|  |  |
| --- | --- |
| 3GPP TR 33.738 V18.1.0 (2023-09) | |
| Technical Report | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects;  Study on security aspects of enablers for network automation for the 5G system phase 3;  (Release 18) | |
|  | |
|  | 3GPP-logo_web |
|  | |
| 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 6

1 Scope 8

2 References 8

3 Definitions of terms, symbols and abbreviations 8

3.1 Terms 8

3.2 Symbols 9

3.3 Abbreviations 9

4 Overview 9

5 Key issues 9

5.1 Key Issue #1: Protection of data and analytics exchange in roaming case 9

5.1.1 Key issue details 9

5.1.2 Security threats 9

5.1.3 Potential security requirements 10

5.2 Key Issue #2: Authorization of selection of participant NWDAF instances in the Federated Learning group 10

5.2.1 Key issue details 10

5.2.2 Security threats 10

5.2.3 Potential security requirements 10

5.3 Key Issue #3: Security for AI/ML model storage and sharing 11

5.3.1 Issue details 11

5.3.2 Security Threats 11

5.3.3 Potential security requirements 11

5.4 Key Issue #4: Anomalous NF behaviour detection by NWDAF 11

5.4.1 Issue details 11

5.4.2 Security Threats 12

5.4.3 Potential security requirements 12

5.5 Key Issue #5: KI on Security for NWDAF-assisted application detection 12

5.5.1 Key issue details 12

5.5.2 Security threats 12

5.5.3 Potential security requirements 12

5.6 Key Issue #6: Key issues on Cyber-attack detection 13

5.6.1 Key Issue Details 13

5.6.2 Security Threats 13

5.6.3 Potential Security Requirements 13

6 Solutions 13

6.0 Mapping Solutions to Key Issues 14

6.1 Solution #1: Authorization of AI/ML model retrieving 14

6.1.1 Introduction 14

6.1.2 Solution details 14

6.1.2.1 Authorization of AI/ML model retrieving from NWDAF 14

6.1.2.2 Authorization of AI/ML model retrieving from ADRF 14

6.1.3 Evaluation 15

6.2 Solution #2: Authorization and Authentication of ML model transfer 15

6.2.1 Introduction 15

6.2.2 Solution Details 15

6.2.3 Evaluation 17

6.3 Solution #3: AI/ML model authorization and retrieval 17

6.3.1 Introduction 17

6.3.2 Solution details 17

6.3.3 Evaluation 19

6.4 Solution #4: AI/ML model storage and sharing security 20

6.4.1 Introduction 20

6.4.2 Solution details 20

6.4.3 Evaluation 23

6.5 Void 23

6.6 Solution #6: Anomalous NF behaviour detection by NWDAF 23

6.6.1 Introduction 23

6.6.2 Solution details 24

6.6.2.1 General 24

6.6.2.2 Collection of security related log data of NFs via NFs EventExposure APIs 24

6.6.2.3 Collection of security related log data of NFs via OAM 25

6.6.3 Evaluation 28

6.7 Solution #7: Secured and authorized AI/ML model transfer and retrieval 29

6.7.1 Introduction 29

6.7.2 Solution details 29

6.7.3 Evaluation 31

6.8 Solution #8: Protection of data and analytics exchange in roaming case 32

6.8.1 Introduction 32

6.8.2 Solution details 32

6.8.3 Evaluation 34

6.9 Solution #9: Anomalous NF behaviour event related data collection and anomalous NF detection 34

6.9.1 Introduction 34

6.9.2 Solution details 34

6.9.3 Evaluation 37

6.10 Solution #10: Authorization of AI/ML model sharing between different vendors and usage of one-time URLs 37

6.10.1 Introduction 37

6.10.2 Solution details 38

6.10.3 Evaluation 40

6.11 Solution #11: Secured and authorized AI/ML model sharing between different vendors 40

6.11.1 Introduction 40

6.11.2 Solution details 40

6.11.3 Evaluation 43

6.12 Solution #12: Secured and authorized AI/ML model sharing between different vendors 43

6.12.1 Introduction 43

6.12.2 Solution details 43

6.12.3 Evaluation 45

6.13 Solution #13: Support authorization of AI/ML model sharing By NWDAF containing MTLF 45

6.13.1 Introduction 45

6.13.2 Solution details 46

6.13.3 Evaluation 48

6.14 Solution #14: Protection of data and analytics exchange in roaming case using Secure Multi-party Computation 48

6.14.1 Introduction 48

6.14.2 Solution details 48

6.14.2.1 Data Privacy Computing Function DPCF 48

6.14.2.2 Data collection protection architecture based on Secure Multi-party Computation 49

6.14.3 Evaluation 50

6.15 Solution #15: Authorization of selection of participant NWDAF instances in the FL group 50

6.15.1 Introduction 50

6.15.2 Solution details 50

6.15.3 Evaluation 51

6.16 Solution #16: FL GROUP AUTHORIZATION OF NWDAF(S) IN 5GC 51

6.16.1 Introduction 51

6.16.2 Solution details 51

6.16.3 Evaluation 53

6.17 Solution #17: Authorization of FL Server and FL Client in FL group 54

6.17.1 Introduction 54

6.17.2 Solution details 54

6.17.3 Evaluation 56

6.18 Solution #18: Solution for authorization in FL 56

6.18.1 Introduction 56

6.18.2 Solution details 56

6.18.3 Evaluation 58

6.19 Solution #19: Access control for user plane data 58

6.19.1 Introduction 58

6.19.2 Solution details 58

6.19.3 Evaluation 60

6.20 Solution #20: Cyber attack detection using an analytics function 60

6.20.1 Introduction 60

6.20.2 Solution details 60

6.20.3 Evaluation 63

6.21 Solution #21: Authorization of data and analytics exchange in roaming case 63

6.21.1 Introduction 63

6.21.2 Solution details 64

6.21.3 Evaluation 65

7 Conclusions 65

7.1 Conclusion on Key Issue #1 "Protection of data and analytics exchange in roaming case" 65

7.2 Conclusion on Key Issue #2 "Authorization of selection of participant NWDAF instances in the Federated Learning group" 65

7.3 Conclusion on Key Issue #3 "Security for AI/ML model storage and sharing" 66

7.4 Conclusion on Key Issue #4 " Anomalous NF behaviour detection by NWDAF " 66

7.5 Conclusion on Key Issue #5 "Security for NWDAF-assisted application detection" 66

7.6 Conclusion on Key Issue #6 "Key issues on Cyber-attack detection" 67

Annex A: Change history 68

# Foreword

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

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

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

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

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

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

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

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

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

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

**should** indicates a recommendation to do something

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

**may** indicates permission to do something

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

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

**can** indicates that something is possible

**cannot** indicates that something is impossible

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

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

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

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

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

In addition:

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

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

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

# 1 Scope

The present document studies the security aspects of enablers for network automation for the 5G system based on the outcome of TR 23.700-81 [6]. More specifically, the present document will identify security issues and requirements and provide corresponding security solutions related to the following scenarios:

- Security aspects of potential architecture enhancement: roaming, supporting federated learning, interaction between NWDAF and MDAS/MDAF, etc.

- Handling of sensitive information inherent to application detection, roaming and location information.

- KIs from R17 which do not have enough time to proceed, e.g. NWDAF detection related issues.

- Any further security enhancements which need to be studied based upon the ongoing SA2 eNA Phase 3 work

# 2 References

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

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 23.501: "System Architecture for the 5G System; Stage 2".

[3] 3GPP TS 23.502: "Procedures for the 5G system, Stage 2".

[4] 3GPP TS 23.503: "Policy and Charging Control Framework for the 5G System".

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

[6] 3GPP TR 23.700‑81: " Study of Enablers for Network Automation for 5G System (5GS); Phase 3".

[7] 3GPP TS 33.501: "Security architecture and procedures for 5G System".

[8] 3GPP TS 29.510: "Network function repository services; Stage 3".

[9] 3GPP TS 28.552: "5G performance measurements".

[10] 3GPP TR 33.875: "Study on enhanced security aspects of the 5G Service Based Architecture (SBA)".

# 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].

## 3.2 Symbols

Void.

## 3.3 Abbreviations

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

# 4 Overview

The architecture for the present study is be based on the existing NWDAF framework as specified in TS 23.288 [5], TS 23.501 [2], TS 23.502 [3] and TS 23.503 [4].

Solutions comply with the 5G System architectural principles in TS 23.501 [2], and network data analytics principles in TS 23.288 [5].

TR 23.700-81 [6] is an enhanced study on eNA Phase 3, for which any security impact will be documented in the present document.

# 5 Key issues

## 5.1 Key Issue #1: Protection of data and analytics exchange in roaming case

### 5.1.1 Key issue details

As per Key Issue #3 in TR 23.700-81 [6], the HPLMN/VPLMN may need to collect data or consume analytics from the VPLMN/HPLMN in roaming scenario. The data or analytics may relate to particular UEs or contain information about all UEs or groups of UEs. Both PLMNs (VPLMN, HPLMN) need the ability to control the amount of data exposed and to abstract or hide network-internal aspects based on operator policy, regulatory constraints and/or roaming agreements.

In the scenario where the NWDAF consumer collects data from multiple PLMNs, PLMNs need the ability to control the amount of data exposed and to abstract or hide network-internal aspects based on operator policy, regulatory constraints and/or roaming agreements. But if the data processed method used by different PLMNs is different, it is possible that the data converged in NWDAF consumer could not be effective used.

As there might be possible architecture enhancements to support this exchange in roaming scenarios and of any necessary enhancements to related NFs in HPLMN and VPLMN. This key issue studies the security aspects of data and analytics exchange in roaming case should be considered.

### 5.1.2 Security threats

If the communication between PLMNs is not confidentiality protected, then sensitive information may be leaked to unauthorized entities.

If the integrity of the data exchanged between PLMNs is not protected, the data may be modified.

If the PLMNs do not have the ability to control the amount of data exposed and to abstract or hide network-internal aspects, the sensitive data may be leaked to other entities.

If exchanged data has not been adequately protected before it is shared between PLMNs, it may be subject to be leaked and abused.

### 5.1.3 Potential security requirements

5GS shall support confidentiality, integrity, and replay protection for data and analytics exchange between PLMNs.

5GS shall be able to control the amount of exposed data and to abstract or hide internal network aspects based on operator policy, regulatory constraints and/or roaming agreements.

Based on the sensitivity of the data, 5GS shall provide confidentiality protection while at rest to certain data based on policies.

NOTE: The user consent for UE data collection is not addressed in the present document.

## 5.2 Key Issue #2: Authorization of selection of participant NWDAF instances in the Federated Learning group

### 5.2.1 Key issue details

3GPP SA2 studies the architecture enhancement to support Federated Learning which allows the cooperation of multiple NWDAFs containing MTLF to train an ML model in 3GPP network. As per KI#8 in TR 23.700-81[6], it will involve selection of participant NWDAF instances in the Federated Learning group.

This key issue studies the authorization aspect of including participant NWDAF instances in the Federated Learning group.

### 5.2.2 Security threats

If a client NWDAF instance joins an unauthorized Federated Learning, it may lead to the following issues:

- Client NWDAF(MTLF)'s resource may be used up by being included into many unauthorized Federated Learning groups.

- Sensitive data may be used to train unauthorized Federated Learning group's ML model.

- Unauthorized Federated Learning group may utilize the local model received from the client NWDAF(MTLF) to infer sensitive training data details.

If a client NWDAF joins a Federated Learning group without being authorized by the server NWDAF, it may lead to the following issues:

- The unauthorized client NWDAF may affect the generation of Federated Learning group's ML model negatively.

- Sensitive training data and the Federated Learning group's ML model may leak out to the unauthorized client NWDAF.

### 5.2.3 Potential security requirements

Authorization of selection of participant NWDAF instances in the Federated Learning group shall be supported:

- A server NWDAF shall be authorized to include a client NWDAF into a Federated Learning group.

- A client NWDAF shall be authorized to join a Federated Learning group.

NOTE: The existing security mechanisms in SBA are reused as much as possible.

## 5.3 Key Issue #3: Security for AI/ML model storage and sharing

### 5.3.1 Issue details

AI/ML model is shared among NWDAFs and/or NFs (i.e., NWDAF to NWDAF, ADRF to NWDAF…). In different scenarios, the NF producer of AI/ML model can store that model in ADRF, NWDAF or other entity.

ADRF (Analytical Data Repository Function) is being enhanced to store AI/ML models to facilitate the distribution and sharing of those models amongst NFs. Since AI/ML models and their algorithms are generally proprietary (i.e., subject to intellectual property rights of the designer), it is imperative to ensure that only the NFs which have been indeed provided by the NF which generated the AI/ML model with access authorization to the AI/ML models can read and use those models. Moreover, the ADRF itself cannot be considered as a fully trusted entity storing the sensitive AI/ML data models. Those models are indeed exposed at rest in ADRF.

The current authorization scheme defined by 3GPP for SBA works only at service level or resource/operation-level scope. This authorization granularity may be not sufficient in the AI/ML model sharing scenario, since the ADRF (Analytical Data Repository Function) or NWDAF, or any other network function which may store the AI/ML model, cannot verify whether the NF consumer is authorized to retrieve the AI/ML model.

### 5.3.2 Security Threats

An unauthorized NFc, in principle which is not eligible to retrieve a particular model stored by a NFp, could have access to the storage entity and retrieve the model.

If there is no protection against accessing and reading an AI/ML model from the ADRF stored by NFp, a compromised ADRF may expose algorithms and sensitive data to a non-authorized entity which can easily misuse it and/or distributed further to other entities, causing a bigger data security breach.

### 5.3.3 Potential security requirements

AI/ML models shall be protected between the entity which produces the ML model or stores the ML model in ADRF (e.g., NWDAF containing MtLF, NFp) and the entity which consumes the model (NFc).

ADRF (Analytical Data Repository Function), or any other network function which may store the AI/ML model, shall be able to check that the NFc is authorized by the NFp (e.g., NWDAF containing MtLF) to retrieve that AI/ML model

NF Service consumers shall be authorized by the NF which generated the AI/ML model to access to the AI/ML models in the ADRF (or any other NF which may store the ML model, for instance NWDAF MtLF).

## 5.4 Key Issue #4: Anomalous NF behaviour detection by NWDAF

### 5.4.1 Issue details

The 5GC supports different NF deployments that could be in distributed or redundant fashion, so that the NFs provides the services from several locations and several execution instances. When these NFs are distributed across diverse cloud infrastructures, it is possible that the NFs may behave in an undefined manner. The undefined behaviour of the NF may be caused by internal errors such as configuration mistakes or internal data corruption. This misbehaviour may impact one or more UE services based on the type of the NF.

In those circumstances, it is imperative that an analytics function such as NWDAF supports the monitoring of the behaviour of all NFs and ensures that the NFs behave as defined/specified. If the NFs behave erroneously, it should be possible to detect the anomaly, so that appropriate steps can be taken, e.g., by an operator to control the potentially damaging behaviour.

### 5.4.2 Security Threats

Different NFs may behave in an undefined manner. Anomalous NF behaviour could include among others, failed attempts to access NF/NF services which was not authorized to a NF as NF/NF service consumer, unusual high consumption of network or compute resources by a particular NF/NF service (consumer or producer), continuous sending of compromised messages to particular NF service producer (DoS), numerous attempts to exhaust connections of a HTTP server, etc.

The above examples of anomalous behaviour can occur due to internal data corruption, configuration errors, communication between NFs from different vendors (i.e., incompatibility issues), etc. Based on the NF type, such behaviour could cause damage to one or multiple UEs. For example, in the case of an AMF or SMF dedicated to a network slice, the service for all UEs within the whole network slice could be affected. Even the complete network slice could get out of service.

An erroneous NF may succeed in outaging the whole network by sending wrong messages to other NFs, causing those NFs to get out of service.

The NFs within the 5GC are already authenticated and allowed to communicate with each other based on successful authentication and authorization. If the NF is misconfigured or its internal data is corrupted or has other reason for anomalous behaviour, the assumption of trust on that NF becomes invalid, even if previously authenticated and authorized, and may cause potential threats and exploits.

### 5.4.3 Potential security requirements

It should be possible for the network to detect anomalous NFs using the data collected from NFs.

NOTE: By this requirement it is only assured that specific data can be collected by and/or reported to an analytics function. Which specific detection mechanism (e.g., AI/ML algorithm) is used is implementation specific and out of the scope in 3GPP.

## 5.5 Key Issue #5: KI on Security for NWDAF-assisted application detection

### 5.5.1 Key issue details

As per Key Issue #2 in TR 23.700-81 [6], the detection of traffic generated by an application can be performed using the application detection filters in the UPF/SMF and those application detection filters may include Packet Flow Description(s), i.e. PFD(s). Depending on service level agreements between the operator and the Application Service Provider (ASP), the ASP may provide PFD(s) for each application identifier maintained by the ASP.

User plane data need to be utilized by NWDAF to output analysis results while it may include sensitive user privacy information. In this case, if the raw user plane data input to NWDAF without privacy protection process or data masking, it may result in user privacy information leakage or violating privacy related regulation.

### 5.5.2 Security threats

If there is no security mechanism applied in user plane data before been utilized by NWDAF, the possibility of user data leakage will increase.

If the privacy related information in user plane data has not been processed before been utilized by NWDAF, the possibility of privacy regulation violation will increase.

### 5.5.3 Potential security requirements

Privacy protection mechanism should be applied in the user plane data before it has been utilized by NWDAF

## 5.6 Key Issue #6: Key issues on Cyber-attack detection

### 5.6.1 Key Issue Details

TR 23.700-91 [x] has identified the use case of NWDAF detecting cyber-attacks by monitoring events and data packets in the UE and the network, with the support of machine-learning algorithms. To achieve cyber-attacks detection, the NWDAF can collaborate with UE and any other NFs to collect related data as inputs and providing alerts of anomaly events as outputs to OAM and other NFs which have subscribed to them so that they could take proper actions.

This key issue describes what kind of cyber-attacks can be detected. In order to mitigate the identified cyber-attacks, the data/parameters collected by the NWDAF or any other NF need to be studied.

The specific cyber-attacks for which an analytics function may provide detection support include but are not limited to the following examples:

(1) MitM attacks on the radio interface: MitM attacks or fraudulent relay nodes may modify or change messages between the UE and the RAN, resulting in failures of higher layer protocols such as NAS or the primary authentication.

(2) DoS attacks: 5G has high performance requirements for system capacity and data rate, improved capacity and higher data rate may lead to much higher processing capability cost for network entities, which may make some network entities (e.g. RAN, Core Network Entities) to suffer from DDoS attack. The NFs may also enable the detection of DDoS attacks.

### 5.6.2 Security Threats

Cyber-attack may not be detected by the 5G network; thus further attacks could be conducted.

Anomaly events may not be detected by the 5G network; thus further attacks could be conducted.

### 5.6.3 Potential Security Requirements

The 3GPP system shall support the detection of cyber-attacks by providing related inputs or collecting output analytics using an analytics function (for e.g., NWDAF).

# 6 Solutions

## 6.0 Mapping Solutions to Key Issues

Table 6.0-1: Mapping of solutions to key issues

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Key Issues | | | | | |
| Solutions | 1 | 2 | 3 | 4 | 5 | 6 |
| 1 |  |  | X |  |  |  |
| 2 |  |  | X |  |  |  |
| 3 |  |  | X |  |  |  |
| 4 |  |  | X |  |  |  |
| 5 | X |  |  |  |  |  |
| 6 |  |  |  | X |  |  |
| 7 |  |  | X |  |  |  |
| 8 | X |  |  |  |  |  |
| 9 |  |  |  | X |  |  |
| 10 |  |  | X |  |  |  |
| 11 |  |  | X |  |  |  |
| 12 |  |  | X |  |  |  |
| 13 |  |  | X |  |  |  |
| 14 | X |  |  |  |  |  |
| 15 |  | X |  |  |  |  |
| 16 |  | X |  |  |  |  |
| 17 |  | X |  |  |  |  |
| 18 |  | X |  |  |  |  |
| 19 |  |  |  |  | X |  |
| 20 |  |  |  |  |  | X |
| 21 | X |  |  |  |  |  |

## 6.1 Solution #1: Authorization of AI/ML model retrieving

### 6.1.1 Introduction

This solution is addressing KI#3. It explains how to use NFProfile->NFServiceList-> allowedOperationsPerNfType and/or NFProfile->NFServiceList-> allowedOperationsPerNfInstance attributes defined in TS 29.510 to authorize a specific NF type and/or a specific instance ID of the NF Service Consumer to retrieve AI/ML models.

### 6.1.2 Solution details

#### 6.1.2.1 Authorization of AI/ML model retrieving from NWDAF

According to TS 23.288, Nnwdaf\_MLModelProvision service enables the consumer to receive a notification when an ML model matching the subscription parameters becomes available, and Nnwdaf\_MLModelInfo service enables the consumer to request and get from NWDAF containing MTLF ML Model Information.

An NWDAF can add Nnwdaf\_MLModelProvision\_Subscribe service operation in "allowedOperationsPerNfType" and/or "allowedOperationsPerNfInstance" for specific NF type and/or specific instance ID of the Consumer, and register its NF profile to NRF. When a NF Service Consumer requests an access token for the Nnwdaf\_MLModelProvision\_Subscribe to retrieve AI/ML models, the NRF will only grant the access token if Nnwdaf\_MLModelProvision\_Subscribe is present in either "allowedOperationsPerNfType", for the NF type of the NF Service Consumer, or in "allowedOperationsPerNfInstance", for the instance ID of the NF Service Consumer. This procedure also applies to Nnwdaf\_MLModelInfo\_Request service operation.

#### 6.1.2.2 Authorization of AI/ML model retrieving from ADRF

As per TR 23.700-81's conclusion to KI: how to Enhance Data collection and Storage, ADRF ML model retrieval service and relevant service operation will be used. Take Nadrf\_MLModelManagement and the service operation Nadrf\_MLModelManagement\_RetrievalRequest as examples.

An ADRF can add Nadrf\_MLModelManagement\_RetrievalRequest service operation in "allowedOperationsPerNfType" and/or "allowedOperationsPerNfInstance" for specific NF type and/or specific instance ID of the Consumer, and register its NF profile to NRF. When a NF Service Consumer requests an access token for the Nadrf\_MLModelManagement\_RetrievalRequest to retrieve AI/ML models, the NRF will only grant the access token if Nadrf\_MLModelManagement\_RetrievalRequest is present in either "allowedOperationsPerNfType", for the NF type of the NF Service Consumer, or in "allowedOperationsPerNfInstance", for the instance ID of the NF Service Consumer.

### 6.1.3 Evaluation

This solution addresses the AI/ML model retrieving authorization requirement of KI#3.

This solution uses the existing security mechanisms in SBA, "allowedOperationsPerNfType" and/or "allowedOperationsPerNfInstance" attributes, to specify NF Service Consumer that is authorized to access AI/ML models from ADRF and NWDAF.

This solution grants access to AI/ML models in general. It cannot grant access to particular AI/ML models respectively.

As NF consumer(s) (MTLF or AnLF) retrieves the ML model or URL (where the model is stored) from ADRF by using ADRF ML model retrieval service, when URL is retrieved, this solution only addresses access to the URLs, not to the models.

## 6.2 Solution #2: Authorization and Authentication of ML model transfer

### 6.2.1 Introduction

The solution proposed below protects AI/ML models between the entity which produces the ML model or stores the ML model in ADRF and the entity which consumes the model (NFc). In this solution, an authorization token is used by ADRF to verify that the NFc is allowed to access the ML model.

### 6.2.2 Solution Details

图示

描述已自动生成

Figure 6.2.2.1-1 Secure ML model transfer

1) The MTLF trains the ML model and sends ML Model to the ADRF by invoking the Nadrf\_DataManagement\_StorageRequest (ML Model) service operation. Along with Model. Metadata of the model is also sent for each model, e.g., ML model ID, analytics ID, Vendor ID, MAC or SHA256 Signature of the Binary of the application, environment required for ML model execution, URL/link to retrieve configuration, and secrets, and/or a signing key, certificate to generate authentication credentials. The NWDAF containing MTLF generates a security context for protecting the ML model information using a logical function or named network function NKGC. MTLF may send an ML model encrypted using a symmetric key (e.g., AES key) before the storage. The security context consists of an encryption key Kenc, an integrity key Kint, and the corresponding security algorithm(s) for encryption and integrity protection. The NWDAF containing MTLF uses the encryption key Kenc and integrity key Kint. to protect the ML model and related information. The NKGC stores the security context.

2. ADRF stores the ML model and response as per TS 23.288 [5], except that the ADRF stores the ML model. Data may be stored in the ADRF as per clause 5B.1 in TS 23.288 [5].

Note 1: Key handling such as key refresh and revocation is outside of the scope of this solution and not addressed in this solution.

3. Consumer, e.g., NWDAF/ANLF, contacts the NRF and requests an access token using existing procedures in 33.501 [2]

4. NRF sends an access token along with MTLF ID using existing procedures per TS 23.288 [5]. According to TS 29.510 [8], An NWDAF can add Nnwdaf\_MLModelProvision\_Subscribe service operation in "allowedOperationsPerNfType" and/or "allowedOperationsPerNfInstance" for specific NF type and/or specific instance ID of the Consumer, and register its NF profile to NRF. When an NF Service Consumer requests an access token for the Nnwdaf\_MLModelProvision\_Subscribe to retrieve AI/ML models, the NRF grants the access token if Nnwdaf\_MLModelProvision\_Subscribe is present in either "allowedOperationsPerNfType", for the NF type of the NF Service Consumer, or in "allowedOperationsPerNfInstance", for the instance ID of the NF Service Consumer. MTLF knows the NF instance IDs of the AnLF as per existing procedures in 23.288[5] (e.g. through OAM).

5. The consumer uses Nnwdaf\_MLModelProvision service operation for ANLF receives ML model ID based on analytics ID and ADRF id to retrieve ML model.

6. MTLF verifies the access token received in step 3. MTLF may send the encryption key used in step 1 to encrypt the ML model, which is stored in ADRF. MTLF also sends one-time credentials to access the ML model from ADRF. One-time credentials may include

a. Nonce, which is shared in step 1 as part of the metadata OR

b. MAC or Hash of a binary or random number shared in step 1 as part of the data OR

c. A signing key as a private key of the. The public part is passed in step 1 OR

d. MTLF uses it is signing key to generate the credentials, e.g., a JWT token or a certificate.

NOTE 2: One-time credentials can be used to limit the number of accesses from the consumer NF. The one-time credential may be used as a regular authorization token for accessing the ML model multiple times, i.e., not only once, as the name suggests otherwise.

7. Consumer of the ML model, e.g., ANLF, uses the ADRF service procedure to request the ML model. It also sends a one-time credential received in step 6.

8. ADRF verifies the one-time credentials (as specified in step 6). If the access token verification is successful, the ADRF provides the stored model to the consumer NF.

### 6.2.3 Evaluation

As per key issue 3, "the ADRF itself cannot be considered as a fully trusted entity storing the sensitive AI/ML data models ML models may be stored in the public cloud for storage". The solution proposes to address this issue that ADRF can not be a fully trusted entity, The solution presents the E2E encryption approach where the consumer and ML model owner can access ML models in clear text. 3rd party public cloud will not have access to the ML model, whose sole purpose is to store the model/files. E2E encryption prevents leakage of IP, i.e., cloning, copying, or internal ML models details by heuristics. in single vendor and multi-vendor environments; furthermore, when ML models are not encrypted, they are cloned ML models may lead to revenue loss or advantage to a competitor. E2E encryption provides security guarantees not provided by just transit security or access level controls to the database.

This solution assumes protection of ML model is vendor specific. ML model storage is done using existing procedures in TS 23.288 [5] e.g.-using Nadrf\_MLModel\_StorageRequest containing the ML model.

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

## 6.3 Solution #3: AI/ML model authorization and retrieval

### 6.3.1 Introduction

This addresses KI#3 on security for AI/ML model storage and sharing, specifically the authorization aspects for the access to the AI/ML models, i.e.:

- ADRF (Analytical Data Repository Function), or any other network function which may store the AI/ML model, authorizes the NFc to retrieve that AI/ML model

- NF Service consumers is authorized to access to the AI/ML models in the ADRF (or any other NF which may store the ML model, for instance NWDAF MtLF).

### 6.3.2 Solution details

The solution defines an authorization schema for retrieval of AI/ML models in 5GS based on OAuth 2.0.

The procedure is as follows:



Figure 6.3.2-1: AI/ML model authorization procedure

1) NF producer which trains the model (e.g., NWDAF MtLF), while storing the ML model in the ADRF, it also appends its metadata. The metadata with respect to each model may contain attributes for identification such as Model Info/ analytics Id(s) supported by the model, and attributes such as NF Type, Instance Id and Interoperability indicator for authorization at the retrieval of a particular identified model. Those attributes are known by the NF producer via the OAM system.

NOTE 1:The referred metadata attributes in the solution are intended to be used in the solution for the identification of the model (Model Info/Analytics Id(s) and NF consumer attributes used for authorization (NF Type, Instance Id and Interoperability indicator).

2) NF producer also registers the model specific information including the model metadata when registering its profile in the NRF. This information includes the Model Info/Analytics ID, allowed NF Type/Id and allowed NF Id per Model.

3) NFc when requesting the access token includes at least the Model Id and/or Analytics Id for which a trained model is needed.

4) NRF when receiving the access token request, verifies that the NFc is authorized to retrieve the model from ADRF.

5) In case of valid authorization, the NRF provides the token with access token claims including the Model Id, and optionally also the Analytics Id, identifying the type of analytics that the model may be providing.

6) NFc now provides the access token with the model retrieval service request to the ADRF.

7) ADRF verifies the access token the ensures that the NFc is indeed authorized for the requested model by verifying the access token claims and then also updates the metadata info of the Model (received in Step 1) to also include the authorized NFc info (NFc Instance ID, NFc type).

8) In case of successful verification ADRF provides the NFc a URI to download the model as a service response.

9) NFc initiates the download of the ML model using the URI.

NOTE2: To ensure the URI is not misused by a rogue or malicious NFc, further verification on the model retrieval should be applied from the ADRF. This verification can be preferably realized via CCA (Client Credentials Assertion) as defined in clause 13.3.8 of [7], or alternatively via other mechanisms subject to the particular implementation. Consequently, if the NFc presents CCA as credentials, the ADRF should support the verification of CCA by caching the information provided in step 7. In case other implementations are in use (e.g., central LDAP), ADRF may need to support the corresponding interfaces to external access control platform.

10) ADRF verifies that the NFc, which is trying to download the model, is indeed authorized by verifying the credentials against the metadata updated in Step 7).

ADRF after a successful verification finally initiates the model download at the NFc.

NOTE: Alignment with TS 23.288[5] is not addressed in the present document.

### 6.3.3 Evaluation

The solution addresses the authorization related aspects when the AI/ML models are shared among NWDAFs and/or NFs.

The following security requirements are addressed by this solution: -

- ADRF (Analytical Data Repository Function), or any other network function which may store the AI/ML model, is able to authorize the NFc to retrieve that AI/ML model

- NF Service consumers are authorized to access to the AI/ML models in the ADRF (or any other NF which may store the ML model, for instance NWDAF MtLF).

The solution introduces the following new procedures in the existing authorization schema based on OAuth 2.0:

- The NF producer of the AI/ML model, while storing the model in the ADRF, is required to append its metadata needed to identify the model, and NF consumer attributes (NF type, NF Instance id and Interoperability indicator) for authorization purposes

- At the registration stage in NRF, the NF producer of the AI/ML model is required to include the Model Info/Analytics ID, allowed NF Type/Id and allowed NF Id per Model.

- The ADRF is required to verify the access token of the AI/ML model consumer and responds (if successful) with an URI to download the model. Furthermore, when the AI/ML model consumer initiates the download of the model, it needs to be authenticated (e.g., CCA) by the ADRF.

- The ADRF additionally verifies if the request to retrieve the model is coming from the same NWDAF AnLF which has been authorized by the NRF. This is just an additional security procedure and can be removed (along with the addition of metadata as defined in step 1a) if it leads to increase in complexity of the solution.

The solution is not fully aligned with procedures defined in TR 23.700-81 [6].

## 6.4 Solution #4: AI/ML model storage and sharing security

### 6.4.1 Introduction

This solution is addressing Key Issue #3: Security for AI/ML model storage and sharing and it is based on the Solution #42: Storage and retrieval of trained ML models to/from ADRF of TR 23.700-81 [6] with the following enhancements:

The data producer (NWDAF containing MTLF) is generating a security context to protect the ML model information which is then stored protected in the ADRF with the data producer identity so that NF consumers (NWDAF containing AnLF), if authorized, can request the protected ML model information from the ADRF as well as the security context from the data producer to unprotect the ML model information for further processing. The protection of the ML model is necessary since the ML model are trained and created with a lot of effort and may be different in terms of performance depending on the vendor. Those vendor specific differences can be seen as a confidential asset that should not be disclosed even to repositories from a different vendor and should be stored only protected, also to avoid the risk that an unprotected model gets leaked from the repository. In case the ADRF is co-located with the MTLF, then no encryption is necessary. The necessity of end-to-end protection is described in the Key Issue #3 "AI/ML models are protected between the entity which produces the ML model or stores the ML model in ADRF (e.g., NWDAF containing MtLF, NFp) and the entity which consumes the model (NFc)".

### 6.4.2 Solution details



Figure 6.4.2-1: ML security context generation in MTLF

1. NWDAF containing AnLF sends Nadrf\_MLModelManagement\_RetrievalRequest which includes Analytics ID(s), ML Model Filter Info (ML model file specific information), optionally Target NF (NWDAF containing MTLF) to subscribe for notifications. The ML model file specific information includes the ML model file serialization format requested by the NWDAF containing AnLF.

2. The ADRF determines if the ML model file for the Analytics ID(s) requested is already stored. If the ML model file for the Analytics ID(s) requested in not stored in ADRF then step 3, 4, 5, 6 are performed, before these steps, the ADRF discovers the target MTLF from the NRF optionally if it is not informed by the AnLF in the step 1. If the ML model file for the Analytics ID(s) requested in stored in ADRF the steps 3, 4, 5, 6 are skipped.

3a. ADRF sends Nnwdaf\_MLModelProvision\_Request with the input parameters defined in TS 23.288 [5] and additional input parameters ML model file specific information (ML model file serialization format).

3b. The NWDAF containing MTLF generates a security context for protecting the ML model information. The security context is per ML model and gets removed once the ML model information is removed from the ADRF. The NWDAF containing MTLF generates a validity time for the security context. The security context consists of an encryption key Kenc and an integrity key Kint as well as the corresponding security algorithm(s) for encryption and integrity protection. The NWDAF containing MTLF uses the encryption key Kenc and integrity key Kint. to protect the ML model and related information. The MTLF stores the security context and the related ML information for identification of the security context.

4. The NWDAF containing MTLF sends a Nnwdaf\_MLModelProvision\_Response with following parameters Analytics ID(s), Protected Trained ML model file(s), NWDAF containing MTLF Identity, validity time for the security context. The validity time indicates to the ARDF when to remove the protected ML model information..

5. ADRF sends Nnwdaf\_MLModelTrainingUpdate\_Subscribe with the input parameters Analytics ID(s), ML model file specific information (ML model file serialization format).

6. When the ML model for which the ADRF has subscribed for ML model training update has been updated, the NWDAF containing MTLF sends Nnwdaf\_MLModelTrainingUpdate\_Notify with the following parameters Analytics ID, Protected Trained ML model(s) file, Notification Correlation ID, NWDAF containing MTLF Identity.

7a. The ADRF sends a response back to NWDAF containing AnLF using Nadrf\_MLModelManagement\_Retrieval Response with the following parameters Protected ML Model File Information (Trained ML model(s) file, ML model file serialization format, Trained ML Model ID per Analytics ID, NWDAF containing MTLF address, validity time for the security context).

7b. NWDAF containing AnLF sends Nnwdaf\_KeyProvision\_Request to the NWDAF containing MTLF with the input parameters Analytics ID(s), Notification Correlation ID. The NWDAF containing AnLF is authorized by the NRF to contact the NWDAF containing MTLF and to retrieve the security context.

NOTE: It is assumed that NWDAF containing AnLF authorization is already performed.

7c. The NWDAF containing MTLF selects the ML model security context based on the related ML information for identification.

7d The NWDAF containing MTLF sends a Nnwdaf\_KeyProvision\_Response to the NWDAF containing AnLF, including the ML model security context. It is assumed that the message is protected with SBA security or NDS/IP.

7e The NWDAF containing AnLF unprotects the ML model data with the received security context.

8. The NWDAF containing AnLF subscribes to ADRF using Nadrf\_MLModelManagement\_RetrievalTrainingUpdate\_Subscribe service operation containing input parameters Trained ML Model ID per Analytics ID.

9. The ADRF sends a notification to NWDAF containing AnLF using Nadrf\_MLModelManagement\_RetrievalTrainingUpdate\_Notify service operation containing following parameters ML Model File Information (Protected Trained ML model(s) file, ML model file serialization format, Trained ML Model ID per Analytics ID, NWDAF containing MTLF Identity).

9b. The validity time for the security context is expired, the ARDF removes the ML model information and the NWDAF containing MTLF removes the security context respectively.

10. NWDAF containing AnLF determines that the ML model training update is no longer required.

11. The NWDAF containing AnLF sends Nadrf\_MLModelManagement\_RetrievalTrainingUpdate\_Unsubscribe with Subscription Correlation ID as input parameters.

12. ADRF determines if any of the NF consumer(s) have subscription for ML Model training update per Analytics ID. If none of the NF consumer(s) have subscription for ML model training update per Analytics ID, the ADRF removes the Protected ML model file and ML model file specific information and proceed to step 9.

13. ADRF sends Nnwdaf\_MLModelTrainingUpdate\_Unsubscribe to ADRF with the Subscription Correlation ID as input parameter. NWDAF containing MTLF removes the security context for the ML model.

NOTE: Alignment with TS 23.288[5] on how to store the ML model in the ADRF not addressed in the present document.

The key management of the security context for the ML model is performed by the NWDAF containing MTLF. The security context consists of an encryption key Kenc and an integrity key Kint as well as the corresponding security algorithm(s) for encryption and integrity protection. The NWDAF containing MTLF assigns a validity time for the security context at the time of its generation, which indicates the time until the security keys become invalid. The validity time is similar to the storage duration in the ADRF, i.e. once the validity time for the security context is expired, the ARDF removes the ML model information and the NWDAF containing MTLF removes the security context respectively. The NWDAF containing MTLF generates for new ML model requests a new security context with respective validity time.

### 6.4.3 Evaluation

This solution is addressing Key Issue #3: Security for AI/ML model storage and sharing by protecting the ML model from the NF producer to the NF consumer, also during the storage time in the ADRF. The data producer (NWDAF containing MTLF) is generating a security context to protect the ML model information which is then stored protected in the ADRF with the data producer identity so that NF consumers (NWDAF containing AnLF), if authorized, can request the protected ML model information from the ADRF as well as the security context from the data producer to unprotect the ML model information for further processing.

The keys sent in steps 7b-d are protected with SBA security, but may not be e2e protected between AnLF and MTLF in case SCP is used in between.

The security context requires a key management with a validity time to indicate when the security keys and thus the storage in the ADRF expires.

The solution needs to be aligned with SA2 that NFc (AnLF) does not request the model directly from the ADRF, but the MTLF decides to store the ML model in the ADRF.

## 6.5 Void

## 6.6 Solution #6: Anomalous NF behaviour detection by NWDAF

### 6.6.1 Introduction

This solution addresses key issue #4.

The 5GC supports distributed NF deployments in order to ensure that the NF can provide services from multiple locations. The infrastructure typically implemented on cloud platforms may be owned and operated by the PLMN operator himself, or they may be run on commercial public cloud platforms. It is always possible that the NFs behave in an anomalous manner in certain sites of such distributed infrastructure. Such anomalous behaviour can arise either due to configuration errors or internal data corruption, or due to an attack.

To allow an operator to manage its NF deployment efficiently, additional relevant log data from the NFs can be provided to the NWDAF, which can then be used by NWDAF in assisting the analysis and detection of the cause for an anomaly.

### 6.6.2 Solution details

#### 6.6.2.1 General

The OAM or some operator defined AF may subscribe/request to the NWDAF to notify/respond when it detects an abnormal behaviour for a particular NF instance (for example excessive resource usage, or consistent failure to provide its service). There can be various causes of abnormal behaviour such as an erroneous operation at a NF, an error at the side of the cloud provider where the NF is deployed, or an impending attack at a particular NF by a malicious entity in the system.

In order to analyse the root cause of the anomaly with certain degree of confidence, more information or related data can help. Thus, NWDAF requests inputs from different entities in the system (such as OAM, NRF and NFs) for detailed analysis. Security related data for analytics can also be collected by NWDAF from the NFs (or via OAM). The final output analytics is then sent to the OAM or the AF for adequate measures.

In order to facilitate the security related log data collection from NFs two different methods are proposed.

NOTE: Defining the protocols used for log collection is not in the scope of the present document.

#### 6.6.2.2 Collection of security related log data of NFs via NFs EventExposure APIs

The procedure depicted in Figure 6.6.2.2-1 allows a consumer to request analytics from NWDAF for anomalous NF behaviour and its root cause.



Figure 6.6.2.2-1: NF anomalous behaviour analytics provided by NWDAF using security logs obtained from NFs EventExposure APIs

1. The OAM or operator's AF sends a request/subscription to the NWDAF for NF anomalous behaviour analytics using either the *Nnwdaf\_AnalyticsInfo\_Request* or *Nnwdaf\_AnalyticsSubscription\_Subscribe* service operation.

2. If the request is authorized, and in order to provide the requested analytics, the NWDAF may subscribe to OAM services to retrieve resource usage and NF resources configuration of all targeted NF instances, following the procedure specified in clause 6.2.3.2 in TS 23.288 [1].

3a. The NWDAF subscribes to NRF to receive notification on changes, e.g., on the load and status of NF instances registered in NRF, using *Nnrf\_NFManagement\_NFStatusSubscribe* service operation for all targeted NF instances. NF instances are identified by their NF id.

3b. NRF notifies NWDAF of changes on the load and status of the requested NF instances by using *Nnrf\_NFManagement\_NFStatusNotify* service operation.

4a. The NWDAF subscribes or requests the additional security specific log info (as specified in the table 6.6.2.3-1) for a particular NF by invoking the *Nnf\_EventExposure\_Subscribe* service operation.

4b. The NF then notifies the NWDAF (e.g. with the complete log report) by invoking *Nnf\_EventExposure\_Notify* service operation.

5. The NWDAF derives the relevant analytics using the inputs provided by the OAM, NRF, and the NF (as specified in the table 6.6.2.3-2).

6. The NWDAF provides requested NF anomalous information along with the corresponding root cause (for instance a malicious NF trying to attack other NF for denial of service, or an erroneous NF unable to provide service to other NFs) using either the *Nnwdaf\_AnalyticsInfo\_Request* response or *Nnwdaf\_AnalyticsSubscription\_Subscribe* response, depending on the service used in step 1.

#### 6.6.2.3 Collection of security related log data of NFs via OAM

The procedure depicted in Figure 6.6.2.3-1 allows a consumer to request analytics from NWDAF for anomalous NF behaviour and its root cause using the security specific NF logs obtained from OAM.



Figure 6.6.2.3-1: NF anomalous behaviour analytics provided by NWDAF using security logs obtained from OAM

1. The OAM or operator's AF sends a request/subscription to the NWDAF for NF anomalous behaviour analytics using either the *Nnwdaf\_AnalyticsInfo\_Request* or *Nnwdaf\_AnalyticsSubscription\_Subscribe* service operation.

2. If the request is authorized, and in order to provide the requested analytics, the NWDAF may subscribe to OAM services to retrieve resource usage and NF resources configuration of all targeted NF instances, following the procedure specified in clause 6.2.3.2 in TS 23.288 [1]. The NWDAF further sends the request to the OAM to provide the security specific log info (as specified in the table 6.6.2.3-1).

3. OAM then collects all the requested information along with the security logs from the NFs and provide it to NWDAF.

4a. The NWDAF subscribes to NRF to receive notification on changes, e.g., on the load and status of NF instances registered in NRF, using *Nnrf\_NFManagement\_NFStatusSubscribe* service operation for all targeted NF instances. NF instances are identified by their NF id.

4b. NRF notifies NWDAF of changes on the load and status of the requested NF instances by using *Nnrf\_NFManagement\_NFStatusNotify* service operation.

5. The NWDAF derives the relevant analytics using the inputs provided by the OAM and NRF (as specified in the table 6.6.2.3-2)

6. The NWDAF provides requested NF anomalous information along with the corresponding root cause (for instance a malicious NF trying to attack other NF for denial of service, or an erroneous NF unable to provide service to other NFs) using either the *Nnwdaf\_AnalyticsInfo\_Request* response or *Nnwdaf\_AnalyticsSubscription\_Subscribe* response, depending on the service used in step 1.

Table 6.6.2.3-1 Security specific parameters provided by NFs (NF/NRF) to assist in detection

|  |  |
| --- | --- |
| Information | Description |
| Timestamp | A time stamp associated with the service request which was sent by the NF Service Consumer to a NF Service Producer. |
| NF Service Consumer / SCP Identifier | The consumer instance or the SCP which sends the service request along with the access token to the NF Service Producer. |
| NF Service Producer Identifier | The producer instance which receives the requests and which verifies the access token received along with the requests |
| Authorization status of NF Service Consumer | Indicated if a given NF Service Consumer is authorized to receive an access token or not, as provided by NRF. |
| Access Token Authenticity | Information such as, if access token provided is for the service request it is sent, and if it is generated for the NF Service Consumer which is requesting the service. |
| Access Token Validity | Verification result, i.e. whether the access token is valid or invalid. |
| Number of requests to access a service | Number of simultaneous requests received at the NF Service Producer for a particular time window. |
| Requested Service Name | Name of the service for which the requests had been received. |
| Service Response Confirmation | Confirmation whether the NF Service Producer was able to fulfil the service requests or not. |

Table 6.6.2.3-2 Inputs provided to NWDAF in assisting the detection of anomalous NF

|  |  |  |
| --- | --- | --- |
| Information | Source | Description |
| Security Log Data | OAM/NF | Additional security relevant log info as described in table 6.6.2.3-1. |
| NF Load | NRF | The load of specific NF instance(s) recorded in their NF profile as defined per TS 29.510 [8]. |
| NF status | NRF | The status of specific NF instance(s) recorded in their NF profile as defined per TS 29.510 [8]. |
| NF resource usage | OAM | The usage of assigned virtual resources for specific NF instance(s) (e.g., mean usage of virtual CPU, memory, disk) as defined in TS 28.552 [9] clause 5.7. |

Table 6.6.2.3-3 and Table 6.6.2.3-4 specifies the output analytics from NWDAF

Table 6.6.2.3-3: Anomalous NF behaviour statistics

|  |  |
| --- | --- |
| Information | Description |
| Exceptions (1..max) | List of observed exceptions |
| > Exception ID | The risk detected by NWDAF |
| > Exception category | Indication if the anomalous behaviour is an attack or genuine error |
| > Exception level | Scalar value indicating the severity of the abnormal behaviour |
| > List of target NF(s) | One or more NFs which are affected due to the anomalous NF in the system |
| > List of anomalous NF(s) | One or more NFs which are the probable cause of the anomalous activity in the system (either because they are malicious or due to internal errors) |

Table 6.6.2.3-4: Anomalous NF behaviour predictions

|  |  |
| --- | --- |
| Information | Description |
| Exceptions (1..max) | List of predicted exceptions |
| > Exception ID | The risk detected by NWDAF |
| > Exception category | Indication if the anomalous behaviour is an attack or genuine error |
| > Exception level | Scalar value indicating the severity of the abnormal behaviour |
| > List of target NF(s) | One or more NFs which are affected due to the anomalous NF in the system |
| > List of anomalous NF(s) | One or more NFs which are the probable cause of the anomalous activity in the system (either because they are malicious or due to internal errors) |
| > Confidence | Confidence of this prediction |

The inputs defined in Tables 6.6.2.3-1 and 6.6.2.3-2 can assist in the detection of anomalies, but how the anomaly is finally detected, e.g., used algorithms and associated procedures, is left to the implementation. Example of use: the authenticity of an access token sent by a NF Service Consumer, and/or the number of services of service requests being sent by the NF Service Consumer, can be instrumental in informing the system about whether a potentially erroneous or malicious NF Service Consumer is trying to access a resource for which it is not authorized, and/or trying to attack a NF Service Producer by multiple simultaneous requests.

Based on the input, an analytics function can monitor and find abnormalities in NF load or NF resource usage, which can result in an alert. Specific security log data can help to understand if this is normal behaviour or could be resulting from a NF that is behaving anomalous.

Thus, when NFs send service requests to other NFs, input data such as the number of service requests sent, the percentage of successful service requests, the percentage of successful access token verification and the serving NF load and resource usage can be used to provide training data for normal service requests reception and load.

Input data such as NF Service Consumer / SCP identifier, the NF Service Producer Identifier and the Requested Service Name can be used to identity the anomalous/erroneous NF in the system.

NOTE: The derivation of output from input depends on the algorithms used or the policy present. How to derive the output from input is up to implementation logic, which is out of scope of 3GPP.

### 6.6.3 Evaluation

The solution addresses the key issue #4 intended to study the detection of anomalous NFs by the network, by enabling the NWDAF in assisting the analytics and detection of the cause for an anomaly in the 5G Core.

The solution proposes NWDAF to collect NF related data such as resource utilization, load information and additional security specific log data from different entities in the 5G system, such as OAM, NRF and NFs, to proceed with the detailed analysis.

The solution introduces two different methods of NF log data collection - by using the NFs EventExposure APIs, and by using OAM. However, the existing event exposure is used for UE related events, not for logs. Using the OAM method requires to first expose security logs of NFs to the OAM and then make them available for the NWDAF, what represents a security risk to be evaluated.

The solution specifies parameters provided by the NFs (NF/NRF) and inputs from OAM to NWDAF to assist in the detection, as well as output analytics from NWDAF in terms of statistics and predictions.

How the anomalies are finally detected and confirmed, i.e., used algorithms and associated procedures, are left to implementation and operator decision and out of scope of the present document. I.e., the performance and accuracy of the anomaly detection relies on the implementation. The NWDAF based solution assists in the detection, but cannot be considered the only source/mechanism in the final decision about the anomaly. The solution does not provide evidence for abnormal NF detection precision, false positive rate, false negative rate. Without that, the administrator may take wrong action for the wrong detection so that the network may face unsecure configuration risk. This solution only focuses on defining the first set of input parameters that can be used to facilitate the detection of abnormal NF behaviour.

## 6.7 Solution #7: Secured and authorized AI/ML model transfer and retrieval

### 6.7.1 Introduction

This addresses KI#3 on security for AI/ML model storage and sharing, specifically the following security requirements:

- AI/ML models are protected between the entity which produces the ML model or stores the ML model in ADRF (e.g., NWDAF containing MtLF, NFp) and the entity which consumes the model (NFc).

- ADRF (Analytical Data Repository Function), or any other network function which may store the AI/ML model, are able to authorize the NFc to retrieve that AI/ML model

- NF Service consumers are authorized to access to the AI/ML models in the ADRF (or any other NF which may store the ML model, for instance NWDAF MtLF).

### 6.7.2 Solution details

The solution defines a secured AI/ML model transfer and retrieval mechanism including confidentiality and integrity protection, along with the authorization schema for retrieval of AI/ML models in 5GS.

The procedure is as follows:



Figure 6.7.2-1: Secured AI/ML model transfer and authorization procedure

1a. NF Service producer which trains the model (e.g., NWDAF MtLF), while storing the encrypted ML model in the ADRF, it also appends its metadata. The metadata with respect to each model may contain attributes for identification such as Model Info/ analytics Id(s) supported by the model, and attributes such as NF Type, NF Instance Id and Interoperability indicator for authorization at the retrieval of a particular identified model.

NOTE 1: The referred metadata attributes in the solution are intended to be used in the solution for the identification of the model (Model Info/Analytics Id(s) and NF consumer attributes used for authorization (NF Type, Instance Id and Interoperability indicator).

1b. ADRF, in case of successful model storage, provides a URI back to NWDAF MtLF, which may be later used to access the model.

2. NF Service producer registers the model specific information including the model metadata when registering its profile in the NRF. This information includes the Model Info/Analytics ID; and allowed NF Type, NF Instance Id and Interoperability indicator of consumers with respect to a particular model.

3. NF Service Consumer (e.g., NWDAF AnLF) when requesting the access token to NRF includes at least the Model Id and/or Analytics Id for which a trained model is needed, along with its Interoperability indicator in addition to the NF type and NF instance Id.

4. NRF when receiving the access token request, verifies that the NF Service Consumer is authorized to consume the model of the NFp.

NOTE 1a: How the NRF knows the vendor ID of the AnLF is not addressed in the present document.5. In case of valid authorization, the NRF provides the token with access token claims including the Model Id, and optionally also the Analytics Id, identifying the type of analytics that the model may provide.

6. NF Service Consumer now provides the access token with the model retrieval service request to the NF Service Producer (e.g. NWDAF MtLF). Specially in indirect communication scenarios CCA may be optionally used for authentication between NF Service Consumer and Network Service Producer.

7. NFp verifies the access token and ensures that the NFc is indeed authorized for the requested model by verifying the access token claims.

8a. [Optional] In the case of successful token verification, NFp initiates an update of the metadata information of the Model in ADRF (sent in Step 1a) to include new authorized NFc info (NFc Instance ID, NFc type) in the ADRF.

8b. [Optional] ADRF sends a confirmation with the details of the NFc which is authorized to consume a particular identified model.

9. NWDAF MtLF (NFp) sends as a service response containing the URI to retrieve the encrypted model, the encryption key ‘K', further encrypted using NFc public key, to NWDAF AnLF (NFc).

10-11. NFc requests an access token from NRF to consume the model retrieval service of ADRF, and receives an access token in response.

12. NFc sends the model retrieval request using the URI (sent by NFp (MtLF) in step 9) to ADRF along with the access token received in Step 11, and additionally its credential information as part of the request (e.g., CCA).

13. ADRF verifies that the NFc, which is trying to download the model, is indeed authorized by verifying the identity against the metadata updated in Step 8a, and verification of credentials of NFc provided in Step 12.

NOTE 2: To ensure the URI is not misused by a rogue or malicious NFc, further verification on the model retrieval should be applied from the ADRF. This verification can be preferably realized via CCA (Client Credentials Assertion) as defined in clause 13.3.8 of [1], or alternatively via other mechanisms subject to the particular implementation.

14. ADRF after a successful verification finally initiates the encrypted model download at the NFc.

15. NFc now decrypts the model using the encryption key ‘K' received in Step 9.

### 6.7.3 Evaluation

The solution addresses the authorization related aspects when the AI/ML models are shared among NWDAFs and/or NFs, and the secure storage of the model in the ADRF.

The solution assumes that there is not an implicit trust among the following logical entities: producer of the AI/ML model (NWDAF MtFL), the storage platform (ADRF) and the AI/ML consumer (AnLF). This assumption implies the need to encrypt the model in the storage platform.

NOTE 3: If AI/ML producer (NWDAF MtLF) and storage platform (ADRF) are part of the same system (e.g., co-located in the same server, belonging in the same security domain), this may imply implicit trust, thus the encryption mechanism can be avoided.

The following security requirements are addressed by this solution:

- AI/ML models shall be protected between the entity which produces the ML model or stores the ML model in ADRF (e.g., NWDAF containing MtLF, NFp) and the entity which consumes the model (NFc).

- ADRF (Analytical Data Repository Function), or any other network function which may store the AI/ML model, shall be able to authorize the NFc to retrieve that AI/ML model

- NF Service consumers shall be authorized to access to the AI/ML models in the ADRF (or any other NF which may store the ML model, for instance NWDAF MtLF).

The solution introduces the following new procedures in the existing authorization schema based on OAuth 2.0:

- The NF producer of the AI/ML model, while storing the model in the ADRF, is required to append its metadata needed to identify the model, and NF consumer attributes (NF type, NF Instance id and Interoperability indicator) for authorization purposes. The AI/ML model is stored in encrypted format.

- At the registration stage in NRF, the NF producer of the AI/ML model is required to include the Model Info/Analytics ID, allowed NF Type/Id, allowed NF Id and Interoperability indicator per Model.

- The NWDAF AnLF also while registering may include vendor information in the form of interoperability indicator, that will be used to authorize the AI/ML model request by the NRF.

- The NWDAF MtLF verifies the access token of the AI/ML model consumer and responds (if successful) with an URI to download the model and the encryption key to decrypt the model. This response is encrypted using the NWDAF AnLF public key.

NOTE 4: The encryption key to decrypt the retrieved model requires to be protected at transit, therefore it is further encrypted using the public key of the NWDAF AnLF. This encryption process will require additional processing at the NWDAF MtLF and AnLF.

NOTE 5: How AnLF retrieves the model from ADRF/MtLF will be aligned with conclusions in TR 23.700-81 [6].

- The ADRF additionally verifies if the request to retrieve the model is coming from the same NWDAF AnLF which has been authorized by the NRF. This is just an additional security procedure and can be removed (along with the addition of metadata as defined in step 1a) if it leads to increase in complexity of the solution.

## 6.8 Solution #8: Protection of data and analytics exchange in roaming case

### 6.8.1 Introduction

This solution is addressing on KI#1.

As per Key Issue #3 in TR 23.700-81 [6], the HPLMN/VPLMN may need to collect data or consume analytics from the VPLMN/HPLMN in roaming scenario. The data or analytics may relate to particular UEs or contain information about all UEs or groups of UEs. Both PLMNs (VPLMN, HPLMN) need the ability to control the amount of data exposed and to abstract or hide network-internal aspects based on operator policy, regulatory constraints and/or roaming agreements.

This contribution proposes a new data exchange protection method by expanding the parameters of token requests, and the verification by NRF, ensuring that only the data permitted by the NRF can be provided to other PLMNs.

### 6.8.2 Solution details

This contribution proposes a new data exchange protection method by expanding the parameters of token requests, and the verification by NRF, ensuring that only the data permitted by the NRF can be provided to other PLMNs.



Figure 6.8.2-1: Protection of data and analytics exchange in roaming case

Step 1: NWDAFc send token request to the vNRF, the message should carry the NF consumer type, target NF type, expected NF Service name and requested data type. The target NF type should be NWDAF.

NOTE: The requested data type that would be verified by hNRF could be the requested data information such as data/analytics specification. The data/analytics specification is specified in TS 23.288[5] clause 5A.2, it is the "Service Operation" specific parameters that identify the collected data (i.e. Analytics ID(s) / Event ID (s), Target of Analytics Reporting or Target of Event Reporting, Analytics Filter or Event Filter, etc.). For example, the hNRF can configure a list to indicate which data or data type can be provided to other PLMNs.

Step 2: vNRF forward the token request and the parameters to hNRF.

Step 3: The home network hNRF verifies the parameters included the requested data type, then determines whether the requested data type can be obtained by the visited operator according to the local policy.

Step 4: If the verification is ok, hNRF returns the token to vNRF, and the data type that can be obtained should be indicated in the token.

Step 5: The vNRF forwards the Token\_Get Response to NWDAFc

Step 6: NWDAFc send Nnwdaf\_DataManagement\_Subscribe message with token and its PLMN ID to NWDAFp of PLMN2.

Step 7: The home NWDAFp verifies the service request, including verifying token, PLMN ID and whether the data type in the token is consistent with the requested data.

Step 8: NWDAFp send Nnf\_EventExposure\_Subscribe message to NF to collect data.

Step 9: NF returns raw data to NWDAFp.

Step 10: Data anonymization or desensitization based on operator's policy.

NOTE: The anonymization or desensitization method of data or analytics is left for implementation.

Step 9: NWDAFp returns the processed data to NWDAFc.

### 6.8.3 Evaluation

The solution fulfils the security requirement of key issue #1.

This solution proposes to extend token-based authorization by adding the requested data type to the access token request and access token. Otherwise, it re-uses existing SBA security. The current SBA mechanism support confidentiality, integrity, and replay protection.

In this solution, NRF is used as the authorization server to determine whether the requested data is allowed to provide. NWDAF is used as central NF or entry point to collect data.

This solution requires that allowed requested data type per NF consumer is configured in the NRF.

## 6.9 Solution #9: Anomalous NF behaviour event related data collection and anomalous NF detection

### 6.9.1 Introduction

The solution addresses KI#4 on Anomalous NF behaviour detection by NWDAF.

The solution describes various anomalous NF behaviour event related data collection and the related anomalous NF behaviour analytics exposure procedure.

### 6.9.2 Solution details

A service consumer (i.e., OAM/AF/NF) can subscribe to the NWDAF and get notified about the anomalous NF behaviour related analytics using the procedure shown in Figure 6.9.2-1.



Figure 6.9.2-1: Anomalous NF behaviour event related network data analytics exposure procedure

The steps shown in figure 6.9.2-1 is described as follows:

1. The service consumer subscribes to the anomalous NF behaviour related analytics information by invoking the Nnwdaf\_AnalyticsSubscription\_Subscribe service operation message, with the analytics ID (set to the anomalous NF behaviour), list of event ID(s) (related to received messages (e.g., malformed messages) violating predefined service operation input or output formats, message requests exceeding configured limits, unintended or unrecognized configuration change/operational change, any errors notification), target of analytics (indicates NF ID(s)/AF ID(s), NF type) and may include other information for the analytics according to TS 23.288 Clause 6.1.3.

If the service consumer is already subscribed to the anomalous NF behaviour related analytics information, the service consumer requests analytics information by invoking Nnwdaf\_AnalyticsInfo\_Request service operation with the respective analytics ID and the target of analytics.

2. When a request for analytics information is received, the NWDAF determines whether triggering new data collection is needed.

3. If the NWDAF determine to perform data collection, it need to subscribe to 5GC NFs or AFs related event exposure services (i.e., via NEF if located externally) to be notified for data on a set of events as described in TS 23.288 Clause 6.2.2.1.

The NWDAF subscribes to set of event IDs (related to the service consumer's anomalous NF behaviour subscription which includes event IDs indicating messages (e.g., malformed messages) violating predefined service operation input or output formats, message requests exceeding configured limits, unintended or unrecognized configuration change/operational change, any error notification) by invoking the Nnf\_EventExposure\_Subscribe service operation.

4. If the NWDAF subscribes to the set of event IDs (as in step 3), the NF manages the inference data which includes all event related information.

5. The NF notifies the NWDAF (e.g., event specific inference data) by invoking the Nnf\_EventExposure\_Notify service operation.

6. The NWDAF performs the requested analytics using the data collected from the NF.

NOTE 1: The Anomalous NF behaviour analytics related implementation logic can be up to Operator's implementation and it is out of 3GPP scope.

NOTE 2: If the data need to be collected from an AF, then in steps 3-6, the target AF will be involved instead of NF.

7. If the service consumer is subscribed to analytics information, the NWDAF notifies the service consumer (i.e., based on the request from the service consumer using Nnwdaf\_AnalyticsSubscription\_Notify or Nnwdaf\_AnalyticsInfo\_Request response service operation), the Analytics Reporting Parameters includes the event specific anomalous NF behaviour analytics information. as shown in the table 6.9.2-1 and 6.9.2-2 respectively.

NOTE 3: If the NF specific resource usage and load information are required additionally to process anomalous NF behaviour related analytics, then steps 2, 3a and 3b (as described in Solution 6) can be additionally performed.

Table 6.9.2-1: Inputs collected by NWDAF from target of analytics (i.e., NFs) for other anomalous NF behaviour detection

|  |  |
| --- | --- |
| Information | Description |
| Target of analytics NF Identifier | The NF identifier indicates the NF from which the data is collected to identify if any anomalous NF behaviour is experienced by the target NF. |
| Predefined service operation violation | Information on messages violating predefined service operation input/output format specified including malformed messages (if any). |
| Message load Information | Information on service requests exceeding configured limit at a time instance or in a period or specific to a slice. |
| Configuration Information | Information related to alerts/triggers associated with unintended configuration or operational change based on operator configuration. |
| NF service consumer identifier(s) | Information on the list of NF service consumers of the target NF. |
| NF service producer identifier(s) | Information on the list of NF service producers from which the target NF consumes services. |

Table 6.9.2-2: Anomalous NF behaviour Statistics

|  |  |
| --- | --- |
| Information | Description |
| Exceptions (1..max) | List of observed exceptions |
| > Exception ID | The risk detected by NWDAF |
| > Exception Level | Scalar value indicating the severity of the abnormal behaviour |
| > Exception trend | Measured trend (up/down/unknown/stable) |
| > Cause | Indicates the cause for the exception and alerts such as configuration issues, type of attack (e.g., cyber attack or any other)/threat, malfunction, overload, or software issues accordingly. |
| > List of anomalous NFs | NF Identification information (i.e., NF IDs) related to NFs that are identified as behaving anomalously |
| > List of impacted NFs | NF Identification information (i.e., NF IDs) for all the impacted NFs associated due to the anomalous NF behaviour. |
| > Amount | Estimated number of NFs affected by the Exception |

Table 6.9.2-3: Anomalous NF behaviour Predictions

|  |  |
| --- | --- |
| Information | Description |
| Exceptions (1..max) | List of predicted exceptions |
| > Exception ID | The risk detected by NWDAF |
| > Exception Level | Scalar value indicating the severity of the abnormal behaviour |
| > Exception trend | Measured trend (up/down/unknown/stable) |
| > Cause | Indicates the cause for the exception and alerts such as configuration issues, type of attack (e.g., cyber attack or any other)/threat, malfunction, overload, or software issues accordingly. |
| > List of anomalous NFs | NF Identification information (i.e., NF IDs) related to NFs that are identified as behaving anomalously |
| > List of impacted NFs | NF Identification information (i.e., NF IDs) for all the impacted NFs associated due to the anomalous NF behaviour. |
| > Amount | Estimated number of NFs affected by the Exception |
| > Confidence | Confidence of this prediction |

NOTE 4: The solution uses the existing method specified in TS 23.288 Clause 6.2.2.1 for data collection from NFs which involves notification of data on a set of events. The events do not indicate threats instead it is related to collection of information related to violations of expected/specified behaviours.

NOTE 5a: This solution proposes to extend event exposure to NF related information, like malformed messages. The solution bases the normal behaviour (e.g., predefined service operation messages related inputs and outputs) as the reference to identify data related to any difference in behaviour i.e., violations, so it is sufficient and feasible. To minimize the impact with usage of NF Event Exposure APIs, the data collection can also be done via OAM.

NOTE 5b: Key Issue #4, covers the scenario where the NFs can behave erroneously or anomalously, but the NWDAF is expected to support monitoring of NFs behaviour to ensure that, NFs behave as defined/specified. In the scope of the key issue #4, the solution assumes that the NWDAF can be trusted (e.g., aspects based on what a NWDAF can be trusted is up to operator implementation).

NOTE 5c: Whether the inputs are sufficient and feasible is not addressed.

NOTE 6: The list of input data and the output analytics (i.e., related to anomalous NF behaviour Statistics and predictions) are shown in Tables 6.9.2-1, 6.9.2-2, and 6.9.2-3. The input-based analytics output generation will depend on the algorithms or intelligence logic used and it is outside scope of 3GPP. The input data provides the information on set of violations attempted by various NFs and the NF lists communicated with the target NF from which the data is collected (e.g., via the OAM). Using the intelligence Algorithm whichever is applied based on operator's implementations can identify if there exist any attack traces, if yes, the source of the attack trace, and the other NFs which have communicated with the impacted NF and so on as listed in Tables 6.9.2-2 and 6.9.2-3etc.,

NOTE 7: The relationship between input and output is not addressed.

### 6.9.3 Evaluation

The solution addresses Key Issue #4 and has the following impacts:

This Solution assumes that NWDAF can be trusted. This solution extends event exposure to NF related information, like malformed messages.

The NWDAF need to support anomalous NF behaviour analytics subscription process, where the NWDAF need to collect data related to new events such as violation of predefined service operation message, message request load information, and security alerts in NFs. Further the NWDAF need to provide output related to anomalous NF behaviour statistics and predictions to detect any anomalous NF(s) and associated risk (if exists) to the subscribed anomalous NF behaviour analytics consumer(s).

To minimize the impact with usage of NF Event Exposure APIs, it is workable to consider Solution 6 as baseline for input data collection as defined in Clause 6.6.2.3. Furthermore, the inputs and output parameters defined in the solution 9 can be merged with the ones proposed in solution 6 to have a comprehensive detection mechanism.

The solution does not provide evidence for abnormal NF detection precision, false positive rate, false negative rate. Without that, the administrator may take wrong action for the wrong detection so that the network may face unsecure configuration risk. As the Intelligence algorithm used in the analytics related to Anomalous NF behaviour is up to operator implementation, it is recommended that a selection of an algorithm can consider their performance metrics e.g., abnormal NF detection precision, false positive rate, false negative rate etc.,

Input about anomalous NF behaviour itself can be sensitive data, which needs to be considered before exposing it to other NFs.

## 6.10 Solution #10: Authorization of AI/ML model sharing between different vendors and usage of one-time URLs

### 6.10.1 Introduction

This solution addresses key issue # 3 (security for AI/ML model storage and sharing) considering that the consumer (NWDAF containing AnLF) and the producer (NWDAF containing MTLF) can be different vendors.

The second requirement of the key issue is about enabling the ML model storage to ensure that the NFc is authorized by the NFp to retrieve the model. According to this solution, the NFp retrieves a one-time URL from the ADRF and then shares the one-time URL with the NFc. To be able to access to the model, the NFc has to know the one-time URL, which implies that the URL has been shared with the NFc by the NFp after authorization of the NFc by the NFp.

### 6.10.2 Solution details

The steps of the procedure depicted in Figure 6.10.2-1 are explained below.



Figure 6.10.2-1: Model sharing between NWDAF containing MtLF and NWDAF containing AnLF from different vendors

Steps:

0a. The NFp registers with the NRF using the interoperability ID. It is assumed that the NRF verifies the vendor ID of the NFp.

0b. The NFc registers with the NRF using the vendor ID. It is assumed that the NRF verifies the vendor ID of the NFc.

Assumption: The NRF can verify the Vendor ID of the NWDAF containing AnLF.

NOTE 1: Alignment with TS 23.288[5] for the overall procedure and usage of vendor ID is not addressed in the present document.

NOTE 2:Alignment with the resolution of Key Issue #11 NRF validation of NFc for access token requests in TR 33.875 [10] on how the NRF verifies the Vendor ID of the NWDAF is not addressed in the present document.

0c. The NFp performs some operations for security of the ML model, such as encryption and integrity protection. These operations are out of scope of the solution.

1. If the NFp wants to store the ML model in the ADRF, the NFp triggers storing the ML model in the ADRF. For downloading the model by the ADRF, the NFp provides the address (URL1) of the model in the NFp. Also, the NFp can send the model correlation ID to the ADRF.

2. ADRF securely fetches the protected model. The fetch method and its security are out of scope.

3. The ADRF sends the location of the model (URL2) to the NFp to be used for the NFp to update or access the model in a later point in time.

4. The NFc executes the discovery procedure with the NRF.

5. The NFc request a token from the NRF indicating the analytics ID.

6. The NRF checks whether the vendor ID of the NFc, stored in the profile of the NFc in the NRF, is one of the vendor ID in the interoperability ID in the NFp profile. If the check is successful, then the NRF issues a token that includes the vendor ID of the NFc.

7. The NFc sends the Model request including the token to the NFp.

8. The NFp validate the token and can perform further authorization using the vendor ID specified in the token.

Steps 9 and 10 are executed if the model is stored in the ADRF.

9. After successful authorization, the NFp request a URL, which can be usable only once, from the ADRF.

10. The ADRF checks if the stored model owner is the NFp. If the check is successful, the ADRF provides a URL (URL3) for the stored model. The URL3 can be a URL that is used only once (one time URL).

NOTE: How to provide one time URL by the ADRF is implementation detail which is out of scope of the solution.

11. The NFp provides the URL3 received from the ADRF to the NFc if the model is stored in the ADRF. If the model is stored in the NFp, then the NFp provides the location of the model in the NFp (URL4).

12. The NFc fetches the protected model from the NFp (step 12a) or ADRF (step 12b). The fetch method and its security are out of scope.

13. The NFc performs some operations considering the interoperability ID on the protected ML model such as decryption and integrity check. These operations and keys are out of scope of the solution.

Note that the one-time URL is protected using the current SBA mechanism, so that it cannot be eavesdropped by unauthorized entities. Using one-time URLs further reduces the risk that URLs are leaked and used by unauthorized entities, since the one-time URL will not be valid after the authorized NFp has used it to retrieve the model.

### 6.10.3 Evaluation

This solution addresses ML model sharing between NFs from different vendors, by allowing the NFp to do further authorization check for model sharing using the vendor id of the NFc. The solution focuses on the ML model downloading via URL case.

## 6.11 Solution #11: Secured and authorized AI/ML model sharing between different vendors

### 6.11.1 Introduction

This addresses KI#3 on security for AI/ML model storage and sharing, specifically the following security requirements.

- AI/ML models shall be protected between the entity which produces the ML model or stores the ML model in ADRF (e.g., NWDAF containing MtLF, NFp) and the entity which consumes the model (NFc).

- ADRF (Analytical Data Repository Function), or any other network function which may store the AI/ML model, shall be able to authorize the NFc to retrieve that AI/ML model

- NF Service consumers shall be authorized to access to the AI/ML models in the ADRF (or any other NF which may store the ML model, for instance NWDAF MtLF).

### 6.11.2 Solution details

The solution defines a secured AI/ML model transfer and retrieval mechanism between different vendors which includes confidentiality and integrity protection, along with the authorization for retrieval of AI/ML models in 5GS.

The procedure is as follows:



Figure 6.11.2-1: Secured and authorized AI/ML model sharing between different vendors

0a. NF Service producer e.g., NWDAF containing MTLF registers its NF profile in the NRF with Analytics ID(s) and Interoperability ID(s). Interoperability ID is a list of Vendor IDs which are allowed to access the ML models per Analytics ID.

0b. NF Service consumer e.g., NWDAF containing AnLF registers its NF profile in the NRF with Analytics ID(s) supported by NWDAF containing AnLF and its Vendor ID;

Assumption: The NRF can verify the Vendor ID of the NWDAF containing AnLF.

NOTE a:Alignment with the resolution of Key Issue #11 NRF validation of NFc for access token requests in TR 33.875 [10] on how the NRF verifies the Vendor ID of the NWDAF is not addressed in the present document.

0c. NWDAF containing MTLF may encrypt the ML model per MTLF vendor's implementation and how the model is encrypted is outside of 3GPP scope.

1. NWDAF containing MTLF triggers the ADRF storage service, with Model address in MTLF(URL1), Analytics ID and Interoperability ID in the message.

2. ADRF sends the response to NWDAF containing MTLF which contains the model address in ADRF(URL2).

NOTE 1: The signalling how the ADRF retrieves the Model from NWDAF containing MTLF is outside of 3GPP scope.

2a. ADRF registers its NF profile in the NRF with Analytics ID, Interoperability ID and NWDAF ID containing MTLF for the stored Model.

3. NF Service consumer e.g., NWDAF containing AnLF performs Nnrf\_NFDiscovery\_Request operation with the requested Analytics ID to select a suitable NF Service Producer e.g., NWDAF containing MTLF.

4a. NF Service consumer e.g., NWDAF containing AnLF requests an access token from the NRF using the Nnrf\_AccessToken\_Get request operation. The token request message contains target NF (NWDAF containing MTLF), NF type and/or NF ID of NWDAF containing AnLF, Vendor ID of NWDAF containing AnLF, Analytics ID.

4b. NRF checks whether the NWDAF containing AnLF is authorized to access the requested service in NWADF containing MTLF and verifies that the NWDAF containing AnLF's Vendor ID is included in the NWADF containing MTLF 's Interoperability ID for the Analytics ID and grants the token (token1), based the information provided in NWADF containing MTLF's NF profile.

5. NWDAF containing AnLF performs Nnwdaf\_MLModelProvision (Analytics ID, Vendor ID and token1) service operation to NWDAF containing MTLF to retrieve ML models for the Analytics ID.

6. NWDAF containing MTLF verifies the access token (token1) and determine the ML model to be shared for the requested Analytics ID. If the determined ML model is stored in ADRF, NWDAF containing MTLF sends Nnwdaf\_MLModelProvision Notification to NWDAF containing AnLF with an indication where the Model is stored (e.g. ADRF) and the address of Model in ADRF(URL2).

7a. NWDAF containing AnLF requests an access token from the NRF to be authorized to retrieve the model stored in ADRF. The token request message contains target NF (ADRF), NF type and/or NF ID of NWDAF containing AnLF, Vendor ID of NWDAF containing AnLF, Analytics ID, the Model producer NF ID i.e. NWDAF ID containing MTLF.

7b. NRF verifies that the NWDAF containing AnLF's Vendor ID is included in the Interoperability ID for the Analytics ID of Model producer NF i.e. NWDAF containing MTLF in the ADRF's NF profile. If verification is successful, NRF grants the token (token2), based the information provided in ADRF's NF profile.

8. NF Service consumer e.g., NWDAF containing AnLF retrieves the target model via the URL2 together with the token2. ADRF verifies the token2 to check if the NF Service consumer e.g., NWDAF containing AnLF is authorized to access the stored model. If authorization is successful, ADRF grants the NWDAF containing AnLF to retrieve the stored Model.

NOTE 2: The signalling how the NWDAF containing AnLF retrieves the Model from ADRF is outside of 3GPP scope.

NOTE 3: Alignment with TS 23.288[5] on the overall procedure and usage of vendor ID is not addressed in the present document.

9. NWDAF containing AnLF decrypts the received model per the vendor's implementation and how the model is decrypted is outside of 3GPP scope.

NOTE 4: If only interoperability indicator is sufficient for ensuring authorization of ML Model is not addressed in the present document.

NOTE 5: How the solution addresses the first security requirement is not addressed in the present document

### 6.11.3 Evaluation

This solution resolves Key Issue #3: Security for AI/ML model storage and sharing.

In this solution it is assumed e2e encryption of the ML model is achieved by the vendor's implementation of the NF producer which generates the AI/ML model. How the model is encrypted in the NF producer or decrypted in the NF consumer is outside of 3GPP scope.

In this solution, when NF producer stores the ML model in ADRF, it provides the URL where the ML model is stored in the NF producer to the ADRF. The NF producer also provides the ADRF with Interoperability ID together with the stored ML Model. And ADRF provides the URL where the ML model is stored in the ADRF back to the NF producer once the ML model is stored in ADRF. ADRF also stores the received Interoperability ID in the NRF which is used by the NRF to perform authorization of token request to retrieve ML model from ADRF.

In this solution, service authorization of ML model retrieval is performed at the NRF using token-based authorization. If NF consumer is requesting token to retrieve ML model from NF producer, NRF checks the vendor ID of the NF consumer is included in Interoperability ID that is stored in NRF by the NF producer. If NF consumer is requesting token to retrieve ML model from ADRF, NRF checks the vendor ID of the NF consumer is included in Interoperability ID that is stored in NRF by ADRF.

When NF consumer requests the ML model from the NF producer, the NF producer checks that if the NF consumer is authorized to retrieve ML model based on the received token and determines the ML model to be shared for the requested Analytics ID. The NF producers provides the URL where the ML model is stored, either in NF producer or ADRF.

When NF consumer retrieves the ML model from the ADRF, the ADRF checks that if the NF consumer is authorized to retrieve ML model based on the received token.

It is assumed the actual signalling how the NF consumer retrieves the Model from ADRF or NF producer, or how the ADRF retrieves the Model from the NF producer is outside of 3GPP scope.

## 6.12 Solution #12: Secured and authorized AI/ML model sharing between different vendors

### 6.12.1 Introduction

This addresses KI#3 on security for AI/ML model storage and sharing, specifically the following security requirements.

- AI/ML models shall be protected between the entity which produces the ML model or stores the ML model in ADRF (e.g., NWDAF containing MtLF, NFp) and the entity which consumes the model (NFc).

- ADRF (Analytical Data Repository Function), or any other network function which may store the AI/ML model, shall be able to authorize the NFc to retrieve that AI/ML model

- NF Service consumers shall be authorized to access to the AI/ML models in the ADRF (or any other NF which may store the ML model, for instance NWDAF MtLF).

### 6.12.2 Solution details

The solution defines a secured AI/ML model transfer and retrieval mechanism between different vendors which includes confidentiality and integrity protection, along with the authorization for retrieval of AI/ML models in 5GS.

The procedure is as follows:



Figure 6.12.2-1: Secured and authorized AI/ML model sharing between different vendors

0a. NF Service producer e.g., NWDAF containing MTLF registers its NF profile in the NRF with Analytics ID(s) and Interoperability ID(s). Interoperability ID is a list of Vendor IDs which are allowed to access the ML models per Analytics ID.

0b. NF Service consumer e.g., NWDAF containing AnLF registers its NF profile in the NRF with Analytics ID(s) supported by NWDAF containing AnLF and its Vendor ID;

Assumption: The NRF can verify the Vendor ID of the NWDAF containing AnLF.

NOTE 1:Alignment with the resolution of Key Issue #11 NRF validation of NFc for access token requests in TR 33.875 [10] on how the NRF verifies the Vendor ID of the NWDAF is not addressed in the present document.

1. NWDAF containing MTLF may encrypt the ML model per MTLF vendor's implementation and how the model is encrypted is outside of 3GPP scope.

2. NF Service consumer e.g., NWDAF containing AnLF performs Nnrf\_NFDiscovery\_Request operation with the requested Analytics ID to select a suitable NF Service Producer e.g., NWDAF containing MTLF.

3. NF Service consumer e.g., NWDAF containing AnLF requests an access token from the NRF using the Nnrf\_AccessToken\_Get request operation with the requested Analytics ID and its own Vendor ID.

4. NRF checks whether the NWDAF containing AnLF is authorized to access the requested service in NWADF containing MTLF and verifies that the NWDAF containing AnLF's Vendor ID is included in the NWADF containing MTLF 's Interoperability ID for the Analytics ID and grants the token (token1), based the information provided in NWADF containing MTLF's NF profile.

5. NWDAF containing AnLF performs Nnwdaf\_MLModelProvision (Analytics ID, Vendor ID, token1,) service operation to NWDAF containing MTLF to retrieve ML models for the Analytics ID.

6. NWDAF containing MTLF verifies the access token (token1) and determine the ML model to be shared for the requested Analytics ID. If the determined ML model is stored in NWDAF containing MTLF, NWDAF containing MTLF sends Nnwdaf\_MLModelProvision Notification to NWDAF containing AnLF with the address of Model file in NWDAF containing MTLF(URL1).

7. NF Service consumer e.g., NWDAF containing AnLF retrieves the target model via the URL1 together with the token1. NWDAF containing MTLF verifies the token1 to check if the NF Service consumer e.g., NWDAF containing AnLF is authorized to access the stored model. If authorization is successful, NWDAF containing MTLF grants the NWDAF containing AnLF to retrieve the stored Model.

NOTE 2: The signalling how the NWDAF containing AnLF retrieves the Model from NWDAF containing MTLF is outside of 3GPP scope.

8. NWDAF containing AnLF decrypts the received model per the vendor's implementation and how the model is decrypted is outside of 3GPP scope.

NOTE 3: Alignment with TS 23.288[5] on the overall procedure and usage of vendor ID is not addressed in the present document.

NOTE 4: How the solution addresses the first requirement is not addressed in the present document

### 6.12.3 Evaluation

This solution resolves Key Issue #3: Security for AI/ML model storage and sharing. It covers the scenario where NF consumer retrieves ML model from the NF producer where the ML model is locally stored in the NF producer.

In this solution it is assumed e2e encryption of the ML model is achieved by the vendor's implementation of the NF producer which generates the AI/ML model. How the model is encrypted in the NF producer or decrypted in the NF consumer is outside of 3GPP scope.

In this solution, service authorization of ML model retrieval is performed at the NRF using token-based authorization. When NF consumer is requesting token to retrieve ML model from NF producer, NRF checks the vendor ID of the NF consumer is included in Interoperability ID that is stored in NRF by the NF producer.

When NF consumer requests the ML model from the NF producer, the NF producer checks that the NF consumer is authorized to retrieve ML model based on the received token and determines the ML model to be shared for the requested Analytics ID.

It is assumed the actual signalling how the NF consumer retrieves the Model from NF producer is outside of 3GPP scope.

## 6.13 Solution #13: Support authorization of AI/ML model sharing By NWDAF containing MTLF

### 6.13.1 Introduction

This solution addresses Key issue #3: Security for AI/ML model storage and sharing.

Since ML models are generally proprietary per provider and subject to intellectual property rights, NF Service consumers are authorized by the producer of the Model, i.e. NWDAF containing MTLF to access to the AI/ML models stored either in NWDAF containing MTLF or ADRF.

This solution proposes that NWDAF containing MTLF to check the Vendor ID of NF consumer with its local authorization policy (e.g., Interoperability Indicator), upon receiving request to retrieve ML model trained by the NWDAF containing MTLF.

### 6.13.2 Solution details

Figure 6.13.2-1 illustrates the high-level procedure of the proposed solution.



Figure 6.13.2-1: High-level procedure for authorization of ML sharing by NWDAF containing MTLF

0a. NWDAF containing AnLF registers its NF profile in the NRF with Analytics ID(s) supported by NWDAF containing AnLF and its Vendor ID.

Assumption: The NRF can verify the Vendor ID of the NWDAF containing AnLF.

NOTE 1:Alignment with the resolution of Key Issue #11 NRF validation of NFc for access token requests in TR 33.875 [10] on how the NRF verifies the Vendor ID of the NWDAF is not addressed in the present document.

0b. NWDAF containing MTLF trains ML model and may encrypt the ML model per its vendor's implementation and how the Model is encrypted is outside of 3GPP scope.

1. NWDAF containing MTLF determines the ML model can be stored in ADRF. NWDAF containing MTLF discovers and selects ADRF via NRF and stores the selected ADRF instance ID for the Model.

2. NWDAF containing MTLF triggers the ADRF storage service to ADRF, with the Model Correlation ID and Model address in MTLF(URL1) in the message.

3. ADRF securely fetches the Model via URL1 from NWDAF containing MTLF. NWDAF containing MTLF verifies that the ADRF is the selected ADRF instance in step 2 and checks that the Model Correlation ID is correct. It is assumed there is no need to verify the vendor ID of ADRF.

NOTE 1: The signalling how the ADRF retrieves the Model from NWDAF containing MTLF is outside of 3GPP scope.

4. ADRF sends the response to NWDAF containing MTLF, with the model address in ADRF(URL2) in the message.

5. NWDAF containing AnLF performs discovery and selection procedure for NWDAF containing MTLF via NRF.

6. NWDAF containing AnLF performs Nnwdaf\_MLModelProvision service operation to NWDAF containing MTLF, with Analytics ID and Vendor ID in the message, to retrieve ML models for the Analytics ID.

7. NWDAF containing MTLF retrieves the NF profile of the NWDAF containing AnLF from NRF and gets NWDAF containing AnLF's Vendor ID from the NF profile. NWDAF containing MTLF verifies that the NWDAF containing AnLF's Vendor ID in the request message is same as the Vendor ID in its NF profile and is included in the Interoperability ID for the Analytics ID.

NOTE 1a: Whether MTLF can also verify the Vendor ID of the AnLF based on the authentication procedure for the AnLF is not addressed in the present document.

If verification is successful, NWDAF containing MTLF determines the ML model to be shared for the requested Analytics ID and stored the NF instance ID of NWDAF containing AnLF as part of allowed NF instance list for the ML model.

8. If the determined ML model is stored in ADRF, NWDAF containing MTLF triggers the ADRF storage Update service to ADRF, with the Model Correlation ID and allowed NF instance list in the message. The ADRF stores the allowed NF instance list for the ML model referenced by the Model Correlation ID.

9. NWDAF containing MTLF sends Nnwdaf\_MLModelProvision Notification to NWDAF containing AnLF with the address of the determined ML model in the message, which can be either the one stored in NWDAF containing MTLF or in ADRF.

10. NWDAF containing AnLF retrieves the target ML model via the URL received from step 9.

If the ML model is stored in NWDAF containing MTLF, NWDAF containing MTLF verifies that the NWDAF containing AnLF is the stored NF consumer instance in step 7. If the ML model is stored in ADRF, ADRF verifies that the NWDAF containing AnLF is part of allowed NF instance list received in step 8. If verification is successful, NWDAF containing AnLF is authorized to retrieve the ML model.

NOTE 2: The signalling how the NWDAF containing AnLF retrieves the Model from NWDAF containing MTLF or ADRF is outside of 3GPP scope.

11. NWDAF containing AnLF decrypts the received ML model per the vendor's implementation and how the ML model is decrypted is outside of 3GPP scope.

NOTE 3: How the solution addresses the first security requirement is not addressed in the present document.

### 6.13.3 Evaluation

This solution resolves Key Issue #3: Security for AI/ML model storage and sharing.

In this solution it is assumed e2e encryption of the ML model is achieved by the vendor's implementation of the NF producer which generates the AI/ML model. How the model is encrypted in the NF producer or decrypted in the NF consumer is outside of 3GPP scope.

In this solution, when NF producer stores the ML model in ADRF, it provides the URL where the ML model is stored in the NF producer to the ADRF. And ADRF provides the URL where the ML model is stored in the ADRF back to the NF producer once the ML model is stored in ADRF.

The NF producer maintains a list of NF consumer instances and/or a list of ADRF instances that are allowed to retrieve the ML model from the NF producer. The NF producer provides and updates ADRF with a list of allowed NF consumer instances for the stored ML model in ADRF.

When NF consumer requests the ML model from the NF producer, the NF producer verifies the vendor ID of the NF consumer based on NF profile of the NF consumer that is stored in NRF and checks the vendor ID is included in Interoperability ID of the NF producer. If the NF consumer is authorized, the NF producer determines the ML model to be shared for the requested Analytics ID and provides the URL where the ML model is stored, either in NF producer or ADRF. The NF producer also adds the NF consumer into the list of allowed NF consumer instances.

When NF consumer retrieves the ML model from the ADRF, the ADRF checks that the NF consumer is included in the list of allowed NF consumer instances for the stored ML model.

It is assumed the actual signalling how the NF consumer retrieves the Model from ADRF or NF producer, or how the ADRF retrieves the Model from the NF producer is outside of 3GPP scope.

## 6.14 Solution #14: Protection of data and analytics exchange in roaming case using Secure Multi-party Computation

### 6.14.1 Introduction

This solution is addressing on KI#1.

In the scenario where the NWDAF consumer collects data from multiple PLMNs, PLMNs need the ability to control the amount of data exposed and to abstract or hide network-internal aspects based on operator policy, regulatory constraints and/or roaming agreements. But if the data processed method used by different PLMNs is different, it is possible that the data converged in NWDAF consumer could not be effective used.

To ensure data provided by multiple data sources can be used and to protect sensitive data not exposed to other operators, this solution proposes a security protection scheme based on secure multi-party computation.

### 6.14.2 Solution details

#### 6.14.2.1 Data Privacy Computing Function DPCF

This contribution introduce a data privacy computing function to proceed data anonymization or desensitization.

When the NWDAF of the visited network initiates a data request across the network, it sets a priority list (1, 2, 3...) according to the number of destination operators, and assigns the priority value to different home operators though encapsulating the value "priority=X (X=1, 2, 3...)" in the service request message. The parameter "priority indicator" includes the priority of the current network, the priority of other networks and the network address. After the home operator NFp receives the service request, the NFp returns the requested data and priority value to the data privacy computing function DPCF. After the DPCF receives the data and the priority value, the DPCF1 of the home network PLMN1 with the higher priority initiates an request to the DPCF2 of the other home network PLMN2 with the lower priority. After the DPCF2 verifies the priority, the DPCF1 and DPCF2 will use secure multi-party computation to proceed data anonymization or desensitization.

#### 6.14.2.2 Data collection protection architecture based on Secure Multi-party Computation

The scenario is NWDAF in visit network collecting data from multiple home operators, DPCF performs a secure multi-party computation process.



Figure 6.14.2.2-1

Step 1: The visiting NWDAFc sends a token request to the hNRF of the home network via vNRF. The message should contain NF consumer type, target NF type, expected NF Service name and requested data type.

Step 2: The home network hNRF verifies the relevant parameters and the requested data type, then determines whether the requested data type can be obtained by the visiting operator according to the local policy.

Step 3: If the verification is ok, hNRF returns the token to NWDAFc, and the data type that can be obtained should be indicated in the token.

Step 4: NWDAFc sends a data collection request to NWDAF1 of PLMN2, carrying parameters including token, PLMN ID, and priority indicator.

Step 5: The home NWDAF verifies the service request, including verifying token, PLMN ID and whether the data type in the token is consistent with the requested data.

Step 6: NWDAF1/NWDAF2 collects data from other NFs

Step 7: NFWDAF1/NWDAF2 returns the raw data and priority indicator to DPCF.

Step 8: DPCF1 initiates secure multi-party computation process with DPCF2 in other home operators network..

Note 1: DPCF2 in PLMN3 needs to receive unprocessed data and priority indicator returned from NWDAF2.

Step 9: DPCF1/DPCF2 returns the processed data to NWDAFC

NOTE 1a: How the precedure align with TS 23.288[5] is not addressed in the present document.

NOTE 2: The participant operators need to agree on secure MPC scheme.

NOTE 3: The necessity of the new interface is not addressed in the present document.

### 6.14.3 Evaluation

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

This solution fulfil the requirement of KI #1 when NWDAF consumer collects data from multiple PLMNs.

This solution requires a new network function and requires the participant operators agree on the secure MPC scheme.

## 6.15 Solution #15: Authorization of selection of participant NWDAF instances in the FL group

### 6.15.1 Introduction

The solution addresses KI#2 on authorization of selection of participant NWDAF instances in the Federated Learning group.

For the client NWDAF's authorization to join a FL group, as per TR 23.700-81's conclusion to KI#8: supporting federated learning in 5GC, server NWDAF selects client NWDAFs from NRF and client NWDAF cannot request to join a FL group. Since client NWDAF joins a FL group only when it is selected by server NWDAF, client NWDAF is authorized to join the FL group by server NWDAF at the same time.

For the server NWDAF's authorization to include a client NWDAF into a FL group, this solution proposes that client NWDAF registers to NRF with the permitted-server-NWDAF list for each supporting Analytics ID. When a server NWDAF discovers client NWDAFs for an Analytics ID from NRF, NRF only returns client NWDAFs whose permitted-server-NWDAF list for this Analytics ID includes the server NWDAF. When server NWDAF requests access token, NRF also checks if the server NWDAF is on the permitted-server-NWDAF list.

NOTE: When new permitted server NWDAFs are introduced, the client NWDAF needs to update its permitted-server-NWDAF list.

### 6.15.2 Solution details

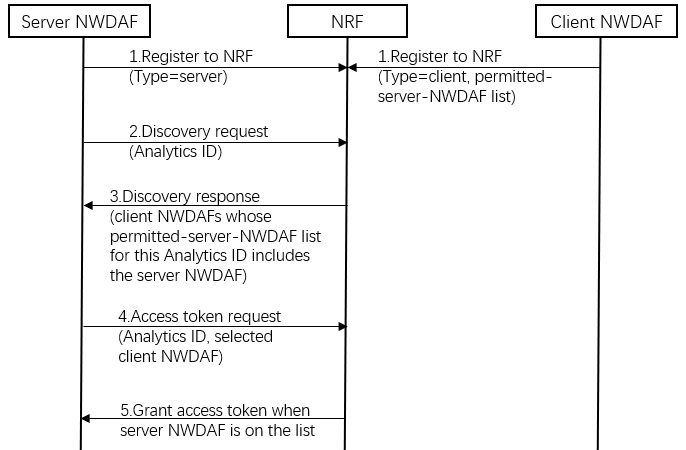


Figure 6.15.2-1

Step 1: NWDAF registers to NRF with its FL related information including Analytics ID(s), FL capability type (i.e. FL server or FL clients), permitted-server-NWDAF list for each Analytics ID when this NWDAF serves as a FL client, etc.

Step 2: Server NWDAF sends discovery request to NRF to find client NWDAFs for an Analytics ID.

Step 3: NRF returns candidate client NWDAFs whose permitted-server-NWDAF list for this Analytics ID includes the server NWDAF.

Step 4: Server NWDAF sends access token requests to NRF for each selected client NWDAFs of the Analytics ID.

Step 5: When the server NWDAF is on the selected client NWDAF's permitted-server-NWDAF list for this Analytics ID, NRF grants access token to the server NWDAF.

### 6.15.3 Evaluation

This solution addresses KI#2: authorization of selection of participant NWDAF instances in the Federated Learning group.

This solution reuses the existing SBA security mechanism and proposes that server NWDAF is authorized based on the permitted-server-NWDAF list which client NWDAF registers to NRF. When new permitted server NWDAFs are introduced, the list needs to be updated.

## 6.16 Solution #16: FL GROUP AUTHORIZATION OF NWDAF(S) IN 5GC

### 6.16.1 Introduction

This solution addresses Key Issue #2 (Authorization of selection of participant NWDAF instances in the Federated Learning group). The proposed solutions use existing SBA architecture for token-based authorization.

### 6.16.2 Solution details

The detailed procedure for NWDAF AnLF/NWDAF MTLF (FL Server) to get token from NRF and receive services from NWDAF MTLF (FL Server)/NWDAF MTLF (FL Client) is depicted in Figure 6.Y.2-1



Figure 6.16.2-1: NWDAF AnLF/NWDAF MTLF (FL Server) Authorization to receive services from NWDAF MTLF (FL Server)/NWDAF MTLF (FL Client)

NOTE 1: Steps 1 to 4 are not specific to FL scenario but details existing mechanisms. Step 5 to 13 are related to FL scenario.

0. NWDAF registers with NRF. If NWDAF MTLF as FL server determines ML model requires FL, the FL Server discovers and selects other NWDAF(s) MTLF as FL Client(s) from NRF. If NWDAF MTLF without FL server capability determines ML model requires FL, the MTLF discovers and selects FL sever from NRF.

1-3. NF(NWDAF AnLF or MTLF) Service Consumer sends a request to the NRF to receive an access token to request NWDAF MTLF (FL Server) services. NRF, after verifying, generates an access token and sends it to the NF(NWDAF ANLF OR MTLF) Service Consumer. Access tokens contain NWDAF MTLF (FL Server specific token).

4. The NF(NWDAF AnLF OR MTLF) Service Consumer initiates an NF service request to the NWDAF MTLF (FL Server), which includes the access\_token\_nwdaf. The NF(NWDAF AnLF OR MTLF) Service Consumer also generates a Client Credentials Assertion (CCA) token (CCA\_NWDAF) and includes it in the request message to authenticate itself towards the NF Service Producers.

5. The NWDAF MTLF (FL Server) verifies if the access\_token\_nwdaf is valid and starts FL group.

6. If The NWDAF MTLF (FL Server) determines to start the FL group for analytics id, The NWDAF MTLF (FL Server) sends a Nnrf\_AccessToken\_Get request to NRF including the information to identify the target NF (NWDAF MTLF (FL Client)), the source NF (NWDAF AnLF OR MTLF) Service Consumer, the NF Instance ID of NWDAF MTLF (FL Server) , Analytics ID, FL local model training service type, FL group ID and the CCA\_NWDAF provided by the NF(NWDAF AnLF OR MTLF) Service Consumer.

7. The NRF checks whether the NWDAF MTLF (FL Server) and the NF(NWDAF ANLF OR MTLF) Service Consumer (e.g. NWDAF) are allowed to access the service provided by the identified NF Service Producers(NWDAF MTLF (FL Client)) for the given Analytics ID included in step 6, and the NWDAF MTLF (FL Server). NRF authenticates both NWDAF MTLF (FL Server) and NWDAF(FL consumer, e.g., AnLF) based on one of the SBA methods described in clause 13.3.1.2 of TS 33.501. NWDAF MTLF (FL Server) may include an additional CCA for authentication.

8. The NRF, after successful verification, then generates and provides an access token to the NWDAF MTLF (FL Server); the claims in the token include the NF Instance Id of NRF (issuer), NF Instance Id of the NF Service Consumer (subject), NF type of the NF Service Producer (audience), expected service name(s), (scope), expiration time (expiration), FL group ID, Analytics ID(s), ML model ID(s) and optionally "additional scope" information (allowed resources and allowed actions (service operations) on the resources), with NF(NWDAF AnLF OR MTLF) Service Consumer Instance (subject), to authorize both NF(NWDAF AnLF OR MTLF) Service Consumer (e.g.. NWDAF) and NWDAF MTLF (FL Server) to consume the services of NWDAF MTLF (FL Client).

NOTE 2: Usage of Model ID and FL group ID and alignment with TS 23.288[5] is not addressed in the present document.

9. NWDAF MTLF (FL Server) finalize the FL group with NWDAF MTLF (FL Client) selected from the list received from NRF.

10. The NWDAF MTLF (FL Server) requests service (local model updates) from the NWDAF MTLF (FL Client). The request also consists of CCA\_NWDAF so that the NF Service Producer(s) authenticates the NF(NWDAF ANLF OR MTLF) Service Consumer (e.g., NWDAF).

11. The NWDAF MTLF(s) (FL client) authenticates the NF(NWDAF AnLF OR MTLF) Service Consumer and verifies the access token, and ensures that the NWDAF MTLF (FL Server) identity, FL group ID, Analytics ID(s), ML model ID(s) is included as an access token additional claim.

12. The NWDAF MTLF(s) (FL client) provides requested data to the NWDAF MTLF (FL Server). Global Model updates/aggregation is done at NWDAF MTLF (FL Server).

13. NWDAF MTLF (FL Server) feedback on the NF Service Response.

### 6.16.3 Evaluation

This solution reuses the existing 5GS security mechanisms, including the current SBA security mechanisms.

NWDAF MTLF (FL Server) can select which Federated Learning task it wants to create by verifying the access token presented by the NWDAF AnLF. NWDAF MTLF (FL Client) can select which Federated Learning group it wants to join by verifying the access token presented by the NWDAF MTLF (FL Server). The solution fulfils the following requirements of KI 1.2:

Authorization of selection of participant NWDAF instances in the Federated Learning group needs to be supported:

A server NWDAF needs to be authorized to include a client NWDAF in a Federated Learning group. A client NWDAF needs to be authorized to join a Federated Learning group.

## 6.17 Solution #17: Authorization of FL Server and FL Client in FL group

### 6.17.1 Introduction

This solution addresses Key issue #2: Authorization of selection of participant NWDAF instances in the Federated Learning group.

As per KI#2, both server NWDAF and client NWDAF need to be authorized for FL procedure.

- A server NWDAF needs to be authorized to include a client NWDAF into a Federated Learning group.

- A client NWDAF needs to be authorized to join a Federated Learning group.

For initiation of FL procedure or reselection of FL client(s) into a FL procedure, the server NWDAF triggers discovery and selection of NWDAF(s) and sends FL request to the client NWDAF.

- At the client NWDAF side, authorization of the server NWDAF that it can include the client NWDAF into the FL group can be achieved by reusing existing SBA token-based authorization upon receiving FL request.

- At the server NWDAF side, authorization of the client NWDAF to be included into a FL group is implicitly achieved as the server NWDAF only triggers the selection of FL client(s) into the FL group that it authorizes to be included.

This solution also considers interoperability of different vendors of Server and Client NWDAF(s).

### 6.17.2 Solution details

Figure 6.17.2-1 illustrates the high-level procedure of the proposed solution.



Figure 6.17.2-1: High-level procedure for authorization of FL Server and FL Client in FL group

0. NWDAF containing MTLF as FL server or FL clients register to NRF with their FL related information, including Analytics ID(s), Interoperability ID, Vendor ID, Address information, FL capability Type (i.e. FL server or FL client), and Service Area etc.

NOTE: The Interoperability indicator indicates a list of NWDAF providers (vendors) that are allowed to retrieve ML models from the NWDAF containing MTLF.

Assumption: The NRF can verify the Vendor ID of the NWDAFs containing MTLF.

NOTE 1:Alignment with the resolution of Key Issue #11 NRF validation of NFc for access token requests in TR 33.875 [10] on how the NRF verifies the Vendor ID of the NWDAF is not addressed in the present document.

1. The Server NWDAF discovers Client NWDAFs from NRF based on FL selection criteria e.g. Federated Learning capability, Interoperability ID, Analytics ID, etc.

Server NWDAF requests tokens for each of selected Client NWDAFs from NRF, with Analytics ID, Vendor ID and FL capability included in the request.

NOTE 2: Inclusion of the vendor ID in the access token request is not addressed in the present document

NRF verifies that the Server NWDAF's Vendor ID is included in the Client NWDAF's Interoperability ID for the Analytics ID and grants the token, based the information provided in Client NWDAF's NF profile.

2. The Server NWDAF sends FL request to the Client NWDAF(s) with the obtained token.

3. Each Client NWDAF checks whether the server NWDAF is authorized based on the token and decides whether to join the FL group.

4. The Client NWDAF(s) sends the response to the Server NWDAF.

5. The FL group is formed.

6. The Server NWDAF registers or updates its registration into NRF about the created FL Group with the following parameters: Federated Learning (FL) Correlation ID. the associated Analytics ID, Interoperability ID, allowed requester NF type or NF instance ID (e.g. NWDAF containing MTLF), allowed FL capability (e.g. FL client) etc.

If the Server NWDAF is about to reselect new Client NWDAF(s) into the FL group during FL execution phase, the Server can either trigger selection procedure again as step 1 to step 4 or receive notification from NRF and select new Client NWDAF(s) available as step 7 to step 9.

7-8. The Server NWDAF gets the information of the new Client NWDAF(s) via notifications from NRF.

9. The Server NWDAF requests tokens for each of selected Client NWDAFs from NRF as step1 and triggers the procedure as step 2-4.

### 6.17.3 Evaluation

This solution resolves Key Issue #2: Authorization of selection of participant NWDAF instances in the Federated Learning group.

This solution proposes to reuse existing SBA token-based authorization for authorization of server NWDAF and client NWDAF for FL procedures.

For initiation of a FL procedure or reselection of FL client(s) into a FL group, the server NWDAF requests token from NRF for each selected client NWDAF, to be authorized to include a client NWDAF into a FL group.

This solution also proposes that FL model sharing is authorized based on the Interoperability ID. When the server NWDAF requests token from NRF to include a client NWDAF into a FL group, NRF verifies that the server NWDAF's vendor ID is included in the client NWDAF's Interoperability ID and grants the token based the information provided in client NWDAF's NF profile.

In this solution, authorization is granted with NWDAF vendor granularity, but not for specific NF instances.

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

## 6.18 Solution #18: Solution for authorization in FL

### 6.18.1 Introduction

Based on existing OAuth 2.0 schema specified for SBA, the solution proposes an authorization mechanism for FL NWDAF server(s) to consume the resources and data of the NWDAFs acting as FL clients, based upon the FL role information and the analytics ID and other data details registered at the NRF during the NF profile registration phase.

### 6.18.2 Solution details

The figure 6.18.2-1 represents the authorization workflow of the solution:



Figure 6.18.2-1: Authorization workflow for FL

Step 1: NWDF FL member registers in the NRF with the following attributes:

a) "FL Process Role", i.e., whether the NWDAF registers as a FL server or as FL client for a particular Analytics Id

b) "Allowed NF(s) (NF type and NF Instance id) as FL server(s) for a particular Analytics Id, if the NWDAF has registered as FL client in NRF

When new allowed FL server NWDAFs are introduced, the FL member NWDAF needs to update its list of allowed server NWDAFs.

NOTE 0: Other data details can be added and are subject to the implementation, such as sensitivity of the data used for training (e.g., it may contain privacy data of subscribers), which are to be decided by the operator.

Step 2: FL server sends a discovery request to the NRF to retrieve the information for the NWDAF registered as FL client for a given Analytics Id.

Step 3: FL server sends an access token request for each required NWDAF FL client (in the picture a single request is shown)

Step 4: NRF authorizes the request of the NWDAF FL server based upon the information registered in Step 1.

Step 5a: If the NWDAF FL server is authorized to request the NWDAF FL clients to perform the FL operations, the NRF provides the corresponding access tokens for the selected NWDAF FL client member(s).

Step 5b: else, NRF would reject the access token requests.

Step 6: The NWDAF FL server sends the service requests to the NWDAF FL clients with the respective access tokens received in Step 5a.

Step 7: NWDAF FL clients can start the FL process for a given request in case of successful access token verification.

NOTE 1: The solution assumes that user consent for model training is provided for the NWDAF FL member as per TS 29.503 (table 6.1.6.3.20-1).

NOTE 2: Authorization process of the NWDAF FL client(s) by the NWDAF FL server is implicit, since the NWDAF FL server sends the requests only to the discovered NWDAF FL clients.

### 6.18.3 Evaluation

The solution addresses the requirement of authorization of the NWDAF FL server to include a client NWDAF into a FL group. The authorization of the NWDAF FL client(s) to join a FL group is implicit, since the NWDAF FL server sends the request only to discovered NWDAF FL clients indicated by the NRF.

The solution proposes:

- An initial registration of the FL members indicating the role in the FL group (client or server), and if the member is a client the list of allowed FL servers for a particular Analytics Id.

- The discovery request of the FL members candidates for the FL group is tackled by NRF, which upon the authorization of the request, delivers the corresponding token used by the FL server to trigger the FL process.

When new allowed FL server NWDAFs are introduced, the FL member NWDAF needs to update its list of allowed server NWDAFs

## 6.19 Solution #19: Access control for user plane data

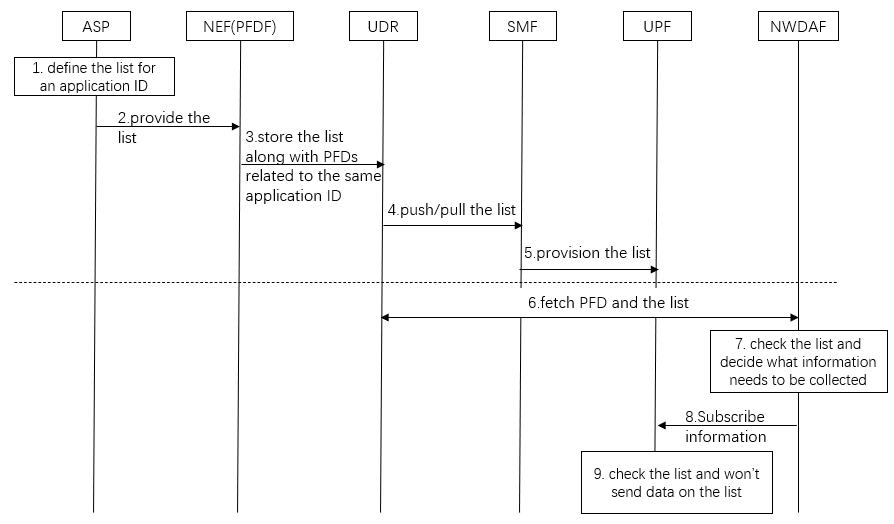
### 6.19.1 Introduction

The solution addresses KI#5 on security for NWDAF-assisted application detection.

When NWDAF provides NWDAF-assisted PFD Determination analytics, it needs to use data extracted from the inspected user plane packets in the flow. This solution proposes a user plane data access control method, where application service provider (ASP) provides a user-plane-data-access-control list of data/data type that should not be provided to NWDAF for each application identifier maintained by the ASP.

### 6.19.2 Solution details

In this solution, ASP defines the user-plane-data-access-control list of data/data type that should not be provided to NWDAF for an application. NWDAF and UPF both check the user-plane-data-access-control list to make sure data that matches the user-plane-data-access-control list will not be collected.



Step 1: ASP defines the user-plane-data-access-control list of data/data type that should not be provided to NWDAF for an application identifier maintained by the ASP.

Step 2: ASP provides the user-plane-data-access-control list to NEF (PFDF).

Step 3: NEF (PFDF) stores the user-plane-data-access-control list along with PFDs related to the same application identifier in UDR.

Step 4: The user-plane-data-access-control list may be retrieved by SMF from NEF (PFDF) in "pull" mode or may be provisioned from NEF (PFDF) to the SMF in "push" mode.

When the "push" mode is used, the NEF (PFDF) retrieves from the UDR the user-plane-data-access-control list for each application identifier and distributes them to those SMFs that enable access to those applications. There are three methods to provision the user-plane-data-access-control list from the NEF (PFDF) to the SMF:

a) Push of whole user-plane-data-access-control lists and PFD(s) that can be accessed by the NEF (PFDF) according to operator configuration in NEF (PFDF) (e.g. provision per day according to operator configuration);

b) Selective push of an ASP change in the user-plane-data-access-control list (i.e. ASP changes the list while operator configuration defines when to push);

c) Selective push of an ASP change in the user-plane-data-access-control list according to ASP request (i.e. ASP indicates to push changes in the list).

When the "pull" mode is used, at the time a PCC Rule with an application identifier for which PFDs are not available is activated or provisioned, the SMF requests all PFDs and the user-plane-data-access-control list for that application identifier from the NEF (PFDF), and NEF (PFDF) retrieves them from the UDR.

Step 5: If there is any user-plane-data-access-control list received and there are still active application detection rules in the UPF for the application identifier, the SMF provisions the user-plane-data-access-control list corresponding to the application identifier to the UPF.

Step 6: When NWDAF does NWDAF-assisted PFD Determination analytics, it fetches currently stored PFD information in use and user-plane-data-access-control list from UDR via NEF(PFDF).

Step 7: NWDAF checks user-plane-data-access-control list and decides what information needs to be collected. Information that matches the user-plane-data-access-control list will not be collected.

Step 8: NWDAF subscribes PFD Determination related information.

Step 9: UPF checks user-plane-data-access-control list. UPF will not send information that matches the user-plane-data-access-control list to NWDAF.

### 6.19.3 Evaluation

This solution addresses KI#5: security for NWDAF-assisted application detection.

This solution proposes that application service provider (ASP) defines the list of user data/data type that should not be provided to NWDAF for NWDAF-assisted application detection. NWDAF and UPF check the list to make sure that user data matching the list will not be collected.

This solution does not provide a method for operator to control access to user data.

## 6.20 Solution #20: Cyber attack detection using an analytics function

### 6.20.1 Introduction

The solution addresses KI#6 on Cyber-attack detection.

KI#6 describes that, to achieve cyber-attacks detection, the NWDAF can collaborate with UE and any other NFs to collect related data as inputs and provide alerts of anomaly events as outputs to OAM and other NFs which have subscribed to them. Further KI#6 also considered two example cases such as (i) fraudulent relay nodes which may modify or change messages between the UE and the RAN and (ii) DoS attacks related to network entities (e.g., RAN and Core network).

### 6.20.2 Solution details

The cyber attack detection solution is described below by using NWDAF as the analytics function, but the solution does not prevent to use any equivalent analytics function if required to be considered.

**Cyber attack detection over the network:**

A service consumer (i.e., OAM/AF/NF) can subscribe to the NWDAF and get notified about the malicious behaviour related analytics associated to one or more UEs using the procedure shown in Figure 6.20.2-1.



Figure 6.20.2-1 Network Data Collection and analytics procedure for cyber-attack detection

The steps shown in figure 6.20.2-1 is described as follows:

1. The analytics service consumer subscribes to the network attack detection related analytics information by invoking the Nnwdaf\_AnalyticsSubscription\_Subscribe service operation message, with the analytics ID (set to the malicious behaviour), list of event ID(s) (related to UE malicious behaviour, attacks/threats (e.g., cyber-attack, DDoS, etc.,), received messages (e.g., malformed messages, malicious data) violating predefined service operation input or output formats, message requests exceeding configured limits, unintended or unrecognized operations, repeated authentication failure), target of analytics (indicates UE ID(s) i.e., such as SUPI or a group of UEs (an Internal Group ID), analytics filter can be optionally included (e.g., as area of interest, S-NSSAIs, DNN, and Application ID) and may include other information for the analytics according to TS 23.288 Clause 6.1.3.

NOTE 1: In the case of untrusted AF the Target of Analytics Reporting can be a GPSI or an External Group Identifier that is mapped in the 5GC to a SUPI or an Internal Group Identifier as described in TS 23.288.

If the service consumer is already subscribed to the network attack detection related analytics information, the service consumer requests analytics information by invoking Nnwdaf\_AnalyticsInfo\_Request service operation with the respective analytics ID and the target of analytics.

2. When a request for analytics information is received, the NWDAF determines whether triggering new data collection is needed.

3. If the NWDAF determine to perform data collection, it need to subscribe to 5GC NFs (i.e., serving AMF, serving SMF, AFs (i.e., via NEF if located externally)) and OAM according to the target of analytics (i.e., UE IDs) for related event exposure services to be notified for data on a set of events (as described in TS 23.288 Clause 6.2.2.1) as requested by the service consumer in step 1.

The NWDAF subscribes to set of event IDs (related to UE malicious behaviour, attacks/threats (e.g., cyber-attack, DDoS, etc.,), received messages (e.g., malformed messages, malicious data) violating predefined service operation input or output formats, message requests exceeding configured limits, unintended or unrecognized operations, repeated authentication failure) by invoking the Nnf\_EventExposure\_Subscribe service operation (in case of AF being the data producer, the NWDAF can use Naf\_EventExposure\_Subscribe service operation) along with the UE IDs.

4. If the NWDAF subscribes to the set of event IDs (as in step 3), the NF manages the inference data which includes all event related information.

5. The NF notifies the NWDAF (e.g., event specific inference data) by invoking the Nnf\_EventExposure\_Notify service operation.

NOTE 2: If the data need to be collected from an AF, then in steps 3-5, the target AF will be involved instead of NF and if the AF is not trusted, then the event exposure is invoked via the NEF.

6. Alternatively, the NWDAF can also collect data from OAM (e.g., for the target of analytics (UE IDs) related to the NFs/AFs to collect event specific information mentioned in step 3 in the form of inference data/logs) using the procedure specified in TS 23.288 Clause 6.2.3.

7. The NWDAF performs the requested analytics using the data collected from the NFs, AFs, and/or OAM.

NOTE 3: The UE malicious behaviour analytics related implementation logic can be up to Operator's implementation and it is out of 3GPP scope.

8. If the service consumer is subscribed to analytics information, the NWDAF notifies the service consumer (i.e., based on the request from the service consumer using Nnwdaf\_AnalyticsSubscription\_Notify or Nnwdaf\_AnalyticsInfo\_Request response service operation), the Analytics Reporting Parameters includes the event specific network attack detection analytics information with the corresponding attack detection and threat alerts respectively as shown in the table 6.Y.2-1 and 6.Y.2-2 respectively.

Table 6.20.2-1: Network attack detection Statistics

|  |  |
| --- | --- |
| Information | Description |
| Exceptions (1..max) | List of observed exceptions |
| > Exception ID | The risk detected by NWDAF |
| > Exception Level | Scalar value indicating the severity of the abnormal behaviour |
| > Exception trend | Measured trend (up/down/unknown/stable) |
| > Cause | Indicates the cause for the exception and alerts such as related to UE malicious behaviour, configuration issues, type of attack (e.g., cyber attack/DoS/DDoS or any other threat), overload/flooding, software issues accordingly. |
| > List of malicious UEs | UE Identification information (i.e., SUPIs/ Internal Group IDs) related to UEs that are identified as behaving maliciously. |
| > List of impacted network functions | Identification information of all the impacted NFs, and AFs due to the UE malicious behaviour. |
| > Amount | Estimated number of NFs, and AFs affected by the Exception |
| > Exception category | Indication if the UE behaviour is an attack or genuine error |

Table 6.20.2-2: Network attack detection Predictions

|  |  |
| --- | --- |
| Information | Description |
| Exceptions (1..max) | List of predicted exceptions |
| > Exception ID | The risk detected by NWDAF |
| > Exception Level | Scalar value indicating the severity of the abnormal behaviour |
| > Exception trend | Measured trend (up/down/unknown/stable) |
| > Cause | Indicates the cause for the exception and alerts such as related to UE malicious behaviour, configuration issues, type of attack (e.g., cyber attack/DoS/DDoS or any other threat), overload/flooding, software issues accordingly. |
| > List of malicious UEs | UE Identification information (i.e., SUPIs/ Internal Group IDs) related to UEs that are identified as behaving maliciously. |
| > List of impacted network functions | Identification information of all the impacted NFs, and AFs due to the UE malicious behaviour. |
| > Amount | Estimated number of NFs, and AFs affected by the Exception |
| > Exception category | Indication if the UE behaviour is an attack or genuine error |
| > Confidence | Confidence of the prediction |

NOTE 4: High false rate/High negative rate are few performance metrics related to the intelligence algorithm used in any analytics process (e.g., here cyber attack detection). As the intelligence algorithm used for the analytics is up to Operator's implementation, the choice of intelligence algorithm is not in the scope of this study. Therefore, it is up to the operator to select the intelligence algorithm by considering various performance metrics based on the decision of the operator policy.

### 6.20.3 Evaluation

This solution describes the methods to collect data from the network to enable cyber attack detection and related output analytics provision in the network.

The cyber attack detection solution is described by using NWDAF as the analytics function, but the solution does not prevent to use any equivalent analytics function if required to be considered.

The existing analytics on ‘abnormal behaviour' in TS 23.288 clause 6.7.5 enables data collection related to mobility and communication related aspects such as unexpected UE location, Ping-ponging across neighbouring cells, unexpected wakeup, unexpected radio link failures, unexpected long-live/large rate flows, unexpected wakeup, suspicion of DDoS attack, wrong destination address, too frequent service access but in addition this solution helps to also collect data on malformed messages violating predefined message input or output formats, message requests exceeding configured limits, unintended or unrecognized operations, and repeated authentication failure. As the Intelligence algorithm used in the analytics related to cyber attack detection is up to operator implementation, it is recommended that a selection of an algorithm can consider their performance metrics e.g., cyber attack detection precision, false positive rate, false negative rate etc., The solution does not provide evidence for cyberattack detection precision, false positive rate, false negative rate. Without that, the administrator may take wrong action for the wrong detection so that the network may face unsecure configuration risk.

## 6.21 Solution #21: Authorization of data and analytics exchange in roaming case

### 6.21.1 Introduction

This solution is addressing on KI#1.

This contribution proposes that NRF is used as the authorization server to issue the allowed data/analytics list for the PLMN in the token for the NWDAF consumer.

### 6.21.2 Solution details



Figure 6.21.2-1: Authorization of data and analytics exchange in roaming case

NRF 2 is preconfigured an allowed data/analytics list to indicate which data/analytics is allowed to be provided to other PLMNs.

NWDAF 1 sends token request to the NRF 1, the message includes the existing IEs, i.e. the NF consumer type, target NF type, expected NF Service name, serving and home PLMN ID, etc.

NRF 1 forwards the token request to NRF 2.

The NRF 2 issues token according to the preconfiguration based on serving PLMN ID. The claim of token contains the allowed data/analytics list related to the serving PLMN ID.

The NRF 2 returns the token to the NRF 1.

The NRF 1 forwards the token to the NWDAF 1.

NWDAF 1 sends Nnwdaf\_RoamingData\_Subscribe or Nnwdaf\_RoamingAnalytics\_Subscribe message with requested data/analytics and the token to the NWDAF 2.

The NWDAF 2 verifies the service request, including verifying token, and whether the allowed data/analytics list in the token covering with the requested data.

The NWDAF 2 returns the data/analytics to the NWDAF 1.

### 6.21.3 Evaluation

The solution addresses the authorization aspects in data and analytics exchange in roaming case described in key issue #1. The solution proposes an authorization schema with analytics id granularity to be implemented in NRF.

In this solution, NRF is used as the authorization server to issue the allowed data/analytics list for the PLMN in the token for the NWDAF consumer. In this way, the NWDAF consumer does not need to request a new token once the NWDAF consumer needs to request other data.

This solution also proposes that all the authorization information related to the NWDAF roaming is configured in the NRF.

The NWDAF profile (and its internal policy) needs to be updated to accommodate for the analytics ID/ data ID level granularity, along with the consumer and PLMN information which can receive this. This information will then be needed to be updated in the NRF, for the NRF to ensure that the NWDAFc is authorized. Finally this will need a change in the access token claims.

# 7 Conclusions

## 7.1 Conclusion on Key Issue #1 "Protection of data and analytics exchange in roaming case"

For Key Issue #1, it is recommended to use the following principles as the baseline for protection of data and analytics exchange in roaming case:

- The NRF uses token-based authorization to restrict the access to services of NWDAF from outside the own PLMN only to the new service(s) of the NWDAF described in TS 23.288 [5], clause 6.1.5.

- Authorization at data and analytics level is enforced by the roaming entry NWDAF producer. The parameters used by NWDAF service consumer to request/subscribe to the services provided by NWDAF producer are defined in TS 23.288 [5], clause 6.1.5. Accordingly, the operator authorization policies can be configured locally in the NWDAF producer. Also, claims in the service access token provided by NRF can be extended to convey Analytics ID parameter, so the NWDAF producer can authorize with analytics granularity (after NRF having checked the Analytics ID in the token get request).

- The roaming entry NWDAF producer is responsible to control the amount of exposed data/analytics and to abstract or hide internal network aspects in the exposed data/analytics. The corresponding mechanisms used to restrict the data/analytics and/or anonymization are subject to the implementation.

- 5GS uses the N32 interconnection security mechanism to support confidentiality, integrity, and replay protection for data and analytics exchange between PLMNs.

## 7.2 Conclusion on Key Issue #2 "Authorization of selection of participant NWDAF instances in the Federated Learning group"

It is recommended to use the following general principles for authorization of participant NWDAFs in a Federated Learning group:

- Initial registration of the NWDAFs involved in FL including Analytics Id and FL capability type (i.e. client, server).

- Authorization of the server NWDAF to include a client NWDAF into a Federated Learning group is done by NRF using SBA OAuth 2.0 token-based authorization.

- The NRF uses the interoperability indicator per Analytics ID, specified in TS 23.288 [5] and provided by the client NWDAF during registration, for authorization of the server NWDAF with vendor granularity.

NOTE: Fine-grained authorization can be done locally at the client NWDAF.

- The server NWDAF includes the desired Analytics ID in the access token request. The NRF checks whether the server NWDAF is authorized to include the client NWDAF in the FL group for this Analytics ID. If the server NWDAF is authorized, the NRF will issue an access token with the Analytics ID included.

- Authorization of the client NWDAF is implicit, since it can join a Federated Learning group only when selected by the server NWDAF.

## 7.3 Conclusion on Key Issue #3 "Security for AI/ML model storage and sharing"

The conclusions for KI#3 are:

- Authorization of the model retrieval at the NRF uses OAuth 2.0 token-based authorization. The NRF uses information provided by the NF service producer (NWDAF MTLF) and information provided by the NF service consumer (e.g. NWDAF AnLF).

- The NF service producer (NWDAF MTLF) needs to be registered in the NRF, indicating the NF service producer information (including interoperability indicator per Analytics ID) that is used by the NRF to decide whether the consumer is authorized. The NF service consumer (e.g. NWDAF AnLF) needs to be registered in the NRF as OAuth 2.0 client, indicating the NF service consumer information (including Vendor ID) that is used by the NRF to decide whether the consumer is authorized. Whether the NRF also uses the interoperability indicator of the NF service consumer is addressed in normative phase.

- The NF service consumer (e.g. AnLF) includes the desired Analytics ID(s) in the access token request. The NRF checks whether the NF service consumer is authorized to retrieve models for this Analytics ID(s). If the NF consumer is authorized, the NRF issues an access token with the Analytics(s) included in the access token.

- The model is stored in encrypted format unless both the AI/ML model producer (NWDAF MtLF) and storage platform (ADRF) are part of the same system and belong to the same vendor and operator security domain.

Storage of the model in encrypted format can be required by the trust model established to store and share AI/ML models. The trust model between AI/ML NF producer (NWDAF MtLF), storage platform (ADRF) and NF consumer (e.g., AnLF) is to be determined during the implementation phase among operator and the providers of the different platforms (MtLF, AnLF, ADRF).

- As per the request of Analytics Id by the NFc, the MTLF performs authorization of the corresponding model retrieval per selected model.

NOTE: The model delivery procedure is to be defined by 3GPP SA2.

- The granularity of the authorization at AI/ML model level is performed at MtLF.

- ADRF verifies that the requested AI/ML model can be retrieved by the NF consumer(s) (MTLF or AnLF), based on the decision by the MtLF.

## 7.4 Conclusion on Key Issue #4 " Anomalous NF behaviour detection by NWDAF "

No normative work is pursued for KI #4.

## 7.5 Conclusion on Key Issue #5 "Security for NWDAF-assisted application detection"

Solution#19 only proposes a method for application service provider to define the user data that should not be provided to NWDAF for NWDAF-assisted application detection. Therefore, no normative work is considered for KI#5 for this release.

## 7.6 Conclusion on Key Issue #6 "Key issues on Cyber-attack detection"

No normative work is pursued for KI#6.

Annex A:  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2022-05 | SA3#107-e | S3-220771 |  |  |  | TR Skeleton | 0.0.0 |
| 2022-05 | SA3#107-e | S3-221279 |  |  |  | S3-220772, S3-220773, S3-221269, S3-221176, S3-221221, S3-221222 | 0.1.0 |
| 2022-07 | SA3#107Adhoc-e | S3-221657 |  |  |  | S3-221453, S3-221454, S3-221603, S3-221595, S3-221639, S3-221615, S3-221591, S3-221616, S3-221617 | 0.2.0 |
| 2022-10 | SA3#108Adhoc-e | S3-223115 |  |  |  | S3-223090, S3-222789, S3-223021, S3-222623, S3-223023,S3-223015,S3-223016, S3-223091,S3-223024, S3-222551, S3-222624, S3-223001 | 0.3.0 |
| 2022-11 | SA3#109 | S3-224178 |  |  |  | S3-224146, S3-224144, S3-224145, S3-224147, S3-224148, S3-224149, S3-224150, S3-224151, S3-224079, S3-224077, S3-224075, S3-224076, S3-223688 | 0.4.0 |
| 2023-01 | SA3#109e-adhoc | S3-230547 |  |  |  | S3-230574,S3-230082,S3-230456,S3-230101,S3-230525,S3-230442,S3-230241,S3-230242,S3-230243,S3-230546,S3-230457,S3-230524,S3-230443,S3-230488, S3-230459,S3-230447,S3-230448,S3-230506,S3-230510 | 0.5.0 |
| 2023-02 | SA3#110 | S3-231500 |  |  |  | S3-231491, S3-231492, S3-231493, S3-231494, S3-231496, S3-231593, S3-231497, S3-230919, S3-231594, S3-231595, S3-230920, S3-231597 | 0.6.0 |
| 2023-03 | SA#99 | SP-230132 |  |  |  | Presented for information | 1.0.0 |
| 2023-04 | SA3#110e-adhoc | S3-232230 |  |  |  | S3-232117, S3-232164, S3-232151,S3-231837,S3-232186,S3-232105 | 1.1.0 |
| 2023-05 | SA3#111 | S3-233264 |  |  |  | S3-233249, S3-233250, S3-233251, S3-233263, S3-233262, S3-233265, S3-233124 | 1.2.0 |
| 2023-06 | SA#100 |  |  |  |  | Presented for approval | 2.0.0 |
| 2023-06 | SA#100 |  |  |  |  | Upgraded to change control version | 18.0.0 |
| 2023-06 | SA#100 |  |  |  |  | EditHelp review | 18.0.1 |
| 2023-09 | SA#101 | SP-230892 | 0002 | 1 | F | Clean up of TR 33.738 | 18.1.0 | |