|  |  |
| --- | --- |
| 3GPP TR 33.875 V18.1.0 (2023-09) | |
| 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 18) | |
|  | |
|  |  |
|  | |
| 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 address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  http://www.3gpp.org |
| ***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.  © 2023, 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 members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 8

Introduction 9

1 Scope 10

2 References 10

3 Definitions of terms, symbols and abbreviations 11

3.1 Terms 11

3.2 Symbols 11

3.3 Abbreviations 11

4 Trust model 11

4.0 General 11

4.1 Actors 12

4.2 Deployment options 12

4.3 Description of the trust assumptions 13

4.3.0 General 13

4.3.1 Trust within one PLMN 13

4.3.2 Trust in Inter-PLMN communication 14

5 Key issues 15

5.1 Key issue #1: Authentication of NRF and NF Service Producer by the NF Service Consumer in indirect communication 15

5.1.1 Key issue details 15

5.1.2 Security threats 15

5.1.3 Potential security requirements 15

5.2 Key issue #2: Need for additional security at operational level among SCP domains 15

5.2.1 Key issue details 15

5.2.2 Security threats 16

5.2.3 Potential security requirements 16

5.3 Key Issue #3: Service access authorization in the "Subscribe-Notify" scenarios 17

5.3.1 Key issue details 17

5.3.2 Security threats 18

5.3.3 Potential security requirements 18

5.4 Key issue #4: Authorization of SCP to act on behalf of an NF or another SCP 18

5.4.1 Key issue details 18

5.4.2 Security threats 19

5.4.3 Potential security requirements 19

5.5 Key issue #5: End-to-end integrity protection of HTTP messages 19

5.5.1 Key issue details 19

5.5.2 Security threats 19

5.5.3 Potential security requirements 19

5.6 Key issue #6: Access token usage by all consumer NFs of an NF Set 19

5.6.1 Key issue details 19

5.6.2 Security threats 20

5.6.3 Potential security requirements 21

5.7 Key issue #7: Authorization mechanism determination 21

5.7.1 Key issue details 21

5.7.2 Security threats 21

5.7.3 Potential security requirements 21

5.8 Key issue #8: Service access authorization requirements in intra-PLMN scenarios for PLMN deploying multiple NRFs (in OAuth2.0 AS role) 21

5.8.1 Key issue details 21

5.8.1.1 Introduction 21

5.8.1.2 Hierarchical NRFs / Deployment model with local NRFs 22

5.8.1.3 Deployment model with NF Service Consumer directly accessing the NRF where the NF Service Producer is registered 22

5.8.2 Security threats 23

5.8.3 Potential security requirements 23

5.9 Key issue #9: Authorization for Inter-Slice Access 24

5.9.1 Key issue details 24

5.9.2 Security threats 24

5.9.3 Potential security requirements 24

5.10 Key issue #10: N32 security in mediated roaming scenarios 24

5.10.1 Introduction 24

5.10.2 Key issue details 25

5.10.3 Security requirements 25

5.11 Key issue #11: NRF validation of NFc for access token requests 26

5.11.1 Key issue details 26

5.11.1.0 General 26

5.11.1.1 NFc registration at NRF not mandatory (Problem 1a) 26

5.11.1.2 IEs mandated in SBA TLS certificate not sufficient for NFc validation (Problem 1b) 26

5.11.1.3 Questions from Problems 1a and 1b 26

5.11.1.4 TLS certificate or NF profile - precedence in NFc validation (Problem 2) 26

5.11.1.5 Questions from Problem 2 26

5.11.2 Security threats 26

5.11.3 Potential security requirements 27

5.12 Key issue #12: Security in Hosted SEPP scenarios 27

5.12.1 Key issue details 27

5.12.1.1 Background 27

5.12.1.2 Issues to be studied 28

5.12.2 Threats 28

5.12.3 Security requirements 28

6 Solutions 30

6.0 Mapping of solutions to key issues 30

6.1 Solution #1: Verification of the entity sending the service response in indirect communication without delegated discovery 31

6.1.1 Introduction 31

6.1.2 Solution details 31

6.1.3 Evaluation 32

6.2 Solution #2: Authorization between NFs and SCP 33

6.2.1 Introduction 33

6.2.2 Solution details 33

6.2.3 Evaluation 34

6.3 Solution #3: Using existing procedures for authorization of SCP to act on behalf of an NF Service Consumer 34

6.3.1 Introduction 34

6.3.2 Solution details 35

6.3.2.1 Request of access token on behalf of the consumer 35

6.3.2.2 Service request on behalf of the consumer 35

6.3.2.4 Protection of the NF Service Consumer's CCA 36

6.3.3 Evaluation 37

6.4 Solution #4: Service request authenticity verification in indirect communication 37

6.4.1 Introduction 37

6.4.2 Solution details 37

6.4.3 Evaluation 38

6.5 Solution #5: End-to-end integrity protection of HTTP body and method 38

6.5.1 Introduction 38

6.5.2 Solution details 39

6.5.3 Evaluation 40

6.6 Solution #6: Verification of Service Response from a NF Service Producer at the expected NF Set 41

6.6.1 Introduction 41

6.6.2 Solution details 41

6.6.2.1 For indirect communication without delegated discovery procedure 41

6.6.2.2 For indirect communication with delegated discovery 42

6.6.2.3 Client credentials assertion of NF Service Producer 43

6.6.3 Evaluation 44

6.7 Solution #7: Access token request for NF Set 44

6.7.1 Introduction 44

6.7.2 Solution details 45

6.7.3 Evaluation 46

6.8 Solution #8: Integrity protection of HTTP message in consideration of update by SCP 47

6.8.1 Introduction 47

6.8.2 Solution Details 47

6.8.3 Evaluation 48

6.9 Solution #9: Authorization mechanism negotiation 48

6.9.1 Introduction 48

6.9.2 Solution details 48

6.9.3 Evaluation 49

6.10 Solution #10: NRF deployment clarifications 49

6.10.1 Introduction 49

6.10.2 Solution details 49

6.10.3 Evaluation 50

6.11 Solution #11: Registered NF Profile changes for Inter-Slice Access 50

6.11.1 Introduction 50

6.11.1 Solution Details 50

6.11.3 Evaluation 50

6.12 Solution #12: Authorization of notification endpoint in "Subscribe-Notify" scenarios 51

6.12.1 Introduction 51

6.12.2 Solution details 51

6.12.3 Evaluation 52

6.13 Solution #13: Authentication of NF Service Producer in Indirect Communication 53

6.13.1 Introduction 53

6.13.2 Solution details 53

6.13.3 Evaluation 54

6.14 Solution #14: SCP trust domain or technical domain grouping 54

6.14.1 Introduction 54

6.14.2 Solution details 54

6.14.3 Evaluation 56

6.15 Solution #15: Authorization mechanism for the involved NFs in the delegated "Subscribe-Notify" scenario. 56

6.15.1 Introduction 56

6.15.2 Solution details 56

6.15.3 Evaluation 58

6.16 Solution #16: Selective End of End Protection of HTTP Request and Response in Indirect Communication 58

6.16.1 Introduction 58

6.16.2 Solution details 58

6.16.3 Evaluation 59

6.17 Solution #17: Authorization mechanism negotiation using existing methods 59

6.17.1 Introduction 59

6.17.2 Solution details 59

6.17.3 Evaluation 61

6.18 Solution #18: Avoiding slice isolation violation 61

6.18.1 Introduction 61

6.18.2 Solution details 61

6.18.3 Evaluation 62

6.19 Solution #19: Hosted SEPP requirements 62

6.19.1 Introduction 62

6.19.2 Solution details 62

6.19.3 Evaluation 63

6.20 Solution #20: PRINS for Roaming Hubs 63

6.20.1 Introduction 63

6.20.2 Solution details 64

6.20.3 RH Proxy Resolves pSEPP Well-Known FQDN 67

6.20.4 Evaluation 67

6.21 Solution #21: Certificate solution for NRF validation of NFc for access token requests 68

6.21.1 Introduction 68

6.21.2 Solution details 68

6.21.2.1 NF Service Consumer information to validate at Service Request Authorization 68

6.21.2.2 Certificates 68

6.21.2.3 NRF validation solution 68

6.21.3 Evaluation 69

6.22 Solution #22: Combined certificate and profile solution for NRF validation of NFc for access token requests 69

6.22.1 Introduction 69

6.22.2 Solution details 69

6.22.2.1 NF Service Consumer information to validate at Service Request Authorization 69

6.22.2.2 O&M Provisioning solution 69

6.22.2.3 Certificates 69

6.22.2.4 NRF validation solution 70

6.22.3 Evaluation 70

6.23 Solution #23: SCP authorization check by NRF 70

6.23.1 Introduction 70

6.23.2 Solution details 70

6.23.2.1 Enabling NRF to check on SCP information 70

6.23.2.2 Including service request information into the CCA 71

6.23.3 Evaluation 72

6.24 Solution #24: Authorization negotiation with bootstrapping mechanism 72

6.24.1 Introduction 72

6.24.2 Solution details 72

6.24.3 Evaluation 73

6.25 Solution #25: Solution on N32 security profiles 73

6.25.1 Introduction 73

6.25.2 Solution details 74

6.25.3 Evaluation 74

6.26 Solution #26: Authorization of NF Service Consumer accessing Nnrf\_AccessToken service 75

6.26.1 Introduction 75

6.26.2 Solution details 75

6.26.3 Evaluation 75

7 Conclusions 77

7.1 KI#1: Authentication of NRF and NF Service Producer in indirect communication 77

7.1.1 Analysis 77

7.1.2 Conclusion 77

7.2 KI#2: Need for additional security at operational level among SCP domains 78

7.2.1 Analysis 78

7.2.2 Conclusion 78

7.3 KI#3: Service access authorization in the "Subscribe-Notify" scenarios 78

7.3.1 Analysis 78

7.3.2 Conclusion 78

7.4 KI#4: Authorization of SCP to act on behalf of an NF or another SCP 78

7.4.1 Analysis 78

7.4.2 Conclusion 79

7.5 KI #5: End-to-end integrity protection of HTTP messages 79

7.5.1 Analysis 79

7.5.2 Conclusion 80

7.6 KI#6: Access token usage by all NFs of an NF set 80

7.6.1 Analysis 80

7.6.2 Conclusion 80

7.7 KI#7: Authorization mechanism determination 80

7.7.1 Analysis 80

7.7.2 Conclusion 81

7.8 KI#8: Service access authorization requirements in intra-PLMN scenarios for PLMN deploying multiple NRFs (in OAuth2.0 AS role) 81

7.8.1 Analysis 81

7.8.2 Conclusion 81

7.9 KI #9: Authorization for Inter-Slice Access 81

7.9.1 Analysis 81

7.9.2 Conclusion 82

7.10 KI #10: N32 security in mediated roaming scenarios 82

7.10.1 Analysis 82

7.10.2 Conclusion 82

7.11 KI #11: NRF validation of NFc for access token 82

7.11.1 Analysis 82

7.11.2 Conclusion 83

7.12 KI #12: Security in Hosted SEPP scenarios 83

7.12.1 Analysis 83

7.12.2 Conclusion 83

Annex A: Change history 84

# 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".

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 security was set on providing solutions for authentication and authorization in direct communication scenarios as well as the N32 roaming 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;

- Access token usage in NF Sets;

- Authorization mechanism determination;

- Security of NRF service management;

- Inter-Slice access authorization;

- N32 roaming security considerations for deployment scenarios including roaming hub and hosted SEPP.

# 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".

[7] 3GPP TS 23.502: "Procedures for the 5G System (5GS); Stage 3".

[8] GSMA NG.113: "5GS Roaming Guidelines"; <https://www.gsma.com/newsroom/wp-content/uploads/NG.113-v6.0.pdf>.

[9] 3GPP TS 29.573: "5G System; Public Land Mobile Network (PLMN) Interconnection; Stage 3".

[10] GSMA PRD FS.34 v5.0, Key Management for 4G and 5G inter-PMN Security.

[11] GSMA PRD IR.67 v20.0, DNS Guidelines for Service Providers and GRX and IPX Provider

[12] 3GPP TS 23.228: " IP Multimedia Subsystem (IMS); Stage 2".

[13] 3GPP TS 29.531: " 5G System; Network Slice Selection Services; Stage 3".

[14] 3GPP TS 33.310: " Network Domain Security (NDS); Authentication Framework (AF)".

[15] IETF RFC 7519: " JSON Web Token (JWT)".

[16] IETF RFC 7515: " JSON Web Signature (JWS)".

[17] IETF RFC 6749: " The OAuth 2.0 Authorization Framework".

[18] 3GPP TS 23.003: "Numbering, addressing and identification".

[19] 3GPP TS 23.288: "Architecture enhancements for 5G System (5GS) to support network data analytics services".

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

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

1. **traffic shaper:** a network device that is used to limit the throughput for individual connections by delaying or dropping packets.

## 3.2 Symbols

Void

## 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and TS 33.501 [2] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1] and TS 33.501 [2].

NFc NF service consumer

NFp NF service producer

SRV service request verification

HL header list

# 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. 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 to fulfil these security tasks 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 services to other NFs. NRF receives discovery requests from NF instances, maintains NF profiles and acts as an OAuth 2.0 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 in intra-PLMN communication:

- 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.

The following deployment options need to be considered in inter-PLMN communication:

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 also provides for authentication between two entities. Thus, securing at transport layer provides hop-by-hop security between two SEPPs.

TLS encryption can be applied for N32-f where the operator only allows for IP level routing by intermediaries. Otherwise, in deployments with intermediaries tasked to provide additional services, hop-by-hop TLS security alone would allow any intermediary to see and modify messages. More precisely,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.

Since there exist also deployments where intermediaries need to see or modify messages, PRINS was designed. The PRotocol for N32 Interconnect Security (PRINS) provides end-to-end application layer security between two SEPPs and hence an operator-controlled possibility for intermediaries to read and modify messages.

## 4.3 Description of the trust assumptions

### 4.3.0 General

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.

### 4.3.1 Trust within one PLMN

This clause describes the existing trust relationships within one PLMN.

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

NRF is the core entity handling management, 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 its service profile 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 TS 33.501 [2] 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 co-located to NFs (service mesh) or standalone.

If a SCP is co-located (e.g., a side-car proxy in service mesh) with a NF, this SCP is performing many of the functionalities on behalf of the NF Service Consumer, which already indicates a certain level of trust between NFs with co-located SCPs.

If a SCP is standalone, this SCP is serving many NFs, not necessarily in the same infrastructure or operated by the same operator than the NFs using the SCP. It still needs to provide secure communication access for all NFs communicating via this SCP.

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

### 4.3.2 Trust in Inter-PLMN communication

With 5G, a new element has been introduced to handle inter-PLMN 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 (such as IPX, RVAS, or roaming hub) 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 by the NF Service Consumer 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 (NFc) and NRF or NFc and NF Service Producer (NFp) 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 or 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

The request by the NF Service Consumer could be forwarded by the SCP to an unintended NF.

The request by the NF Service Consumer could be forwarded by the SCP within the validity of an authorization token more than once to the same NF, which could result in a deny of service attack.

In indirect communication scenarios an NF Service Consumer cannot verify whether the service response was sent by a legitimate NF or NRF. NFc can only authenticate the SCP, but not NFp or NRF. Thus, 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.

NOTE: It needs to be taken into account that producer reselection by SCP can be a desired feature.

## 5.2 Key issue #2: Need for additional security at operational level among SCP domains

### 5.2.1 Key issue details

TS 23.501 [3] addresses the aspects of handling multiple SCPs in indirect communication without and with delegated discovery. A SCP domain comprises a configured group of one or multiple SCPs that can reach certain NF instances or SCPs directly.

SCPs can register a SCP profile in NRF. TS 23.502 [7] describes in the SCP profile SCP domain registration details about interconnected SCPs and also identifies SCPs that interconnect domains. Table 6.1.6.2.2-1 of TS 29.510 [6] provides a detailed description. 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.

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, the domain concept could be used for establishing operational restrictions by defining SCP domains. Or, there can be several technical SCP 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.

The figure below gives an idea of structuring or grouping a PLMN in several SCP domains, possibly interconnecting with other SCP domains via dedicated SCPs.

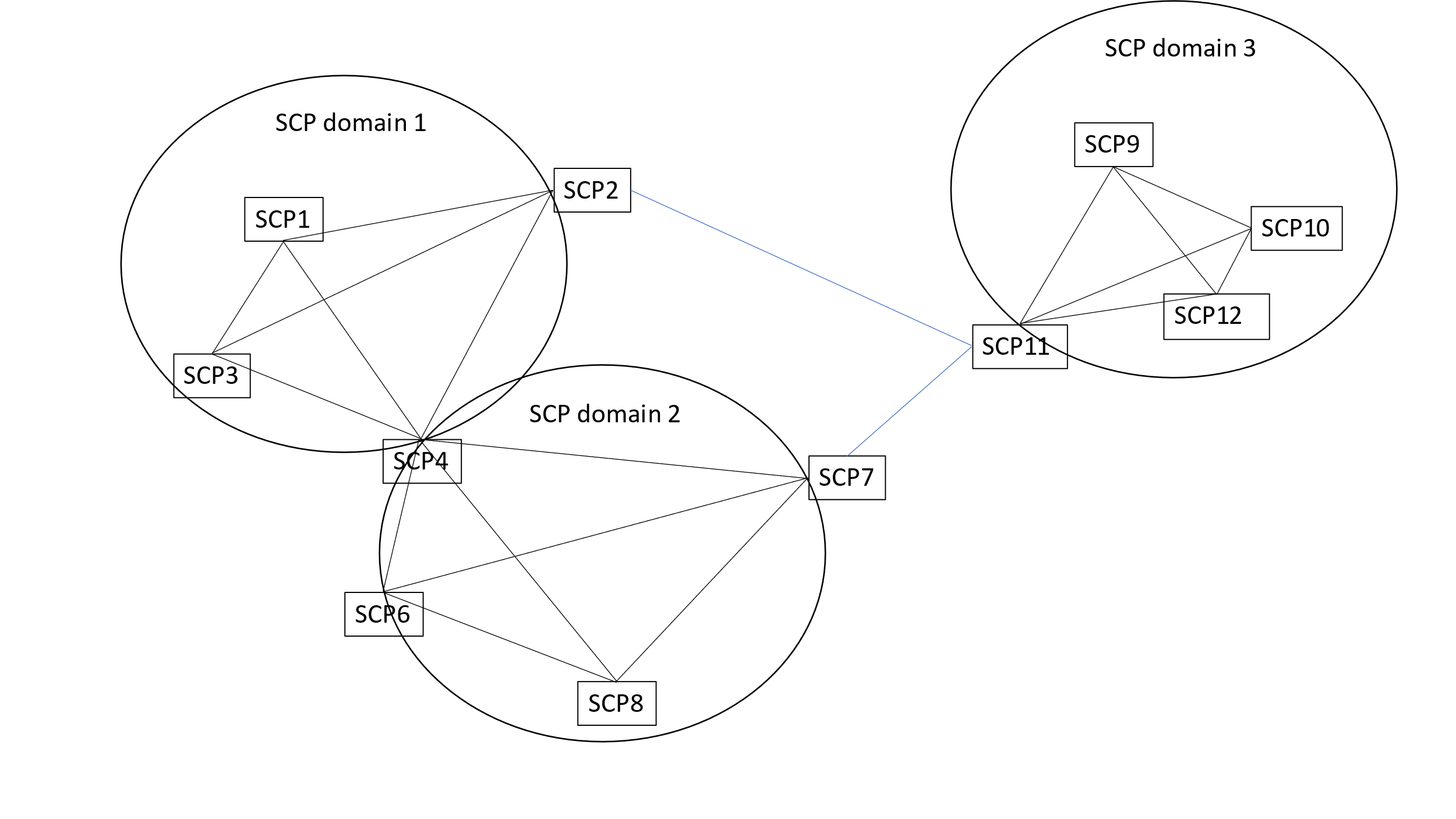


Figure 5.2.1-1: Illustration of SCP domains connecting via dedicated SPCs

This key issue is to study the concept 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

In large PLMNs operated by different organizations in different regions/provinces or by slices operated by a tenant, PLMN wide trust between NFs and SCPs could result in messages traversing over several SCP domain boundaries, even though this is not wanted. It could be not possible to shield, e.g., regions or organizations against each other, resulting in availability of information to unauthorized parts of the network.

If an access token has been requested by one SCP on behalf of the NF Service Consumer and granted by NRF, it is not guaranteed that the same SCP is forwarding the service request to the NF Service Producer. Instead, the SCP could forward the service request including the access token to another SCP first. If the new SCP is not be part of the same SCP domain, the trust in this SCP could be less than in SCPs within the same SCP domain.

### 5.2.3 Potential security requirements

Potential security requirements are not addressed in the present document. Please refer to clause 5.2.4.

5.2.4 Potential architectural/operational security requirements

The 5G system shall allow restricting network topology so that requests between certain NFs or SCPs or SCP domains are not allowed.

## 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 [3], clause 7.1.2, allows one NF (e.g., NF\_A) to subscribe to notifications of NF Service 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 [3], clause 7.1.2, allows one NF (e.g., NF\_A) to subscribe the service of NF Service 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 [7] 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 arises because of the subscribe notify use cases 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.

Authorization for the subscribe-notify scenarios is not described explicitly in TS 33.501 [2]. However, it is described in clause 6.7.3 of TS 29.500 [5]: "For request/response semantics service operations and for the subscribe and unsubscribe operations of subscribe/notify semantics service operations, an NF service consumer may use OAuth 2.0 for the authorization of the API access, based on local configuration." Further: "For the notify operation of subscribe/notify semantics service operations, in this release of this specification OAuth 2.0 access token is not used."

This key issue seeks for solutions whether additional means are needed on how to assure that the notification messages could be only forwarded to an authorized NF by the NRF and that notifications cannot be leaked.

### 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 [3], "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 this information can be leaked to an unauthorized NF, while according to TS 23.288 [19] 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.

Sensitive information in the notification could be disclosed to an unauthorized entity.

An NF could receive unwanted notifications, leading to a denial of service.

The sender of the notification could be misled to disclose information to an NF in a different PLMN that the sender is not permitted to disclose to that PLMN.

### 5.3.3 Potential security requirements

The same security requirements and means as specified for request/response shall apply.

NOTE 1: TS 33.501[2] lacks security requirements for subscribe/notify, which can be the same as request/response in most use cases. If requirements are already fulfilled by existing means in clause 13.4.1 of TS 33.501[2], this can be stated as part of the solution part.

In detail:

It shall be possible for the 5G system to ensure notification service is only provided to an authorized NF to which a notification message is routed (identified by the URI in the subscribe request message). I.e., an NF shall be authorized to receive notification messages. The usage of notification URI shall be authorized.

It shall be possible for the 5G system to ensure that the subscriber of notification messages is authorized to subscribe for notifications on behalf of the NF Service Consumer.

The 5G system should mitigate information disclosure of sensitive information to an unauthorized entity.

The 5G system should mitigate denial of service due to unwanted notifications.

NOTE 2: The following requirement may not be fulfilled, because it is out of hand if the an authorized notification recipient is disclosing the received information.

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

## 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 Service 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 need to be considered.

### 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.

NOTE: Critical HTTP elements would need to be determined by stage 3.

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 consumer 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, whereby 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:**

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 TS 23.501 [3]/TS 29.500 [5], but the related security aspects of using such optimization have not been addressed in TS 33.501 [2]. 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/useable 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 useable for achieving this. Otherwise, it could end in massive access token requests that are used in the same context of service consumption.

If an access token cannot be used by any NF 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 targeting 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 each time 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

In general, the concept of access token is already in place. No new threats need to be listed.

### 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.

If an access token provided for all NFs of the same NF Set is used, its usage shall be limited to this specific 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 a 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 oauth2Required 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 should 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 [3] 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 2.0 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 2.0 client needs to be authenticated and by which NRF an OAuth 2.0 client gets the access token after authorization.

Only the NRF where the NF Service Producer has registered its services can act as the OAuth 2.0 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 [2] 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 considers 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 2.0 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 OAuth 2.0 Authorization Server role), i.e., with an NFc registered as OAuth 2.0 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 OAuth 2.0 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 2.0 client at a NRF that is not necessarily close to NFc, e.g., an AMF registered in a PLMN wide NRF.

#### 5.8.1.3 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 OAuth 2.0 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 OAuth 2.0 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 should use for a specific network slice (see TS 29.531 [13]).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 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.



Figure 5.8.1.3-1: From TS 23.502 [7], 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, TS 23.502 [7];

- 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, TS 23.502 [7].

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 TS 29.531 [13], clause 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 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 OAuth 2.0 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

In the current SBA authorization framework, the mechanisms specified to validate the S-NSSAI of NF Service Consumer are not fully specified and therefore, sometimes an NF Service Producer and/or NRF is dependent on local configuration or proprietary mechanisms to verify if the NF Service Consumer really belongs to the S-NSSAI it is claiming to belong to. While an NRF could validate the S-NSSAI by comparing the S-NSSAI provided in, e.g., Access-Token-Get request with the S-NSSAI registered in the NF-Profile; this is not always possible, as it is not necessary that every NF registers its profile with NRF, unless the information is available to NRF by other means (e.g., local configuration). This results in a NF Service Consumer gaining access to slices it should not have been allowed to access.

Additionally, currently there is no mechanisms specified whereby an NRF could validate if an NF Service Consumer is allowed to access resources belonging to a different S-NSSAI in the NF Service Producer. While an NRF could specify, in the Access Token Grant, the S-NSSAIs an NF Service Consumer is allowed to access in the NF Service producer, it is assumed that NRF uses local configurations while providing such information.

This Key Issue will study how to prevent any malicious entity (for instance a NF Service Consumer) from accessing a slice it is not authorized to access, or from requesting a service from a slice which it is not authorized to access.

NOTE: Assuming that an NF within a 3GPP network can be fully compromised would be 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

A malicious entity (for instance a NF Service Consumer) can request an access token for the slice it is not authorized to access.

A malicious entity (for instance a NF Service Consumer) can send a service request for the slice which it is not authorized to access.

### 5.9.3 Potential security requirements

The 5GS should provide a mechanism that allows the NF Service Producer not to provide service to NF Service Consumers, which are not authorized to access a slice.

The 5GS should provide a mechanism that allows NRF not to provide access tokens to NF Service Consumers which are not authorized to access a slice.

## 5.10 Key issue #10: N32 security in mediated roaming scenarios

### 5.10.1 Introduction

In roaming scenarios, the communication between the visited network and the home network is sometimes mediated and routed through intermediaries. The following types of intermediaries exist.

**IPX providers:** While, traditionally, for some roaming relations, they simply route the traffic, for others they actively modify certain messages in order to establish or enhance interoperability.

**Roaming hubs:** There are two types of roaming hub, as follows.

**-** **Public Roaming Hubs** offer a contractual and technical framework that enables operators to avoid entering a very large number of roaming agreements with individual other operators in order to achieve a large roaming coverage. By contracting the roaming hub, the operator obtains access to a large roaming footprint, without individually contracting the operators that are mediated through the hub. Apart from mediating the signalling, roaming hubs typically also mediate the billing.

**- Operator Group Roaming Hubs** offer similar services as public roaming hubs, and are special in that they are used by group network operator companies that reside in the same security domain to consolidate and secure operator group roaming.

The key issue looks at N32 security in intermediary scenarios, in particular roaming hub scenarios that have not been addressed in TS 33.501[2] so far.

### 5.10.2 Key issue details

The GSMA is working on best practice guidance and related specifications for how operators interact with both IPX and roaming hubs. However, the pre-5G architecture of these methods uses the LTE hop-by-hop security paradigm. In 5G, SEPP communication for roaming relies on end-to-end security by design as specified in TS 33.501 [2], i.e., messages are authenticated and integrity-protected between the SEPP of the visited network and the SEPP of the home network. This creates the need to study in more detail how the migration away from hop-by-hop security can be achieved without loss of the benefits provided by IPX providers and roaming hubs while the 5G security requirements on N32 are met.

In certain scenarios, IPX providers and roaming hubs would need to modify certain IEs. This requires the setup of appropriate modification policies between the roaming partners. Given that, according to TS 33.501 [2],

- all attributes transferred over the N32-f interface shall be integrity protected (clause 5.9.3.3),

- a default data-type encryption policy, which provides confidentiality protection for authentication vectors, cryptographic material and location data, and optionally for SUPI, applies (clause 5.9.3.3), and

- any roaming partner-specific data-type encryption policy takes precedence over the default policy (clause 13.2.3.6),

and further given that:

- it is assumed that the set of IEs to be modified depends on the deployment scenario; and

- no default modification policy has been specified;

- maintaining a large set of roaming partner-specific modification policies could cause avoidable complexities. It is, therefore, unclear whether the current specification needs to be extended in order to avoid such complexities.

Furthermore, a roaming hub may need to prevent the setup of N32-c between a home network SEPP and a visited network SEPP, e.g., in case the respective operators have not established a commercial roaming relation via the roaming hub.

### 5.10.3 Security requirements

Existing requirements in TS 33.501[2] on the SEPP should apply in the roaming intermediary scenario unless there are explicit exceptions.

Specifically, the requirements on edge protection of the PLMN should still apply, including end-to-end security for N32-c.

Existing NFs and SCPs should be impacted as least as possible.

For the scenarios involving intermediaries (IPX providers, roaming hubs), requirements are as follows.

- The intermediaries (IPX providers, roaming hubs) shall use their own unique credentials to authenticate themselves.

## 5.11 Key issue #11: NRF validation of NFc for access token requests

### 5.11.1 Key issue details

#### 5.11.1.0 General

There are several procedures for access token requests in TS 33.501 [2] clause 13.4.1, each for different scenarios such as direct or indirect communication. The description of how the NRF authorizes the access token request are more or less detailed. The most detailed description is in clause 13.4.1.1.2, Access token request for accessing services of NF Service Producers of a specific NF type: "The NRF may verify that the input parameters (e.g., NF type) in the access token request match with the corresponding ones in the public key certificate of the NF Service Consumer or those in the NF profile of the NF Service Consumer. The NRF checks whether the NF Service Consumer is authorized to access the requested service(s)." Other procedures either refer to clause 13.4.1.1.2 or contain even less detailed descriptions.

There are several issues with how authorization of access token requests is described in TS 33.501 [2].

#### 5.11.1.1 NFc registration at NRF not mandatory (Problem 1a)

Not all NF Service Consumers register an NF profile at the NRF. This results in no NF profile is available. Furthermore, the NF profile is provided by the NF itself, which makes it a less reliable source for deciding whether the NF is authorized.

#### 5.11.1.2 IEs mandated in SBA TLS certificate not sufficient for NFc validation (Problem 1b)

It is not a mandatory requirement that all IEs defined in the SBA TLS certificate has been filled in with information presented in the TLS certificate. For example, NF Type is only "should". Other kind of information, like slice information, is not contained in the TLS certificate profile at all. So, there may be situations when certificates are available, but information may not be sufficient for NFc validation in NRF. It is not specified how the NRF shall validate the NFc request in all situations.

#### 5.11.1.3 Questions from Problems 1a and 1b

The study needs to elaborate on the following questions:

1. What IEs are required to validate as a minimum?

2. What information shall be used in NRF to validate the NFc?

3. How is this information provisioned in NRF?

#### 5.11.1.4 TLS certificate or NF profile - precedence in NFc validation (Problem 2)

It is also possible that the NFc TLS certificate is available together with the NF profile registered in the NRF. In this case it is unclear what precedence the NRF shall use when performing the NFc validation. The information in the profile is provided by the NFc itself, but the information in the certificate is provided outside of the NFc, i.e., the CA.

#### 5.11.1.5 Questions from Problem 2

The study needs to elaborate on the following questions:

1) What information shall have precedence when NFc certificate is available as well as the NF profile?

2) What information shall be cross-checked between the certificate and the profile?

### 5.11.2 Security threats

The security level may be NRF vendor specific if not specified.

If there is no standardized agreement of the solution to the Key Issue the interoperability between different vendors of NFc and NRF may result in being too allowing from a security perspective.

### 5.11.3 Potential security requirements

This KI does not add any new security requirements but is focused on to develop solutions to the problems stated above.

## 5.12 Key issue #12: Security in Hosted SEPP scenarios

### 5.12.1 Key issue details

#### 5.12.1.1 Background

It has been discussed that SEPPs can be deployed in different ways:

- Local SEPP: The SEPP is deployed within the PLMN. This is the deployment as specified by TS 33.501 [2].

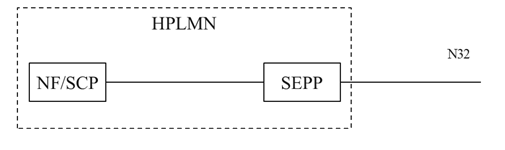


Figure 5.12.1-1 Local SEPP deployment

- Outsourced SEPP: The SEPP is outsourced by the PLMN and deployed within the PLMN infrastructure and considered within the PLMN secure domain Several PLMNs which form an Operator Group can use the same outsourced SEPP. This scenario is described as an Operator Group Roaming Hub in KI#10 and GSMA NG.113 [8].

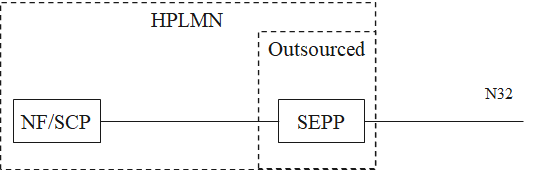


Figure 5.12.1-2 Outsourced SEPP deployment

- Hosted SEPP: The SEPP is outsourced by the PLMN and hosted by an external entity and deployed outside the PLMN. Such a SEPP could be used by different PLMNs.

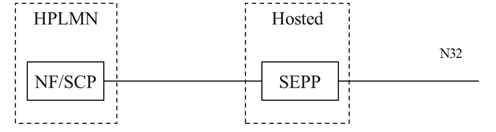


Figure 5.12.1-3 Hosted SEPP deployment

The "Hosted SEPP" scenario is subject of this key issue.

#### 5.12.1.2 Issues to be studied

In a Hosted SEPP scenario it is necessary to consider different perspectives when assessing the security perimeter boundaries. From the roaming partners' perspective, the security perimeter of the PLMN extends to the Hosted SEPP provider, as described in TS 33.501, clause 4.2.1. From the perspective of the PLMN itself, the Hosted SEPP provider operates within its own security domain. Since the Hosted SEPP potentially serves multiple PLMNs as the Hosted SEPP is a service by a service provider, the requirements for introducing such Hosted SEPP need to be clarified.

Nevertheless, in the Hosted SEPP deployment scenario, the Hosted SEPP is similarly to any SEPP responsible for

- Terminating the N32c/f interface (TLS, PRINS) - N32 side

- Providing topology Hiding - N32 side

- Providing filtering criteria and capabilities – N32 side

- Filtering criteria and possibly a firewall – Home PMN side.

The following issues are required for studying:

- Whether the external entity, e.g., the IPX, which hosts the SEPP, uses certificates from its own CA or uses certificates issued from the Home PLMN (i.e., all PLMNs that want to use the Hosted SEPP service) and whether the Home PLMN is willing to consider and use, e.g., the IPX's CA as trusted CA;

- Whether the Hosted SEPP can handle private addresses of the Home PMN;

- Whether the Hosted SEPP is trusted to enforce market specific regulatory requirements;

- In addition, the connection between Hosted SEPP and the Home PMN needs protection, e.g., Topology Hiding, filtering criteria and firewall, etc.

The PKI infrastructure can be deployed in different ways for issuing the digital identities to the hosted SEPP's PLMN facing SBI and external facing N32.

A PLMN can use both Local SEPPs (managed by PLMN) and Hosted SEPPs, i.e., the PLMN has Local SEPPs each handling a set of roaming relations, while the Hosted SEPPs handle a different set of roaming relations.

### 5.12.2 Threats

In a scenario of Hosted SEPP additional risks arise. These include:

- attacks on the traffic between PLMN's NFs and the PLMN's Hosted SEPP,

- insufficient protection of the PLMN core,

- lack of a way to attribute the cause of a security issue to a specific actor (e.g., the operator or the Hosted SEPP provider),

- an unauthorized actor claiming to be a Hosted SEPP provider of an operator towards its roaming partners, and

- risks due to one actor operating Hosted SEPPs for multiple PLMNs (overlapping PLMN security domains).

### 5.12.3 Security requirements

The Hosted SEPP scenario introduces security requirements for the connection between PLMNs and the Hosted SEPP operator.

Existing requirements in TS 33.501[2] on the SEPP shall apply in the Hosted SEPP scenario, unless there are explicit exceptions. Specifically, edge protection requirements can differ from the existing requirements since the hosted SEPP is not deployed at the edge of the PLMN.

Existing NFs and SCPs should be impacted as least as possible.

Further, the following requirements shall apply:

- The Hosted SEPP providers shall use their own unique credentials to authenticate themselves. Moreover, it shall be possible for operators that receive signalling from a Hosted SEPP provider pertaining to a roaming partner to verify that the Hosted SEPP provider has been authorised by the roaming partner.

- The connection between the PLMN and the Hosted SEPP provider shall be confidentiality, integrity, and replay protected.

- The solution should enable the Hosted SEPP providers to operate SEPPs for multiple PLMNs in a way that isolates the SEPP instances operated for different PLMNs from each other for security reasons. Specifically, if one of the instances gets corrupted or otherwise malfunctions, other instances should remain unaffected as much as possible.

# 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 | #7 | #8 | #9 | #10 | #11 | #12 |
| #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 Service 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 |  |  |  |  |  |  |
| #8: integrity protection of HTTP message in consideration of update by SCP |  |  |  |  | X |  |  |  |  |  |  |  |
| #9: Authorization mechanism negotiation |  |  |  |  |  |  | X |  |  |  |  |  |
| #10: NRF deployment clarifications |  |  |  |  |  |  |  | X |  |  |  |  |
| #11: Registered NF Profile changes for Inter-Slice Access |  |  |  |  |  |  |  |  | X |  |  |  |
| #12: Authorization of notification endpoint in "Subscribe-Notify" scenarios |  |  | X |  |  |  |  |  |  |  |  |  |
| #13: Authentication of NF Service Producer in Indirect Communication | X |  |  |  |  |  |  |  |  |  |  |  |
| #14: SCP trust domain or technical domain grouping |  | X |  |  |  |  |  |  |  |  |  |  |
| #15: Authorization mechanism for the involved NFs in the delegated "Subscribe-Notify" scenario |  |  | X |  |  |  |  |  |  |  |  |  |
| #16: Selective End of End Protection of HTTP Request and Response in Indirect Communication |  |  |  |  | X |  |  |  |  |  |  |  |
| #17: Authorization mechanism negotiation using existing methods |  |  |  |  |  |  | X |  |  |  |  |  |
| #18: Avoiding slice isolation violation |  |  |  |  |  |  |  |  | X |  |  |  |
| #19: Hosted SEPP requirements |  |  |  |  |  |  |  |  |  |  |  | X |
| #20: PRINS for Roaming Hubs |  |  |  |  |  |  |  |  |  | X |  |  |
| #21: Certificate solution for NRF validation of NFc for access token requests |  |  |  |  |  |  |  |  |  |  | X |  |
| #22: Combined certificate and profile solution for NRF validation of NFc for access token requests |  |  |  |  |  |  |  |  |  |  | X |  |
| #23: SCP authorization check by NRF |  |  |  | X |  |  |  |  |  |  |  |  |
| #24: Authorization negotiation with bootstrapping mechanism |  |  |  |  |  |  | X |  |  |  |  |  |
| #25: N32 security profiles |  |  |  |  |  |  |  |  |  | X |  |  |
| #26: Authorization of NF Service Consumer accessing Nnrf\_AccessToken service |  |  |  |  |  |  |  |  |  |  | X |  |
| #27: Dedicated N32 connections per PLMN in hosted SEPP deployment |  |  |  |  |  |  |  |  |  |  |  | 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 TS 33.501 [2] 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 6.1.1-1: 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 6.1.2-2: 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 [2]. 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 needs to 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 communication

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 [14]. 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

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 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 Service 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 Service Consumer, i.e., to request access tokens or services on behalf of the consumer. The solution relies on implicit authorization of the SCP using mutual TLS between NF Service Consumer and SCP.

### 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 Service 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 Service 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 Service Consumer. For Scenario C, the same principles hold.



Figure 6.3.2.2-1: Service request of SCP on behalf of an NF Service 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 [2]. 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 [2].

#### 6.3.2.4 Protection of the NF Service 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.

### 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. No e2e authorization information is provided from the NFc to NRF or NFp. 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.

NOTE 1: To avoid backward compatibility issues and allow to distinguish between releases, a new API URI with the respective release version can be used.

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. If not the hash but the whole message or header is included, impact on the throughput is expected.

- 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 2: 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.

NOTE 3: A keyed hash value is not necessary, because the hash is signed digitally in the CCA.

- 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 should be transmitted to the NF Service Producer to inform HTTP headers and order among HTTP headers which should 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 should 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.

## 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 Service 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 TS 29.500 [5] Table 5.2.3.2.11-1, this solves the backwards compatibility with Rel-16 NF Service Producers supporting only existing CCA. A Rel-16 NF Service 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 NF Service 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 behavior can be specified for IEs in the CCA, see Table 6.5.2-1 below.

### 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 TS 33.501 [2], 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.

The updated definition of type ClientCredentialsAssertion in TS 29.500 [5] is (additions in **bold** style):

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 [15], 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 [15], 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 [15], 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 [15], 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 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.

### 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. It extends solution #1. In order to verify the message from NF Service Producer in indirect communication, it is proposed to append CCA of NFp. An NF Service Consumer may accept the certificate if it is verified well and NF Service Producer instances belongs to the expected NF Service 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 Service 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 Service 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 TS 33.501 [2]). 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 TS 33.501 [2]). 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 TS 33.501 [2]). 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 TS 33.501 [2]). 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 TS 33.501 [2]). 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 Consumer.

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, and may verify whether the CCA of the NF Service Consumer in the CCA of NF Service Producer is the same with the CCA of the NF Service Consumer sent in Step 1.

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

CCAs shall be JSON Web Tokens as described in RFC 7519 [15] and are secured with digital signatures based on JSON Web Signature (JWS) as described in RFC 7515 [16].

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;

- The CCA of the NF Service Consumer received in the service request message;

- A timestamp and an expiration time.

The NF Service Consumer digitally signs the generated CCA based on its private key as described in RFC 7515 [16]. The signed CCA includes 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 or a token alike also for the NF Service Producer. The assertion includes NFp Instance ID, NFc Instance ID, and a signature using the 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 that SCP only re-selects another NF Service Producer within the NF Set.

NFc can only trust NFp if there has been a direct possibility to verify NFp's authenticity. This is done by proposing an assertion token (such as CCA) to be also used by NFp.

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 address 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.

To support the NF Set concept, i.e., a group of interchangeable NF instances or NF Service instances providing the same service, the context needs to be accessible to all NFs of the set.

The solution addresses NF Sets or NF Service Sets on the consumer side. It assumes that each NF Service Consumer of a set is known 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 Service Consumer as OAuth 2.0 client when registering at NRF. Thus, if NRF is then issuing an access token with a distinct set id related to the NF Service Consumer's NF Set IF or NF Service 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 1: 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 targeting a service of an existing resource it can use the access token provided to a NF Set of NF Service Consumers. All NFs of a NF Set need to retrieve the current status from the same place to allow for the same access token to be used by different NFs of the set.

NOTE 2: 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 useable by all NF Service Consumer instances within the NF Set. If NRF authorization of the NF Service Consumer is successful, i.e., 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. The request needs to put the context information of such common token at a place from which all NFs of the NF Set can retrieve it.



Figure 6.7.2-1: 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.

There is always one NF representing a set at a given time. Only one NF of the NF Set or NF Service Set shall be able to use the access token issued for a set at a given time. The management of which NF within the set is currently the leader and using the OAuth token is for implementation and out of scope of the present document.

### 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 enables 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 [17]. 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 [17], 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 intervention 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 should be destroyed and not used. A new access token is required.

Since NF Service Producers can indicate in their profile allowedxxx IEs, configuration effort is necessary so that they are consistent with the NF Sets.

This solution requires each NF Service Consumer to register its profile and maintains it always updated at all the time with the NRF.

The usage of an access token consumed from a single NF Service Consumer instance by any NF of a NF Set of Service Producers is allowed per 3GPP specification (first case). The solution described in this clause is about Service Consumers of a NF Set to be able to use the same token for consuming a service from a NF Service Producer. From a security perspective, having an access token issued for a specific NF Service Consumer that authorizes a single NF Service Consumer instance to consume a service from any NF Service Producers of an NF set of is quite different than having a single access token issued for a group of NF Service Consumers belonging to the same NF Set to consume a service from a specific NF Service Producer or a NF Set of Service Producers.

In the first case, only one NF Service Producer of the NF Set receives the access token at a time to validate the NF Service Consumer instance whether it is allowed to receive the service or not. Each NF Service Producer will have the ability to validate whether the NF Service Consumer instance was authorized or not via the access token validation process. There is no security risk in this scenario as the access token is issued to authorize a single entity, the NF Service Consumer instance. It is important to note that the NF Service Producer is NOT required to maintain or save the access token.

In the case described by this solution, i.e., a set of NF Service Consumers sharing the same access token, additional access token management is needed. Otherwise the access token may end up being saved at multiple places rather than one which causes a security risk. In addition, the access token does not reflect the identity of the NF Service Consumer instance that is requesting access to the service resource at the NF Service Producer and this way the NF Service Producer is not able to validate whether the NF Service Consumer instance requesting the service is authorized to receive service or not.

In comparison, that is why no certificate is issued for multiple entities to be used by a group of entities, e.g., NF Service Consumer set to be used to blindly authenticate a member of the group.

## 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 should 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 should be positioned as the last once. When an SCP add HTTP custom headers, those should come after 3GPP-Sbi-Client-Credentials header from NF Service Consumer 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 6.8.1-1: HTTP message with hash value in CCA for end to end message protection

### 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 custom 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 custom 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 custom 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 custom headers appended by SCP and hash value of HTTP body by comparing the calculated value and the ones in CCA.

### 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 proposes 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.

NOTE: Whether a token similarly to CCA is used for NF Service Producer authentication is per decision of key issue #1.

## 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 6.9.2-1: Authorization mechanism negotiation

Pre-requisite:

- the vPLMN and hPLMN support at least static authorization dependent on the roaming partner.

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. The hPLMN, i.e., all NF Service Producers of the hPLMN that handle service requests from other PLMN, need to support at least static authorization. In this way, the hNRF can always select an authorization method that is supported by both the vPLMN and the hPLMN.

NOTE: The Final authorization mechanism selected by hNRF depends on the 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

This solution addresses the threats and requirements of Key issue #7: "Authorization mechanism negotiation".

The pre-requisite is that the hPLMN supports at least static authorization.

This solution addresses for the inter-PLMN scenario the use case that the vPLMN has only implemented static authorization. The NF Service Producer needs to support whichever authorization mechanism is determined by the hNRF, i.e., using different authorization mechanisms depending on the roaming partner.

In the solution proposed, the VPLMN supporting static authorization only could enforce the HPLMN to overcome the mandated support for OAuth. I.e., to demand static authorization by only providing this specific capability to the roaming partner.

## 6.10 Solution #10: NRF deployment clarifications

### 6.10.1 Introduction

This solution addresses key issue #8. It provides clarification of handling access token requests in different NRF deployment scenarios.

### 6.10.2 Solution details

There are different deployment options for NRFs, as described in TS 23.501 [3] (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 OAuth 2.0 client registration, when information about the NF Service Consumer instance and its NF Type is made available in the OAuth 2.0 Authorization server, i.e., the slice specific NRF.

This registration process is subject to implementation procedures of the operator, with the following consideration on authentication procedure: OAuth 2.0 clients are capable to authenticate securely with the authorization server, i.e., client type as specified in RFC 6749 [17] is "confidential".

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

The solution proposes that information about the consumer instance and its type is made available in the slice specific NRF. This is technically possible, potentially quite complicated. It is applied to an optimization in TS 23.501on a slicing-related procedure. Whether the procedure with this addition is still an optimization, would need further evaluation.

This proposal requires pre-configuration of the slice-specific NRF for solving a very specific issue of an optimization in TS 23.501 [3] on a slicing-related procedure.

## 6.11 Solution #11: Registered NF Profile changes for Inter-Slice Access

### 6.11.1 Introduction

This solution proposes to address following security requirement of KI #9:

- The 5GS should provide a mechanism that allows the NF Service Producer not to provide service to NF Service Consumers, which are not authorized to access a slice.

This solution gives the option for NF Service Producers to indicate which consumer's S-NSSAIs are allowed to access which producer's S-NSSAIs, and by which operations in its profile.

### 6.11.1 Solution Details

Using individual or combination of IEs like *allowedNfTypes* and *allowedNssais,* NRF can already filter the requests from NF Service Consumers not allowed to access any services of an NF Service Producer.

To address the issue of static configurations at NRF for restricted Inter-slice access, it is proposed that:

- When an NF Service Producer registers its profile in NRF, it includes a new information element, "*allowedSnssaisPerSnssai*", which specifies, for each NF-Service Consumer's S-NSSAI, the list of NF-Service Producer's S-NSSAIs it is allowed to access.

- When an NF Service Producer registers its profile in NRF, it includes a new information element, "*allowedOperationsPerSnssai*", which specifies allowed operations on its resources, for NF Service Consumers belonging to the given S-NSSAI.

These attributes can then be used by NRF to determine the "producerSnssaiList" to be included in the AccessTokenClaims. Additionally, NRF can determine whether a given resource/operation-level scope can be granted to an NF Service Consumer that requested an OAuth 2.0 access token with a specific scope, or limit the scope or authorization irrespective of the request.

Additionally, to allow NF Service Producers (themselves) validate the "Inter-Slice" access requests from the NF Service consumers, it is proposed to include "Requester-NSSAI" in the access-grant, indicating the NRF validated NF-Consumer's S-NSSAIs.

The NF Service Producer can thus check the decision of the NRF to assign an access token, but is still dependent on the NRF validating the consumer's S-NSSAI and including it in the access token.

### 6.11.3 Evaluation

This solution proposes to address the following security requirement of KI #9:

The 5GS should provide a mechanism that allows the NF Service Producer not to provide service to NF Service Consumers, which are not authorized to access a slice.

1. Finer granularity for NF Service Producer's profile:

The solution gives the option for NF Service Producers to indicate which consumer's S-NSSAIs are allowed to access which producer's S-NSSAIs, and by which operations, in its profile.

2. NFp authorizing NF Service Consumer's S-NSSAI:

In this solution, the NF Service Producer is still dependent on NRF to include the validated S-NSSAIs, how NRF validates the NF Service Consumers is not covered in this solution.

## 6.12 Solution #12: Authorization of notification endpoint in "Subscribe-Notify" scenarios

### 6.12.1 Introduction

This solution addresses key issue #3: Service access authorization in the "Subscribe-Notify" scenarios by including "notification URI" in the token request.

This solution proposes to include "notification URI" and CCA of the NF which owns the "notification URI" at the access token request. The NRF verifies whether the notification URI is authorized to receive the service/notification that is requested in the access token request.

### 6.12.2 Solution details

During access token request process, NF Service Consumer include "notification URI" in Nnrf\_Access Token\_Get Request when operation semantics of the requested service type is "Subscribe/Notify". NRF may verify whether the notification URI in the access token request match the corresponding information in the public key certificate of the NF Service Consumer or those in the NF profile of the NF Service Consumer. In addition, when "notification URI"(s) are included in the access token request, NRF may verify whether the "notification URI"(s) locate its location(s) with the same address(es) which are indicated by its own FQDN or IP address as registered in NRF. The NRF checks whether the NF Service Consumer is authorized to access the requested service(s).

When the NF Service Consumer is authorized, the NRF shall then generate an access token with appropriate claims which may include "notification URI" as requested by NF Service Consumer.



Figure 6.12.2-1: Access token request for "Subscribe-Notify" NF Service illustration 1

A NF Service Consumer (NF\_A) may request to subscribe to NF Service offered by a NF Service Producer (NF\_B) on behalf of NF\_C according to the Service Request from NF\_C (e.g., Figure 4.15.3.2.2-1: Nudm\_EventExposure\_Subscribe, Unsubscribe and Notify operation in TS 23.502[7]).

The Figure 6.12.2-2 describe the solution to verify the service request from NF\_A including Notification URI of NF\_C.



Figure 6.12.2-2: Access token request for "Subscribe-Notify" NF Service illustration 2

Step 0. The NF\_C sends a Service Request to the NF\_A for notification service on an event with an access token including notification URI and CCA of the NF\_C.

Step 1. The NF\_A decides to subscribe a service of the NF\_B associated to the Service Request received at step 0 and sends an Access Token Request to the NRF for a Service Request toward the NF\_B including notification URI and CCA of the NF\_C.

Step 2. The NRF verifies whether the NF\_A is authorized and whether the NF\_C identified by the appended CCA is a valid Network Function and authorized to receive the requested NF Service from the NF\_B. NRF may verify whether the notification URI in the access token request match the corresponding information in the public key certificate of the NF\_C of the CCA of the NF\_C, or those in the NF profile of the NF\_C.

Step 3. Upon successful verification, the NRF publishes an access token for a Service Request of the NF\_A toward the NF\_B. The access token includes notification URI of the NF\_C in the claims as requested in step 2.

Step 4. The NF\_A sends a Service Request for a notification service toward NF\_B including the access token received at step 3 and CCA of the NF\_A.

Step 5. The NF\_B verifies the access token and CCA of the NF\_A. During the verification, the NF\_B can check whether notification URI in the Service Request are included in the access token. If the verification is successful, the NF\_B may respond the NF\_A and provide the NF\_C with the notification service as requested by the NF\_A. In case of notification target reselection as described in clause 6.3.1.0 of TS 23.501[3], binding can be used to indicate suitable NFs other than NF\_C (see TS 23.501 [3], Table 6.3.1.0-1 on binding, selection and reselection).

### 6.12.3 Evaluation

The solution extends the existing token-based authorization method by including the notification URI of the NF\_C in the token claims.

The token is used to indicate towards NF\_B (the producer and sender of the notification) that NF\_C (the notification receiver) authorizes NF\_A (the subscriber) to subscribe services on behalf of NF\_C (the notification receiver). The token is also used to indicate towards NF\_B (the producer and sender of the notification) that the NRF authorizes the notification URI contained in the token to receive notifications from NF\_B.

The solution has impact on the access token generation by adding the notification URI of the NF\_C into the access token, and token verification.

Further evaluation would be necessary. The solution does not address the case of notification target reselection as described in clause 6.3.1.0 of TS 23.501 [3].

## 6.13 Solution #13: Authentication of NF Service Producer in Indirect Communication

### 6.13.1 Introduction

This solution addresses KI#1.

This solution proposes a simple authentication procedure which allows an NF Service Consumer or any other entity (e.g., a delegated SCP) to authenticate an NF Service Producer before sending an actual service request. This ensures that an actual service request with request parameters, some of which may be sensitive (e.g., SUPI), is only sent to an NF Service Producer that has already been authenticated.

This property is consistent with the behaviour of HTTPS/TLS in which a client certificate with client information is sent to the server only after the authentication of a server certificate, and an HTTPS request is sent to the server only after the authentication of the server in TLS.

The procedure can be executed by an NF Service Consumer or by an SCP which has been delegated by an NF Service Consumer to perform service discovery and re-selection on its behalf.

### 6.13.2 Solution details

A Client (e.g., an NF Service Consumer) first performs the service discovery to obtain information about an NF Service Producer (e.g., NF Service Producer Instance ID). Before the Client sends a service request to the NF Service Producer, it performs the following procedure to obtain a Server Credentials Assertion (SCA) from the producer. Similar to CCA, which allows to verify the identity of a consumer, SCA allows to verify the identity of the producer.

This procedure can be performed by an NF Service Consumer to verify the identity of an NRF or an NF Service Producer. For example, it can be performed to verify the identity of an NRF before an NF Service Consumer sends an access token request to an NRF.

In delegated discovery, this procedure can be performed by an NF Service Consumer before sending an actual service request. This procedure will trigger the SCP to perform the first service discovery and eventually return the SCA to the NF Service Consumer. Alternatively, if a delegated SCP is trusted by an NF Service Consumer to verify the identity of an NF Service Producer, an SCP will perform this procedure on the behalf of NF Service Consumer to verify the identity of an NF Service Producer before it forwards a service request from an NF Service Consumer to the NF Service Producer.

When a re-selection of an NF Service Producer is performed, this procedure can be re-performed against the newly selected NF Service Producer. If reselection by the SCP is a desired feature, the SCP also needs to be trusted, because the NF Service Consumer cannot distinguish whether the SCP is acting maliciously or as wanted.



Figure 6.13.2-1: Authentication Procedure for NF Service Producer in Indirect Communication

1. A client (e.g., an NF Service Consumer or an SCP) sends an HTTP request to a server (e.g., an NRF or an NF serviced producer) to obtain its Server Credentials Assertion (SCA). This HTTP request can be a simple HTTP Get request to a well-known resource (e.g., /SCA) or a service request (e.g., NFp\_SCA\_Get\_Request) without any requesting parameter. This ensures that no client information is exposed to the server in such request.

2. The SCP forwards the request (NFp\_SCA\_Get\_Request, or HTTP Get) from the Client to the Server.

3. The Server (e.g. an NRF or an NF Service Producer) receives the request and generates Server Credential Assertion (SCA). The SCA has the same format as CCA and includes the identity of the server (e.g., NF Service Producer instance ID) and associated proof, which allows the verification of the server identity.

4. The Server sends the SCA in the response (e.g., NFp\_SCA\_Get\_Response or HTTP response) back to the SCP. The response may include an indicator to request for the CCA from the Client.

5. The SCP forwards the response including the SCA back to the Client.

6. The Client verifies the SCA cryptographically and then verifies the server identity (e.g., NF Service Producer Instance ID) in the SCA against the corresponding identity of server where a service request is to be sent.

### 6.13.3 Evaluation

This solution allows a NF Service Consumer to identify the NRF or NF Service Producer. However, only in limited cases the procedure successfully addresses the threats and requirement, i.e., only in the cases when the consumer discovers the producer itself and if the SCP is not reselecting the producer afterwards.

When discovery is delegated to the SCP or the SCP is doing a reselection of the producer, then the SCP needs to be trusted, which is not in line with key issue threat assumption of a misbehaving SCP.

## 6.14 Solution #14: SCP trust domain or technical domain grouping

### 6.14.1 Introduction

This solution is addressing key issue #2. It provides an optimization for SCP domain operation in trusted environments.

### 6.14.2 Solution details

According to TS 23.502 [7], SCP domains can be registered at the NRF. A NF Service Producer can register within its profile its SCP domain.

A SCP domain is called trusted if it is a group or subset of available SCPs within one PLMN that includes several SCPs operated in the same trust environment according to a security policy. Such technical SCP trust domains can be defined in large PLMNs, for example, based on computer centre boundaries, operators of subnetworks, regions/provinces, etc.

SCP trust domains could also be formed equipment-specific or release-specific. Large operators could have involved different organizations to operate the complete network and hence, structure their regional availability by SCP domains, where additional security means between these organizations could be needed for achieving trust.

For a SCP trust domain, one or several SCP domains can be grouped, i.e., a SCP trust domain is a superset of a distinct set of SCP domains. A SCP trust domain could be isolated from other SCP domains. It could be registered or configured together with a set of policies at NRF.

If the SCP is part of several SCP domains, all SCP domains have to be part of the same SCP trust domain as illustrated in the figure below.

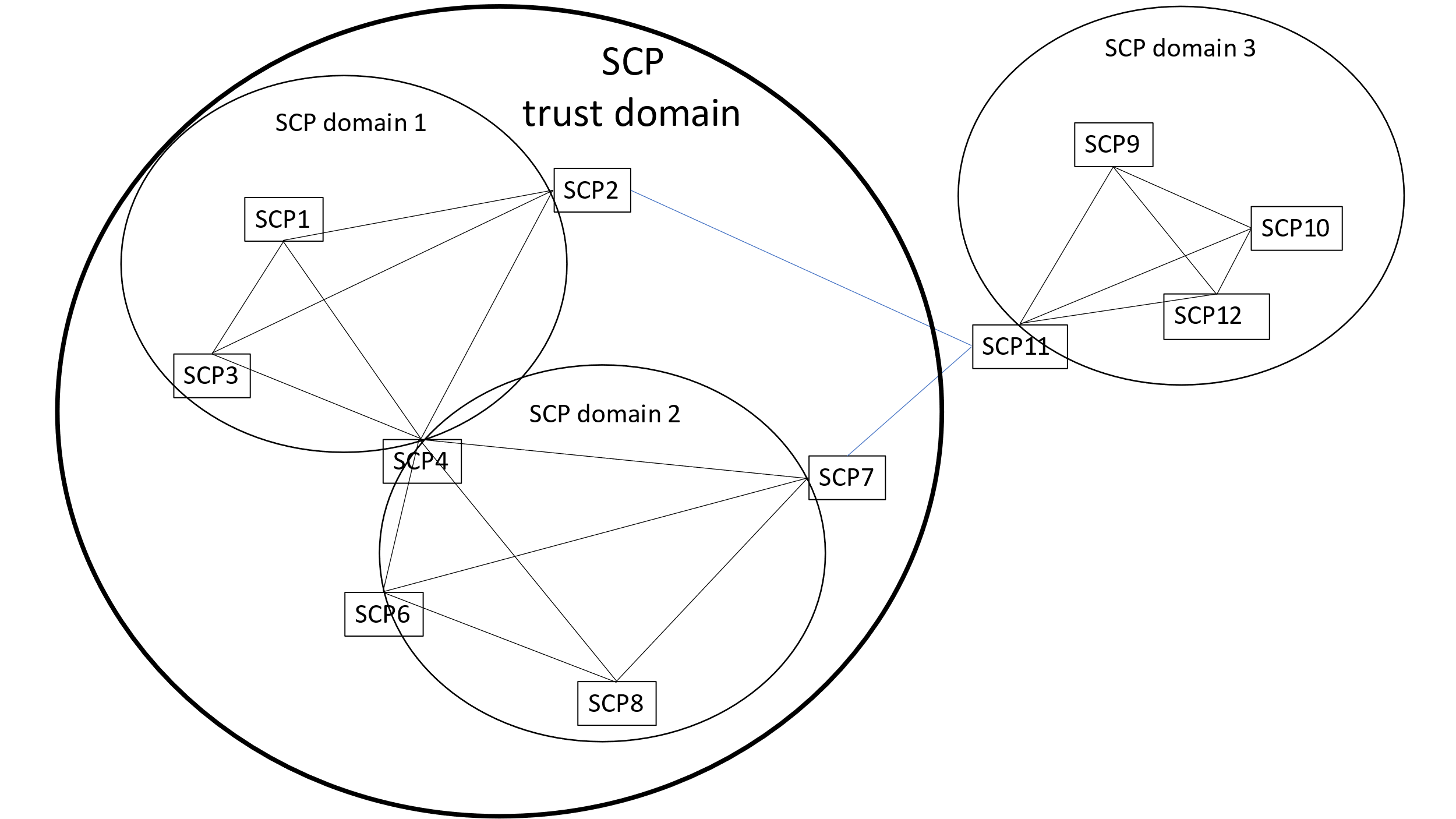


Figure 6.14.2-1: Illustration of SCP trust domain

SCPs can discover boundaries of trust or technical domains based on:

- the SCP domain, from which a request is received from and is forwarded to, and

- the configured knowledge of the relationship of trust or technical domains and SCP domains.

Policies for a SCP trust domain could include information such as which domain can communicate with which domain or whether NFs from one SCP trust domain can access NFs of another SCP trust domain.

The policies are related to restriction in the communication between trust domains that the SCP enforces, as outlined in the examples below.

Some examples of policies for a SCP trust domain are as follows:

- Only SCPs from a particular (usually the same) SCP trust domain would be allowed to discover the NF Service Producer.

- An access token could be provided for all NFs of the same SCP trust domain.

- A service is provided only for NFs of a distinct NF Type within a specific SCP trust domain.

- The SCP domain of NF Service Consumer and SCP are the same as of the NF Service Producer.

- Access to certain services is not permitted over trust domain boundaries.

If the SCP discovers that a request message traverses a boundary between trust/technical domain, it can perform also policing on traffic between SCP trust domains, for example:

- Checking target URIs

- Checking delegated discovery parameters

- Checking routing binding indications

- Checking whether a request contains a CCA

The following are examples of using SCP trust domains:

- The access from NF or SCP to the SCPs of a SCP trust domain can be limited to this domain only. Hence, all SCPs forming a trust domain could be allowed to communicate with NRF and with NF Service Producers on behalf of a NF Service Consumer from the same trust domain. By this, SCPs forwarding a request to another SCP, e.g., for load balancing, have a mean to check whether the forwarding SCP is a member of the same SCP trust domain.

- If a policy for SCP trust domain is registered and the NF profile includes SCP trust domain information in the NRF, the NRF can authorize or reject a discovery request from NF or SCP based on whether the expected NF Service Producer or another SCP instance is part of the same SCP trust domain as the NF Service Consumer.

- If a NF Service Consumer has included in its request a target SCP trust domain, the NRF can provide an access token that provides authorization for all NFs within this domain.

- If the NF Service Producer receives a service request, it could provide the service only, when contacted by an SCP of the trust domain indicated in the access token (e.g., AMF getting a token from NRF granting access to any SMF in the same SCP domain).

- If domain information of the NF Service Consumer is included in the token, it enables the NF Service Producer or SCP to verify whether the requesting NF Service Consumer is part of the same SCP trust domain.

SCPs of a trust domain connecting to another SCP domain outside the trust domain shall provide confidentiality, integrity and replay protection.

### 6.14.3 Evaluation

The solution addresses SCP domains. Trust domains do not need to be necessarily identical to SCP domains. Restricting network topology so that requests between certain elements are not allowed could be generalized as provided by the potential security requirement in clause 5.2.3.

Further evaluation is needed for deciding on the usefulness of the concept in general, such as addressing the moving of UEs between regions with different trust requirements.

## 6.15 Solution #15: Authorization mechanism for the involved NFs in the delegated "Subscribe-Notify" scenario.

### 6.15.1 Introduction

This solution addresses KI#3 on how to assure that the notification messages could be only forwarded to an authorized NF identified by its NF instance ID in the delegated "Subscribe-Notify" scenarios. This solution does not address authorization of the notification URI.

The solution is based on the authorization of NF Service Consumers for data access via DCCF specified in Annex X.2 of TS 33.501 [2]. It proposes to include two instance IDs in the access token request, allowing the NRF to check whether one NF (e.g., NF\_C) is authorized to subscription and whether another NF (e.g. NF\_A) as the proxy is allowed to request the service from the identified NF Service Producer on behalf the NF\_C. The NF Service Producer verifies the access token is valid according to these instance IDs. After successful verification, the NF Service Producer should provide the notification service.

### 6.15.2 Solution details

In the "Subscribe-Notify" NF service illustration 2 (delegated scenario) specified in TS 23.501 [3], clause 7.1.2, an NF Service Consumer (e.g. NF\_C) may subscribe the service of an NF Service Producer (e.g., NF\_B) on behalf of another NF Service Consumer (e.g. NF\_A).

The Figure 6.15.2-1 describes the detailed solution for authorization mechanism for the involved NFs in the delegated "Subscribe-Notify" scenarios.



Figure 6.15.2-1: Authorization mechanism for the involved NFs in the delegated "Subscribe-Notify" scenario

1) The NF\_A decides to subscribe the service of NF Service Producer (e.g. NF\_B) on behalf of NF\_C and gets an access token from the NRF for a Service Request toward the NF\_A. The NF\_C initiates an NF service request to the NF\_A which includes the notification URI, the access\_token\_NF\_C and the CCA of NF\_C to be used for subscription.

2) The NF\_A should verify if the access token and the CCA of the NF\_C is valid and executes the service.

3) The NF\_A sends a Nnrf\_AccessToken\_Get request to NRF including the information to identify the target NF (NF Service Producer), the source NF (NF Service Consumer e.g., NF\_C), the NF Instance ID of NF\_A and the CCA\_NF\_C received at step 2.

4) The NRF should check whether the NF\_C are allowed to access the service provided by the identified NF Service Producer, and whether the NF\_A as the proxy is allowed to request the service from the identified NF Service Producer on behalf the NF\_C. The NRF authenticates NF\_C based on the CCA of NF\_C.

5) If the authorization is successful, the NRF shall then generate an access token with the identity of the NF\_A and the identity of the NF\_C, NF type of the NF Service Producer (audience), subscribe service name(s), (scope).

6) The NRF sends access token to the NF\_A in the Nnrf\_AccessToken\_Get response operation.

7) The NF\_A requests service to the NF Service Producer. The Service Request also consists of the CCA\_NF\_C, so that the NF Service Producer authenticates the NF \_C.

8) The NF Service Producer authenticates the NF\_C and verifies the access token to ensure that the access token is valid. After authentication and authorization is successful, the NF Service Producer assures that the NF\_A as the proxy is allowed to receive the response message on behalf the NF\_C, and executes the subscribe service.

9) The NF Service Producer should respond the NF\_A and provide the NF\_C with the notification service based on the Notification URI.

### 6.15.3 Evaluation

The solution extends the existing token-based authorization method by including the instance ID of the NF\_A and the instance ID of the NF\_C in the token claims.

The token is used to indicate towards NF\_B (the producer and sender of the notification) that NF\_C (the notification receiver) authorizes NF\_A (the subscriber) to subscribe services on behalf of NF\_C (the notification receiver). The token is also used to indicate towards NF\_B (the producer and sender of the notification) that the NRF authorizes NF\_C (the notification receiver) to receive notifications from NF\_B.

The solution has impact on the access token generation by adding the instance ID of delegated NF\_C into the access token, and token verification.

How the solution works in case of notification target reselection as described in clause 6.3.1.0 of TS 23.501 [3] is not addressed in the present document.

NOTE: Further evaluation is necessary and not addressed in the present document.

## 6.16 Solution #16: Selective End of End Protection of HTTP Request and Response in Indirect Communication

### 6.16.1 Introduction

This solution addresses KI#5.

It allows an NF to verify the integrity of selected attributes in a service request or response consumer. It requires the use of CCA and Server Credential Assertion (SCA) so that a recipient NF has access to the public key certificate of a sending NF. The solution does not propose any change to CCA or SCA.

Instead of protecting the integrity of entire request or response, which has impact on the functioning of SCP based on the feedback from CT, this solution proposes to protect only selected fields in a request or response that are considered critical and should not be modified by any intermediate entities such as SCP. Note there is a trade-off. The more of the message is protected, the fewer attacks are possible, but there is also a higher risk that the SCP may need to modify integrity protected parts of the message.

Further, to avoid complexity of signalling which attributes are to be integrity protected inline request or response, this solution proposes to use static configuration to indicate which attributes require integrity protection.

### 6.16.2 Solution details

Each NF, either NF Service Consumer or NF Service Producer, is provided with a local integrity protection policy, describing which attributes or information elements within a service request or service response require integrity protection. For example, AUSF and UDM may be configured with the security policy requiring the integrity protection of authentication method, and authentication vector.

Local security policy can be dynamically updated as with other local configuration of the NFs. NF Service Consumer and NF Service Producer can also communicate its security policy to each other. The security policy of an NF Service Producer can be sent to the NF Service Consumer along with the SCA during the authentication procedure as described in solution #13. NF Service Consumer can send its security policy to the NF Service Producer in its service request.

Based on the security policy, an NF creates one (or more) JWT tokens for the information elements in a service request or service response that require integrity protection. The JWT token is digitally signed by the private key of the NF. The JWT token may be included in an information element in a service request or service response, or in an HTTP header.

The public key certificate of the sending NF that is required to verify a received JWT token is communicated to a receipt NF in CCA or SCA.

When a recipient NF receives a service request or service response, it checks its security policy to see if any information elements in the received service request or service response require integrity protection. If yes, it extracts and verifies the JWT token to ensure that protected information elements have not been tampered with.

Since the JWT with the protected information is not bound to the message in which it is sent, attacks are possible where the JWT is inserted or replayed into another message.

Replay of a JWT token can be mitigated in the same way as replaying of any other security token such as CCA, e.g., using a timestamp.

### 6.16.3 Evaluation

No evaluation is provided in the present document.

## 6.17 Solution #17: Authorization mechanism negotiation using existing methods

### 6.17.1 Introduction

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

The solution assumes that an operator has to support both methods, i.e., token-based authorization and static authorization, but one operator only uses static authorization. Hence, for this solution both networks need to understand OAuth 2.0.

### 6.17.2 Solution details

The solution assumes that discovery is used by both PLMNs according to the standard. Thus, both PLMNs are able to understand the OAuth2Required indication as specified in TS 29.510 [6] when NRFs communicate with each other, and when a NF service is discovered at NRF.

The key issue use case assumption is that one PLMN uses only static authorization. Based on TS 33.501 [2] (clause 13.4.1.0): "The authorization framework uses the OAuth 2.0 framework as specified in RFC 6749 [43]. […] The authorization framework described in clause 13.4.1 is mandatory to support for NRF and NF." and clause 13.4.0, static authorization can be used. "It can be used when token-based authorization is not used."

This solution follows the standard, i.e., that at least the support for OAuth2.0 is provided, thus, even if not used, NRF and NFs of a PLMN using only static authorization need to at least understand the attributes (IEs) provided during discovery.

This solution suggests that from the oAuth2Required indication, the vNRF can imply, whether OAuth2.0 or static authorization is to be used within one PLMN. This covers the use case, where within one PLMN maybe not yet all NFs use OAuth2.0.

For inter-PLMN stage 3 (TS 29.510 [6] Table 6.1.6.2.3-1) "oauth2Required" can be used to handle the authorization method setting by the hNRF. Another type for NFService, the "perPlmnOauth2ReqList", is also specified and includes the OAuth 2.0-based authorization requirement supported by the NF Service Instance per PLMN of the NF Service Consumer.

Table 6.17.2-1: The IEs oauth2Required and perPlmnOauth2ReqList, part of TS 29.510 [6] Table 6.1.6.2.3-1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| oauth2Required | boolean | O | 0..1 | It indicates whether the NF Service Instance requires Oauth2-based authorization.  Absence of this IE means that the NF Service Producer has not provided any indication about its usage of Oauth2 for authorization. |
| perPlmnOauth2ReqList | PlmnOauth2 | O | 0..1 | When present, this IE shall include the Oauth2-based authorization requirement supported by the NF Service Instance per PLMN of the NF Service Consumer.  This IE may be included when the Oauth2.0 authorization requirement supported by the NF Service Instance for different PLMN is different. When the requester PLMN Id is available in perPlmnOauth2ReqList IE, this IE shall override the oauth2Required IE. If the requester PLMN ID is not present in perPlmnOauth2ReqList IE, then the value of oauth2Required IE shall be applicable if available. |

In addition, stage 3 has specified for inter-PLMN usage the type PlmnOauth2 (see clause 6.1.6.2.102, TS 29.510 [6]. If the optional attribute "oaut2NotRequiredPlmnIdList" is used, the solution suggests that in this case pre-configured information by HPLMN (how to apply static authorization with a specific roaming partner) can be used.

Stage-3 details could be improved by stating explicitly that oauth2NotRequired refers to static authorization.

Table 6.17.2-2: The IEs oauth2RequiredPLMNIdList and oauth2NotRequiredPLMNIdList, part of TS 29.510 [6] Table 6.1.6.2.3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| oauth2RequiredPlmnIdList | array(PlmnId) | O | 1..N | It shall indicate the consumer PLMN ID list for which NF Service Instance requires Oauth2-based authorization.  (See NOTE 1) |
| oauth2NotRequiredPlmnIdList | array(PlmnId) | O | 1..N | It shall indicate the consumer PLMN ID list for which NF Service Instance does not require Oauth2-based authorization.  (See NOTE 1) |
| NOTE 1: The same PLMN Id shall not be present in both oauth2RequiredPlmnIdList and oauth2NotRequiredPlmnIdList. | | | | |

Further, the IE oauth2Required used in the bootstrapping information is used to indicate whether NRF requires OAuth 2.0 based authorization for accessing its services:

Table 6.17.2-3: oauth2Required for service access from NRF, part of TS 29.510 [6], see Table 6.4.6.2.2-1: Definition of type BootstrappingInfo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| oauth2Required | map(boolean) | O | 1..N | When present, this IE shall indicate whether the NRF requires Oauth2-based authorization for accessing its services.  The key of the map shall be the name of an NRF service, e.g. "nnrf-nfm" or "nnrf-disc".  The value of each entry of the map shall be encoded as follows:  - true: OAuth2 based authorization is required.  - false: OAuth2 based authorization is not required.  The absence of this IE means that the NRF has not provided any indication about its usage of Oauth2 for authorization. |

If set to false, OAuth 2.0 based authorization is not required. This solution proposes to clarify stage 3 that this mean, the only other available authorization mechanism "static authorization" is the default in this case.

GSMA has provided the following recommendations given in NG.113 [8], clause 7.6.3.4:

"*It is recommended that both VPMN and HPMN use either static authorization or  
authorization using*  *OAuth 2.0 access token.  
Note: Authorization is not possible in case the HPMN only uses authorization  
using*  *OAuth 2.0 access token and the VPMN only uses static authorization.*

*If using authorization using*  *OAuth 2.0 access token it is recommended that both VPMN and HPMN support oauth2Required IE as specified in 3GPP Release 16 TS 29.510 [16].*

*If the HPMN wants to use authorization using*  *OAuth 2.0 only for some VPMNs then HPMN must support perPlmnOauth2ReqList IE as specified in 3GPP Release 17 TS 29.510 [16].*"

How to configure the NF profile in case the operator wants to change to another authorization method is out of scope of this solution. By OAM this could be automated for each NF affected.

If the operator of one PLMN does not want to follow the specification and only supports static authorization, and the other operator requires the usage of OAuth, then in a consequence, those operators cannot have a roaming agreement. The service request has to be rejected if one PLMN does not accept a request without an OAuth.

### 6.17.3 Evaluation

Several means to allow a PLMN's hNRF to provide to the requesting vNRF information on the authorization method used exist in the current specification TS 29.510 [6]. If in the array(PlmanId) of hPLMN NRF a roaming partner is on the "oauth2NotRequiredPlmnIdList", static authorization can be used with this roaming partner. When operators follow the recommendations given in NG.113 [8], the key issue seems to be covered sufficiently by the existing methods.

Further evaluation would be needed, but the study has been stopped.

## 6.18 Solution #18: Avoiding slice isolation violation

### 6.18.1 Introduction

This solution is addressing KI#9.

### 6.18.2 Solution details

To avoid network slice isolation violation as described in detail by KI#9, clarification of the specification clause 13.4.1.1.2 is required to mitigate potential slice related attacks.

The solution proposes

- in step 1a of clause 13.4.1.1.2 of TS 33.501 [2], NRF to verify that the S-NSSAI of the NF Service Consumer in the access token request is consistent. Specification text is proposed as follows:

*"The NRF may additionally verify the S-NSSAIs of the NF Service Consumer."* The NRF checks whether the NF Service Consumer is authorized to access the requested service(s). *"For example, the NRF may verify that the NF Service Consumer can serve a slice which is included in the allowed slices for the NF Service Producer."*

- in step 2 of clause 13.4.1.1.2 of TS 33.501 [2], NF Service Producer checking the audience claim to verify that the producer NSSAIs in the access token (received in the service access request) contains the allowed NSSAI of a UE if the request is for UE related information. Specification text is proposed as follows:

It checks that the audience claim in the access token matches its own identity or the type of NF Service Producer. If a list of NSSAIs or list of NSI IDs is present, the NF Service Producer shall check that it serves the corresponding slice(s). *"If applicable (e.g., when the request is for information related to a specific UE), the NF Service Producer may check that the NF Service Consumer is allowed to access (as indicated by the NF Service Producer's NSSAIs in the access token presented by the NF Service Consumer) at least one of the slice(s) that the UE is currently registered to, e.g., by verifying that the UE's allowed NSSAI(s) intersect with the NF Service Producer's NSSAIs in the access token. As the NSSAI supplied by NF Service Consumer is forged, NF Service Producer (e.g. ,AMF) should obtain the NSSAI of the UE from the UDM for verification."*

### 6.18.3 Evaluation

The text proposed provides input to normative work on this topic. It avoids slice isolation violation.

## 6.19 Solution #19: Hosted SEPP requirements

### 6.19.1 Introduction

This solution addresses key issue #12. It provides input for text that needs to be adapted for clarification of a deployment option for SEPP of a PLMN, if hosted by an entity external to the PLMN.

### 6.19.2 Solution details

This solution is addressing the Hosted SEPP as described in KI#12.

If the Hosted SEPP provider is contracted by several PLMNs, it hosts several distinct Hosted SEPP instances for the different PLMNs.

A PLMN can have a Local SEPP (managed by the PLMN) and a Hosted SEPP (instance), both handling disjoint sets of roaming relations.

It is proposed to add the following definitions to TS 33.501 [2]:

**"Hosted SEPP:** A SEPP that is hosted and operated by a provider outside the PLMN, for example an IPX provider. From the perspective of roaming partners, PLMN trust extends to the Hosted SEPP instance representing this PLMN. A Hosted SEPP provider can operate Hosted SEPPs for multiple PLMNs.

**Local SEPP:** A SEPP of a PLMN that is operated by an MNO.

The Hosted SEPP serves a set of roaming relations that is disjoint from the set of roaming relations served by the Local SEPP."

It is further proposed to have an addition to the general requirements in TS 33.501[2] clause 5.9.3 (Requirements for e2e core network interconnection security) as follows:

"From the roaming partner's point of view, a Hosted SEPP shall behave in the same way as the SEPP.

All SBI messages transmitted between the PLMN and the Hosted SEPP deployment shall be confidentiality, integrity, and replay protected.

A Hosted SEPP provider shall isolate the processing for each PLMN from the processing for other PLMNs.

A Hosted SEPP provider shall use a different TLS certificates per PLMN, as the MNC and MCC of that PLMN are encoded in the certificate (as specified in TS 23.003 [18] and TS 33.310 [14], Table 6.1.3c.3-1). This certificate shall be issued to the Hosted SEPP provider under its own unique name, and hence shall contain both the PLMN-ID and its own unique identifier.

A Hosted SEPP shall only be able to obtain a certificate with an MCC/MNC combination if the affected MNO has previously authorized this."

As roaming with hosted SEPPs potentially increases the number of root CAs in the overall system, the need to support certificate pinning also increases. It is proposed to add the following requirement to TS 33.501 [2].

*"The SEPP shall support Certificate Pinning in the following way: It shall maintain a set of lists containing root CA certificates, as well as a mapping of PLMN-IDs to this set. The mapping associates each PLMN-ID with a given root CA certificate list. During N32-c connection setup, the SEPP shall map the PLMN-ID of the remote SEPP leaf (server or client) certificate to the associated root CA certificate list for the purposes of certificate chain verification. Only the root CA certificates in the associated list shall be treated as trusted during certificate chain verification. If the remote SEPP certificate contains multiple PLMN-IDs that are mapped to different root CA certificate lists, then that certificate shall be rejected."*

Finally, it is proposed to have an additional NOTE referring to Hosted SEPP concept in TS 33.501 [2] clause 13.1.2 (Protection between SEPPs) as follows:

"*If a Hosted SEPP is deployed as the edge of one PLMN and in the same security zone, the same protection requirements apply to this Hosted SEPP as they apply to a SEPP in general. Furthermore, the Hosted-SEPP-specific protection requirements in clause 5.9.3 apply*."

### 6.19.3 Evaluation

Whether to use a Hosted SEPP is a business decision. Nevertheless, it is important to provide a minimal set of requirements covering this deployment option.

Guidelines for naming are provided for Hosted SEPP by GSMA already in [8] and TS 33.310 [14] may need to be checked for consistency with respect to new security requirements. Such requirements may need to be communicated to GSMA as well.

NOTE: Further evaluation is not addressed in the present document.

## 6.20 Solution #20: PRINS for Roaming Hubs

### 6.20.1 Introduction

This solution addresses key issue #10.

There may be one or two roaming hubs involved in the communication between two PMNs. The roaming hubs require the ability to allow roaming relations between these PMNs via these roaming hubs. Furthermore, roaming hubs with financial liability require the ability to shape roaming traffic between the PMNs in order to control how much roaming traffic they will have to pay for.

PRINS can be used to achieve this, allowing and disallowing establishment of N32c through standard HTTP proxy functionality. PRINS can also be used to dynamically introduce a traffic shaper under control of the RH into the data plane.

### 6.20.2 Solution details



Figure 6.20.2-1: Establishment of N32c via roaming hubs

0: The cSEPP is configured to make use of RH1

1: After receiving an initial request, the cSEPP may set up a TLS connection with RH1's proxy. The cSEPP then sends a CONNECT command to RH1's proxy indicating the destination to be pSEPP. If cSEPP does not have access to the pSEPP FQDN, the cSEPP follows the procedure in 6.20.2.**3** to find the pSEPP FQDN.

2: RH1's proxy shall verify that cSEPP is allowed to set up a roaming relation with pSEPP. If not, the flow continues with the error message of step 6. Otherwise, it continues with step 3.

3: If pSEPP requires another roaming hub to be reached, RH1's proxy sets up a TLS connection with RH2's proxy. R1's proxy then sends a CONNECT command to RH2's proxy indicating the destination to be pSEPP. RH2's proxy shall verify that RH1 is allowed to set up a roaming relation with pSEPP. If not, the flow continues with the error message of step 5. Otherwise, it continues with step 4.

4: If RH2 is present, RH2's proxy, otherwise RH1's proxy shall set up a TCP connection to pSEPP. This connection may be protected by TLS.

5: If successful, if RH2 is present, RH2 shall return success with code 200.

6: If successful, RH1 shall return success with code 200.

If successful, pSEPP and cSEPP shall establish a TLS connection via the proxy/proxies for N32c and establish an N32f context, selecting PRINS as the protection mechanism.



Figure 6.20.2-2: N32f via roaming hubs

1: cSEPP may either initiate a new TLS connection to RH1's proxy, or reuse the TLS connection from setting up N32c. In any case, communication between cSEPP and RH1's proxy shall be TLS protected.

2: After receiving a request from an NF, the cSEPP shall encapsulate the request as defined in TS29.573 [9] subclause 5.3.2.3 and send the reformatted message to RH1's proxy. The URL to send the request to shall be located on RH1's proxy, thus RH1's proxy is able to read the request.

3: If another RH is present, RH1's proxy shall forward the message to RH2's proxy, including a via header and an asserted identity header asserting the identity of the cSEPPs PMN. If RH2 is not present, then this request shall be sent directly to pSEPP.

4: If another RH is present, RH2's proxy shall forward the request to pSEPP, including a via header giving both RH proxies. An asserted identity header is not necessary as pSEPP can identify the source PMN from the N32f context.

5: pSEPP shall decapsulate the request, forward it to the pNF and receive the response.

6: pSEPP shall reformat the response and forward to RH2 (if present) or to RH1 (if RH1 is not present).

7: If RH2 is present, then RH2's proxy shall forward the response to RH1's proxy.

8: RH1's proxy shall forward the response to cSEPP.

9: cSEPP shall decapsulate the response, forward it to the pNF

All communication shall be TLS protected on the hops between cSEPP and RH1's proxy, RH1's proxy and RH2's proxy, RH2's proxy and pSEPP.

RH1's proxy and RH2's proxy both may include modifications, or return errors in case of roaming policy violations.

The following two messages sequence charts give examples:

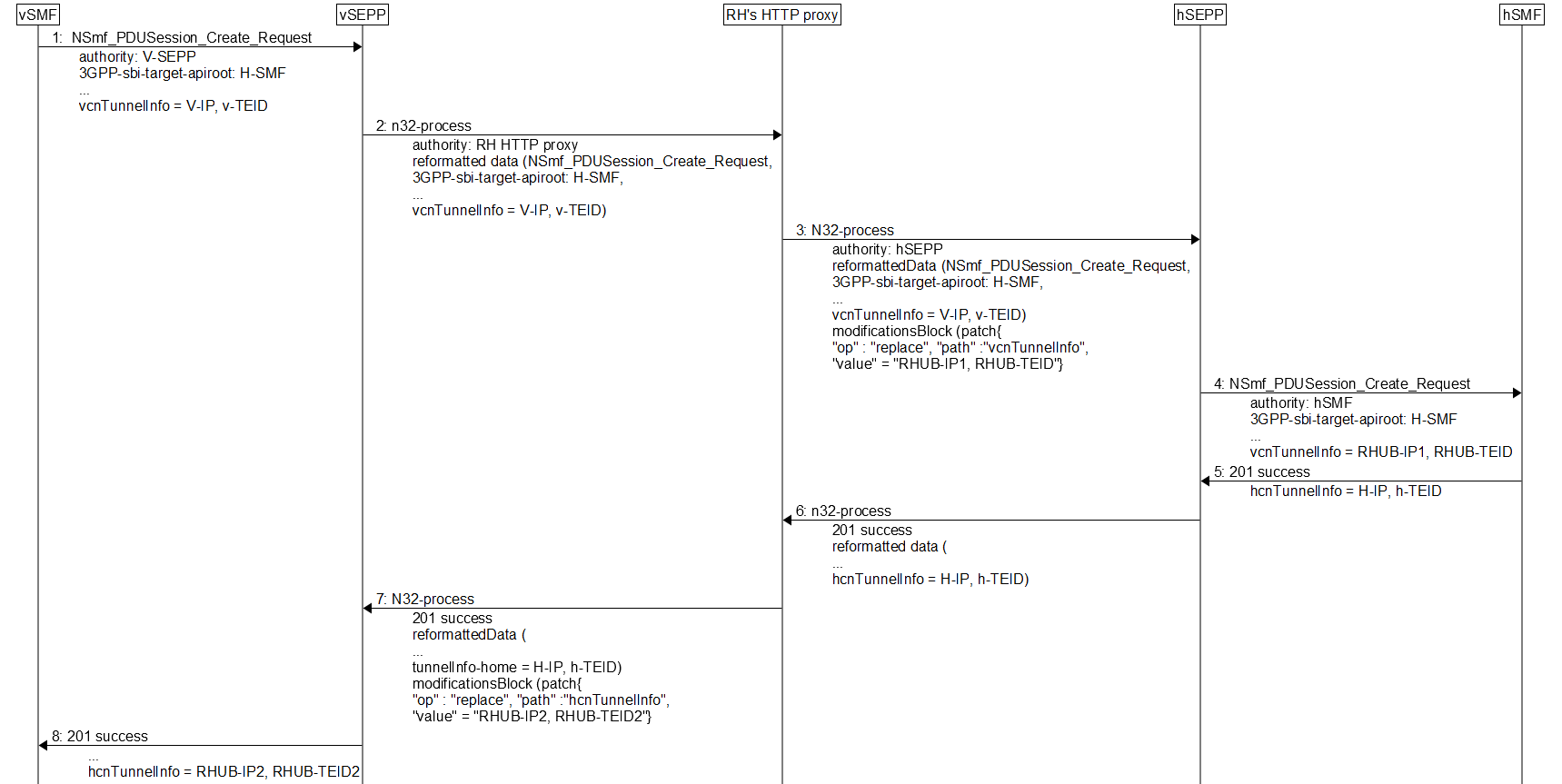


Figure 6.20.2-3: Roaming hub inserting a traffic shaper

To insert a traffic shaper into the roaming user data plane, the RH may include a modifications block that rewrites the N9 endpoints IP address and TEID for the visited networks UPF in message 3, and another modifications block that rewrites the N9 endpoints IP address and TEID for the visited networks UPF in message 7. As these modifications blocks are signed, the modifications are attributable.



Figure 6.20.2-4: Roaming rejecting a session creation request

A roaming hub may also reject a PDU session setup request by returning an error in message 3. This error is forwarded to the vSMF in message 4. This may be triggered by some information in the request visible to the RH. If different SUPIs require different handling, then of course, SUPI shall not be encrypted in the request.

The examples can be extended to include two roaming hubs and potentially even two traffic shapers.

### 6.20.3 RH Proxy Resolves pSEPP Well-Known FQDN

When the cSEPP does not have access to the pSEPP FQDN, the cSEPP follow the following procedure to get the pSEPP specific FQDN:

1. The cSEPP is configured to make use of RH1.

2. After receiving an initial request, the cSEPP may set up a TLS connection with RH1's proxy. The cSEPP then sends a CONNECT command to RH1's proxy indicating the destination to be the pSEPP well-known FQDN as described in [x].

3. RH1's proxy shall verify that cSEPP is allowed to set up a roaming relation with pSEPP based on the pSEPP well-know FQDN. If the cSEPP is allowed to setup a roaming relationship with pSEPP, the RH1's HTTP proxy shall resolve the pSEPP well-known FQDN to a specific pSEPP FQDN in the HPLMN following GSMA PRDs [10] and [11].

4. The RH shall respond by a 307 HTTP/2 response with the Location response header containing the https scheme, the discovered pSEPP FQDN, and the discovered n32-c port, if non-standard port is discovered.

5. cSEPP shall validate that the received pSEPP FQDN belongs to the HPLMN that the cSEPP is trying to reach.

### 6.20.4 Evaluation

The solution allows a RH to allow or deny roaming traffic between operators based on the contractual relations with the RH. In addition, it allows traffic shaping of the user plane roaming traffic between two PMNs making use of roaming hubs by dynamically introducing a traffic shaper.

The solution exposes user plane traffic to roaming hub's traffic shapers (when present). However, at both PMNs, the presence of the traffic shapers is visible in the modifications blocks. Furthermore, modifications performed by the roaming hubs are attributable.

NOTE: Further evaluation is necessary but not addressed in the present document.

## 6.21 Solution #21: Certificate solution for NRF validation of NFc for access token requests

### 6.21.1 Introduction

This potential solution addresses the KI #11, NRF validation of NFc for access token requests. It describes what entities that need to be validated, what needs to be in place from a certificate perspective, and finally some additional details how the verification is performed.

The basic idea of this solution is that the NRF uses the information in the NF Service Consumer's certificate to base its authorization decision on.

### 6.21.2 Solution details

#### 6.21.2.1 NF Service Consumer information to validate at Service Request Authorization

The authorization of a NF Service Consumer needs to be performed by validating verified information about the NF Service Consumer stored in NRF. This solution proposes that the NRF validates NF Type, NF Instance ID, PLMN-ID, and FQDN.

#### 6.21.2.2 Certificates

This solution proposes that the NRF uses the NF Service Consumer information in the NF Service Consumer's certificate to authorize the NF Service Consumer. The public key certificate can be the TLS certificate of the NF Service Consumer, if the NF Service Consumer itself requests the access token, or the certificate used to sign the CCA of the NF Service Consumer, if the SCP requests the access token on behalf of the NF Service Consumer.

If the TLS certificate of the NF Service Consumer is used, this solution requires that the NF Service Consumer information described in clause 6.21.2.1 is available in the TLS certificate.

Today the following entities, relevant for authorization purposes, are mandatory in the TLS certificate: PLMN-ID and FQDN.

The following are currently only optional, but need to be present in the certificate, otherwise this solution will not work:

- NF instance ID

- NF type

Having this information present in the certificate can be reached in different ways. It could either be mandated in the specification or required to be present in deployments in order for this procedure to work. If it is not mandated that the information is present in the certificates, this may lead to interoperability problems between different vendor implementations.

#### 6.21.2.3 NRF validation solution

The NRF checks whether the NF Service Consumer is authorized to access the requested service(s) by performing the following validation:

- The NRF retrieves the available NF Service Consumer information in the public key certificate of the NF Service Consumer to decide whether the NF Service Consumer is authorized to invoke the NF Service Producer's service.

In deployment scenarios with only one NRF, the NRF does not use the information about the NF Service Consumer in the access token request to base its authorization decision on, since this information is provided by the NF Service Consumer itself and therefore not reliable.

In hierarchical NRF deployments, additional measures are necessary. The NRF that receives the access token request needs to verify the information about the NF Service Consumer in the access token request with the certificate. The NRF that issues the token can then base its authorization decision on the information about the NF Service Consumer in the access token request, since the NRF that received the access token request has verified the information.

### 6.21.3 Evaluation

NOTE1: Evaluation for this solution is not addressed in the present document.

NOTE 2: This solution assumes that it is not clearly specified which mechanism the NRF uses to validate whether the NF Service Consumer is authorized to receive the requested service. It is not addressed in the present document whether the mechanism in this solution is needed, or whether the NRF can use the NF Service Consumer profile to validate the NF Service Consumer in the process of authorization of requested services including access token request.

## 6.22 Solution #22: Combined certificate and profile solution for NRF validation of NFc for access token requests

### 6.22.1 Introduction

This potential solution addresses the KI #11, NRF validation of NFc for access token requests. It describes what entities need to be validated, what needs to be in place from a provisioning perspective, and finally some additional details how the validation is performed.

The basic idea of the solution is that the NRF uses the information in the NF Service Consumer's certificate and additional information in the NF profile as basis for its authorization decision.

### 6.22.2 Solution details

#### 6.22.2.1 NF Service Consumer information to validate at Service Request Authorization

The authorization of a NF Service Consumer needs to be performed by validating verified information about the NF Service Consumer stored in NRF. This solution proposes that the NRF validates NF Type, NF Instance ID, PLMN-ID, and FQDN.

#### 6.22.2.2 O&M Provisioning solution

Any Network Function is allowed to use the NF management API to register its profile at the NRF. However, it can happen that NF Service Consumers do not use the NF management API to register their profile at the NRF. In this case, this solution proposes that the NRF is provisioned with the NF profiles for these NF Service Consumers, containing the NF Service Consumer information described in clause 6.22.2.1, by an O&M provisioning operation.

The O&M provisioning operation does not need to be standardized.

#### 6.22.2.3 Certificates

In this solution, the public key certificate can be the TLS certificate of the NF Service Consumer, if the NF Service Consumer itself requests the access token, or the certificate used to sign the CCA of the NF Service Consumer, if the SCP requests the access token on behalf of the NF Service Consumer.

If the TLS certificate of the NF Service Consumer is used, this solution requires that the NF instance ID is available in the TLS certificate, so that the certificate can be linked to the NF profile.

Having the NF instance ID present in the certificate can be reached in different ways. It could either be mandated in the specification or required to be present in deployments in order for this procedure to work. If it is not mandated that the information is present in the certificates, this may lead to interoperability problems between different vendor implementations.

#### 6.22.2.4 NRF validation solution

The NRF checks whether the NF Service Consumer is authorized to access the requested service(s) by performing the following validation(s):

1. The NRF retrieves the available NF Service Consumer information in the public key certificate of the NF Service Consumer.

2. The NRF uses the NF instance ID of the NF Service Consumer that was obtained during authentication of the NF Service Consumer to retrieve the NF profile.

3. The NRF uses the information about the NF Service Consumer obtained in steps 1 and 2 to decide whether the NF Service Consumer is authorized to invoke the NF Service Producer's service.

In deployment scenarios with only one NRF, the NRF does not use the information about the NF Service Consumer in the access token request to base its authorization decision on, since this information is provided by the NF Service Consumer itself and therefore not reliable.

In hierarchical NRF deployments, additional measures are necessary. The NRF that receives the access token request needs to verify the information about the NF Service Consumer in the access token request with the certificate and/or the NF profile. The NRF that issues the token can then base its authorization decision on the information about the NF Service Consumer in the access token request, since the NRF that received the access token request has validated the information.

### 6.22.3 Evaluation

NOTE1: Evaluation for this solution is not addressed in the present document.

NOTE 2: This solution assumes that it is not clearly specified which mechanism the NRF uses to validate whether the NF Service Consumer is authorized to receive the requested service. It is not addressed in the present document whether the mechanism in this solution is needed, or whether the NRF can use the NF Service Consumer profile to validate the NF Service Consumer in the process of authorization of requested services including access token request.

## 6.23 Solution #23: SCP authorization check by NRF

### 6.23.1 Introduction

This potential solution addresses KI#4.

### 6.23.2 Solution details

#### 6.23.2.1 Enabling NRF to check on SCP information

NOTE 1: This solution assumes that the SCP is not used for the registration of the NF profile itself, in which the NF Service Consumer indicates which SCPs are allowed to request an access token on its behalf.

Additional information about SCPs is provided by any NF when registering its profile at the NRF. In particular, it is proposed that the NF adds the SCP Instance ID and the SCP FQDN to provide NRF with the knowledge of which SCP is allowed to request an access token on its behalf.

Many of the NFps registering are also acting as NFcs. If this is the case, the NRF profile information of a requesting NF can be used to verify, if the requester NF has allowed a particular SCP to contact the NRF on its behalf.

In the case, the NF does not register its profile information at the NRF, the NF can provision the SCP authorization information in the OAM.

The solution enables the NRF to provide SCP with an access token for NFc that is bound to be requested only by the SCPs listed in the NF profile, i.e., for which (a) the NF Service Consumer has agreed to act on its behalf and (b) the NF Service Producer has agreed to provide access tokens and later the service, when SCP is acting on behalf of the requesting NFc.

In the following the steps for an access token request are sketched:

1. NF Service Consumer sends its requests to SCP along with its CCA\_NFc.

2. SCP requests the access token from NRF (including the CCA\_NFc) and optionally also includes in own CCA\_SCP. CCA\_SCP is optional in the case, that the SCP Instance ID is present in its TLS certificate (and there is only one SCP in between NFc and NRF).

3. NRF verifies whether the SCP information, received either via its CCA\_SCP or via the TLS certificate, matches the SCP information that is provided by the NF during its profile registration. NRF can also verify by CCA\_NFc, if the SCP is listed in the NFc profile as being allowed to act on behalf of NFc.

4,5. In case of a successful verification, NRF generates the access token, also adding the SCP information (i.e., SCP Instance ID and SCP FQDN) in the claims, and sends it to SCP.

In this way, NRF first checks if NFc has authorized the SCP to receive the access token on its behalf, and then adds SCP details to the claim, so that the NFp then in the later stage can verify if the NFc and NRF has authorized the SCP to receive the service response on the behalf of NFc or not)

Once the access token has been received by SCP the following steps sketch the service request part:

6. SCP then sends the service request to NFp (along with this enhanced access token, CCA\_NFc, and its own CCA\_SCP).

7. NFp after verification of the received access token, matches the SCP info in the claims, with either the subject of the CCA\_SCP, or with the Instance ID of SCP present in its TLS certificate (if there is a direct connection between SCP and NFp, and there is only one SCP between NFc and NFP).

8. If the verification is successful, NFp sends the service response to SCP, else an error code is sent instead.

NOTE 2: In the case where there are multiple SCPs in between either NFc and NRF, or NFc and SCP, NFc adds the information for the first SCP which is in direct communication with the NFc. For the subsequent SCPs, the initial SCP adds its information in the ‘via or other custom header'.

NOTE 3: The presence of original (initial) SCP information in the via or custom header is providing at least a certain level of trust in the case multiple SCPs are involved. The NRF can verify if the original SCP information is present in the NF profile. The NFp, can verify if the origin SCP information is present in the access token claim.

In the multiple NRF scenario where NFc is registered in NRF1 and NFp is registered in NRF2, the following applies:

In this case, when access token request is received at the first NRF (NRF1), the NRF1 can validate the access token request and match CCA and NFc NF profile. If it is validated, the NRF can add custom header that validation is successful. In this case, NRF2 can trust the information received from the NRF1 and can generate the token accordingly.

To be valid for all the NFs (including the ones which have not registered their profile at the NRF), step 3,4 can be modified as follows:

3,4. In the case the NFc is not registered at the NRF and thus the profile information is not present, the NRF sends a request to OAM to request for the authorized SCP info. The OAM, after validating the request, sends the SCP info to NRF.

NOTE 4: If following up this solution, it needs evaluation whether it is reasonable to have a real-time communication between NRF and OAM.

#### 6.23.2.2 Including service request information into the CCA

CCA currently does not have any information regarding the service request for which the CCA is sent. In this solution the CCA can also be enhanced to include a service request info to further decrease the attack surface.

Thus, it is proposed to couple the CCA with the service request and/or the access token request along with which the CCA is sent. For example, a new parameter 'serviceRequestInfo' can be added in the CCA which contains the request info and/or the scope information for which the access token is generated.

When the NRF/NFp receives the request from a SCP, they compare the CCA 'serviceRequestInfo' parameter against the request which they have received. In case of successful verification, the response is sent.

### 6.23.3 Evaluation

NOTE: Evaluation for this solution is not addressed in the present document.

## 6.24 Solution #24: Authorization negotiation with bootstrapping mechanism

### 6.24.1 Introduction

This solution addresses Key Issue #7 "Authorization mechanism negotiation". It is proposed for authorization mechanism negotiation with bootstrapping mechanism.

### 6.24.2 Solution details



Figure 6.24.2-1: Authorization mechanism negotiation with bootstrapping mechanism

Pre-requisite:

- the vPLMN and hPLMN support at least static authorization dependent on the roaming partner

The Figure 6.24.2-1 describes the detailed solution for authorization negotiation.

1) The vNRF sends a Nnrf\_Bootstrapping\_Get request to the hNRF to configure addressed of the bootstrapping service instance.

2) The hNRF responds with all the service instances of the hNRF and their endpoint addresses. In addition, the hNRF adds hPLMN authorization capability into the bootstrapping response. The hPLMN authorization capability indicates the supported authorization mechanisms, i.e., static, OAuth, or both.

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

4) The vNRF selects the final authorization mechanism supported by both the received hPLMN authorization capability and the vPLMN authorization capability.

NOTE: The final authorization mechanism selected by vNRF is depended on operator.

5) The vNRF sends the final authorization mechanism to the NFc.

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

### 6.24.3 Evaluation

This solution addresses the threats and requirements of Key issue #7: "Authorization mechanism negotiation".

The pre-requisite is that the hPLMN supports at least static authorization.

This solution addresses the inter-PLMN scenario where the vPLMN has only implemented static authorization. The hPLMN sends its authorization mechanism by bootstrapping mechanism. The NF Service Producer needs to support which authorization mechanism is determined by the vNRF.

## 6.25 Solution #25: Solution on N32 security profiles

### 6.25.1 Introduction

This solution is addressing aspects of KI#10, in the attempt to improve practicability of the N32 PRINS solution for IPX or roaming hub as intermediaries. It proposes to introduce profiles for the N32 security solution.

The following requirements on roaming via N32 are defined in TS 33.501 [2]:

*- The solution shall support application layer mechanisms for addition, deletion and modification of message elements by intermediate nodes except for specific message elements described in the present document.*

*NOTE: Typical example for such a case is IPX providers modifying messages for routing purposes.*

- Th*e solution shall provide confidentiality and/or integrity end-to-end between source and destination network for specific message elements identified in the present document. For this requirement to be fulfilled, the SEPP – cf [2], clause 6.2.17 shall be present at the edge of the source and destination networks dedicated to handling e2e Core Network Interconnection Security. The confidentiality and/or integrity for the message elements is provided between two SEPPs of the source and destination PLMN–.*

*- The destination network shall be able to determine the authenticity of the source network that sent the specific message elements protected according to the preceding bullet. For this requirement to be fulfilled, it shall suffice that a SEPP in the destination network that is dedicated to handling e2e Core Network Interconnection Security can determine the authenticity of the source network.*

*- The solution should have minimal impact and additions to 3GPP-defined network elements.*

*- The solution should be using standard security protocols.*

*- The solution shall cover interfaces used for roaming purposes.*

*- The solution should take into account considerations on performance and overhead.*

*- The solution shall cover prevention of replay attacks.*

*- The solution shall cover algorithm negotiation and prevention of bidding down attacks.*

*- The sol*ution should take into account operational aspects of key management.

3GPP has defined two security mechanisms for 5G Roaming Security, specifically for N32-f interface protection: TLS (Transport Layer Security) and PRINS (ALS: Application Layer Security).

Different stakeholders voiced that they may require the flexibility provided by 3GPP specs, selecting between transport layer security and application layer security, based on the interconnection schema or business needs.

This solution provides answers to the question for simplification. One of the arguments given was that the security model needs to avoid cumbersome negotiation procedures, which however in PRINS are relevant, as security configuration parameters and protected information elements need to be agreed between roaming partners and provided to IPX providers.

### 6.25.2 Solution details

To facilitate and simplify the deployment and operation of PRINS as one N32 security solution, it is proposed to introduce security profiles.

N32-c negotiation for PRINS is enhanced to allow selecting the existing scheme (for backward compatibility and high security requirements voiced in discussions) or selecting one or several security profiles.

- Only by selecting "full PRINS", negotiation of a cipher suite and exchange of modification and encryption policies is needed (current schema).

- If a pre-defined profile, e.g. "profile A" or "profile B", is chosen, a pre-defined security profile will be negotiated between SEPPs, and IPX can be instructed equally.

- "integrity-only PRINS" could be then one option, which in current understanding means, that JSON objects are created without encryption policies but integrity protected.

NOTE: Integrity-only PRINS may however not be preferable, since AVs and authorization tokens need protection

With this information, during N32-c handshake, if the PRINS enhanced profile, e.g., "B", is chosen, then both SEPPs (VPLMN and HPLMN) know how to handle the communication on the N32-f interface and the intermediary IPX providers as well. I.e., a profile indicator during N32-c negotiation phase can be propagated as an indication of the selected PRINS profile to the IPX; since only PRINS can be chosen, N32-f will always be based on application layer.

Table 6.25.2-1: Example of N32 security profile

|  |  |
| --- | --- |
| Enumeration value | Description |
| TLS | TLS security |
| PRINS  Profile full  Profile integrity-only  Profile A  Profile B  …  Operator defined profile | Protocol for N32 Interconnect security with subcategories to indicate full usage of PRINS, with integrity protection only or specific profiles |

If PRINS with "full PRINS" is chosen, configuration parameters can still be negotiated/exchanged, which keeps market open to those, really wanting this high security option

a. Modification policy. A modification policy indicates which IEs can be modified by an IPX provider of the sending SEPP.

b. Data-type encryption policy. A data-type encryption policy indicates which types of data will be encrypted by the sending SEPP.

c. Cipher suites for confidentiality and integrity protection, when application layer security is used to protect HTTP messages between them.

d. N32-f context ID. The N32-f context ID identifies the set of security related configuration parameters applicable to a protected message received from a SEPP in a different PLMN.

If PRINS with any other profile is chosen, the following configuration parameters need to be negotiated/exchanged and profiles need to be defined.

a) A PRINS profile indicating a predefined set of one or more of the above policies.

NOTE: Data type encryption policy for integrity-only PRINS profile: this policy will not specify any data type to be confidentiality protected; Modification policy for integrity-only PRINS profile: this policy will not specify any IE subject to be modifiable. Still, integrity protection is provided.

### 6.25.3 Evaluation

NOTE: Evaluation for this solution is not addressed in the present document.

## 6.26 Solution #26: Authorization of NF Service Consumer accessing Nnrf\_AccessToken service

### 6.26.1 Introduction

The following text sketches the proposed text for normative work needed to address KI#11.

### 6.26.2 Solution details

NRF (Network Repository Function) main and sole functionality as defined in TS 33.220 [4] can be summarized in two main functionalities:

- NRF is the 5GC repository for all 5GC Network Functions that belong to the operator 5GC network.

- NRF is the Authorization server within the operator 5GC network.

Therefore, in order for NRF to be the Repository of the operator 5GC Network, the NRF needs to have a record of all NFs that are present within the operator 5GC network. Not only 3GPP defined 5GC NFs but also it should include all those non-3gpp network functions.

- There is only one mechanism that is used to allow any NF to register its profile with the NRF. This procedure is described under clause 5.2.2.2. in TS 29.510 [3].

- NRF shall trust the information of registered NFs that is contained in the NF profile within the NRF. It is assumed that NF profile registration has been done in a secure manner. The NRF shall trust any information that is registered and contained within this NF profile.

- When a NF service consumer sends an access Token Request to the NRF, after the NRF authenticates the NF Service consumer using any valid secure mechanism as described in TS 33.501 [2], the NRF (as captured in [2]) shall validate whether the NF service consumer is authorized to receive the service it is requesting or not.

- As part of the authorization process of the NF Service Consumer, the NRF shall use the following information of the NF Service Consumer: NF Type, NF Instance ID, PLMN-ID, and FQDN.

- The NRF shall perform the above part of the authorization in the following order:

a. The NRF shall check if there is a registered profile for the NF service consumer.

If there is a registered profile, the NRF shall use such profile information to validate the NF Service Consumer access token request content, i.e., the following fields: NF Type, NF Instance ID, PLMN-ID, and FQDN.

If successfully validate, the NRF shall proceed with the rest of the authorization mechanism against the requested service.

b. If there is no NF profile registered with the NRF for the NF Service Consumer, the NRF shall validate the parameters from the Access token request against the parameters in the NF Service Consumer certificate. If successfully validated, the NRF shall proceed with the rest of the authorization mechanism against the requested service.

c. If the validation of the NF Service Consumer Access token Request content has not been successfully authenticated as per a or b above, the NRF shall reject the Access Token Request with the appropriate error code "Unauthorized".

### 6.26.3 Evaluation

NOTE: Evaluation for this solution is not addressed in the present document.

6.27 Solution 27: Dedicated N32 connections per PLMN in hosted SEPP deployment

6.27.1 Introduction

This solution addresses key issue #12. This solution describes the requirements for allowing several PLMNs to be customer of a SEPP provider and use the hosted SEPP functionality.

6.27.2 Solution details

The connection between the customer PLMN and the Hosted SEPP provider is protected in terms of confidentiality, integrity, and against replays by means of a secure channel. Several options exist for this secure channel, for example NDSIP (TLS, IPsec).

To fulfil the isolation requirement, the Hosted SEPP provider operates a dedicated SEPP instance for each PLMN it serves. This includes the usage, for each customer PLMN, of at least one separate N32c (and N32f) connection(s) towards the roaming partners. That is, it is prohibited to mix the signalling of multiple customer PLMNs over a single N32 connection due to the isolation requirement. Moreover, the TLS certificates used for N32 connections contain both the PLMN ID(s) of the customer PLMN and the own unique identifier of the hosted SEPP operator.

NOTE: The details of identifying the subject in the TLS certificate are out of scope of 3GPP.

It is further required that the secure channel endpoint (e.g., NDSIP endpoint) at the PLMN acts as a firewall that drops SEPP traffic that does not conform to configured filtering rules.

6.27.3 Evaluation

As the N32-c connections of a customer PLMN terminates at the hosted SEPP, the hosted SEPP represents the endpoint of the security relation with that PLMN's roaming partners.

Compared to the setting with a local, i.e., non-hosted SEPP, the attack surface in the setting of a hosted SEPP is increased because

- the Hosted SEPP provider is put into a position where it can inject signalling associated with one of its customers into a connection associated with another, and

- the source of unwanted traffic cannot be immediately and uniquely attributed as being injected by the Hosted SEPP provider or one of its customers' PLMNs (such attribution works only in a mediated way, i.e., requiring the hosted SEPP operator to cooperate in tracing back the source of signalling)

It is not possible to fully eliminate these shortcomings, because the operator of the hosted SEPP is explicitly trusted to represent the PLMN on the N32 interface as its network edge. These shortcomings can, however, be addressed by a robust governance framework which

- specifies a strict naming scheme that uniquely identifies the operator of the hosted SEPP

- defines conditions under which the hosted SEPP operators are granted the necessary TLS certificates

- installs monitoring mechanisms with rapid reaction, e.g., TLS certificate revocation makes use of the isolation requirement to enable roaming partners to selectively disable N32 connections mediated by the hosted SEPP provider without risking a situation of "overblocking".

The above mitigations are out of scope of 3GPP.

# 7 Conclusions

## 7.1 KI#1: Authentication of NRF and NF Service Producer in indirect communication

### 7.1.1 Analysis

The key issue addresses the scenario of an intermediary such as a standalone SCP to be compromised. In this case, the NF Service Consumer is not able to verify if the NRF response or the NF Service Producer response has been received by a legitimate entity. However, it also needs to be considered that reselection of the NF Service Producer by the SCP can be a desired feature.

3 solutions are presented to cover this key issue.

Solution #1 and extended solution #6 are based on using the concept of CCA for the NF Service Producer or the SCP, i.e., similar to the CCA used for the NF Service Consumer as specified in TS 33.501 [2]. Such a token is introduced to allow a client to validate the sender of a response directly, even if an SCP is in between.

Solution #1 has a limited scope as provided in the respective evaluation part in clause 6.1.

Solution #6 supersedes solution #1, overcoming some of the limits of solution#1 in case of Model C, also addressing the scenario of reselection of the target NF.

The optional inclusion of such a token including a NF Set Id allows a NF Service Consumer to validate if the NF sending the response is the producer that NFc has selected by itself or if it is a producer of the same NF Set or NF Service Set as indicated in the OAuth token received from NRF. The token cannot be used, if SCP has applied reselection of NFp outside of an NF Set.

Solution #6 addresses Model C with direct TLS between NF Service Consumer and NRF for discovery. It does not address Model D or Model C without direct TLS between NF Service Consumer and NRF for discovery. It requires that the NF Service Consumer has knowledge about which NF Service Producers are in the NF Set of the producer.

Solution #13 addresses similarly to solution #6 Model C. If the NF Service Consumer knows the producer, it can use the procedure described in #13 to have a verification of the producer sending the service response. However, in cases of delegated discovery and re-selection of an NF Service Producer, the SCP needs to be trusted since the NF Service Consumer cannot distinguish, if the SCP is acting maliciously or as wanted due to network performance issues. Thus, the behaviour on consumer side is hard to specify, i.e., accepting the response, because SCP was doing its job of reselection or rejecting the response, because the producer identity is not matching the one certified before.

The key issue assumed a compromised SCP, but the cases of when an SCP is compromised or when an SCP is just fulfilling its duty (i.e., re-selection), are not clear and were questioned during this study. As a consequence, this key issue is not followed up normatively.

### 7.1.2 Conclusion

It was decided that SCPs can only be used in situations in which all parties communicating through the SCP trust the SCP to correctly handle the messages passing through it.

No normative solution is pursued for addressing this KI.

## 7.2 KI#2: Need for additional security at operational level among SCP domains

### 7.2.1 Analysis

The key issue provides for operational aspects of SCP domains. Usually within an SCP domain, trust is assumed. If several SCP domains form a wider SCP trust domain, additional means for securing such trust domain against other SCP domains could be considered. However, further elaboration would be needed, whether such operational concept is useful.

### 7.2.2 Conclusion

The topic is not followed up normatively.

## 7.3 KI#3: Service access authorization in the "Subscribe-Notify" scenarios

### 7.3.1 Analysis

In TS 33.501[2] there is no reference to subscribe/notify, but it is clear that it has to be handled in the same manner as subscribe/request. Whether additional requirements would be needed for a notification recipient being different as the subscriber for the notifications can be checked in normative work, but the assumption is that the same mechanisms as for request/response apply.

Solution#12 is the only solution provided for this key issue. It only addresses the potential information leakage threat to an unauthorized NF. However, leakage of notifications from an authorized compromised recipient to an unauthorized recipients cannot be avoided, which is out of scope of the solution. It is a matter of trust that a recipient that has been beforehand authorized to receive notifications, is not forwarding or leaking it to other NFs. The study identified that no token-based authorization method for notification messages has been specified. It is agreed to not specify a solution for callback-URI authorization as part of the normative work following this study, since it is not clear whether the information leakage can be prevented by the proposed solution and no further solutions have been identified.

NOTE: How to assure that the notification message is sent to authorized NF is not addressed in the present document.

### 7.3.2 Conclusion

A conscious decision is made to not apply token-based authorization to notification operations.

No normative work on the solutions described in this study is pursued, but TS 33.501[2] needs to be updated with a clarification that token-based authorization also applies to subscribe and unsubscribe operations, which is described in detail in stage 3, but is missing in stage 2 requirements.

## 7.4 KI#4: Authorization of SCP to act on behalf of an NF or another SCP

### 7.4.1 Analysis

One of the main 5G security features is the usage of OAuth 2.0 to authorize service requests. In indirect communication scenarios, however, an SCP is not explicitly authorized, it could request a service or a token without the knowledge of the NF Service Consumer (NFc).

Two solutions (Sol#2 and Sol#3) have been proposed, mandating the usage of CCA, by which the NFc can at least "implicitly authorize" the SCP. This achieves a minimum level of trust, but cannot avoid the threat described in clause 5.4.2 within the validity time of the CCA, i.e., that "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". Thus, CCA validity time need to be chosen very carefully to minimize this possibility. A third solution (Sol#23) proposes that NRF has in its database knowledge about the SCPs that could request a service on behalf of a consumer.

Solution #3 argues that existing mechanisms are sufficient: a NF sending a request (with its CCA) to SCP, implicitly authorises SCP against NRF to process this request. The limit of this solution is that the CCA can be stolen or misused and, without NFc having requested a service, NRF could provide an access token to SCP or NFp could provide a service to SCP.

Solution #2 proposes to include the SCP identifier (i.e., either SCP Instance and/or Domain info) in the CCA generated by the NF Service Consumer. Thus, only the SCP, which SCP Identifier is included in the CCA from NFc is able to get a response from NRF to its request. This ensures that the CCA cannot be misused by a MitM and thus achieves an additional level of trust, when the CCA is more than once useable. This however comes with a trade-off, since it impacts existing implementation (CCA generation by NFc and NRF handling of CCA). Further, if more than one SCP is in the path, with this solution SCP needs to create its own CCA including the SCP identifier from the requesting SCP in order to keep the chain of trust.

Solution #23 provides SCP with an access token for NFc that is bound to be requested only by the SCPs listed in the NF profile as registered by the consumer at NRF. It also proposes to include information about the service request in the CCA, which would allow the final recipient of the query to verify if the SCP forwarding the request on behalf of the consumer not changed anything. This solution would require an update to CCA and additions in the NF profile. This solution only works if there is a secure way to provision the knowledge about the SCP(s) on the path to the NRF. Including information about the service request in the CCA can lead to service interruption if a well-behaving SCP modifies the service request according to its policies.

Any conclusion to this key issue needs to weight the level of security and trust achieved against the impact of suggested enhancements to the current release.

Another aspect, which requires further analysis is whether explicit authorization of the SCP would be needed. NRF can only authorize NFs. NRF needs to trust that any request forwarded by SCP was initiated by a NFc. Thus, in general only implicit authorization of SCP by the NF Service Consumer is possible with the current solutions proposals based on CCA.

In summary, it is a trade of between security enhancement versus complexity added.

### 7.4.2 Conclusion

It was decided that SCPs can only be used in situations in which all parties communicating through the SCP trust the SCP to correctly handle the messages passing through it.

No normative solution is pursued for addressing this KI.

## 7.5 KI #5: End-to-end integrity protection of HTTP messages

### 7.5.1 Analysis

The KI #5 is about end-to-end protection of HTTP message for indirect communication. The key issue seek for solutions on how an NF Service Producer can verify that a service request of the NF Service Consumer received via SCP has not been modified tampered with.

Parameters in the request messages may be modified by the intermediate node, and end-to-end protection of HTTP headers and bodies is discouraged, i.e., end-to-end protection of HTTP headers and bodies and segregating the parameters based on SCP (or SEPP) modification criteria. There can be multiple SCPs and/or SEPPs involved in the service request/response path where SCP can modify the HTTP headers, and SEPP (or IPX) can modify the HTTP headers and/or HTTP body attributes. In addition, there may be potential compatibility issues with end-to-end protection of HTTP headers and bodies.

### 7.5.2 Conclusion

One of the fundamental functionalities of SCP is message mediation which may require SCP to modify certain headers and/or content of the HTTP message, typical end-to-end message integrity protection between the far end NFs seems not feasible.

For this reason, 3GPP so far relies on hop-by-hop security.

No normative solution to be pursued for addressing this KI.

## 7.6 KI#6: Access token usage by all NFs of an NF set

### 7.6.1 Analysis

Only solution #7 is provided for this key issue. When implementing NF Sets a common context storage needs to be organised, from which all NFs of the set can retrieve the same status. There is no difference in this handling for NF Service Producer and NF Service Consumer.

The main hurdle is that a NF Service Consumer is not necessarily a NF Service Producer, in which case no NF profile registration at NRF is needed. However, NF Service Consumers need to register with the OAuth 2.0 server and as described in the solution, the NF Set ID or NF Service Set ID is needed to be known at the NRF for the solution to work.

The NF Set management is out of scope but not different from using services of different NF Service Producers within the same set by the same token.

Further evaluation would be necessary, but this is not addressed in the present document.

### 7.6.2 Conclusion

No normative solution to be pursued for addressing this KI.

## 7.7 KI#7: Authorization mechanism determination

### 7.7.1 Analysis

The key issue is for studying that 5GS should provide mechanisms to handle the case that one operator uses token-based authorization and its roaming partner uses static authorization. Solutions (Solution #9, and #17) were proposed in this regard.

TS 33.501 [2] mandates the support of OAuth2.0; and GSMA recommends (NG.113 [8], clause 7.6.3.4) that roaming partners support the same authorization method. However, if one operator uses token-based authorization and its roaming partner uses static authorization, TS 29.510 [6] is not clear, because only the OAuth2required use case is described.

Solution #9 provides a negotiation method for the case that one operator uses token-based authorization and its roaming partner uses static authorization. The usage of static authorization only by VPLMN seems to involve additional management effort on the HPLMN hNRF side for defining authorization policies per roaming partner. It further involves the risk that a vNRF can dictate the hNRF its own conditions on which authorization method to use.

Solution #17 is using existing stage 3 methods, which allow hNRF to configure per PLMN whether OAuth2.0 method is required. However, also for this approach some management effort is needed. Further, the existing stage 3 methods emphasize on OAuth2.0 only. The solution proposes to provide an explicit statement on static authorization if OAuth2.0 is not required. Solution #17 requires that one network knows the capability of the other network.

Solution #24 provides a negotiation method with bootstrapping mechanism. The solution proposes a simple procedure to handle the authorization mechanism negotiation. It is applicable for the case that one operator uses token-based authorization, and its roaming partner uses static authorization.

### 7.7.2 Conclusion

No normative work is recommended. This key issue is related to Inter-PLMN roaming discussion and is not addressed in the present document.

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

### 7.8.1 Analysis

As described in clause 6.2.6.1 of TS 23.501 [3], an operator network can deploy multiple NRFs, for example due to network slicing or network segmentation.

A clause on handling access token requests in deployments with several NRFs is missing in TS 33.501 [2].

Solution #10 addresses one part, i.e., making NF Service Consumer instance and type available to a slice specific NRF. As provided by evaluation, the solution is technical possible but seems to have high impact in realization.

For normative work, rather a generic description is desired how to handle multiple NRF scenarios.

### 7.8.2 Conclusion

A new clause to TS 33.501 [2] along the lines of the following text should be added to clarify NRF deployment scenarios.

*If an NRF receives an access token request for an NF Service Producer that is not registered at this NRF, the NRF has to determine the target NRF where the NF Service Producer is registered. Only afterwards it can forward the access token request to the target NRF.*

*There can also be several hops of NRFs between the NRF that receives the access token request from the NF Service Consumer and the target NRF where the NF Service Producer is registered.*

*One option of hierarchical NRF deployment is the local NRF deployment, where the local NRF checks if the NF Service Consumer is authorized to receive the requested service and, if yes, issues and signs the access token. In the case when the access token request from the NF Service Consumer was forwarded by another NRF, the local NRF of the NF Service Producer needs to trust the NRF which forwarded the access token request.*

The part of Solution #10 of making NF Service Consumer instance and type available to a slice specific NRF has not been concluded for normative work since there was no common view on whether the solution provides an optimization.

## 7.9 KI #9: Authorization for Inter-Slice Access

### 7.9.1 Analysis

Solution #11 and solution #18 address this key issue.

Solution #11 proposes to include "*allowedSnssaisPerSnssai"* or "*allowedOperationsPerSnssai"* in the NF/NF-Service profile registered by the NF Service Producers in NRF. The parameter "*allowedOperationsPerSnssai"* allows finer control of scopes allowed to a consumer from a different slice.

Additionally, use of parameter "Requester-NSSAI" is an enhancement to existing Access-Token Grant, and allows NF-producers to authorize the NF-Consumer's S-NSSAI if the NRF has validated the S-NSSAIs.

Solution #18 provides a minimal update as normative text proposal by demanding the NRF to check on the NF Service Consumer being authorized to access a slice of the NF Service Producer.

### 7.9.2 Conclusion

It is proposed to add the following requirements in the normative phase:

- The 5GS should provide a mechanism that allows the NF Service Producer not to provide service to NF Service Consumers, which are not authorized to access a slice.

- The 5GS should provide a mechanism that allows NRF not to provide access tokens to NF Service Consumers which are not authorized to access a slice.

Further, normative text in clause 13.4.1.1.2. of TS 33.501 [2] should be added to update in line with solution #18.

## 7.10 KI #10: N32 security in mediated roaming scenarios

### 7.10.1 Analysis

Further elaboration is needed on this topic. It is therefore decided to outsource this key issue to a dedicated study on N32 roaming topics.

### 7.10.2 Conclusion

This key issue cannot be concluded yet. It is continued to be studied in a dedicated study.

## 7.11 KI #11: NRF validation of NFc for access token

### 7.11.1 Analysis

Several problems have been identified in this key issue and are analysed in the following.

**Regarding problem 1a**

The problem description is about an NFc profile not being available. This problem statement would need further differentiation. In order to get authenticated and authorized for service consumption at least the OAuth2.0 client registration information of the NF Service Consumer needs to be available at the NRF.

In detail: If the NF Service Consumer is not acting as resource server (NF Service Producer), i.e., being a pure consumer NF, this NFc will not register services at NRF. Hence, no NF profile with services will be available at NRF. However, in order to request an authorization token, the NF Service Consumer needs to be known to NRF as OAuth 2.0 client (see TS 33.501 [2], clause 13.4.1). Thus, at minimum a consumer NF needs to be known to the NRF by its OAuth2.0 client registration information. Even though TS 33.501 [2] lists the usage of NF Service Registration procedure for registering the OAuth2.0 client as an option, using NF Service Register is only sensible if the NF is also providing services (i.e. as producer). TS 33.501 [2] lacks a description, how to do OAuth client registration if no NF profile is registered. OAM can be used for this.

**Regarding problem 1b**

This problem is about how NRF is validating the NFc, i.e., which information to use if the NF has also registered its NRF profile or the OAuth 2.0 client information and then providing its TLS certificate with the access token request. In the problem statement it was asked whether one takes precedence over the other.

The registering entity needs to assure that information registered at the NRF is in line with the information provided in the NF TLS certificate or CCA or the information provided by SCP in the service request, at minimum NF Instance ID and PLMN ID. The local NRF, where the NFc is registered with its OAuth client registration needs therefore to validate and match accordingly before requesting the NRF where the services of a producer are registered.

**Regarding problem 2**

This problem requests for clarification on which of the two takes precedence: the NF profile information or OAuth client registration information known by the NRF, or the NFc TLS certificate received by NRF, when NFc is asking for an OAuth 2.0 access token.

The situation can apply, if the NF Service Consumer has already registered as NF Service Producer or when the NF Service Consumer was registered as OAuth2.0 client. In both situations, the registering entity needs to assure that the information in the NRF is in line with the information issued in the NF's TLS certificate, thus none takes precedence, but in case of a mis-match, the authorization request cannot be granted. There are also situations that the NRF has no TLS certificate, i.e., if SCP has authenticated NFc. In this case the NRF either uses CCA if available or trusts the SCP and relies on the information in the access token request.

**Solution summary**

Solution #21 proposes that NRF uses the information in the NF Service Consumer's certificate to base its authorization decision on.

Solution #22 proposes that the NRF uses the information in the NF Service Consumer's certificate and additional information in the NF profile as basis for its authorization decision. Mandating of NF instance ID being part of the certificate profile is a pre-condition to not run into interoperability issues.

Solution #26 proposes NFc authorization by NRF is done using the existing profile information ( OAuth 2.0 client profile only or if available NFp profile).

**Solution evaluation comments**

TLS certificate usage by NRF is only possible, if available, i.e., in direct communication. In indirect communication the certificate is used by SCP for authentication. NRF can either trust SCP authentication or use CCA for validation. If information elements are part of both, certificate or CCA and in the request, they need to match.

### 7.11.2 Conclusion

A normative clarification is needed along these lines:

The NRF can validate the NF instance ID in access token requests to the registered OAuth 2.0 client instance ID.

If the certificate and/or NF profile is available to the NRF, the NRF validates information presented in the access token request (i.e., NF instance ID, NF type, PLMN ID) against the same information in the certificate and/or profile, which addresses Problem 2.

If the validation fails, NRF rejects access token request.

NOTE: It is assumed that NF instance ID, NF type and PLMN ID are mandatory parameters in the certificate. By this problem 1b is also addressed.

Additions to TS 33.501 [2] are necessary such that OAuth client registration information could be provided also by OAM, which would address problem 1a.

## 7.12 KI #12: Security in Hosted SEPP scenarios

### 7.12.1 Analysis

Further elaboration is needed on this topic. It is therefore decided to outsource this key issue to a dedicated study on N32 roaming topics.

### 7.12.2 Conclusion

The conclusion is not addressed in the present document.

Annex A:  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2023-09 | SA#101 | SP-230895 | 0001 | 1 | F | Editorial cleanups | 18.1.0 |
| 2023-09 | SA#101 | SP-230895 | 0002 | - | F | Scope alignment with key issues addressed | 18.1.0 |
| 2023-09 | SA#101 | SP-230895 | 0003 | - | F | Clarifications on deployment options | 18.1.0 |
| 2023-09 | SA#101 | SP-230895 | 0004 |  | F | Key issue 11 editorial updates | 18.1.0 |
| 2023-09 | SA#101 | SP-230895 | 0005 | - | F | Alignment of key issue 12 with GSMA input | 18.1.0 |
| 2023-09 | SA#101 | SP-230895 | 0006 | - | F | Evaluation clarification to solution for access tokens for NFc Sets | 18.1.0 |