5G AKMA lacks resilience

Max Duparc[1,2] and Colin Barschel[1]

[1] Cyber-Defence Campus, armasuisse Science and Technology, Thun, Switzerland
colin.barschel@armasuisse.ch
[2] EPFL, Lausanne, Switzerland
max.duparc@epfl.ch

Abstract. In release 16, the 3rd Generation Partnership Project (3GPP) introduced the Authentication and Key Management for Applications mechanism (AKMA) as a new security feature of 5G networks, enabling the bootstrapping of secure connections between User Equipment (UEs) and Application Functions (AFs) by leveraging the primary authentication of UEs to the network. We study the resilience of AKMA and show that it is unsecure considering its scheduled usage. In its current form, AKMA holds security weaknesses that enable adversaries, under reasonable assumptions, to perform wiretapping, Man-in-the-Middle (MitM) attacks and to recover the identity of UEs trying to connect to an AF. For improving AKMA resilience and in addition to providing the details of those attacks, this paper propose mitigations to thwart them.

Keywords: 5G · AKMA · Resilience · Security · Wiretap · Privacy

1 Introduction

The creation of 5G StandAlone (SA) by 3GPP and its Service Based Architecture (SBA) comes as an answer to the ever increasing need for better connections in our day-to-day life. It must enable greater data rate, lower latency, better reliability, more flexible networks, increased network density and more, including improved security.

Precisely, on the side of security issues, much publicity was shed to 5G-AKA and EAP-AKA′, which are improvement of the primary authentication. However, these are not the sole modifications that took places in the mobile network security architecture. In particular, a detailed list of new authentication mechanisms can be found in [16].

Among these new authentication mechanisms lies AKMA, a marginally studied security mechanism of 5G SA that enable the bootstrap of secure connections between UEs and AFs. It does so by leveraging the primary authentication of the UE to the 5G network. In the eyes of 3GPP, it is an essential aspect of the future 5G Internet of Things (IoT) security paradigm, although it is not restricted to that usage and can be used by any kind of UE. Due to its relative anonymity, few literature over AKMA security is available, though analysis using formal verification tools can be found in [19,10], the former using Tamarin and

the later ProVerif. Our work expends on their results and considers the resilience of AKMA.

The security notion of resilience, as defined in [11], describes "the ability of an infrastructure to continuously deliver its intended outcome, despite adverse cyber attacks". In the case of the AKMA mechanism, the intended outcome refers to the capacity to enable secure connections between UEs and AFs.

Contributions In this paper, we show that AKMA holds worrisome problems of resilience. The security provided by AKMA connections between a given UE and AF is indeed strongly impacted by other AFs, as its is compromised in the case of any AF inside and outside the network going rogue.

Toward that end, we present the following contributions:

- We discuss the importance of considering, due to their nature, usages, lack of standardisation and multiplication in 5G SA, the security risk of AF compromising.
- We analyse both standard and roaming architecture of AKMA, the later lacking any literature on its security and inducing additional vulnerabilities.
- To do so, we propose 3 threat models, all corresponding to real threat scenarios and enabling the study of AKMA resilience.
- We list AKMA's security weaknesses, distinguishing between those that are consequences of AKMA design and the other that are due to the underspecification of the standard.
- We explain how these weaknesses can be combined together in order to enable attacks like wiretapping, MitM and recovering the identity of UEs using AKMA. We also details additional vulnerabilities that further ease these attacks.
- We finally propose modifications of the AKMA mechanism that solve some of the found security weaknesses in order to prevent our attacks and vulnerabilities.

Our definition of AKMA is based on v18.0 of [6] and v18.0 of [9]. These versions where the latest available when we disclosed with 3GPP our findings, together with our concerns. We are happy to claim that some of our identified weaknesses are now fixed in the latest version, with more coming in November of 2023.

2 AKMA

The full technical specifications of AKMA, as detailed by 3GPP, are available in [6], with additional information (especially regarding roaming) detailed in [9]. It can be seen as the fusion of the 4G Generic Bootstrapping Architecture (GBA) and of the Battery Efficient Security for very low Throughput (BEST) Machine Type Communication. Its goal is to be an universal mechanism that enable secure connections (denoted Ua*) between all types of UEs (ranging from

mobile handsets to extremely low power IoT devises) and AFs by reusing the primary authentication of 5G networks in an explicit manner, meaning that unlike GBA, it does not require a dedicated handshake between the UE and the network. AKMA can be used as:

- An OAuth-like delegation authentication protocol.
- A bootstrapping mechanism based on PreSharedKey (PSK).

This paper will focus on the second usage, as it is the one enabling our attacks.

2.1 AKMA Architecture

Due to space considerations, we omit a presentation of the general 5G SA SBA architecture to focus on the AKMA mechanism. It is nonetheless excellently described in [4]. Fig.1 describes the global architecture of AKMA and the different Network Functions (NF) of the 5G SA core that are used, together with their dedicated interfaces. They are:

- AAnF: AKMA Anchor Function
- AMF: Application Management Function
- AF: Application Functions
- AUSF: AUthentication Server Function
- NEF: Network Exposure Function
- UDM: Unified Data Management
- UE: User Equipment

Details on their respective roles in the 5G SA architecture are available in [5] and [1].

2.2 Key requests

Fig.2 depicts essential exchanges performed in order to setup AKMA and to request the application key (K_{AF}), depending if the AF is internal (meaning that it interacts with Network Functions (NFs) directly) or external (meaning that it interacts with NFs via the NEF). Some additional details are available in section 6.2 and 6.3 of [6]. Elements in *italics* are send depending on the local policy of the network.

3GPP provides in annex B of [6] detailed specifications on how to adapt AKMA key requests to version 1.2 and 1.3 of the Transport Layer Security protocol (TLS) [15,17]. It is done as:

- A shared key-based UE authentication with certificate-based AF authentication (the aforementioned OAuth-like protocol).
- A shared key-based mutual authentication between UE and AF (the PSK based bootstrapping mechanism).

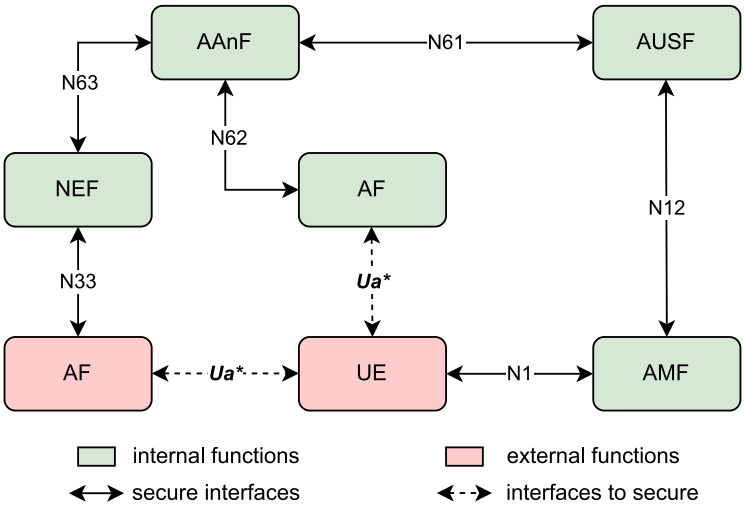

Fig. 1. AKMA standard architecture

In the latter case and if TLS 1.3 is used, the Application_Session_Establishement_Request (4. and 9. of Fig.2) is in fact a TLS_PSK_ClientHello together with the A-KID. Similarly, the Application_Session_Establishement_Response (8. and 16. of Fig.2) is a TLS_PSK_ServerResponse.[3]

In this paper, we distinguish between two families of TLS "cipher-suites" using PSK.

- TLS_(EC)DHE_PSK: Where PSK are only use to ensure the authentication, while the confidentiality derives from a Diffie-Hellman key exchanges.
- TLS_PSK : Where PSK are use to ensure *both* authentication and confidentiality.

A fully detailed technical overview of the inner workings of those cipher-suites is available in [18].

2.3 AKMA specific keys and identifiers

AKMA ensure its security by utilising Key Derivation Functions (KDFs) over the following keys and identifiers:

- The SUbscription Permanent Identifier (SUPI) and Generic Public Subscription Identifier (GPSI) are permanent UE identifiers. The first is used inside networks while the second is used outside.
- The K_{AUSF} is derived from the primary authentication (see A.2 in [5]) and last up until a new primary authentication is performed.

[3] The case of TLS 1.2 is globally similar and is available in section B.1.3.2.1 of [6].

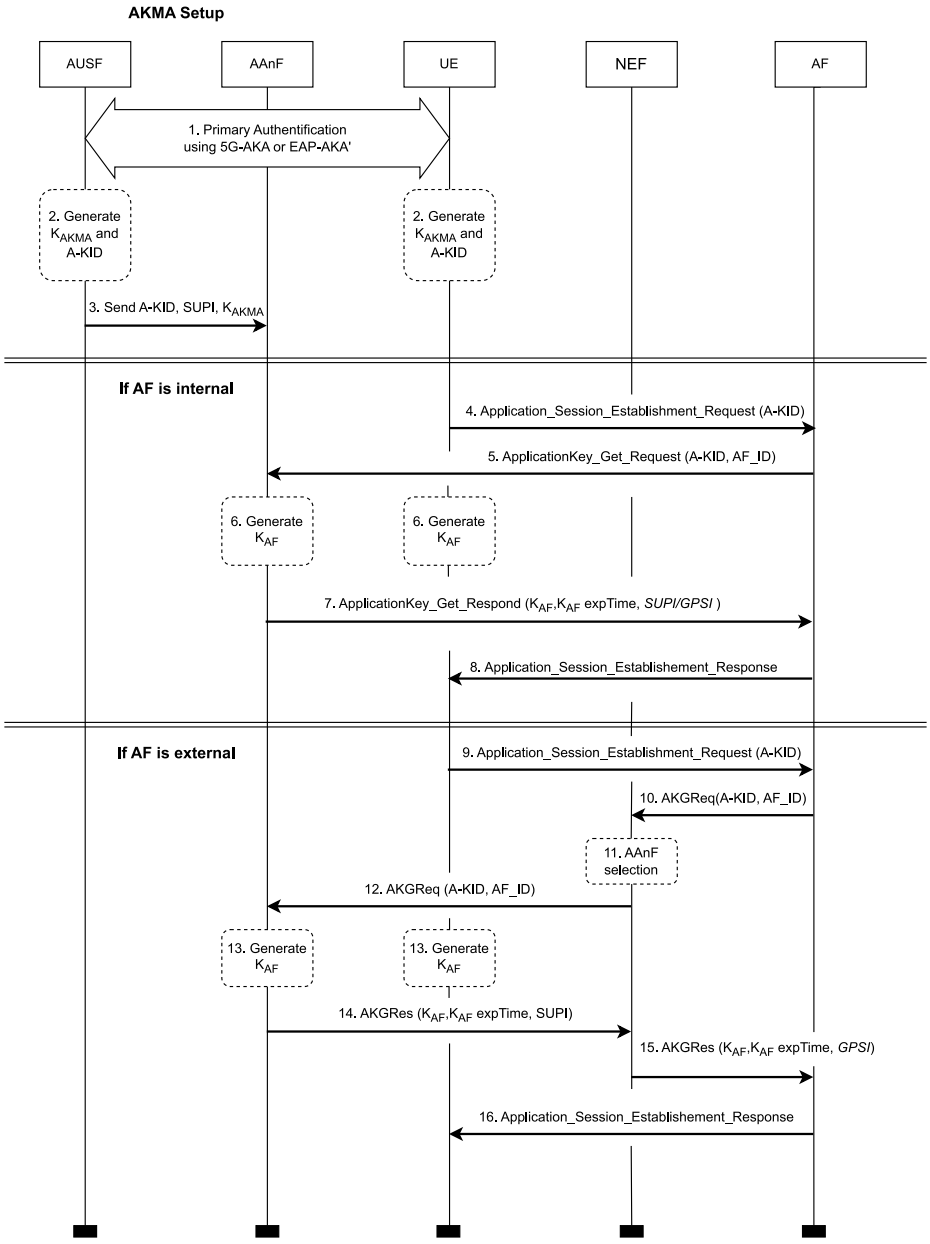

Fig. 2. AKMA procedures for key requests

– $\mathsf{K_{AKMA}}$ is derived as:

$$\mathsf{K_{AKMA}} := \mathsf{KDF}(\mathsf{K_{AUSF}}, \mathsf{S_0})$$

$$\mathsf{S_0} := \mathsf{0x80} \| \text{“AKMA”} \| \mathsf{0x0004} \| \mathsf{SUPI} \| \mathsf{length}(\mathsf{SUPI})$$

It can only be changed:

- By running a new primary authentication.
- By using the AKMA context removal procedure. (Protocol specified in section 6.6 of [6].)

– The AKMA-Key IDentifyer (A-KID) is defined as:

$$\mathsf{A\text{-}KID} := \mathsf{A\text{-}TID} \| \mathsf{RID} \| \mathsf{HNI}$$

$$\mathsf{A\text{-}TID} := \mathsf{KDF}(\mathsf{K_{AUSF}}, \mathsf{S_1})$$

$$\mathsf{S_1} := \mathsf{0x81} \| \text{“A-TID”} \| \mathsf{0x0005} \| \mathsf{SUPI} \| \mathsf{length}(\mathsf{SUPI})$$

with HNI the Home Network Identifier and RID being the Routing InDicator. A-KID is updated only when:

- A new primary authentication is performed.
- The context removal procedure is used.

– AF-ID is defined as:

$$\mathsf{AF\text{-}ID} := \text{AF's FQDN} \| \text{Ua*\_security\_protocol\_ID}$$

with FQDN standing for Fully Qualified Domain Name and the Ua*_security_protocol_identifier being specified in [6], with additional rules available in [1].

– $\mathsf{K_{AF}}$ is defined as:

$$\mathsf{K_{AF}} := \mathsf{KDF}(\mathsf{K_{AAnF}}, \mathsf{S_2})$$

$$\mathsf{S_2} := \mathsf{0x82} \| \mathsf{AF\text{-}ID} \| \mathsf{length}(\mathsf{AF\text{-}ID})$$

$\mathsf{K_{AF}}$ can be changed using different methods:

- By running a new primary authentication.
- By using the AKMA context removal procedure.
- The AF *may* trigger a re-keying, when $\mathsf{K_{AF}}$ lifetime is passed. This is up to its implementation.

the KDFs used by 3GPP and additional technicalities are specified in annex B of [2].

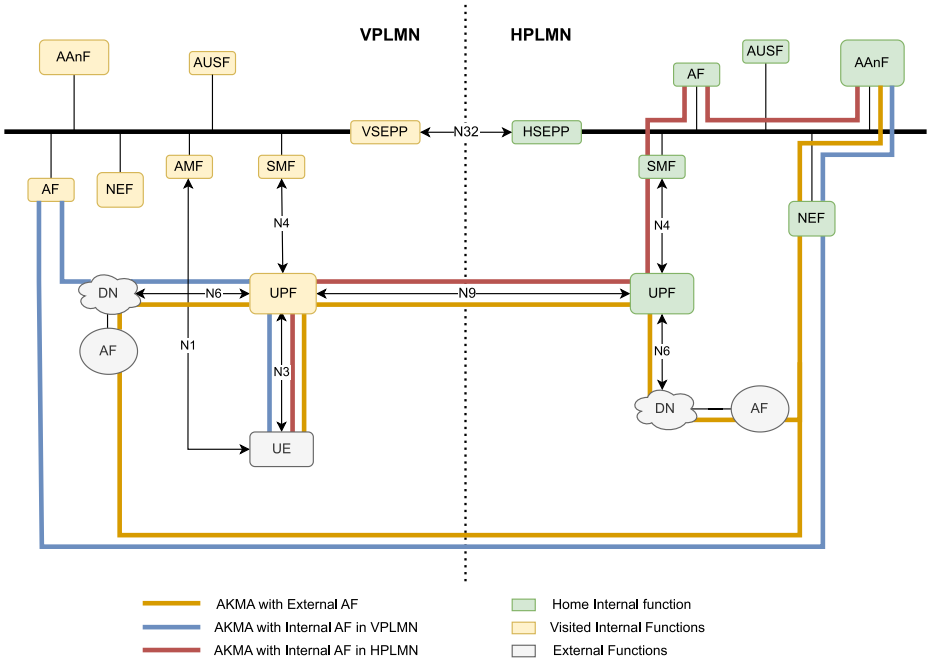

Fig. 3. AKMA architecture in a roaming scenario

2.4 Roaming

The use of AKMA in roaming scenarios (meaning that the UE utilizes the user plane of a Visited Public Land Mobile Network (VPLMN) distinct from its Home Public Land Mobile Network (HPLMN)) was just freshly fixed by 3GPP in [9]. In its current form, it works as depicted in Fig.3:

There are 3 cases to consider, depending on AF's nature.

- **Internal AFs in the HPLMN**: This is handled by deploying a home routed PDU session[4] to enable the Ua* protocol between the UE and the AF. This session passes via the N9 and N4 interfaces. These AFs connect to the HAAnF (the AAnF of the HPLMN) directly.
- **External AFs that are trusted by the HPLMN**: The UE initiates the Ua* protocol with the AF via the Data Network (DN). This connection can be either home-rooted or using a local breakout. These AFs connect to the HAAnF via the HNEF.
- **Internal AFs in the VPLMN**: The UE initiates the Ua* protocol with the AF using a local breakout. These AFs also connect to the HAAnF via the HNEF.

It must be noted that external AFs that are trusted by the VPLMN but not the HPLMN cannot use AKMA with the UE. This therefore follows the

[4] i.e. an end-to-end connection from the UE to the control plain through the UPF.

principle of trust non-transitivity.[5] Additionally, the Security Edge Protection Proxies (SEPPs) and the N32 interface are never used for AKMA.

2.5 AFs

The definition of an AF given in [5] is "a system that interacts with the 3GPP Core Network in order to provide services". AF's expansion is currently an important field, as they should provide many new functionalities of 5G SA like Applications Servers (AS), the Quality of Service (QoS) hubs, the Multi-access Edge Computing (MECs), etc.

Speaking of MECs, they are of particular interest to us. As defined by the European Telecommunications Standards Institute (ETSI) in [13], they "enables the implementation of applications as software-only entities that run on top of a virtualisation infrastructure, which is located in or close to the network edge". Use cases are massive and listed in [14], ranging from Vehicle to Infrastructure (V2I) to third party cloud provider, passing through deployment in dense-network environment. They are a central part of the solution for 5G networks to provide the promised lower latency, higher data rate and increased network flexibility.

Importantly MECs can utilise AKMA and request information regarding UE identity, as specified in [8]. MECs are thus:

- Numerous.
- Outside or at the edge of the network, possibly next to the Radio Access Network (RAN), therefore in potentially unsecured environments.
- Potentially handled by 3rd parties.
- Potentially trusted by the 5G SA network.
- No strict standard due to the diversity of their usages.

It is therefore apparent from this list of properties that MEC compromising (and by extension AF compromising) is a serious security concern that should be considered very closely.

3 Threat models

Although we now have described the AKMA mechanism, we have not yet detailed in which environment it must operate, which is equivalent to ask what is our assumed security? 3GPP security requirements is detailed in section 4.4 of [6], with supplementary information on the NEF-AF interface (N33) available in section 12 of [4]. AKMA was therefore conceived and studied in literature [19,10] using the following threat model:

T0: - The adversary carries messages.

[5] This means that there is no automatic trust between a network and an element that is trusted by another network itself trusted by the initial network.

- All connections in the core are secure[6] except for Ua*.
- All NFs are honest.
- In case of roaming, the VPLMNs is trusted and follows 3GPP guidelines.
- The connections between VPLMNs and HPLMN are secure.

Although perfectly valid, for the reasons stated earlier, we disagree on the realism of this threat model as, by their number, lack of strict standardisation and because they can be handled by third-party, AFs are very likely candidates for holding vulnerabilities and being hacked. We therefore propose to examine what would be the adversarial capacity if, in addition, he had access to one malicious AF.

We thus propose 3 new threat models:

T1: **T0** + the adversary has access to a malicious external AF.

T2: **T0** + the adversary has access to a malicious internal AF inside the HPLMN.

T3: **T0** + the adversary has access to a malicious internal AF inside the VPLMN.

All three represent real threat scenarios that are more in line with the expected development of the 5G SA. Indeed:

T1 is the inevitable consequence of the massive IoT revolution promised by 5G SA, as we shall see numerous MECs in unsecured environments, increasing to near certainty the probability that one is hacked.

T2 can be seen as the capacity of a government on its own national networks, but is not limited to this setting. As detailed in [12], internal MECs exists and could therefore be corrupted. This is worsen by the fact that this threat model enables the most powerful attacks, making internal AFs valuable targets.

T3 is likewise the representation of the capacity of external government on their national 5G networks but considers more generally if it is interesting, in order to attack a given network, to attack a weaker one and then to use roaming. It must be noted that in this model, apart from the malicious AF, the VPLMN is assumed to be honest and to follow 3GPP guidelines.

Ideally, as we would like for the security provided by AKMA for a connection between a given UE and AF to not be impacted by other AFs, those three threat models should have the same potency than **T0**. We will show that it is not the case.

[6] Secure connections shall ensure confidentiality, integrity, replay protection and authentication.

4 Security Weaknesses

We now list all security weaknesses that we found during our analysis. These weaknesses will then be combined in order to enable our vulnerabilities and attacks. We distinguish between two types of weaknesses:

- Those that are consequences of AKMA design.
- Those that are caused by the underspecification of the standard.

This separation in two categories will be important when we discuss mitigations. Indeed, if fixing the latter ones is relatively easy, fixing the former requires significant modifications that are sometimes close to impossible as some weaknesses are natural consequences of desired properties of AKMA.

4.1 Security weaknesses by design

1A K_{AF} is deterministically derived.
If we make two key requests with the same K_{AKMA}, A-KID and AF-ID, then we will receive the same K_{AF} twice. Furthermore, AKMA does not forbid the re-sending of K_{AF}. This comes from the deliberate decision to make AKMA an explicit bootstrapping protocol, as no specific handshake between the AAnF and the UE can be done, preventing the usage of randomness.

1B K_{AKMA} and A-KID are AF-independent and are rarely renewed.
A-KID and K_{AKMA} are only renewed by running the primary authentication or by using the context removal procedure. If the latter depends on the network policy, the former can be considered a rare event to limit resource usage. Therefore, both A-KID and K_{AKMA} can be considered as fixed in the order of one hour. An intercepted A-KID could be misused during this time frame.
Having K_{AKMA} and A-KID independent from AF is core to AKMA, as it was one of its design requirement.

1C TLS_(EC)DHE_PSK and TLS_PSK send the A-KID and AF-ID in clear.
By the design of TLS (both 1.2 and 1.3), the cipher-suite, the A-KID and the FQDN of the AF are available in cleartext inside in the clientHello message of the UE to the AF. These information are necessary for authentication and sometimes confidentiality. They therefore can not be sent in an encrypted extension.
Interestingly, TLS 1.2 is more secure that TLS 1.3 on that aspect as there exists a scenario where using TLS 1.2 does not output the A-KID, while TLS 1.3 does. This is when the UE propose to use TLS using AKMA based PSK and the AF refuses and switch to a non AKMA based TLS.

1D Using TLS_PSK does not provide forward security.
This is an old and widely known security weaknesses of TLS_PSK but that must be kept in order to enable secure connection for extremely low capacity UEs, which are one of AKMA's targets.

1E The privacy of the UE is only dependent on the AF and the 5G network.
An UE has no say whether or not an AF is authorised to query its SUPI/GPSI, as it only depends on the 5G network policy regarding the AF. Furthermore, the UE is not informed when it occurs.

4.2 Security weaknesses by underspecification

2A The AAnF does not check that the send AF-ID correspond to the sending AF.
We consider two AFs (AF1 and AF2), with AF2 internal and with a valid A-KID and AF1-ID. Then AF2 can send an ApplicationKey_Get_Request(A-KID, AF1-ID) to the AAnF and will receive a valid key. Indeed, the AAnF does not verify that AF1-ID correspond to AF2 and considers the request valid.
This weakness was, up until v17.7 of [6], possible for any AFs but was patched in the NEF but not in the AAnF, therefore solving only half of the problem, as not providing security for AFs that can connect directly to AAnF.

2B The AKMA initiation protocol is underspecified.
The specification of the AKMA initiation protocol (6.5 in [6]) is unclear on its limitations and thus on what it allows and what it does not. In its current state, it enables the AKMA initiation message to contain a different FQDN from the requested AF. The UE will use this information to derive the $\mathsf{K_{AF}}$ key on its side. This could be catastrophic if done without having the AF authenticate, as it enables AF impersonation.
This scenario could be set up by a VPLMN that does not want other PLMNs to know the FQDN of its AFs. This could indeed reveal its internal architecture to competitors. In this case, It would use a specific AF to handle all roaming AKMA demands.

2C The AAnF and NEF do not know the UE's serving PLMN.
The AAnFs and NEFs are ignorant of the serving PLMN ID in which the UE is currently. This means that key requests are handled independently of UE position.

2D The HNI inside the A-KID is not checked by the AAnF and the NEF.
The NEF and AAnF do not check during a key request that the HNI inside the A-KID corresponds to their respective PLMN. This becomes problematic when considering that AAnFs, when receiving an A-KID, will just check that they hold a $\mathsf{K_{AAnF}}$ identified by this A-KID to know whether or not AKMA is enabled. This starts getting problematic when $\mathsf{K_{AAnF}}$ is distributed in several PLMN.

5 Vulnerabilities & Attacks

We now details vulnerabilities and attacks on the AKMA protocols enabled by the security weaknesses that we listed.

5.1 Attacks via T3 are always possible, independently of UE position.

Any AF in a VPLMN that has a roaming agreement with the HPLMN of a given UE and gains access to its A-KID can request the HAAnF through the HNEF. This is possible independently of the fact that the UE is actually roaming inside the VPLMN. Indeed, both HNEF and HAAnF do not know the UE's current serving PLMN, i.e. location. This means the UE could be in its HPLMN, and a key request from an AF in the VPLMN would be considered valid and answered positively.

Used weaknesses 2C

Consequences This increases the potency of attacks using AFs inside VPLMN, as there is no need for the UE to be inside the malicious AF's PLMN to perform some attacks. This makes attacks usable as long as there exists a malicious AF in *any* VPLMN that has a roaming agreement with the HPLMN of the UE. This is especially problematic as operators have hundreds of roaming agreements.

5.2 Wiretap attack on AKMA

This attack enables an adversary to break the confidentiality of AKMA by retrieving the master key of the TLS tunnel. It is described in Fig. 4. We add that this attack is invisible to the UE and AF1.

Needed conditions

- An UE want to connect to AF1 using TLS_PSK.
- The adversary is able to *listen* at the exchanges between UE and AF1.
- The adversary has control of AF2, an internal AF in the HPLMN.

Used weaknesses 2A 1A 1B 1C 1D

Feasibility T2 (with **T1** and **T3** up until v17.7 of [6])

Modus Operandi

(i) The adversary starts listening to traffic.
(ii) The UE initiates a Ua* connection to AF1 using TLS_PSK.
(iii) The adversary *listen* to the first message (in clear text), with the A-KID and AF1-ID.
(iv) The adversary sends the A-KID and AF1-ID to AF2.
(v) AF2 sends a key request to the network (using A-KID and AF1-ID).
(vi) AF2 receives the $\mathsf{K}_{\mathsf{AF1}}$ from the network.
(vii) AF2 sends back the $\mathsf{K}_{\mathsf{AF1}}$ information to the adversary.
(viii) The adversary derives the TLS master key K_{MK} following rule in [18].
(ix) The adversary uses this key to decipher all messages exchanged between the UE and the AF using Ua*.

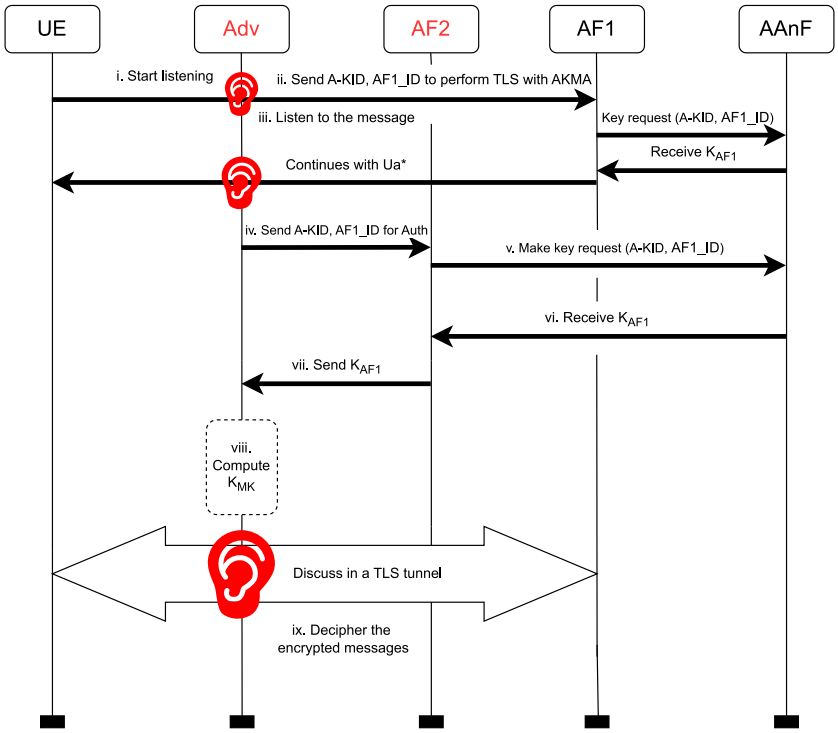

Fig. 4. Wiretap attack

Remark The query from AF2 to the AAnF must be done before the A-KID is no longer valid, i.e. before the next primary authentication or AKMA context removal.

5.3 MitM Attack on AKMA

We now detail how an adversary can perform a man-in-the-middle attack between the AF and the UE. It is described in Fig. 5. [7]

Necessary conditions

- An UE want to connect to AF1 using TLS_DHE_PSK or TLS_PSK.
- The adversary is able to *intercept* the exchanges between UE and AF1.
- Adversary have the control of AF2, an internal AF of the HPLMN.

Used weaknesses 2A 1A 1B 1C

[7] This attack is based on the idea found in [10] which depicted how an adversary was able in their formal model to impersonate the AF. We have expended it in the form of a full MitM attack.

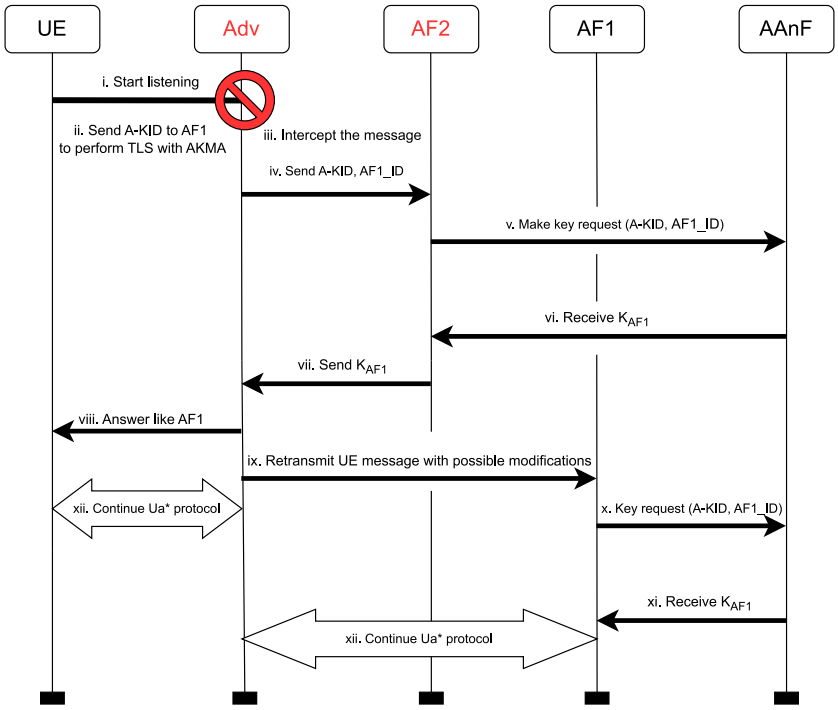

Fig. 5. Man-in-the-Middle attack

Feasibility T2 (with **T1** and **T3** up until v17.7 of [6])

Modus Operandi

(i) The adversary starts listening to traffic.
(ii) UE initiates an Ua* connection to AF1 using TLS_DHE_PSK or TLS_PSK.
(iii) The adversary *intercepts* the first message (in clear text), with the A-KID and AF1-ID.
(iv) The adversary sends the A-KID and AF1-ID to AF2.
(v) AF2 sends a key request to the network (using A-KID and AF1-ID).
(vi) AF2 receives the $\mathsf{K}_{\mathsf{AF1}}, \mathsf{K}_{\mathsf{AF1}}$expTime from the network.
(vii) AF2 sends back the $\mathsf{K}_{\mathsf{AF1}}$ to the adversary.
(viii) The adversary answers like AF1 would using $\mathsf{K}_{\mathsf{AF1}}$ and continue the Ua* connection.
(ix) The adversary then send the initial request of the UE to AF1 and modify the secret part of the Diffie-Hellman in case AKMA is using TLS_DHE_PSK.
(x) AF1 performs the key request to the AAnF/NEF.
(xi) It receives $\mathsf{K}_{\mathsf{AF1}}$ from the AAnF/NEF.
(xii) $\mathsf{K}_{\mathsf{AF1}}$ is then used to continue the Ua* connection to finish the MitM attack.

Remarks

- All this protocol must be performed before the timeout of the UE that waits for an answer.
- We can of course perform only half of this attack and impersonate the AF to the UE or the UE to the AF, depending on the situation.

5.4 Privacy breaking attack via AKMA

We now explain how an adversary can retrieve the unique identifier (SUPI/GPSI) of an UE trying to enter in contact with an AF. It is described in Fig. 6 and represents a real danger to privacy. We add that this attack is invisible for UE and AFs and difficult to spot for the 5G network.

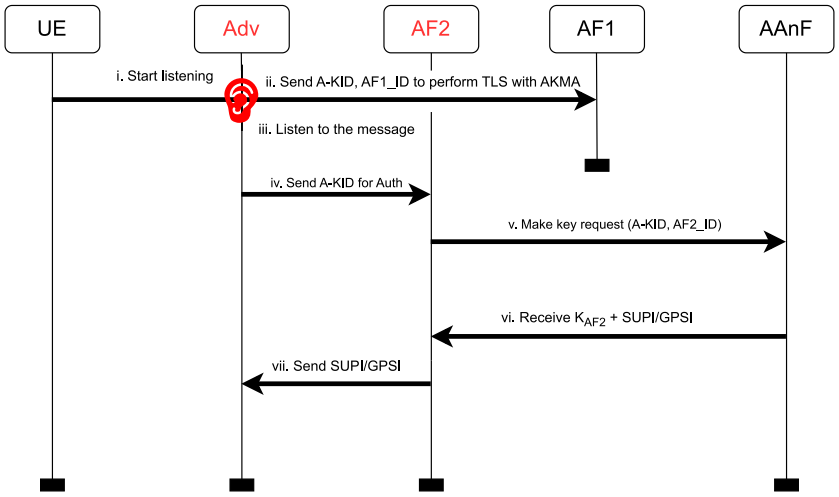

Fig. 6. Anonymity breaking attack

Needed conditions

- An UE want to connect to AF1 using TLS_DHE_PSK or TLS_PSK.
- The adversary is able to *listen* at the exchanges between UE and AF1.
- Adversary has access to AF2, an AF authorised to receive SUPI/GPSI.

Used weaknesses 1B 1C 1E

Feasibility T1 T2 T3

Modus Operandi

(i) Adversary starts listening to traffic.
(ii) UE initiates an Ua* connection to AF1 using TLS_DHE_PSK or TLS_PSK.
(iii) The adversary *listen* their first message.
(iv) The adversary sends the A-KID to AF2.
(v) AF2 sends a key request to the HPLMN (using A-KID and AF2-ID), asking for the SUPI/GPSI and for all additional personal information that it is allowed.
(vi) AF2 receives the desired information from the AAnF.
(vii) AF2 sends back these information to the adversary.

Remarks

- This attack it totally independent of the capabilities of AF1, meaning that it only depends on which personal information AF2 is allowed to ask the HPLMN.
- Here, the attack is presented for a fixed AF1, but as AF1-ID can be found in the first message of the UE, the adversary could listen to traffic and identify Ua* connections that pass through.
- As there exists AFs like MECs that have the additional property that UEs connect to those AFs only if they are in a very specific localisation, attacking those AFs can be seen as SUPI/GPSI catchers. In fact, this attack that transform the A-KID into the SUPI/GPSI is somewhat similar in idea to having a method that transform GUTI into SUPI.

Unlike previous attacks, this one is solely based on design weaknesses. This therefore makes it a "feature" of AKMA. In fact, due to its efficiency, simplicity and complex mitigations, this attack is the one that worries us the most, as it will likely not be fixed. Thus, as its scope is seemingly immense, we believe that it will be widely used.

5.5 Giving T3 the same potency as T2 using roaming LI

Like every part of 5G, AKMA must comply with the Lawful Interception (LI) requirements, detailed in [3]. The AKMA LI architecture and targets are available in section 7.15.3 of [7]. In case of roaming, the conclusions are available at the end of [9]. Note that they lack standardisation, meaning this will most likely result in different protocols being used. They could be incompatible with each other and also hold dangerous vulnerabilities.

On that note, we now detail why, for LI reasons, sharing the $\mathsf{K}_{\mathsf{AKMA}}$ with the VPLMN where the UE is roaming enables AFs inside this VPLMN to bypass the HNEF for any key request. This gives **T3** the same potency as **T2**. Note that this scenario is a real possibility due to the lack of standardisation of LI roaming architecture and objects.

Used weaknesses 1A 2C 2D

Modus Operandi If, as proposed in some LI roaming solutions[8], we send $\mathsf{K}_{\mathsf{AKMA}}$, SUPI and A-KID to the VPLMN and stored them in the VAAnF[9], we enable AFs inside the VPLMN to query for K_{AF} not only via the HNEF but also via the VAAnF. Indeed, for the VAAnF, a $\mathsf{K}_{\mathsf{AKMA}}$ is identified with the received A-KID, meaning that AKMA is enabled. It therefore derive K_{AF} and provides it to the AF. This also works because the VAAnF does not check the HNI in the A-KID. The malicious AF may not know in which AAnF the $\mathsf{K}_{\mathsf{AKMA}}$ is stored, but it just can try all AAnFs.

Remark In some specific cases (if using the RID inside the A-KID, the VNEF is able to find in which VAAnF are the LI information stored), an external AF could also obtain K_{AF} by making a key request to the VNEF. This would thus break the trust non-transitivity, as external AF need not to be trusted by the HPLMN to use AKMA.

6 Proposed Mitigations & Recommendations

We now discuss on how we can fix AKMA security weaknesses in order to prevent the aforementioned attacks and vulnerabilities. We propose to fix all weaknesses by underspecification and we also propose a solution in order to fix weakness **1E**. Our proposed solutions try to make the least changes possible to AKMA, therefore increasing the odds of implementations. All other weaknesses by design are necessary consequences of desired AKMA properties can therefore not be mitigated.

We additionally provide recommendations on how to use AKMA in order to mitigate its security risks and ensure its secure usage.

6.1 Mitigations

2A. The AAnF should check that the send AF-ID corresponds to the sending AF too. As specified in [4], there are three different mechanisms for authorisation for 5G SBA:

- Using OAuth, where a token must be acquired in order to request services from other NFs. In this case, the AAnF should check that the FQDN in the AF-ID corresponds to one of the tokens.
- Using TLS certificates. Similarly to before, we can check that the FQDN of the certificate is in fact the one inside AF-ID.
- Using "trusted network", meaning that the NFs accept anything from inside the network. In order to be secure, this type of authorisation often relies on a secure environment, meaning that all internal AFs should be known. It may therefore be possible to check that the IP address of the asking AF and FQDN in the AF-ID matches, using a whitelist.

[8] 6.5, 6.8, 6.10, 6.11 and 6.12 of [9]

[9] This is logical as AAnFs are designed to store $\mathsf{K}_{\mathsf{AKMA}}$ and LI systems do not interact with each others.

2B. The AKMA initiation protocol should be more thoroughly specified on its limitations. We think 3GPP should provide detailed specifications like it did for Ua*. A crucial point to consider is to know whether or not AFs can indicate inside their AKMA initiation message a FQDN used do derived K_{AF} and what are the limitations. (We recommend that the AF be authenticated by the UE).

2C. The UE-Serving PLMN IDentifier (UE SN-ID) used during the latest primary authentication should be sent to the HAAnF by the AUSF and should be checked in case of a key request. This means that the HAAnF should ensure that the only VPLMN able to use AKMA is the UE-Serving PLMN. This could be done using the HNEF. When receiving a key request from an AF that is internal in a VPLMN, the HNEF forwards the AF's VPLMN ID (noted AF SN-ID) to the AAnF which would then check if it agrees with UE SN-ID.
This is an easy fix as all connections between NEF and AFs shall be authenticated with TLS certificates, following section 12 of [4].

2D. The HNI inside the A-KID should be checked by both the AAnF and NEF in order for them to refuse all A-KIDs that are not from the AAnF's or NEF's network.

1E. This weakness requires changing the design of the AKMA mechanism. Our best idea is to make use of the fact that AKMA requires that we previously finished a primary authentication, meaning that there exists a way to create a secure tunnel between the UE and the HPLMN (this can be done by using the shared secret between UE and AAnF that is $\mathsf{K}_{\mathsf{AKMA}}$). Therefore, we propose that before sending the A-KID to an AF, UE send *securely* to the AAnF the AF's FQDN and waits for the acknowledgement. Note that the UE must know the AF's FQDN as it is necessary to compute K_{AF}. We then follow the AKMA key request up until the AAnF has to derive K_{AF}. At that time, the AAnF checks that the AF-ID agrees with the FQDN sent by the UE. Our solution can be criticised because it increases both the Round-Trip Time (RTT) and the complexity of the implementation on the UE and HPLMN side. However, it guarantees that only AFs selected by the UE can use AKMA.
In any case, this security weakness requires greater attention.

We represent in Fig. 7 all our proposed modifications to AKMA key request protocol.

Finally, we are strongly positive about the necessity for 3GPP to write a proper AKMA LI roaming standard.

6.2 Recommendations

– In general, if the AF is authenticated with a certificate, we recommend using AKMA as a shared key-based UE authentication with certificate-based AF authentication mechanism, as detailed in B.1.2 of [6]. Indeed, none of the aforementioned attacks are possible when using such protocols. Additionally,

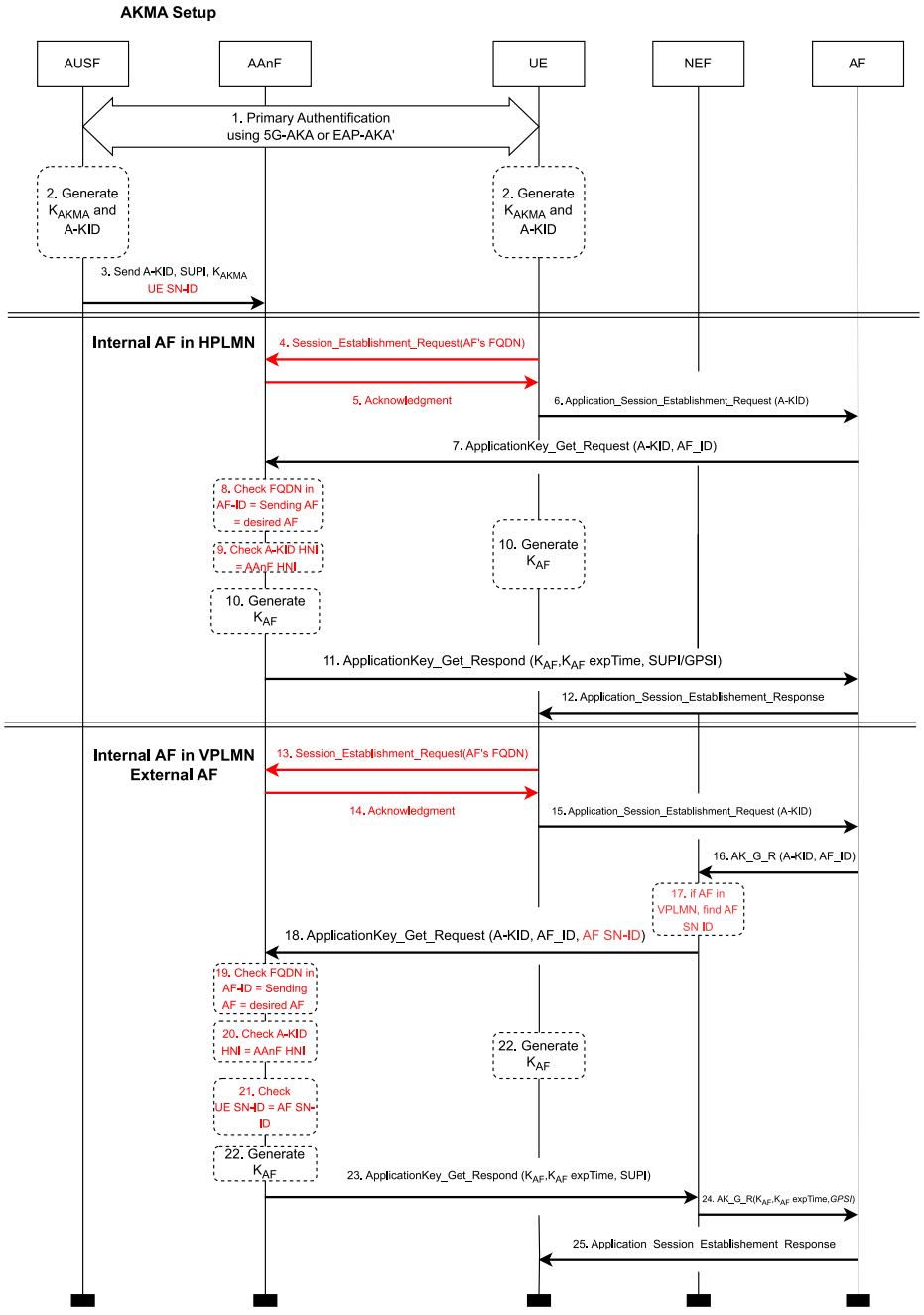

Fig. 7. Our modified version of AKMA (simplified version)

we think that TLS_PSK should not be used, unless for extremely low power devises and when it is an absolute necessity.
- A 5G SA network should *not* rely on trusted network as its authorisation mechanisms for its SBA requests. OAuth token or TLS certificates should *strongly* be preferred.
- We recommend not to give MECs the right to receive SUPI/GPSI via AKMA, especially for those that are in insecure environment.
- Additionally, we think that all MECs should be treated as external AFs and the only internal AFs should be physically inside the core network.
- Until AKMA LI roaming is properly standardized and its security is thoroughly studied, we recommend to not use roaming AKMA.

7 Conclusion

Throughout this paper, we demonstrated that, in its current state, the AKMA mechanism strongly lacks resilience, for the reasons that the compromising of any AF in any 5G network has an impact on the security of potentially *all* connections between AFs and UEs. This impact ranges from catastrophic, with wiretap and MitM capabilities that simply destroy every security of Ua*, to the problematic with the possibility to identify every UE that utilises AKMA as a shared key-based mutual authentication TLS. The latter is especially worrisome because of its easy access. Our work exemplifies how important it is to think carefully of the ways protocols can be used and how slightly modifying there usage domain may yield serious security issues. More generally, based on the scheduled deployment of 5G SA, we think that AKMA does not currently offer an acceptable security paradigm and should therefore be handled with great care.

Further work Our results follow from 3GPP technical specifications and our attacks should therefore be valid on any 5G Core. Sadly, we were unable to verify that last point as AKMA, roaming and LI are not available on open source core and that their implementation was beyond our capacity. This is nonetheless an issue that we look forward to investigating, together with the coming fixes of AKMA by 3GPP. Another point of interest is the adaptation of AKMA to Object Security for Constrained RESTful Environments (OSCORE) that could hold similar or new weaknesses.

References

1. 3GPP: Generic authentication architecture (gaa); access to network application functions using hypertext transfer protocol over transport layer security (https). Tech. Rep. 33.222 v17.2, 3rd Generation Partnership Project (3GPP) (Jun 2022)
2. 3GPP: Generic authentication architecture (gaa); generic bootstrapping architecture (gba). Tech. Rep. 33.220 v17.4, 3rd Generation Partnership Project (3GPP) (Sep 2022)

3. 3GPP: Lawful interception requirements. Tech. Rep. 33.126 v18.0, 3rd Generation Partnership Project (3GPP) (Aug 2022)
4. 3GPP: Security architecture and procedures for 5g system. Tech. Rep. 33.501 v18.0, 3rd Generation Partnership Project (3GPP) (Dec 2022)
5. 3GPP: System architecture for the 5g system (5gs). Tech. Rep. 23.501 v18.0, 3rd Generation Partnership Project (3GPP) (Dec 2022)
6. 3GPP: Authentication and key management for applications (akma) based on 3gpp credentials in the 5g system (5gs). Tech. Rep. 33.535 v18.0, 3rd Generation Partnership Project (3GPP) (Jun 2023)
7. 3GPP: Lawful interception (li) architecture and functions. Tech. Rep. 33.127 v18.3, 3rd Generation Partnership Project (3GPP) (Mar 2023)
8. 3GPP: Mec(23) 000134r4 ls to 3gpp sa3 on requiring accurate transmission of derived. Tech. Rep. S3-232430, 3rd Generation Partnership Project (3GPP) (May 2023)
9. 3GPP: Study on authentication and key management for applications phase 2. Tech. Rep. 33.737 v18.0, 3rd Generation Partnership Project (3GPP) (Jun 2023)
10. Akman, G., Ginzboorg, P., Damir, M.T., Niemi, V.: Privacy-enhanced akma for multi-access edge computing mobility. Computers **12**(1), 2 (2022)
11. Björck, F., Henkel, M., Stirna, J., Zdravkovic, J.: Cyber resilience–fundamentals for a definition. In: New Contributions in Information Systems and Technologies: Volume 1. pp. 311–316. Springer (2015)
12. ETSI: Mec security; status of standards support and future evolutions. Tech. Rep. 979109262041, European Telecommunications Standards Institute (ETSI) (Sep 2022)
13. ETSI: Multi-access edge computing (mec); framework and reference architecture. Tech. Rep. GS MEC 003 v3.1.1, European Telecommunications Standards Institute (ETSI) (Mar 2022)
14. ETSI: Multi-access edge computing (mec); phase 2: Use cases and requirements. Tech. Rep. GS MEC 002 v2.2.1, European Telecommunications Standards Institute (ETSI) (Jan 2022)
15. Group, N.W.: The transport layer security (tls) protocol, version 1.2. Tech. Rep. 5246, RFC Editor (2008)
16. Huang, X., Yoshizawa, T., Baskaran, S.: Authentication mechanisms in the 5g system. Journal of ICT Standardization **9**(2), 61–78 (2021)
17. IETF: The transport layer security (tls) protocol, version 1.3. Tech. Rep. 8446, RFC Editor (2018)
18. Li, Y., Schäge, S., Yang, Z., Kohlar, F., Schwenk, J.: On the security of the pre-shared key ciphersuites of tls. In: Public-Key Cryptography–PKC 2014: 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26-28, 2014. Proceedings 17. pp. 669–684. Springer (2014)
19. Yang, T., Wang, S., Zhan, B., Zhan, N., Li, J., Xiang, S., Xiang, Z., Mao, B.: Formal analysis of 5g authentication and key management for applications (akma). Journal of Systems Architecture **126**, 102478 (2022)