
| Data type | Clause defined | Description |
| SearchResult | 6.2.6.2.2 | Contains the list of NF Profiles returned in a Discovery response. |
| NFProfile | 6.2.6.2.3 | Information of an NF Instance discovered by the NRF. |
| NFService | 6.2.6.2.4 | Information of a given NF Service Instance; it is part of the NFProfile of an NF Instance discovered by the NRF. |
| StoredSearchResult | 6.2.6.2.5 | Contains a complete search result (i.e. a number of discovered NF Instances), stored by NRF as a consequence of a prior search result. |
| PreferredSearch | 6.2.6.2.6 | Contains information on whether the returned NFProfiles match the preferred query parameters. |
| NfInstanceInfo | 6.2.6.2.7 | Contains information on an NF profile matching a discovery request. |
| ScpDomainRoutingInfo | 6.2.6.2.8 | SCP Domain Routing Information |
| ScpDomainConnectivity | 6.2.6.2.9 | SCP Domain Routing Information |
| ScpDomainRoutingInfoSubscription | 6.2.6.2.10 | SCP Domain Routing Information Subscription  |
| ScpDomainRoutingInfoNotification | 6.2.6.2.11 | Notification for SCP Domain Routing Information Update |
| NoProfileMatchInfo | 6.2.6.2.ww | No Profile Matching information |
| QueryParamCombination | 6.2.6.2.xx | Contains a list of query parameters |
| QueryParameter | 6.2.6.2.yy | Contains name and value of a query parameter |

Table 6.2.6.1-2 specifies data types re-used by the Nnrf\_NFDiscovery service-based interface protocol from other specifications, including a reference to their respective specifications and when needed, a short description of their use within the Nnrf\_NFDiscovery service-based interface.

Table 6.2.6.1-2: Nnrf\_NFDiscovery re-used Data Types

|  |  |  |
| --- | --- | --- |
| Data type | Reference | Comments |
| Snssai | 3GPP TS 29.571 [7] |  |
| PlmnId | 3GPP TS 29.571 [7] |  |
| Dnn | 3GPP TS 29.571 [7] |  |
| Tai | 3GPP TS 29.571 [7] |  |
| SupportedFeatures | 3GPP TS 29.571 [7] |  |
| NfInstanceId | 3GPP TS 29.571 [7] |  |
| Uri | 3GPP TS 29.571 [7] |  |
| Gpsi | 3GPP TS 29.571 [7] |  |
| GroupId | 3GPP TS 29.571 [7] |  |
| Guami | 3GPP TS 29.571 [7] |  |
| IPv4Addr | 3GPP TS 29.571 [7] |  |
| IPv6Addr | 3GPP TS 29.571 [7] |  |
| UriScheme | 3GPP TS 29.571 [7] |  |
| Dnai | 3GPP TS 29.571 [7] |  |
| NfGroupId | 3GPP TS 29.571 [7] | Identifier of a NF Group |
| PduSessionType | 3GPP TS 29.571 [7] |  |
| AtsssCapability | 3GPP TS 29.571 [7] |  |
| PlmnIdNid | 3GPP TS 29.571 [7] |  |
| NfSetId | 3GPP TS 29.571 [7] |  |
| NfServiceSetId | 3GPP TS 29.571 [7] |  |
| ExtSnssai | 3GPP TS 29.571 [7] |  |
| DurationSec | 3GPP TS 29.571 [7] |  |
| RedirectResponse | 3GPP TS 29.571 [7] | Response body of the redirect response message. |
| MbsSessionId | 3GPP TS 29.571 [7] | MBS Session Identifier |
| IpAddr | 3GPP TS 29.571 [7] | IP Address |
| AreaSessionId | 3GPP TS 29.571 [7] | Area Session Identifier used for an MBS session with location dependent content |
| Fqdn | 3GPP TS 29.571 [7] | Fully Qualified Domain Name |
| EventId | 3GPP TS 29.520 [32] | Defined in Nnwdaf\_AnalyticsInfo API. |
| NwdafEvent | 3GPP TS 29.520 [32] | Defined in Nnwdaf\_EventsSubscription API. |
| ExtGroupId | 3GPP TS 29.503 [36] |  |
| SharedDataId | 3GPP TS 29.503 [36] |  |
| ExternalClientType | 3GPP TS 29.572 [33] |  |
| SupportedGADShapes | 3GPP TS 29.572 [33] | Supported GAD Shapes |
| DefaultNotificationSubscription | 3GPP TS 29.510 | See clause 6.1.6.2.4 |
| IPEndPoint | 3GPP TS 29.510 | See clause 6.1.6.2.5 |
| NFType | 3GPP TS 29.510 | See clause 6.1.6.3.3 |
| UdrInfo | 3GPP TS 29.510 | See clause 6.1.6.2.6 |
| UdmInfo | 3GPP TS 29.510 | See clause 6.1.6.2.7 |
| AusfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.8 |
| SupiRange | 3GPP TS 29.510 | See clause 6.1.6.2.9 |
| AmfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.11 |
| SmfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.12 |
| UpfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.13 |
| PcfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.20 |
| BsfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.21 |
| ChfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.32 |
| NFServiceVersion | 3GPP TS 29.510 | See clause 6.1.6.2.19 |
| PlmnSnssai | 3GPP TS 29.510 | See clause 6.1.6.2.44 |
| NwdafInfo | 3GPP TS 29.510 | See clause 6.1.6.2.45 |
| NFStatus | 3GPP TS 29.510 | See clause 6.1.6.3.7 |
| DataSetId | 3GPP TS 29.510 | See clause 6.1.6.3.8 |
| ServiceName | 3GPP TS 29.510 | See clause 6.1.6.3.11 |
| NFServiceStatus | 3GPP TS 29.510 | See clause 6.1.6.3.12 |
| LmfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.46 |
| GmlcInfo | 3GPP TS 29.510 | See clause 6.1.6.2.47 |
| NefInfo | 3GPP TS 29.510 | See clause 6.1.6.2.48 |
| PfdData | 3GPP TS 29.510 | See clause 6.1.6.2.49 |
| AfEventExposureData | 3GPP TS 29.510 | See clause 6.1.6.2.50 |
| PcscfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.53 |
| HssInfo | 3GPP TS 29.510 | See clause 6.1.6.2.57 |
| ImsiRange | 3GPP TS 29.510 | See clause 6.1.6.2.58 |
| VendorSpecificFeature | 3GPP TS 29.510 | See clause 6.1.6.2.62 |
| ScpInfo | 3GPP TS 29.510 | See clause 6.1.6.2.65 |
| NefId | 3GPP TS 29.510 | See clause 6.1.6.3 |
| VendorId | 3GPP TS 29.510 | See clause 6.1.6.3 |
| AnNodeType | 3GPP TS 29.510 | See clause 6.1.6.3.13 |
| SuciInfo | 3GPP TS 29.510 | See clause 6.1.6.2.71 |
| SeppInfo | 3GPP TS 29.510 | See clause 6.1.6.2.72 |
| NsacfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.81 |
| NsacfCapability | 3GPP TS 29.510 | See clause 6.1.6.2.82 |
| MlAnalyticsInfo | 3GPP TS 29.510 | See clause 6.1.6.2.84 |
| MbSmfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.85 |
| TsctsfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.91 |
| MbUpfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.94 |
| TrustAfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.96 |
| CollocatedNfInstance | 3GPP TS 29.510 | See clause 6.1.6.2.99 |
| NssaafInfo | 3GPP TS 29.510 | See clause 6.1.6.2.104 |
| IwmscInfo | 3GPP TS 29.510 | See clause 6.1.6.2.108 |
| MnpfInfo | 3GPP TS 29.510 | See clause 6.1.6.2.109 |

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

##### 6.2.6.2.2 Type: SearchResult

Table 6.2.6.2.2-1: Definition of type SearchResult

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute name | Data type | P | Cardinality | Description |
| validityPeriod | integer | M | 1 | It shall contain the time in seconds during which the discovery result is considered valid and can be cached by the NF Service Consumer. This value shall be the same as the value contained in the "max-age" parameter of the "Cache-Control" header field sent in the HTTP response. |
| nfInstances | array(NFProfile) | M | 0..N | It shall contain an array of NF Instance profiles, matching the search criteria indicated by the query parameters of the discovery request. If the nfInstancesList IE is absent, an empty array means there is no NF instance that can match the search criteria. |
| searchId | string | O | 0..1 | This IE may be present if the NRF stores the result of the current service discovery response in a given URL (server-side caching), to make it available in the future to NF Service Consumers without having to compute the whole search process again. |
| numNfInstComplete | Uint32 | O | 0..1 | This IE may be present when the total number of NF Instances found by NRF, as the result of the service discovery process, is higher than the actual number of NF Instances included in the attribute nfInstances of the SearchResult object. This may happen due to the NF Service Consumer including in the discovery request parameters such as "limit" or "max-payload-size". |
| preferredSearch | PreferredSearch | C | 0..1 | This IE shall be present to indicate whether all the returned NFProfiles match the preferred query parameters, if the discovery request contain any of the query parameter defined in the PreferredSearch data type. |
| nrfSupportedFeatures | SupportedFeatures | C | 0..1 | Features supported by the NRF for the NFDiscovery service (see clause 6.2.9).This IE should be present if the NRF supports at least one feature. |
| nfInstanceList | map(NfInstanceInfo) | O | 1..N | This IE may be present if the NF Discovery request indicated support of the Enh-NF-Discovery feature.When present, this IE shall contain a map of NfInstanceInfo of NF instance profiles matching the search criteria indicated by the query parameters of the discovery request. The key of the map shall be the NF instance ID.  |
| alteredPriorityInd | boolean | O | 0..1 | This IE shall indicate whether the NRF altered the priority values returned in the search result or not. (NOTE)When present, this IE shall be set as following:- true: NF instances with NRF altered priority are returned in the search result.- false: the NRF has not altered priority values in any NF instance returned in the search result  |
| noProfileMatchInfo | NoProfileMatchInfo | O | 0..1 | This IE may be present if an empty array of nfInstances is conveyed and the nfInstancesList IE is absent; otherwise it shall be absent. If present, it shall indicate the specific reason for not finding any NF instance that can match the search criteria. |
| NOTE: If NRF altered priority values are returned in the search result, when the NF consumer receives a different priority value in a subsequent NF Profile change notification for NF instance(s) returned in the search result, the NF consumer should not overwrite the NRF altered priority in the cached search result. |

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

##### 6.2.6.2.ww Type: NoProfileMatchInfo

Table 6.2.6.2.ww-1: Definition of type NoProfileMatchInfo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute name | Data type | P | Cardinality | Description |
| reason | NoProfileMatchReason | M | 1 | This IE shall indicate the specific reason for not finding any NF instance that can match the search criteria.  |
| queryParamCombinationList | array(QueryParamCombination) | C | 1..N | This IE shall be present if the reason IE is set to "QUERY\_PARAMS\_COMBINATION\_NO\_MATCH". It may be present otherwise.If present each QueryParamCombination indicates that no NF Instance matching the QueryParamCombination has registered. |

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

##### 6.2.6.2.xx Type: QueryParamCombination

Table 6.2.6.2.xx-1: Definition of type QueryParamCombination

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute name | Data type | P | Cardinality | Description |
| queryParams | array(QueryParameter) | M | 1..N | This IE shall contain a list of query parameters. |

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

##### 6.2.6.2.yy Type: QueryParameter

Table 6.2.6.2.yy-1: Definition of type QueryParameter

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute name | Data type | P | Cardinality | Description |
| name | string | M | 1 | name of the query parameter |
| value | string | M | 1 | value of the query parameter |

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

##### 6.2.6.3.zz Enumeration: NoProfileMatchReason

The enumeration NoProfileMatchReason indicates the specific reason for not finding any NF instance that can match the search criteria.

Table 6.2.6.3.zz-1: Enumeration NoProfileMatchReason

|  |  |
| --- | --- |
| Enumeration value | Description |
| "REQUESTER\_PLMN\_NOT\_ALLOWED" | NF profiles are not allowed to be discovered by the requester's PLMN |
| "TARGET\_NF\_SUSPENDED" | Target NF exists with NFStatus or NFServiceStatus "SUSPENDED"  |
| "TARGET\_NF\_UNDISCOVERABLE" | Target NF exists with NFStatus or NFServiceStatus "UNDISCOVERABLE"  |
| "QUERY\_PARAMS\_COMBINATION\_NO\_MATCH" | No NF instance matching the Query Parameter Combination has registered |
| "UNSPECIFIED" | Other reasons |

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

# A.3 Nnrf\_NFDiscovery API

openapi: 3.0.0

\*\*\*\*\*\*\*\*text not shown for clarity\*\*\*\*\*\*\*\*\*\*\*\*\*

[…]

 SearchResult:

 description: Contains the list of NF Profiles returned in a Discovery response

 type: object

 required:

 - nfInstances

 properties:

 validityPeriod:

 type: integer

 nfInstances:

 type: array

 items:

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

 searchId:

 type: string

 numNfInstComplete:

 $ref: 'TS29571\_CommonData.yaml#/components/schemas/Uint32'

 preferredSearch:

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

 nrfSupportedFeatures:

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

 nfInstanceList:

 description: List of matching NF instances. The key of the map is the NF instance ID.

 type: object

 additionalProperties:

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

 minProperties: 1

 alteredPriorityInd:

 type: boolean

 noProfileMatchInfo:

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

[…]

 NoProfileMatchInfo:

 description: Provides the reason for not finding NF matching the search criteria

 type: object

 required:

 - reason

 properties:

 reason:

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

 queryParamCombinationList:

 type: array

 items:

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

 minItems: 1

 QueryParamCombination:

 description: Contains a list of Query Parameters

 type: object

 required:

 - queryParams

 properties:

 queryParams:

 type: array

 items:

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

 minItems: 1

 QueryParameter:

 description: Contains the name and value of a query parameter

 type: object

 required:

 - name

 - value

 properties:

 name:

 type: string

 value:

 type: string

 NoProfileMatchReason:

 description: No Profile Match Reason

 anyOf:

 - type: string

 enum:

 - REQUESTER\_PLMN\_NOT\_ALLOWED

 - TARGET\_NF\_SUSPENDED

 - TARGET\_NF\_UNDISCOVERABLE

 - QUERY\_PARAMS\_COMBINATION\_NO\_MATCH

 - UNSPECIFIED

 - type: string

[…]

\*\*\*\*\*\*\*\*text not shown for clarity\*\*\*\*\*\*\*\*\*\*\*\*\*

\* \* \* End of Change \* \* \* \*