|  |
| --- |
| 3GPP TR 33.875 V0.4.0 (2021-08) |
| Technical Report  |
| 3rd Generation Partnership Project;Technical Specification Group Services and System Aspects;Study on enhanced security aspects of the 5G Service Based Architecture (SBA);(Release 17) |
|   |
|  |  |
|  |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP.The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented.This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification.Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. |

|  |
| --- |
|  |
| ***3GPP***Postal address3GPP support office address650 Route des Lucioles - Sophia AntipolisValbonne - FRANCETel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16Internet |
| ***Copyright Notification***No part may be reproduced except as authorized by written permission.The copyright and the foregoing restriction extend to reproduction in all media.© 2021, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).All rights reserved.UMTS™ is a Trade Mark of ETSI registered for the benefit of its members3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational PartnersLTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational PartnersGSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 6

Introduction 7

1 Scope 8

2 References 8

3 Definitions of terms, symbols and abbreviations 8

3.1 Terms 8

3.2 Symbols 9

3.3 Abbreviations 9

4 Trust model 9

4.0 General 9

4.1 Actors 9

4.2 Deployment options 10

4.3 Description of the trust assumptions 10

4.3.1 Trust within one PLMN 10

4.3.2 Trust in Inter-PLMN communication 11

5 Key issues 12

5.1 Key issue #1: Authentication of NRF and NF Service Producer in indirect communication 12

5.1.1 Key issue details 12

5.1.2 Security threats 12

5.1.3 Potential security requirements 12

5.2 Key issue #2: SCP security domains 12

5.2.1 Key issue details 12

5.2.2 Security threats 13

5.2.3 Potential security requirements 13

5.3 Key Issue #3: Service access authorization in the "Subscribe-Notify" scenarios 13

5.3.1 Key issue details 13

5.3.2 Security threats 14

5.3.3 Potential security requirements 14

5.4 Key issue #4: Authorization of SCP to act on behalf of an NF or another SCP 15

5.4.1 Key issue details 15

5.4.2 Security threats 15

5.4.3 Potential security requirements 15

5.5 Key issue #5: End-to-end integrity protection of HTTP messages 15

5.5.1 Key issue details 15

5.5.2 Security threats 15

5.5.3 Potential security requirements 15

5.6 Key issue #6: Access token usage by all NFs of an NF set 16

5.6.1 Key issue details 16

5.6.2 Security threats 17

5.6.3 Potential security requirements 17

5.7 Key issue #7: Authorization mechanism determination 17

5.7.1 Key issue details 17

5.7.2 Security threats 17

5.7.3 Potential security requirements 17

5.8 Key issue #8: Service access authorization requirements in intra-PLMN scenarios for PLMN deploying multiple NRFs (in OAuth2.0 AS role) 18

5.8.1 Key issue details 18

5.8.1.1 Introduction 18

5.8.1.2 Hierarchical NRFs / Deployment model with local NRFs 18

5.8.1.2 Deployment model with NF Service Consumer directly accessing the NRF where the NF Service Producer is registered 19

5.8.2 Security threats 20

5.8.3 Potential security requirements 20

5.9 Key issue #9: Authorization for Inter-Slice Access 20

5.9.1 Key issue details 20

5.9.2 Security threats 20

5.X.3 Potential security requirements 20

5.X Key issue #X: <distinct KI name> 20

5.X.1 Key issue details 20

5.X.2 Security threats 20

5.X.3 Potential security requirements 20

6 Solutions 21

6.0 Mapping of solutions to key issues 21

6.1 Solution #1: Verification of the entity sending the service response in indirect communication without delegated discovery 21

6.1.1 Introduction 21

6.1.2 Solution details 22

6.1.3 Evaluation 23

6.2 Solution #2: Authorization between NFs and SCP 24

6.2.1 Introduction 24

6.2.2 Solution details 24

6.2.3 Evaluation 25

6.3 Solution #3: Using existing procedures for authorization of SCP to act on behalf of an NF Consumer 25

6.3.1 Introduction 25

6.3.2 Solution details 26

6.3.2.1 Request of access token on behalf of the consumer 26

6.3.2.2 Service request on behalf of the consumer 27

6.3.2.4 Protection of the NF consumer's CCA 27

6.3.3 Evaluation 28

6.4 Solution #4: Service request authenticity verification in indirect communication 28

6.4.1 Introduction 28

6.4.2 Solution details 28

6.4.3 Evaluation 29

6.5 Solution #5: End-to-end integrity protection of HTTP body and method 30

6.5.1 Introduction 30

6.5.2 Solution details 30

6.5.3 Evaluation 32

6.6 Solution #6: Verification of Service Response from a NF Service Producer at the expected NF Set 32

6.6.1 Introduction 32

6.6.2 Solution details 32

6.6.2.2 for indirect communication with delegated discovery 34

6.6.2.3 Client credentials assertion of NF Service Producer 35

6.6.3 Evaluation 35

6.7 Solution #7: Access token request for NF Set 36

6.7.1 Introduction 36

6.7.2 Solution details 36

6.7.3 Evaluation 37

6.8 Solution #8: integrity protection of HTTP message in consideration of update by SCP 37

6.8.1 Introduction 37

6.8.2 Solution Details 38

6.8.3 Evaluation 39

6.9 Solution #9: Authorization mechanism negotiation 39

6.9.1 Introduction 39

6.9.2 Solution details 39

6.9.3 Evaluation 40

6.10 Solution #10: NRF deployment clarifications 40

6.10.1 Introduction 40

6.10.2 Solution details 40

6.10.3 Evaluation 40

6.Y Solution #Y: <distinct solution name> 40

6.Y.1 Introduction 40

6.Y.2 Solution details 40

6.Y.3 Evaluation 40

7 Conclusions 41

7.X <distinct KI name> 41

Annex A (informative): Change history 42

**Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.Error! Bookmark not defined.**Error! Bookmark not defined.

# Foreword

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

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

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

Y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

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

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

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

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

**Should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

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

**Can** indicates that something is possible

**cannot** indicates that something is impossible

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

**Will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

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

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

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

In addition:

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

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

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

# Introduction

The 5G core network introduced a Service-Based Architecture (the so-called SBA). This brought fundamental impacts on the way new services are created and how the individual Network Functions (NF) communicate. A more open and adaptable system design necessitated to study different approaches to enforce the security requirements of 3GPP systems, whilst not impeding flexible service creation and future innovations. Along with these architectural challenges, SBA further introduced changes to the protocol stack and serialization format of the 5G core network.

The SBA was set on providing solutions for authentication and authorization in direct communication scenarios as well as the N32 security. Later on enhancements were introduced for indirect communication scenarios as well as the concept of Client Credential Assertion to allow NRF/NF Service Producer to directly authenticate a NF Service Consumer.

While the SBA provides a good level of security, several additional aspects have been identified that may bring new potential threats. This will be documented by the present document.

# 1 Scope

The present document studies enhanced security aspects of the 5G Service Based Architecture. It will analyse potential threats, study necessary security enhancements, and document decisions of solutions to be adopted or not adopted after evaluating the risks versus the complexity.

In particular, the following topics are addressed:

- Need and mechanism of enabling end to end authentication in roaming case if no cross-certification between operators is enabled;

- Need and mechanism of enabling NF Service Consumer authentication of NRF and the NF Service Producer;

- Need for addressing potential security impact of different deployment scenarios including the several SCPs;

- Verification of URI in subscription/notification;

- Dynamic authorization between SCPs or NF and SCP;

- End-to-End Critical HTTP headers/body parts integrity protection;

- Security of NRF service management.

# 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 33.501: "Security architecture and procedures for 5G System".

[3] 3GPP TS 23.501: "System architecture for the 5G System (5GS); Stage 2".

[4] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA)".

[5] 3GPP TS 29.500: "5G System; Technical Realization of Service Based Architecture; Stage 3"

[6] 3GPP TS 29.510: "5G System; Network function repository services; Stage 3"

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

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

Definition format (Normal)

**example:** text used to clarify abstract rules by applying them literally.

## 3.2 Symbols

For the purposes of the present document, the following symbols apply:

Symbol format (EW)

<symbol> <Explanation>

## 3.3 Abbreviations

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

Abbreviation format (EW)

<ABBREVIATION> <Expansion>

# 4 Trust model

## 4.0 General

With introduction of the service-based architecture and moving at the same time to cloud deployments, new attack vectors such as that for NFs deployed in clouds give ground to vulnerabilities and, thus, can impact the mobile operator domain. As more important it is to assure the trust also within one operator's network. For this, security concepts have been introduced.

A service request requires mutual authentication, thus all network functions support mutually authenticated TLS and HTTPS. After registration and discovery, any service request of a network function needs also to be authorized by an authorization server (NRF) before a NF Service Consumer can consume the services of a NF Service Producer. For this 5G introduces the concept of authorization token utilizing the Oauth 2.0 authorization framework.

With Rel-16 indirect communication via a Service Communication Proxy (SCP) is possible. SCPs can be operated in a fully service-meshed environment or as standalone entity.

In the following the trust relationships between the entities of an operator network are described.

## 4.1 Actors

The following actors within one PLMN are considered: NF, NRF, SCP.

- NFs can provide services or consume services.

- NRF is a repository capturing NF profiles of NFs offering its services to other NFs. It receives discovery requests from NF instances, maintains NF profiles and acts as an authorization server. NRF responds to authorization requests by NF Service Consumers by providing Oauth 2.0 access tokens to authorize a NF Service Consumer for gaining access to a service from a NF Service Producer.

- SCP is a service communication proxy used in indirect communication to interact with NFs and other SCPs within the PLMN. SCP also communicates with the SEPP.

The following types of actors when requesting services from another PLMN are considered: SEPP.

- SEPP sits at the edge of one operator's network to allow for a secured communication with the other operator's network’s SEPP.

## 4.2 Deployment options

The following deployment options need to be considered:

- Direct communication within the same PLMN vs direct communication in Inter-PLMN scenario, i.e. without SCP behind SEPP.

- Indirect communication in the same PLMN vs indirect communication in Inter-PLMN scenario: For both, SCP standalone and service mesh need to be considered.

SEPP to SEPP communication is secured on N32-c via TLS and on N32-f via TLS (i.e. transport layer security) or PRINS (i.e. application layer security on top of NDS/IP or TLS). TLS provides for authentication between two entities. Thus, securing at transport layer provides hop-by-hop security between two SEPPs. PRINS provides end-to-end application layer security between two SEPPs.

Hop-by-hop security introduces the additional risk of allowing an entity on the path to gain full access to signalling messages exchanged. An intermediary node can read, hide, or modify the originator information.

## 4.3 Description of the trust assumptions

### 4.3.1 Trust within one PLMN

This clause describes the existing trust relationships within one PLMN. Trust among the entities within one PLMN is required whether the NF Service Consumer (NFc) and NF Service Producer (NFp) are within the same PLMN or not. The trust relationships described here can be replaced by security mechanisms.

NOTE: Whether the list of existing trust relationships described below is complete, depends on deployment choices.

NRF is the core entity handling managment, discovery and authorization requests by NFs or SCP. The operator needs to apply necessary security measures to secure these operations. It is assumed that there is only one NRF, or all NRFs are within the same trust domain, i.e. all NRFs are in the same security domain and the same entity(-ies) are responsible for all NRFs.

**Registration Management:**

An NF Service Provider needs to trust the NRF that no other NF can register with the identity of NFp.

If there is no direct communication between NF and NRF, an NF Service Provider needs to trust that the SCPs forward NFp profiles unmodified.

If there is no direct communication between NF and NRF, an NF Service Provider needs to trust the SCPs that no other NF can impersonate the identity of NFp towards the SCP, i.e. tempting the SCP to register an NF with the false identity.

**Discovery:**

An NF Service Consumer needs to trust NRF to provide profiles of authenticated NF Service Providers that offer their services to the requesting consumer.

 An NF Service Consumer needs to trust SCP to correctly forward the profiles of authenticated NF Service Providers that offer their services to the requesting consumer.

**Access token request:**

Trust in direct communication between NFs, NFs and SCP/SEPP, as well as SCP and SEPP is assumed per 33.501 with mandatory mutual authentication using TLS.

An NF Service Provider needs to trust NRF to provide access tokens for consumption of its services only to those NF Service Consumers that have requested for it and only for those services that are allowed by the registered NRF policy and the registered NF Service Provider policy.

Authentication and confidentiality protection in indirect communication is only achieved between NF and SCP, (potentially between multiple SCPs), SCP and NRF as well as SCP and SEPP, but additional considerations are needed for achieving trust between NFs, NF and NRF, as well as NRF and SEPP, NF and SEPP, when an SCP is on the path. This is because all traffic in indirect communication passes through SCPs, and TLS terminates at SCPs.

Thus, the SCP needs to be trusted by NFc and NFp, to only forward authentication tokens or CCA with the original request, as well as to forward information only between the legitimate endpoints of the communication.

An NF Service Provider needs to trust NRF to provide access tokens for consumption of its services only to those SCPs that are authorized by the NF Service Consumers that have requested for it and only for those services that are allowed by the registered NRF policy and the registered NF Service Provider policy.

It also needs to be distinguished if SCP is collocated to NFs (service mesh) or standalone.

For both standalone and service-mesh, the NFs sending their service requests via an SCP need to trust the SCP to which they send their service requests.

If a SCP is co-located (e.g. a side-car proxy) with a NF, trust of the NF in the SCP is implicit by its co-location. This is, because this SCP is performing many of the functionalities on behalf of the consumer, which already indicates a certain level of trust between NFs with co-located SCPs. How the security between a SCP as a side-car proxy and its NF is implemented, is out of 3GPP scope. When the SCP is implemented as a service mesh, the security solution between the side-car proxies is out of scope of 3GPP and left to the SCP implementation.

A standalone SCP is serving many NFs, not necessarily in the same infrastructure.

### 4.3.2 Trust in Inter-PLMN communication

With 5G, a new element has been introduced to handle inter-PLM communication. The SEPP, i.e. the Secure Edge Protection Proxy acting as perimeter of PLMN, is responsible to secure the signalling message exchange with the SEPP of another PLMN via the Internet.

The SEPP of the sending PLMN needs to trust the SEPP of the receiving PLMN that no other entity on the path has unauthorized access or can modify signalling messages if not permitted to do so by policy.

**Discovery:**

The NRF in the NF Service Consumer PLMN needs to trust the cSEPP to route the request to the pSEPP representing the target PLMN and apply the correct protection policies to the discovery request.

The NRF in the NF Service Provider PLMN needs to trust the pSEPP to authenticate the origin network of the discovery request and ensure that this origin network is correctly represented in the request arriving at the pNRF.

**Access token request:**

When requesting an access token from the NRF in another PLMN, there is always an indirect communication involving the cSEPP and pSEPP. In addition, SCPs can be involved in either network.

An NF Service Provider needs to trust pNRF to provide access tokens for consumption of its services only to those NF Service Consumers in another PLMN and only for those services that are allowed by the registered NRF policy and the registered NF Service Provider policy.

An NF Service Provider needs to trust cNRF to provide access tokens for consumption of its services only to those NF Service Consumers in another PLMN that have requested for it and only for those services that are allowed by the registered NRF policy and the registered NF Service Provider policy.

An NF Service Provider needs to trust SCP in the Service Consumer PLMN to only forward authentication tokens or CCA with the original request, as well as to forward information only between the legitimate endpoints of the communication.

**Service request:**

An NF Service Provider needs to trust pSEPP to authenticate and verify the NFc's PLMN included in the request in order to be able to perform dynamic authorization.

A pSEPP needs to trust that the cSEPP is not forwarding requests on behalf of foreign PLMNs.

# 5 Key issues

## 5.1 Key issue #1: Authentication of NRF and NF Service Producer in indirect communication

### 5.1.1 Key issue details

When SCP is present, the TLS between an NF Service Consumer and NRF/NF Service Producer can be split into at least two segments (NFc-SCP, SCP-NRF or SCP-NFp). In this case, the NF Service Consumer and NRF/NF Service Producer do not directly authenticate each other via TLS.

Client Credentials Assertion (CCA) has been specified to allow NRF or another NF to directly authenticate an NF Service Consumer in the presence of an SCP, but direct authentication of the NRF/NF Service Producer by the NF Service Consumer has not been addressed in indirect communication. The key issue will investigate solutions allowing the NF Service Consumer to directly authenticate the NRF/NF Service Producer in indirect communication.

### 5.1.2 Security threats

Editor’s note: The threats need to be further clarified and studied

An NF Service Consumer could send service requests to an unintended NF.

An NF Service Consumer could receive service responses from an unintended NF.

### 5.1.3 Potential security requirements

The 5GS should provide a mechanism that allows an NF Service Consumer to authenticate an NRF or an NF Service Producer during an indirect communication with them via an SCP.

## 5.2 Key issue #2: SCP security domains

### 5.2.1 Key issue details

Editor’s note: SCP security domains to be defined.

TS 23.501 [3] addresses the aspects of handling multiple SCPs in indirect communication without and with delegated discovery and introduced SCP domains, which comprises multiple SCPs. NF Service Consumers or/and SCPs need to request NRF to discover the next hop SCP to route a service request from the NF Service Consumer to a NF Service Producer via multiple SCPs. 23.502 describes in the SCP profile SCP domain registration details about interconnected SCPs to and thus also identifies SCPs that interconnect domains.

The primary purpose of SCP domains is to describe the connectivity topology within a network. All SCPs within an SCP domain can directly interconnect. One SCP can be part of multiple SCP domains. In fact, the primary purpose of intermediate SCPs in the path is to interconnect SCP domains, thus, there are boundaries between SCP domains at each SCP in the path.

PLMN-wide trust between NFs and SCPs is an option, but more restrictions could be desirable in complex networks with SCP domains, e.g. if SCPs are operated in different regions/provinces. There can be several technical domains within a PLMN, where equipment with different capabilities is deployed and signalling also varies in some respects, e.g., if equipment upgrade is performed in a stepwise manner. Such technical domains can be defined based on computer centre boundaries, based on operators of subnetworks, based on regions/provinces, etc.

Figure 5.2.1-1: Illustration of SCP domains connecting via dedicated SPCs

This key issue is to study whether there is a need of one or several SCP domains becoming regions of trust of finer granularity than PLMN and whether there is a necessity of trust and policing of communication within or among such domains, i.e. for the case that request messages traverse a boundary between trust domains.

### 5.2.2 Security threats

Editor's Note: FFS. Maybe not applicable if only architectural security requirements are specified.

### 5.2.3 Potential security requirements

Editor's Note: FFS. Maybe not applicable if only architectural security requirements are specified.

## 5.3 Key Issue #3: Service access authorization in the "Subscribe-Notify" scenarios

### 5.3.1 Key issue details

"Subscribe-Notify" NF Service illustration 1 specified in TS 23.501, clause 7.1.2, allows one NF (e.g. NF\_A) to subscribe to notifications of NF producer (e.g. NF\_B). The subscription request includes the notification endpoint (e.g. the notification URL) of the NF Service Consumer. In this scenario, NF\_A subscribes the service of NF\_B for itself.



Figure 5.3.1-1: "Subscribe-Notify" NF Service illustration 1 (non-delegated scenario)

"Subscribe-Notify" NF Service illustration 2 specified in TS 23.501, clause 7.1.2, allows one NF (e.g. NF\_A) to subscribe the service of NF producer (e.g. NF\_B) on behalf of another NF (NF\_C), in which the notification URI of NR\_C is included. It means the NF\_C will receive the notification message even though the subscribe request is sent by NF\_A.



Figure 5.3.1-2: "Subscribe-Notify" NF Service illustration 2 (delegated scenario)

For instance, as defined in TS 23.502 clause 4.15.3.2.2, UDM could send subscribe request including the UDM URI and NEF URI to the AMF to subscribe service on behalf of the NEF, i.e. Namf\_EventExposure\_subscribe request. If the monitored event occurs, the AMF will send the event report to the associated notification URI endpoint of the NEF. Here the location report of the UE is one of the potential event reports, which can be provided by the AMF during in the above procedure. It means that the UE location report will be transmitted to the NF\_C according to the subscribe request sent by NF\_A.

The security issue of "Subscribe-Notify" NF Service illustration 1 and 2 is that NF\_B may redirect the Notification message to an unauthorized NF if the Notification URI in the subscribe message is not authorized. The issue now also arrises because of the subscribe notify usecases that have been defined with respect to DCCF and MFAF, wherein both the DCCF and the MFAF are only provided with the URI where the notification has to be sent, and therefore an unauthorized consumer can receive the notifications if the URI is not authorized.

This key issue seeks for solutions on how to assure that the notification messages could be only forwarded to an authorized NF by the NRF.

### 5.3.2 Security threats

When a malicious NF or a compromised NF tries to access an unauthorized service, in “Request-Response” scenario, NRF can verify and prevent it during access token process. But, in “Subscribe-Notify” scenario, a compromised NF can subscribe a notification service from a NF Service Producer to notify data to an unauthorized NF (possibly, a malicious NF) by setting address of notification endpoint (e.g. “Notification URI”) with address of the unauthorized NF. In this case, the NF Service Producer cannot ensure that the NF, whose URI is mentioned, is authorized to receive the notification. Thus, a malicious NF can force the NF Service Producer to send notifications to arbitrary consumers, which can e.g. result in information leakage.

According to TS 23.501, “Subscribe-Notify” scenario are used not only for subscriber’s mobility, session and subscription related events but also for NF’s own event (e.g. AMF Status change) and those information can be leaked to an unauthorized NF, while according to TS 23.288 Clause 6.2.6 subscribe-notify is used in order to enable the data consumer to receive the data from DCCF and MFAF. On the other hand, the Notification message that may include the sensitive information (e.g. location report), may expose to an unauthorized network function routed by the URI in the subscribe request message.

### 5.3.3 Potential security requirements

It shall be possible for 5G system to ensure notification service is only provided to an authorized NF routed by the URI in the subscribe request message.

It shall be possible for 5G system to prevent information disclosure to an unauthorized NF routed by the URI in the subscribe request message.

Editor’s Note: It is ffs whether these are the correct requirements.

## 5.4 Key issue #4: Authorization of SCP to act on behalf of an NF or another SCP

### 5.4.1 Key issue details

This key issue is about authorization of SCP to request services on behalf of an NF or of another SCP and how this authorization is verified by the NRF or NF Service Producer.

### 5.4.2 Security threats

If the NRF cannot verify if the SCP has been authorized by the NF Service Consumer, the SCP can send a service request and receive a valid service response on behalf of NF Service Consumer, even though the NF Service Consumer has not authorized the SCP.

If the NF Service Producer cannot verify if the SCP has been authorized by the NF Service Consumer, the NF Service Producer can provide a service response to an unauthorized entity.

### 5.4.3 Potential security requirements

The 5GS should provide a mechanism for how an NRF or NF Service Producer can verify an SCP has been authorized by an NF Consumer to request access tokens or services on behalf of the consumer.

## 5.5 Key issue #5: End-to-end integrity protection of HTTP messages

### 5.5.1 Key issue details

Currently, in the case of indirect communication with an SCP in the path between an NF Service Consumer and an NF Service Producer, the integrity protection of the HTTP messages is provided by TLS for each hop but not end-to-end between the NF Service Consumer and the NF Service Producer. Since an SCP may need to change the content of an HTTP message, this KI is to investigate how end-to-end integrity protection of HTTP messages can be achieved while at the same time continue to allow the SCP to perform necessary mediation of HTTP messages.

NOTE: Potential issues with backwards compatibility with existing procedures are to be considered during the study.

### 5.5.2 Security threats

Critical elements of an HTTP message that are not end-to-end integrity protected could be modified by an attacker. In more detail, a service request in indirect communication could lead to attacks by Man in the Middle, which for instance can intercept the service request and try to modify the content of the message or HTTP (custom) header. This could cause communication failure, lead to DoS attacks.

### 5.5.3 Potential security requirements

In the case of indirect communication with an SCP in the path between an NF Service Consumer and an NF Service Producer, the 5GS should support end-to-end integrity protection of critical elements of an HTTP message while allowing the SCP to continue to perform necessary HTTP message mediation.

Editor's Note: Collaboration with CT4 is needed in identifying critical HTTP elements that need not be mediated by an SCP.

The NF Service Producer should be able to verify that critical elements of a service request of the NF Service Consumer received via the SCP have not been modified.

## 5.6 Key issue #6: Access token usage by all NFs of an NF set

### 5.6.1 Key issue details

SBA introduces the concepts of NF Set and NF Service Set, i.e. sets of functionally equivalent and inter-changeable NFs or NF services. 5G SBA architecture design further allows for the concept of stateless NFs, where by binding indication the NF Service Resource owner can indicate to the NF Service Consumer, for a particular resource, whether it is to an NF Service Instance, NF Instance, NF Service Set or NF Set.

**Access token usage for NF Service Producer Set:**

As specified in Rel-16, an access token can be provided by NRF for consuming a service from a dedicated producer with a distinct NF Instance Id or a specific NF type or a NF Set Id for a NF Set of NF Service Producer instances. Thus, if the NF Service Producer belongs to a NF Set, the access token can be consumed by a NF Service Consumer from any of the NF Service Producers within the set.

**Stateless NFs:**

NF Set concept supports stateless NF implementations i.e. an NF Service Producer or NF Service Consumer in a NF Set can take over at any time the control of respectively resource contexts (e.g PDU session contexts) or session contexts to receive notifications. NFs typically produce and consume services (e.g. an SMF producing the PDUSession service to establish PDU session also needs to consume services to render its PDU session service, e.g. it consumes PCF and CHF services), taking over the control at any time allows for reliability of NF instances within the same NF Set (e.g. when an NF instance fails or is scaled-in).

If an access token is granted to a specific NF Service Consumer instance, other NF Service Consumer instances in the same NF Set currently need to request always a new access token, whenever a request is sent by a different NF Service Consumer instance.

For example, a connection is released since the NF Service Consumer is stateless, then another NF Service Consumer of the NF Set can be assigned to continue subsequent communication. This optimization is part of 23.501/29.500 specifications, but the related security aspects of using such optimization have not been addressed in 33.501 Rel-16. Thus, any NF in NF Set issuing a service request targeting an existing context need to request a new access token. Further, any subsequent request may be sent to any other NF than the initiator NF of the NF Set; and also in this case, a new access token is needed.

**Examples:**

The following examples show, why it is useful to have an access token also be valid/usuable for any NF in the NF Set during its validity time.

1) A SMF instance can wish to remain the SMF (binding to itself), but at end of procedure, i.e. non-moving uEs anymore foreseen. Thus, this SMF gets stateless because it considers it is a long time before next SMF involvement. Thus, if another SMF than the service request originating SMF would get involved later, it would either need a new token or it could re-use the non-expired access token, the other SMF instance of the NF Set received earlier.

2) In stateless UDM, the binding within UDM set can be used. When UDM instance of UDM Set initially creates an AMF event subscription, it has to request an access token to be able to access the corresponding AMF service. However, the UDM instance that created the subscription may be a completely different UDM instance of the UDM Set that is later deleting the subscription. Thus, the same token within the NF Set should be usuable for achieving this. Otherwise we end up at massive access token requests that are used in the same context of service consumption.

If an access token canot be used by any ND in the NF Set during its validity time, the need for access token requests is multiplied, because every time there is a different NF instance in the NF Set that is requesting from the existing resource would need a new access token, while this is not necessarily required.

**Key issue scope:**

This key issue proposes to study the advantages and disadvantages from security perspective that any NF in a NF Set targetting a service of an existing resource can use an access token provided to a NF Set.

If acceptable from security point of view, the benefit of this concept would be that it maps with the 5G SBA architecture design, the concept of stateless NF, and the binding level of NF Set, where any NF instance can serve subsequent request without everytime requesting a new access token.

Thus, this key issue studies the security implications of a stateless NF Service Consumer belonging to a NF Set requesting an access token on behalf of and for usage by all NF instances of the NF Set.

### 5.6.2 Security threats

Not applicable, since concept of access token is already in place.

Editor's Note: possibly threats resulting of the usage of the same access token by different NFs of the same NF set can be captured here.

### 5.6.3 Potential security requirements

All NF Service Consumers of an NF Set shall be authorized to use the access token requested by one NF Instance of the NF Set, if the access token is issued for NF Set.

The 5GS may provide means to authorize a NF Service Consumer of the NF Set to request and/or use an access token requested by another NF Service Consumer of the same NF Set.

## 5.7 Key issue #7: Authorization mechanism determination

### 5.7.1 Key issue details

It is specified in TS 33.501 [2] clause 13.3.0 that static authorization can be used for authorization when token-based authorization is not used. However, two PLMNs may have the roaming issue if the authorization mechanism is not aligned between them. For example, when the NF service consumer (NFc) deployed in one PLMN only supports the usage of static authorization, and the NF service producer (NFp) deployed in the other PLMN only supports the usage of OAuth authorization, the NFp will reject the NF service consumer.

On the other hand, TS 29.510 [6] defined an oauth2Requried indicating that OAuth authorization is required for the NFp service access, which will be sent back to the NFc via the discovery response. Accordingly, NFc shall get the token before consuming the NFp services. Hence, NFc that only supports the usage of the static authorization will not be able to consume the service provided by the NFp. However, how to handle the failure issue when the NFc only supports the usage of static authorization is not clarified.

The key issue will investigate solutions allowing the two operators to handle the case that one operator uses token-based authorization and its roaming partner uses static authorization.

### 5.7.2 Security threats

The SBA service authorization will fail in the roaming case if the authorization mechanism is not aligned between them.

### 5.7.3 Potential security requirements

The 5GS should provide mechanisms to handle the case that one operator uses token-based authorization and its roaming partner uses static authorization.

## 5.8 Key issue #8: Service access authorization requirements in intra-PLMN scenarios for PLMN deploying multiple NRFs (in OAuth2.0 AS role)

### 5.8.1 Key issue details

#### 5.8.1.1 Introduction

Multiple NRFs can be deployed in a PLMN, optionally using a hierarchical structure whereby an NRF may redirect or forward service requests to another NRF. One (or more) NRF can serve the entire PLMN, a set of network slices, or a single network slice.

TS 23.501 states:

In the context of Network Slicing, based on network implementation, multiple NRFs can be deployed at different levels (see clause 5.15.5):

- PLMN level (the NRF is configured with information for the whole PLMN),

- shared-slice level (the NRF is configured with information belonging to a set of Network Slices),

- slice-specific level (the NRF is configured with information belonging to an S-NSSAI).

One PLMN with several NRFs can be deployed in many ways: NRFs can have all the same data or could hold different subset of data. NRFs could all be OAuth 2.0 servers or only some of them, e.g. having one NRF being the central OAuth 2.0 server.

To receive an access token, the OAuth client need to be known to the NRF issuing the token. But looking at the different deployment options, the NRF knowing the client could be different from the NRF authorizing and issuing the access token. This raises the question, by which NRF an OAuth client needs to be authenticated and by which NRF an OAuth client gets the access token after authorization.

Only the NRF where the NF Service Producer has registered its services can act as the OAuth authorization server, i.e. to provide an access token. But the requesting NF Service Consumer is not necessarily known to this OAuth authorization server in deployment scenarios with multiple NRFs. For instance, an AMF may be registered in a PLMN-wide NRF while SMFs supporting specific network slice(s) may be registered in a slice(s) specific NRF. How does the AMF get an access token to access the SMF services in such deployment?

This key issue will clarify the service access authorization requirements and call flows, for the different NRF deployment models in case of multiple NRFs in the PLMN, including when the access token request is sent to a different NRF than the NRF where the NF Service Producer has registered its services.

TS 33.501 does only cover the inter-PLMN case, where vNRF authenticates the NF Service Consumer and hNRF provides the access token after the hNRF authorized the NF Service Consumer. How the trust between vNRF and hNRF is assured needs further clarification. For the intra-PLMN case, in particular slice specific authorization, such clause is missing.

Therefore, this key issue takes into account the different deployment models in intra-PLMN authorization requests.

#### 5.8.1.2 Hierarchical NRFs / Deployment model with local NRFs

This deployment model assumes that NFc needs to be registered at a local NRF or that NFc is known (as Oauth client) at a local NRF. It also assumes that one NRF is trusting the other NRF in the same PLMN.

When requesting an access token, NFc goes first to its local NRF, which authenticates NFc and then forwards or redirects the request to the target NRF, where a NFp has registered its services. In this case the local NRF authenticates the NFc and the target NRF (holding the policy for NFp services) provides the access token for NFp service.

Comment: This variant uses the model of inter-PLMN service access authorization also for intra-PLMN cases with multiple NRFs (with OAuth2 Authorization Server role), i.e. with an NFc registered as OAuth2 client to one NRF (local NRF) and with access token requests issued by this NFc always going through this specific/local NRF and being forwarded or redirected to the target NRF (with OAuth2 Authorization Server role) where the NFp has registered its services.

This deployment model can also apply to deployments where NFc is registered or known as Oauth client at a NRF that is not necessarily close to NFc, e.g. an AMF registered in a PLMN wide NRF.

#### 5.8.1.2 Deployment model with NF Service Consumer directly accessing the NRF where the NF Service Producer is registered

There can be centralized NRF(s) or distributed NRFs in OAuth2 Authorization Server role. An NRF can be configured by OAM with OAuth clients/ access token policies enabling a consumer to get access tokens from different NRFs (in OAuth2 Authorization Server role).

A NF (e.g. AMF) can register and/or be known as OAuth 2.0 client to a PLMN-wide NRF, but can also address a specific NRF directly, e.g. AMF can be configured with or can retrieve from the NSSF the NRF Access Token URI to use for a specific network slice: AMF may retrieve from the NSSF the NRF Access Token URI it shall use for a specific network slice (see 29.531).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **nrfAmfSetAccessTokenUri** | Uri | O | 0..1 | When present, this IE shall contain the **API URI of the NRF Access Token Service** (see clause 6.3.2 of 3GPP TS 29.510 [13]). |

Thus, an AMF can send the Access Token Request directly to the slice specific NRF, there is no need to go via a "local" NRF, where the AMF could be registered.



23.502, Figure 4.3.2.2.3.2-1: SMF selection for non-roaming and roaming with local breakout scenarios

This procedure may be skipped altogether if SMF information is available in the AMF by other means (e.g. locally configured); otherwise:

- **when the serving AMF is aware of the appropriate NRF to be used to select NFs/services within the corresponding Network Slice instance based on configuration or based on the Network Slice selection information received during Registration,** only steps 3 and 4 in the following procedure are executed as described in Figure 4.3.2.2.3.2-1;

- when the serving AMF is not aware of the appropriate NRF to be used to select NFs/services within the corresponding Network Slice instance, all steps in the following procedure are executed as described in Figure 4.3.2.2.3.2-1.

1. The AMF invokes the Nnssf\_NSSelection\_Get service operation from the NSSF in serving PLMN with the S-NSSAI of the Serving PLMN from the Allowed NSSAI requested by the UE, PLMN ID of the SUPI, TAI of the UE and the indication that the request is within a procedure of PDU Session establishment in either the non-roaming or roaming with local breakout scenario.

2. **The NSSF in serving PLMN selects the Network Slice instance, determines and returns the appropriate NRF to be used to select NFs/services within the selected Network Slice instance, and optionally may return a NSI ID corresponding to the Network Slice instance.**

See 29.531, 6.1.6.2.7 which includes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| nrfAccessTokenUri | Uri | O | 0..1 | When present, this IE shall contain the API URI of the NRF Access Token Service (see clause 6.3.2 of 3GPP TS 29.510 [13]). |

Thus, based on configuration or based on the Network Slice selection information received during Registration or PDU session establishment, the AMF is aware of the appropriate NRF to be used to select NFs/services within the corresponding Network Slice instance. There is no way to pass this URI to a “local NRF” (where the AMF would be known as OAuth2 client), as opposed to the Inter-PLMN case, where the NRF Access Token Request supports the AMF providing the hnrfAccessTokenUri.

### 5.8.2 Security threats

Not applicable, since this key issue is for clarifying missing specification text.

### 5.8.3 Potential security requirements

Not applicable, since this key issue is for clarifying missing specification text.

## 5.9 Key issue #9: Authorization for Inter-Slice Access

### 5.9.1 Key issue details

GSMA LS [S3-211383](https://www.3gpp.org/ftp/TSG_SA/WG3_Security/TSGS3_103e/Docs/S3-211383.zip%22%20%5Ct%20%22_blank) on “*Prevention of attacks on sliced core networks*” identifies a number of issues related to SBA authorization framework.

This Key Issue studies SBA related aspects of the attack papers mentioned in the GSMA LS.

NOTE: In GSMA LS to SA3, it is assumed that an NF within a 3GPP network can be fully compromised, which is a rather strong assumption. This Key Issue only aims to strengthen the authorization mechanism for granting access to an NF within SBA.

### 5.9.2 Security threats

TBD

### 5.X.3 Potential security requirements

TBD

## 5.X Key issue #X: <distinct KI name>

### 5.X.1 Key issue details

TBD

### 5.X.2 Security threats

TBD

### 5.X.3 Potential security requirements

TBD

# 6 Solutions

## 6.0 Mapping of solutions to key issues

**Table 6.0-1: Mapping of solutions to key issues**

|  |  |
| --- | --- |
| **Solutions** | **Key Issues** |
|  | #1 | #2 | #3 | #4 | #5 | #6 |  |  |
| #1: Service response verification in indirect communication without delegated discovery | X |  |  |  |  |  |  |  |
| #2: Authorization between NFs and SCP |  |  |  | X |  |  |  |  |
| #3: Using existing procedures for authorization of SCP to act on behalf of an NF Consumer |  |  |  | X |  |  |  |  |
| #4: Service request authenticity verification in indirect communication |  |  |  |  | X |  |  |  |
| #5: End-to-end integrity protection of HTTP body and method |  |  |  |  | X |  |  |  |
| #6: Verification of Service Response from a NF Service Producer at the expected NF Set | X  |  |  |  |  |  |  |  |
| #7: Access token request for NF Set |  |  |  |  |  | X |  |  |

## 6.1 Solution #1: Verification of the entity sending the service response in indirect communication without delegated discovery

### 6.1.1 Introduction

This solution is addressing KI#1.

This solution allows the NF Service Consumer (NFc) to verify the genuineness of the NF Service Producer (NFp), which is sending the response, when an SCP is used in between and the discovery of NFp has not been delegated to the SCP (see 3GPP TS 33.501 [X] Annex R, model C). I.e. The deployment scenario addressed is indirect communication (via SCP) without delegated discovery without re-selection.

The solution counters a malicious SCP or a Man in the Middle (MitM) that could forward the service request to a malicious or unauthorized NF Service Producer, i.e. a NFp that was not intended to provide a response. Especially where multiple SCPs are involved, and the NF Service Consumer does not know whether the right entity or some malicious entity is responding its request, this situation can occur.

Currently there is no means to prevent a malicious SCP or Man in the Middle to forward the service request to a different NF Service Producer.

The example shows a Service request with a token for service consumption from NFp that has been redirected by a malicious MitM to a different NF Service Producer, which can be a rogue NF cooperating with the MitM.



**Figure** 6.1.1-A**:** Example of a potential attacking scenario

This solution avoids that a service response is returned back to the NF Service Consumer by an unauthenticated and/or unauthorized MitM.

### 6.1.2 Solution details

NFc discovers NFp at NRF and requests an access token for a specific NFp Instance ID for consuming a service from NFp.

If indicated by NFc in the service request, the NFp provides back its CCA\_NFp. Thus, the NFc can compare the NFp instance ID in the CCA\_NFp with the selected NF instance ID when NFc requested the service. I.e. NFc can check if the NFp ID that the access token was provided for by NRF is matching the NFp ID present in the subject of CCA\_NFp. Since NFp provides its CCA\_NFp, this comparison is even possible if the response is sent via SCP.

If the NFp includes its own CCA\_NFp in the service response, by this the NFc can verify that NFp, the sender of the service response, is the one that NFc's service request was sent to.

This allows authentication of NFp by NFc, i.e. by NFc verifying the CCA\_NFp against the original NFp Instance ID, for which NRF provided the access token. NFp is authenticated, if the certificate NFp used to sign CCA has been verified by NFc. In case of failure, error messages can be triggered and reported to the operator.

In the following, the steps are described in detail.



Figure 6.1.2-1: Flow chart for allowing verification of NF sending the service response

Step1,2: NFc selects NFp to send a service request along with the token. To allow NFc to validate the service response, it will require validation of the producer's identity via CCA as part of the response.

Step 3: SCP intends to forward the service request to further SCPs. If SCP or some proxy is malicious (or MitM), it forwards the service request to a rouge NFp instead.

Step 4,5: A rouge NF can try to send the service response without performing the authorization. As the service request requires validation, the NFp has to add its CCA header, CCA\_NFp.

Step 6: SCP will relay back the response to NFc including the CCA\_NFp.

Step 7: NFc compare the NFp instance ID received and Set ID (if present) in the CCA\_NFp with the one used for service request. If it is the same, then NFc is assured the service response is received from a genuine NFp.

Step 8: If it does not match, the NFc can also raise an alarm and revert the transaction at NFc.

### 6.1.3 Evaluation

This solution proposes an enhancement at the NF Service Producer to use the CCA as defined in TS 33.501. It provides an approach how an NF Service Consumer can authenticate NF Service Producer, from which NF Service Consumer received a service response, as intended NF for Service Response in indirect communication without delegated discovery.

This solution introduces Client credentials assertion of NF Service Producer which includes NFp Instance ID and signature using certificate of NFp. The NF Service Consumer can validate the CCA sent by the NF Service Producer and ensure that no rogue or malicious SCP or MitM has sent a service request to a malicious NF Service Producer.

This solution is only applicable in a very limited scope, it does not cover model D and the case when SCP reselects another NF as NF Service Producer which is different from the targeted NF Service Producer by NF Service Consumer. Therefore, it is possible for NF Service Consumer to reject the received service response from a legitimate NF Service Producer and may induce service unavailability. Further, the NF Service Producer cannot determine if the request coming from SCP which is using model D or SCP using model C or a re-selected by SCP, so the producer cannot determine, when to generate CCA\_NFp.

## 6.2 Solution #2: Authorization between NFs and SCP

### 6.2.1 Introduction

This potential solution addresses KI#4.

### 6.2.2 Solution details

When sending the service request to SCP in delegated discovery, the NF Service Consumer must authorize the SCP to act on its behalf. Thus, NRF needs to be provided with evidence by NFc about the SCP instance ID.

NRF knows implicit the SCP instance ID because of direct TLS between SCP and NRF, when SCP ID would be added in a SCP TLS certificate. But this still does not assure that NFc as sent its request to this SCP. Thus, authorization between NF Service Consumer and SCP, when sending the service request to SCP in delegated discovery, has to be explicit. The solution proposes to do so by enhancing the CCA by inserting either the SCP Instance ID or the SCP Domain Info in CCA\_NFc, and therefore the NF Service Consumer can authorize SCP.

NOTE: Since in model D the NF Service Consumer is delegating the discovery, as well as access token request, service request and receiving service response to SCP, the NF Service Consumer authorizes the SCP to perform these actions on its behalf.

The SCP also generates its own enhanced CCA\_SCP including its Instance ID and/or its Domain Info and sends it along with access token request and the enhanced CCA\_NFc as received from NF Service Consumer.



Figure 6.2.2-1: Authorization of SCP by NFc in indirect communiation

NFc

SCP

NRF

1. Service Request (optionally includes enhanced CCA')

**\* CCA' additionally includes Authorized SCP ID**

2. Nnrf\_AccessToken\_Get\_Request (**includes CCA' and CCA,**

CCA contains the SCP Instance ID in the subject parameter)

3. NRF analyzes the request, and authorizes the SCP

4. Nnrf\_AccessToken\_Get Response

(access token)

The NRF verifies that the Target SCP Instance ID and/or SCP Domain info present in the CCA\_NFc matches the Instance ID/Domain Info of SCP as also being part of the subject of the CCA\_SCP. A successful verification of CCA(s) by NRF ensures that the SCP has been authorized by the NF Service Consumer.

Thus, the NRF needs to know the SCP Instance ID. One way for the NRF to learn the SCP instance ID is to use direct TLS between SCP and NRF, this requires SCP ID in the SCP TLS certificate which is currently not specified in TS 33.310. Another way for the NRF to learn the SCP instance ID is by CCA\_SCP.

If authentication was successful and the NF Service Consumer is authorized based on the NRF policy and the SCP requesting the access token has been explicitly authorized by NF Service Consumer, the NRF issues an access token.

A similar solution is also applicable for authorizing SCP by NFc to request a service and receive a response from NFp on its behalf. The NFp then may perform similar verification and, in case of successful verification, can send the service response to SCP.

However, even if the TLS certificate of the NFc would mandate the usage of NFc Instance ID, another problem still needs to be solved in case the SCP selects another SCP. This is because if the NRF or the NF Service Producer do not know the SCP domain, to which the SCP belongs to, the SCP\_CCA included by NFc does not help NRF. Thus, for this reason it is suggested that the TLS certificate needs in addition to SCP Instance Id also to hold the SCP domain identifier for allowing NRF to verify that NFc authorized one SCP of a SCP domain.

If the SCP, that NF Service Consumer delegated the authorization token request to, is not serving the NF Service Consumer request by itself, but demands another SCP' to do so, then the same procedure is needed between SCP and SCP'. Hence, SCP forward the service request to SCP' with its own CCA\_SCP including into it the SCP' ID.

The verification of the CCA shall be performed by the receiving node as described in clause 13.3.8.3, but verifying that the SCP instance ID in the CCA is matching the SCP instance ID in the public key certificate used for signing the CCA. This is either done by an SCP, in case there are several SCPs in between, or by the NF Service Producer.

In practice, one would expect one to three SCPs between consumer and producer. But it needs to be noted, if the NF Service Producer wants to have verification of the full chain of trust via several SCPs, all CCAs and certificates from the NF Service Consumer and the intermediary SCPs need to be available to the NF Service Producer.

### 6.2.3 Evaluation

Editor's Note: Provide an analysis of the risks of threats mitigated by this solution. Provide a statement on complexity/impact/backward compatibility if one would follow this solution.

This solution fulfils requirement on KI#4. The SCP can be authorized to act on behalf of an NF Service Consumer and to request access tokens by NFc, because the NFc is including the SCP ID or SCP Domain ID into CCA\_NFc. With NFc providing the SCP ID in the CCA, authorization of that particular SCP is given, because NRF or NF Service Producer can with assurity verify that the SCP, which provides CCA\_NFc, is indeed the one SCP to which the NF Service Consumer sent its CCA and has authorized that SCP to request services and receive response on its behalf.

Thus, this solution counters a potential attack of SCP stealing a CCA and using it for requesting an access token without being requested by a NF Service Consumer.

The concept of CCAs is already known. In addition to verifying the NFc CCA the NF Service Producer needs to verify also the CCAs of one or several SCPs. In addition to the verification of the NF Service Consumer's CCA the NRF needs to verify also the CCAs of one or several SCPs. Further, SCPs need to be able to create their own CCAs.

## 6.3 Solution #3: Using existing procedures for authorization of SCP to act on behalf of an NF Consumer

### 6.3.1 Introduction

This solution addresses Key Issue #4 "Authorization of SCP to act on behalf of an NF or another SCP". It explains how token-based authorization and CCAs as currently specified in TS 33.501 [2] can be used to authorize the SCP to act on behalf of an NF Consumer, i.e. to request access tokens or services on behalf of the consumer.

### 6.3.2 Solution details

#### 6.3.2.1 Request of access token on behalf of the consumer

The SCP requests access tokens on behalf of the consumer in Scenario D (indirect communication with delegated discovery) and in Scenario C (indirect communication without delegated discovery) without mutual authentication between NF and NRF at the transport layer. The following procedure describes token requests for Scenario D, and particularly how CCAs are used to authorize the SCP to request access tokens on behalf of the NF Consumer. For Scenario C without mutual authentication between NF and NRF at the transport layer, the same principles hold.



Figure 6.3.2.1-1: Access token request of SCP on behalf of an NF Consumer

1. The NF Service Consumer sends a service request to the SCP. The consumer includes a CCA signed by the consumer. The CCA includes the NF Instance ID of the consumer. The consumer's certificate used for signing the CCA also contains the consumer's NF Instance ID.

2. The SCP sends an access token request to the NRF. The SCP includes the CCA received by the consumer in step 1.

3. The NRF verifies the CCA as described in clause 13.3.8.3 of TS 33.501 [2] and thus obtains the NF Instance ID of the consumer that signed the CCA. Besides authentication of the consumer, the CCA also implicitly authorizes the SCP to act on behalf of the NF Service Consumer.

The NRF authorizes the NF Service Consumer as described in TS 33.501 [2].

4.-8. The remaining steps of the access token request and service request procedure are exactly as described in TS 33.501 [2].

#### 6.3.2.2 Service request on behalf of the consumer

The SCP requests services on behalf of the consumer in all indirect communication scenarios. The following procedure describes access token and service requests for Scenario D, and particularly how CCAs and access tokens are used to authorize the SCP to request services on behalf of the NF Consumer. For Scenario C, the same principles hold.



Figure 6.3.2.2-1: Service request of SCP on behalf of an NF Consumer

1.-4. Service request and access token request and response are performed as described in the previous clause, clause 6.3.2.1.

5. The SCP sends a service request to the NF Service Producer. The service request contains the access token and optionally the CCA received in step 1. The access token contains the NF instance ID of the NF Service Consumer.

6. The NF Service Producer validates the access token as described in TS 33.501 [x]. Because the network implements the procedures described in the previous clause, clause 6.3.2.1, the NRF has already verified that the SCP was authorized to request the access token on behalf of the NF Service Consumer. Hence the access token does not only authorize the consumer, but also implicitly authorizes the SCP to act on behalf of the NF Service Consumer.

7.-8. The remaining steps of the access token request and service request procedure are exactly as described in TS 33.501 [x].

#### 6.3.2.4 Protection of the NF consumer's CCA

The CCA is protected in transport and storage by the following methods, partly in and partly out of 3GPP scope:

- Transport protection: The CCA is protected in transport by TLS or other means, as specified in TS 33.501 [2], clause 13.1.0. Thus, it is protected between NF and SCP, and between SCP and NRF or NFp.

- Storage protection: Although CCAs are expected to be short-lived, they could be cached for a short period of time at the NF Service Consumer. Similar as for other data handled at the NF Service Consumer, e.g., sensitive UE data, storage protection mechanisms outside of 3GPP scope need to be in place.

If used according to the procedure describes in clause 6.3.2.1, only the NF Service Consumer itself, the SCP and the NRF will obtain the CCA that allows to request access tokens on behalf of the NF Service Consumer. This solution assumes, that the SCP is authorized by the NF Service Consumer to request access tokens on behalf of it, the NF Service Consumer indicates that by sending the CCA to the SCP. The NRF is itself the entity that issues access tokens for the NF Service Consumer. Hence, if used according to the procedure described in clause 6.3.2.1, only entities that are authorized by the NF Service Consumer to request access tokens on behalf of it obtain the CCA.

Editor's Note: Whether an implicit authorization of the SCP by sending the CCA to the SCP is sufficient, is ffs.

### 6.3.3 Evaluation

The solution addresses the threats and requirements of Key issue #4: Authorization of SCP to act on behalf of an NF or another SCP.

The solution relies on token-based authorization and CCAs as currently specified in TS 33.501 [2].

It proposes that authorization of the SCP by the CCA is implicit by sending the CCA to the SCP, i.e. by presenting the CCA\_NFc received by the NF Service Consumer, the SCP shows it is authorized to act on behalf of the Consumer and to request access tokens on behalf of it. However, authorization is not explicitly stated in the CCA. Hence an entity that is not authorized by the NF Service Consumer but somehow has obtained a valid CCA signed by the consumer could use it to request access tokens on behalf of the consumer. Thus, in this case the NRF or the NFp can provide the service response to an unauthorized consumer.

## 6.4 Solution #4: Service request authenticity verification in indirect communication

### 6.4.1 Introduction

This solution addresses the KI#5.

### 6.4.2 Solution details

This solution allows the NF Service Producer to verify that a service request of the NF Service Consumer received via SCP has not been modified.

In case of CCA is used for authentication, the service request received by NRF or NF Service Producer can be verified as the one to be originally sent by the NF Service Consumer. This would guarantee that in indirect communication no intermediary can modify the service request unrecognized.

Editor's Note: Backwards compatibility with Rel-16 NF producers supporting only existing CCA is ffs.

For this, the CCA is enhanced with a new payload value for 'service request verification' and a protected header list.

- The 'service request verification' (SRV) includes the service request message (or a hash of it) as one of the payload values.

Editor's Note: If not the hash but the whole message or headers is included, impact on throughput needs to be considered and is ffs.

Editor's note: It is ffs how the SCP can perform necessary message modifications, if the (hash of the) whole service request is included in CCA.

- The protected header list (HL) includes custom headers that shall be integrity protected and thus not be modifiable undetected by SCP.

If present, the NF Service Producer or the NRF can verify whether these data included in the CCA are matching the service request as sent by the NF Service Consumer. I.e. the NF Service Producer verifies that the data included in the payload is matching the service request received together with the CCA. The receiver also verifies that the headers in the protected header list are not modified.

Since CCA is digitally signed by the NF Service Consumer, thus the recipient can verify that the service request received from SCP is the original one as provided by the NF Service Consumer. The additional SRV payload provides authenticity of the service request.

NOTE: This solution assumes that an SCP does not need to modify service request details for providing its service of delegated discovery and access token request to NRF or transferring a service request to the NF Service Producer. If there are headers that need to be modified by SCP/Proxy, then those headers cannot be considered as payload of SRV. The NF Service Consumer provides in this case a separate list of headers (HL) to explicitly state what is covered under SRV. The destination endpoint (NRF or NF) can take them in consideration while verifying the received data.

In detail:

- NF Service Consumer creates a service request and creates a keyed hash value about those parts of the service request, that are not to be modifiable by the SCP, and generates CCA including a 'service request verification' (SRV) payload with the keyed hash value. If necessary, a protected HL is included.

Editor's Note: CT4 feedback is needed on which headers are not subject to modification, mediation, or alteration by the SCP and can be delivered as is to the other far end of the indirect communication.

Editor's Note: It is ffs if a keyed hash is necessary and if yes how the key is obtained or derived.

- NRF, after verifying the authenticity of NF Service Consumer by checking the CCA, it checks SRV, i.e. it verifies the authenticity of the service request by creating a hash of the service request and comparing it with the received SRV value. It also verifies that the headers in the protected HL are not modified.

- NF Service Producer, after receiving an access token and CCA/SRV from the SCP, it verifies the NF Service Consumer by checking the CCA, it checks whether the NF instance id for which the access token was provided, matches the identity in CCA and it verifies the authenticity of the service request by creating a hash of the service request and comparing it with the received SRV value. It also verifies that the headers in the protected HL are not modified.

### 6.4.3 Evaluation

This solution provides an approach how an NF Service Producer can verify that a service request of the NF Service Consumer received via SCP has not been modified.

This solution extends Client credentials assertion to include new payload value for service request verification and a protected header list.

When the service request verification includes whole service request message, which may double the size of the message and may impact on system throughput.

When the service request verification includes hash value of service request message, additional information shall be transmitted to the NF Service Producer to inform HTTP headers and order among HTTP headers which shall be considered in calculation of hash value.

When SCP appends HTTP standard header(s) such as Via header and Authenticate header, in this solution, NF Service Producer cannot recognize those headers shall not be considered in calculation of hash as those are added by SCP and NF Service Producer will fail to calculate correct hash value of HTTP message.

This solution proposes to include keyed hash value of service request in CCA, but the necessity and benefit of keyed hash value of service request in CCA are not well identified.

## 6.5 Solution #5: End-to-end integrity protection of HTTP body and method

### 6.5.1 Introduction

This solution addresses the key issue #5 (End-to-end integrity protection of HTTP messages).

The core steps of this solution are:

- Use Client credentials assertions (CCAs) based authentication as specified in TS 33.501 [2] Clause 13.3.8 for NF-NRF or/and NF-NF communication.

- Enhance the Client credentials assertions (CCAs) to optionally include a hash of the HTTP body and HTTP method to protect the message itself.

- The receiving node (NRF or NF producer) computes the hash of the HTTP body and HTTP method and validates that it is identical to the hash received in the Client credentials assertions (CCAs).

Since the added hash is an optional field in the ClientCredentialsAssertion as specified in 3GPP TS 29.500 [5] Table 5.2.3.2.11 -1, this solves the backwards compatibility with Rel 16 NF producers supporting only existing CCA. A Rel-16 producer will verify the signature of the CCA correctly but ignore the optional field that it does not recognize. The behaviour is similar to Rel-15 producers' behaviour for IEs in access tokens that were introduced in Rel-16. As specified in TS 29.510 [6], Table 6.3.5.2.4-1 "Definition of type AccessTokenClaims", if an NF service producer receives an IE in the access token that it does not understand, the NF service producer ignores the IE. Similar behaviour can be specified for IEs in the CCA, see Table 6.5.2-1 below.

Editor's Note: It needs to be clarified whether the handling for access tokens is aplicable for CCAs.

Editor's Note: This solution has dependency on CT4 feedback on what SCP exactly needs to modify.

### 6.5.2 Solution details



Figure 6.5.2-1 CCA based Authentication with HTTP hash enhancement

1. NF service consumer sends a service request including a signed Client credentials assertion (CCA) token to authenticate against NF service producer or NRF as described in TS 33.501 [2] Clause 13.3.8. But for this solution it is also proposed to add an optional field in CCA to protect the part of the message itself. The added field is a hash of HTTP body and HTTP method.

2. NF service producer or NRF validates the CCA as described in 3GPP 33.501 Clause 13.3.8.3. But since one optional field is supposed to be added to the CCA, the receiving end point (NF service producer or NRF) also needs to compute the hash of the HTTP body and HTTP method and validates that it is identical to the hash received in the Client credentials assertion.

Table 6.5.2-1: Updated CCA based on Table 5.2.3.2.11 -1: Definition of type ClientCredentialsAssertion

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute name | Data type | P | Cardinality | Description |
| sub | NfInstanceId | M | 1 | This IE shall contain the NF instance ID of the NF service consumer, corresponding to the standard "Subject" claim described in IETF RFC 7519 [41], clause 4.1.2. |
| iat | integer | M | 1 | This IE shall indicate the time at which the JWT was issued, corresponding to the standard "Issued At" claim described in IETF RFC 7519 [41], clause 4.1.6. This claim may be used to determine the age of the JWT. |
| exp | integer | M | 1 | This IE shall contain the expiration time after which the client credentials assertion is considered to be expired, corresponding to the standard "Expiration Time" claim described in IETF RFC 7519 [41], clause 4.1.4.  |
| aud | array(NFType) | M | 1..N | This IE shall contain the NF type of the NF service producer and/or "NRF", for which the claim is applicable, corresponding to the standard "Audience" claim described in IETF RFC 7519 [41], clause 4.1.3.  |
| **hash** | **string** | **O** | **0..1** | **This IE contains a hash of the body of the HTTP message and HTTP method. If an NF service producer that receives this IE in the CCA included in the** **3gpp-Sbi-Client-Credentials header does not understand this IE, it shall be ignored.** |
| **halg** | **string or integer** | **O** | **0..1** | **This IE contains the hash algorithm information that is used by NF service consumer to compute the hash of the HTTP message. If an NF service producer that receives this IE in the CCA included in the** **3gpp-Sbi-Client-Credentials header does not understand this IE, it shall be ignored.** |

The details of the hash are proposed to be specified as following:

Option 1: For computation of the hash of the HTTP body and HTTP method for inclusion into the Client credential assertion, the input S to the KDF specified in Annex B of 3GPP TS 33.220 [4] is computed as follows:

 - P0 = HTTP body;

- L0 = length of the HTTP body;

- P1 = HTTP method;

- L1 = length of HTTP method.

The input key KEY is equal to null. Note that the FC value will be allocated in the normative phase.

Option 2: Alternatively to using the fixed KDF as hash function, the choice of hash function can also be done similar as in JWT or JWS. The hash algorithm is chosen by NF service Consumer. The selection of hash algorithm needs to be aligned between HTTP message sender and HTTP message receiver, i.e., mandatory to support algorithms need to be specified in a 3GPP profile. This option provides more crypto agility and is better aligned with JWT and JWS. For ease of implementation in initial deployments, the 3GPP profile for the hash algorithm could mandate the usage of a specific hash function, e.g. SHA256. This is similar to the JOSE profile of PRINS as specified in TS 33.501 [2], clause 13.2.4.9, which specifies the usage of specific AEAD and signature algorithms, but still provides crypto agility if changes should be necessary in the future.

Editor's Note: It needs to be clarified whether the usage of a new hash algorithm can also be indicated by the length.

### 6.5.3 Evaluation

This solution provides an approach how an NF Service Producer can verify that a service request of the NF Service Consumer received via SCP has not been modified.

This solution extends Client credentials assertion to include hash value of HTTP body and HTTP methods.

This solution does not handle integrity protection of HTTP headers.

## 6.6 Solution #6: Verification of Service Response from a NF Service Producer at the expected NF Set

### 6.6.1 Introduction

This solution addresses key issue #1. In order to verify the message from NF Service Producer in indirect communication, it is proposed to append CCA of NFp. And NF Servcie Consumer may accept the certificate if it is verified well and NF Service Producer instances belongs to the expected NF Producer instance(s).

### 6.6.2 Solution details

6.6.2.1 For indirect communication without delegated discovery procedure

Figure 6.6.2.1-1: With mutual authentication between NF and NRF at the transport layer

**Discovery of the NF Service Producer:**

0. When a NF Service Consumer discover a NF Servcie Producer for a service, NRF provides information of target NF set and candidate target NF instance IDs belonging to the target NF set.

The NF set information in the discovery response from NRF to NF consumer needs to be end to end integrity protected, by e.g. TLS or solution to Key Issue #5, so that the SCP cannot modify the NF set information in the discovery response.

**NF Service Consumer authorization:**

1-2. After mutual authentication between NF Service Consumer and NRF at the transport layer, the NF Service Consumer and NRF perform the "Access token request before service access" procedure. If the NF Service Consumer has already discovered the NF Service Producer, it can also perform the "Access token request for a specific NF Service Producer/NF Service Producer instance" procedure.

**Service Request:**

4. Among the candidates NF instances list, the NF Service Consumer may select an NF instance for a Service Request. And the NF Service Consumer keep the list of candidate NF instances and NF set for verification of expected Service Response.

After acquiring an access token from the NRF, a NF Service Consumer may send a Service Request to the SCP. The service request includes the access token and CCA of the NF Service Consumer.

The service request includes the 3gpp-Sbi-Routing-Binding header and/or 3gpp-Sbi-Discovery header in order to specify target NF Service Producer and/or target NF Set, so that the SCP is instructed to perform the NFp reselection within the scope of NF Set.

5. An SCP forward a Service Request to the NF Service Producer. If needed, the SCP may reselect another NF Service Producer belonging to the same NF set.

**Service Response:**

6-7. After receiving a Service Request, the NF Service Producer may verify the Service Request and may respond with a Service Response with CCA of the NF Service Producer. CCA of NF Service Producer includes NF instance ID of NF Service Producer and NF instance ID of NF Service Consumer.

8-9. When receiving a Service Response, the NF Service Consumer may verify whether the NF instances ID of NF Service Producer which sends the Service Response is in the list of candidate NF instances for the Service Request.

#### 6.6.2.2 for indirect communication with delegated discovery



Figure 6.6.2.2-1: for indirect communication with delegated discovery

1. The NF Service Consumer sends a service request to the SCP. The service request shall include NF service discovery factors such as target NF type and the NF Service Consumer's CCA as defined in clause 13.3.8.

2. The SCP may perform a NF discovery operation with the NRF using NF service discovery factors received in step 1.

3. (same with step 3 in subclause 13.4.1.3.2 TS33.501.) The SCP sends an access token request (Nnrf\_AccessToken\_Get Request) to the NRF. The access token request includes parameters as defined in clause 13.4.1.1. The access token request may include the NF Service Consumer's CCA if received in Step 1.

4. (same with step 4 in subclause 13.4.1.3.2 TS33.501.) The NRF authenticates the NF Service Consumer using one of the methods described in clause 13.3.1.2. If NF Service Consumer authentication is successful and the NF Service Consumer is authorized based on the NRF policy, the NRF issues an access token as described in clause 13.4.1.1. The NRF uses the NF Service Consumer instance ID as the subject of the access token.

5. (same with step 5 in subclause 13.4.1.3.2 TS33.501.) The NRF sends the access token to the SCP in an access token response (Nnrf\_AccessToken\_Get Response).

6. (same with step 6 in subclause 13.4.1.3.2 TS33.501.) The SCP sends the service request to the NF Service Producer. The service request includes an access token (i.e., received in Step 1, received in Step 5, or previously cached), and may include the NF Service Consumer's CCA if received in Step 1.

7. (same with step 7 in subclause 13.4.1.3.2 TS33.501.) The NF Service Producer authenticates the NF Service Consumer by one of the methods described in clause 13.3.2.2 and if successful, it validates the access token as described in clause 13.4.1.1.

8. If the validation of the access token is successful, the NF Service Producer may respond with a Service Response with CCA of the NF Service Producer. CCA of NF Service Producer may include NF type and NF instance ID of NF Service Producer and NF instance ID of NF Service Consuer.

9. The SCP forwards the service response to the NF Service Consumer. The SCP may include the access token in the service response to NF Service Consumer for possible re-use in subsequent service requests.

10. When receiving a service response, the NF Service Consumer may verify whether the NF Service Producer belongs to the target NF type and authenticate NF Service Producer using CCA and X.509 certificate of the NF Service Producer.

#### 6.6.2.3 Client credentials assertion of NF Service Producer

CCAs shall be JSON Web Tokens as described in RFC 7519 [44] and are secured with digital signatures based on JSON Web Signature (JWS) as described in RFC 7515 [45].

The CCA of NF Service Producer may include:

- the NF instance ID of the NF Service Producer;

- the NF set information of the NF Service Producer;

- the NF instance ID of the NF Service Consumer;

- The NF type of the NF Service Producer;

- A timestamp and an expiration time, and

The NF Service Consumer shall digitally sign the generated CCA based on its private key as described in RFC 7515 [45]. The signed CCA shall include one of the following fields:

- the X.509 URL (x5u) to refer to a resource for the X.509 public key certificate or certificate chain used for signing the client authentication assertion, or

- the X.509 Certificate Chain (x5c) include the X.509 public key certificate or certificate chain used for signing the client authentication assertion.

### 6.6.3 Evaluation

 This solution provides an approach how an NF Service Consumer can authenticate NF Service Producer, from which NF Service Consumer receives a service response, as intended NF for Service Response in indirect communication without delegated discovery and with delegated discovery.

This solution introduces Client credentials assertion of NF Service Producer which includes NFp Instance ID, NFc Instance ID, and signature using certificate of NFp.

In indirect communication without delegated discovery, by reusing existing HTTP custom headers, it can also cover the case when SCP reselect another NF as NF Service Producer. This solution works with assumption that the discovery results from NRF to NF Service Consumer are protected to detect any harmful modification in the middle. And it also assumes that NRF will inform NF Service consumer about which NF Service Producers are in the NF Set and SCP only re-selects another NF Service Producer within the NF Set.

Editor’s Note: How to assure by the NFc that the NFp is origianl NFp which received the service request is FFS.

In indirect communication with delegated discovery, this solution requires extension of CCA and/or X.509 Certificate of NF Service Producer to include NF type of NF Service Producer.

This solution is to address KI#1 which basically assumes that the SCP and NFp are compromised or at least the SCP is compromised. If that the threat this solution is trying to address, thus the proposed solution only prevents such attack in the case when NF and NRF are mutually authenticated using TLS over direct communication without SCP being present. This means this solution does not addres KI#1 in the following cases:

- Delegated Discovery, Model D,

- Model C when the NF service consumer communicates with NRF over indirect communication via SCP.

## 6.7 Solution #7: Access token request for NF Set

### 6.7.1 Introduction

This solution addresses KI#6.

3GPP introduces the concepts of NF Set and NF Service Set which allows essentially for a group of interchangeable NF instances/NF Service instances of the same type, supporting the same services and the same Network Slice(s). Rel-16 also allows re-selection of a NF instance or a NF Service instance within the Set for subsequent transaction.

5G SBA architecture design allows for the concept of stateless NFs.

The solution assumes that each NF of a set has registered at NRF also with its NF Set ID or the NF Service Set ID. Thus, verification of the correctness of a set id is done when authenticating the NF when registering at NRF. Thus, if NRF is then issuing an access token with a distinct set id, the NF Service Producer can trust the correctness, or do another verification, if the set id is also included in CCA or NF certificate.

A NF Service Producer can also indicate in its profile, if it is allowing the NRF to provide access tokens for NF Sets or NF Service Sets.

NOTE: Whether to have this feature allowed per operator policy configured at NRF or per NF Service Producer or NF Service Producer Set is a deployment decision.

The solutions objective is to avoid that a NF from a NF Set needs to request a new access token, when targeting a service of an existing resource requested before by another NF of the NF Set, it is proposed that any NF in a NF Set can request an access token for the NF Set. Thus, any NF Service Consumer targetting a service of an existing resource it can use the access token provided to a NF Set of NF Service Consumers.

NOTE: For any NF to make use of this solution, that NF is required to register its profile with the NRF.

### 6.7.2 Solution details

The NF Service Consumer belonging to a NF Set, it includes its NF Set ID in the Access Token Request message to NRF and also in the CCA or the NF certificate.

When the Access Token Request is processed by the NRF and a NF Set ID is included, the NRF knows that the NF Service Consumer requests an access token to be usuable by all NF Service Consumer instances within the NF Set. If NRF authorization of the NF Service Consumer is successful, ie. the NF Service Producer has indicated that an access token for a NF Set or NF Service Set can be issued, and the NF Set ID in the CCA matches the NF Set ID in the access token or in the NF certificate, NRF includes as claim the NF Set ID of the expected NF Service Consumer instances to allow the access token generated for usage by all NF Service Consumers in the NF Set. NRF sends the access token back to the requester.



Figure 1 – Access Token Request procedure (TS 33.501 Figure 13.4.1.1.1-1) enhanced with NF Set ID in the Access Token Request message

How NFs of a NF Set or a NF Service Set manage the distribution of an access token issued for set or service set and their availability to other NFs within the NF Set, is for implementation and out of scope.

When a service is requested, the requester (NF Service Consumer or SCP) includes the NF Set ID of the NF Service Consumer in the Service API Request, as well as in the CCA, if the CCA is sent, in addition to the access token obtained from the NRF. NF Set ID in CCA is only reliable if the NF Set ID is included in the certificate related to the private key that the NF Service Consumer used to sign the CCA.

The NF Service Producer checks whether the Consumer NF Set Id in the Service Request matches with the NF Set ID claim in the Access token. If CCA is sent, it also verifies, if the NF Set ID matches the NF Set ID in the CCA. If included in NF certificate, it can also match the NF Set ID with the NF Set ID in the NF certificate. If yes, it proceeds with serving the request, otherwise it rejects the request.

Editor's Note: Clause 5.21.3.2 of TS 23.501 states "Furthermore, for a given UE and PDU Session any SMF in the SMF Set should be able to control the N4 session with the UPF (however, at any given time, only one SMF in the SMF Set will control the UPF for a given UE's PDU Session)." It is ffs whether only one NF consumer in the NF set can use the same token to request service from NFp at the same time, i.e. if only one NFc can represent the NF set at any given time.

### 6.7.3 Evaluation

The solution proposed allows the authorization server, i.e., NRF, to issue an access token that can be used by all members of an NF Set or NF Service Set. The concept of NF Set and NF Service Set has been introduced by 3GPP. This solution enable optimization that is sought from the mutual redundancy among the NF instances of the set. It would be less optimized if each instance needs to request its own token.

Using the same access token for a NF Service Consumers belonging to one NF Set is not explicitly described by RFC 6749. Other literature mentions group access tokens, but further investigation on the impact managing an access token used by NF Service Consumers of the same set is needed.

According to RFC 6749, each NF instances needs to register with the authorization server (NRF) as a separate OAuth2.0 client before the authorization server is able to issue such a token which can be used by all members of the NF Set.

Since CCA is used for Indirect communications when SCP in the path between the NF Service Consumer and the NF Service Producer, including NF set ID and/or NF service set ID into the CCA only work for the case of indirect communication but not in the direct communication case.

Including NF set ID in the NF certificate is not a flexible mechanism which requires an intervension in case of the NF instance is removed from a specific NF set ID and/or NF service set ID or added to another NF set ID or NF service set ID. On the other hand, if any of these operations are done to any NF instance, the NF instance will update its profile with the NRF automatically and the update is almost dynamic for the rest of the processes.

This solution requires that in case of any change to the list of members of the NF set, all existing access token with the impacted NF set ID and/or NF service set ID shall be destroyed and not used. A new access token is required.

## 6.8 Solution #8: integrity protection of HTTP message in consideration of update by SCP

### 6.8.1 Introduction

This solution addresses key issue #5.

It is proposed to use enhance CCA to include hash value of HTTP headers and HTTP body.

In the enhanced CCA, hash value of HTTP headers and hash value of HTTP body are included, separately.

For calculation of hash value of HTTP headers, Via and Authorization headers are not included.

There shall be a rule for ordering HTTP headers. HTTP standard headers come first and HTTP custom headers come after HTTP standard headers. Among HTTP custom headers from NF Service Consumer or NF Service Producer, 3GPP-Sbi-Client-Credentials header shall be positioned as the last one. When an SCP add HTTP custom headers, those shall come after 3GPP-Sbi-Client-Credentials header from NF Service Consuner of NF Service Producer.

When NF Service Consumer of NF Service Producer receive an HTTP message, they compute the hash value of HTTP headers and HTTP body and compare those with the values in CCA for checking of integrity protection.



Figure 6.X.1-1: HTTP message with hash value in CCA for end to end message protection

NFc

SCP

NRF

1. Service Request (optionally includes enhanced CCA')

**\* CCA' additionally includes Authorized SCP ID**

2. Nnrf\_AccessToken\_Get\_Request (**includes CCA' and CCA,**

CCA contains the SCP Instance ID in the subject parameter)

3. NRF analyzes the request, and authorizes the SCP

4. Nnrf\_AccessToken\_Get Response

(access token)

### 6.8.2 Solution Details

This solution enables the NF Service Producer and NF Service Consumer to verify a HTTP message received via SCP has not been modified.

NF Service Consumer calculate hash value of HTTP standard headers except Authenticate header and HTTP custom headers and hash value of HTTP body and include in the CCA.

NF Service Consumer append CCA as the last of HTTP custom headers.

For this, the CCA is enhanced with additional payload values.

- hash value of HTTP headers including HTTP standard headers and HTTP custom headers generated by originating Network Function.

- hash value of HTTP body generated by originating Network Function.

After SCP receives HTTP message from NF Service Consumer, it may append Via header and Authenticate header(only for delegated discovery case) and some HTTP custome headers after HTTP standard headers and HTTP custom headers at the received HTTP message, respectively.

NF Service Producer verify hash value of HTTP standard headers and HTTP extended headers except Via header, Authorization header and HTTP custome headers appended by SCP and hash value of HTTP body by comparing the calculated value and the ones in CCA.

Similarly, NF Service Producer can calculate hash value of HTTP standard headers and HTTP custom headers and hash value of HTTP body and include in the CCA if CCA is used.

After SCP receives HTTP message from NF Service Producer, it may append Via header and some HTTP custome headers after HTTP standard headers and HTTP custom headers at the received HTTP message, respectively.

NF Service consumer can verify hash value of HTTP standard headers and HTTP extended headers except Via header and HTTP custome headers appended by SCP and hash value of HTTP body by comparing the calculated value and the ones in CCA.

Editor's Note: Whether CCA of NF Service Producer is used is per decision of key issue #1.

### 6.8.3 Evaluation

This solution provides an approach how an NF Service Producer can verify that a service request of the NF Service Consumer received via SCP has not been modified.

This solution extends Client credentials assertion to include hash value of HTTP headers and HTTP body.

This solution propose a rule how to compose the HTTP headers and how to calculate hash value of a HTTP message and it allows NF Service Producer to calculate hash value of HTTP headers without any extra information.

This solution provides how to calculate hash value of HTTP headers and HTTP body even though SCP modify HTTP messages in NF Service Producer. And when SCP modifies illegally a HTTP message, NF Service Producer can detect it.

## 6.9 Solution #9: Authorization mechanism negotiation

### 6.9.1 Introduction

This solution addresses Key Issue #7 "Authorization mechanism negotiation". It is proposed to use the two NRFs for the authorization mechanism negotiation.

### 6.9.2 Solution details



Figure 6.x.2-1: Authorization mechanism negotiation

1. The NF Service Consumer sends a discovery request to the vNRF.

2. The vNRF sends a discovery request to the hNRF. In addition to the discovery request from the NFc, the vNRF add the vPLMN authorization Capability into the discovery request. The vPLMN authorization Capability indicates the supported authorization mechanisms, i.e., static, OAuth, or Both.

3. The hNRF selects the Final authorization mechanism supported by both the received vPLMN authorization Capability and the hPLMN authorization Capability.

 Note: The Final authorization mechanism selected by hNRF is depend on operator.

4-5. The hNRF sends the Final authorization mechanism to the NFc.

Then, if the Final authorization mechanism indicates static authorization, then the NFc could use the static authorization to access the NFp service. If the Final authorization mechanism indicates OAuth authorization, then the NFc could get the token from the NRF before consuming the service from the NFp.

### 6.9.3 Evaluation

TBD

## 6.10 Solution #10: NRF deployment clarifications

### 6.10.1 Introduction

This solution addresses key issue #X. It provides input for text that needs to be adapted for clarification of handling access token requests in different NRF deployments

### 6.10.2 Solution details

The following text outlines a potential update to TS 33.501 in a new clause (e.g. 13.4.1.1.1a) on "NRF deployments" with clarification text for NF Service Consumer behaviour and local NRF deployments along the lines:

There are different deployment options for NRFs, as described in TS23.501 (see clause 5.15.5).

The NF Service Consumer may have discovered a specific NRF in advance, e.g. a slice specific NRF, and can send its request directly to this specific NRF. In this case, if the specific NRF is not the NF Service Consumer's local NRF, the authorization server part of this NRF does not have a record of this NF Service Consumer's Oauth2.0 client registration.

Editor’s Note: It is FFS how the specific NRF, e.g., a slice specific NRF, authorizes the NF Service Consumer before offering the requested service.

If the NF Service Consumer requests an NRF, where the NF Service Producer is not registered (see NRF deployment options), the requested NRF needs to redirect/forward the service request to that NRF.

In a local NRF deployment, the NF Service Producer only gets the certificate of its local NRF. Thus, the local NRF of the NF Service Producer would need to trust the forwarding NRF that has authenticated the NF Service Consumer before the local NRF be able to authorize the NF Service Consumer.

### 6.10.3 Evaluation

TBD

## 6.Y Solution #Y: <distinct solution name>

### 6.Y.1 Introduction

Editor's Note: Motivate how the potential security requirements of one or several key issues are addressed by this solution proposal.

### 6.Y.2 Solution details

TBD

### 6.Y.3 Evaluation

Editor's Note: Provide an analysis of the risks of threats mitigated by this solution. Provide a statement on complexity/impact/backward compatibility if one would follow this solution.

# 7 Conclusions

Editor's Note: The purpose of this TR is to make conscious decisions whether 5G SBA security needs to be enhanced to address specific threats and to which price (complexity versus security gain) this is possible. The clause will provide conclusive statements per key issue, i.e. whether and how to move forward with normative work and, if yes, which solutions are endorsed.

## 7.X <distinct KI name>

TBD

Annex A (informative):
Change history

|  |
| --- |
| **Change history** |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2021-01 | SA3#102-e | S3-210420 |  |  |  | Skeleton of TR eSBA SEC | 0.0.0 |
| 2021-01 | SA3#102-e | S3-210679 |  |  |  |

|  |  |
| --- | --- |
| S3-210562 | Introduction |
| S3-210422 | Scope |
| S3-210564 | Authentication of NRF and NFp in indirect communication |
| S3-210565 | SCP deployment models |
| S3-210653 | KI on Verification of UE in subscription and notification in the delegated “Subscribe-Notify” scenarios |
| S3-210566 | KI on Dynamic authorization between SCPs or NF and SCP |
| S3-210567 | End-to-End Critical HTTP headers and body parts integrity protection |

 | 0.1.0 |
| 2021-03 | SA3#102bis-e | S3-211344 |  |  |  |

|  |  |
| --- | --- |
| S3-211224 | Rapporteurs update to 33.875 |
| S3-211217 | Service response verification in indirect communication |
| S3-211218 | More details on SCP deployment models |
| S3-211046 | New Solution to KI#4: Using existing procedures for authorization of SCP to act on behalf of an NF Consumer |
| S3-211220 | NF-SCP authorization |
| S3-211221 | KI details added to End-to-end integrity protection of HTTP messages |
| S3-211205 | New Solution to KI#5: End-to-end integrity protection of HTTP body and method |
| S3-211223 | Service request authenticity verification in indirect communication |
| S3-211225 | Mapping of solutions to key issues |
| Rapporteur additional work done | Updating references, heading numbers and mapping tables in line with TR implementation, updating adding missing ed notes in TBD/empty clauses  |

 | 0.2.0 |
| 2021-05 | SA3#103-e | S3-212297 |  |  |  |

|  |  |
| --- | --- |
| S3-212292 | New solution for key issue#1: Authentication of NF service producer in indirect communication |
| S3-212298 | Sol1 deployment scenarios |
| S3-212299 | ENs on Sol1 Service response verification in model C |
| S3-212300 | EN resolution on sol 2 - NFc authorizing SCP to act on its behalf |
| S3-212301 | Evaluation on sol 2 - NFc authorizing SCP to act on its behalf |
| S3-211973 | KI on Access token usage by all NFs of an NF Set |
| S3-212394 | Solution on Access token request for NF Set |
| S3-212303 | Trust model |
| S3-212372 | Evaluation of Solution #3 "Using existing procedures for authorization of SCP to act on behalf of an NF Consumer" |

 | 0.3.0 |
| 2021-08 | SA3-104-e | S3-21XXXX |  |  |  |

|  |  |
| --- | --- |
| S3-213053 | Sol 1 NFp verification – EN resolutions and evaluation |
| S3-213141S3-213142 | Update on Solution 6Evaluation for solution 1 |
| S3-213166 | Requirement of subscribe-notification key issue |
| S3-213054S3-212888S3-212763S3-213043 | SCP authorizationSCP authorization solution evaluationCorrection of implementation of S3-211046Update to Solution #3 "Using existing procedures for authorization of SCP to act on behalf of an NF Consumer" |
| S3-212764 | Update Solution #5: End-to-end integrity protection of HTTP body and method |
| S3-212930 | Evaluation for solution 4 |
| S3-213143 | Evaluation for solution 5 |
| S3-212928 | New solution on key issue #5 |
| S3-213055 | Access token request for NF Set – EN resolution |
| S3-213056 | Access token request for NF Set – RFC clarification |
| S3-213120 | New Key issue on authorization mechanism negotiation |
| S3-213121 | New solution for the authorization mechanism negotiation |
| S3-213057 | KI and solution to NRF deployments |
| S3-213139 | KI on Authorization for Inter-Slice Access |
| S3-212883 | Editorial update on trust clause |
| S3-213058 | EN resolution on trust model for SCP |

 | 0.4.0 |