University of Padova

Department of Mathematics "Tullio Levi-Civita"

Master Thesis in Cybersecurity

BARON:
Base-Station Authentication Through Core Network for Mobility Management in 5G Networks

Supervisor
Prof. Mauro Conti
University of Padova

Co-supervisor
Prof. Alessandro Brighente
University of Padova

Master Candidate
Alessandro Lotto

Student ID
2019184

Academic Year
2022-2023

"L'esperienza è come argilla,
Passano anche i guai:
Quello che resta ce l'hai nelle mani
Poi conta che forma ci dai"
— MezzoSangue

Abstract

The Fifth-Generation (5G) cellular communication networks are nowadays being deployed on applications that go beyond mobile phone devices, including vehicular networks and industry automation, for instance. Despite their increasing popularity, 5G networks, as defined by the Third Generation Partnership Project (3GPP), have been shown to be vulnerable against *Fake Base Station* (FBS) attacks. An adversary carrying out an FBS attack emulates a target legitimate base station by setting up a *rogue base station*, which is assumed to have the same capabilities as the legitimate base stations. This enables the adversary to control the connection of any user equipment that (inadvertently) connects with the rogue base station. As a consequence of a successful FBS attack, the adversary not only can gather sensitive information belonging to the user, but also affect the reliability of the network itself.

Despite there is a large body of work focused on the development of tools to detect FBSs, these solutions do not actually prevent the FBS attack success, as they do not address the vulnerability cause of such an attack. Therefore, the user equipment will continue to remain vulnerable to an FBS attack. On the other hand, solutions in the literature that are specifically designed to address the FBS attack majorly consist on protecting the broadcast messages transmitted by the base stations. This can be achieved through integrity protection or digital signature mechanisms. However, solutions following these two approaches can be made ineffective, and may lead to a possible increase of manufacturing costs.

In this thesis, we present BARON, a new base station authentication methodology to enable the user equipment to determine whether a target base station that it is connecting to is legitimate or rogue. BARON accomplishes its objective by ensuring that the user receives an *authentication token* from the target base station which can be computed only by a legitimate and trusted entity. As a consequence, receiving such an authentication token from a base station ensures its legitimacy. BARON does not require any additional infrastructure for its deployment, making it being fully backward compatible with the current standard 5G networks. We evaluate BARON through extensive experiments on the *handover process* between base stations in 5G networks. Our experimental results show that BARON introduces an overhead of less than 1% during handover completion, which is 10000× lower than the overhead reported by a state-of-the-art solution, making its adoption practical. BARON is also effective in thwarting an FBS attack and quickly recovering connection to a legitimate base station.

Contents

Listing of figures

Listing of tables

Listing of acronyms

3GPP . . . . . Third Generation Partnership Project

4G-LTE . . . . . Fourth Generation Long Term Evolution

5G . . . . . Fifth Generation

ACK . . . . . Acknowledgement

AKA . . . . . Authentication and Key Agreement

AMF . . . . . Access and Mobility Function

AT . . . . . Authentication Token

AUSF . . . . . Authentication Server Function

AV . . . . . Authentication Vector

BS . . . . . Base Station

CN . . . . . Core Network

DoS . . . . . Denial of Service

DAPS . . . . . Dual Active Protocol Stack

EAP-AKA . . . . . Extensible Authentication Protocol - AKA

FBS . . . . . Fake Base Station

ID . . . . . Identifier

IoT . . . . . Internet of Things

MAC . . . . . Message Authentication Code

MIB . . . . . Master Information Block

MitM . . . . . Man in the Middle

mmWave . . . . . Millimeter Wave

MR . . . . . Measurement Report

NAS . . . . . Non-Access Stratum

NFV . . . . . Network Function Virtualization

PCI . . . . . Physical Cell Identity

PKG . . . . . Public Key Generator

PKI . . . . . Public Key Infrastructure

PRACH . . . . . Random Access Channel Preamble

RA . . . . . Random Access

RACH . . . . . Random Access Channel

RAN . . . . . Radio Access Network

RAT . . . . . Radio Access Technology

rBS . . . . . Rogue Base Station

RLF . . . . . Radio Link Failure

RNTI . . . . . Radio Network Temporary Identifier

RRC . . . . . Radio Resource Control

RT . . . . . Re-connection Token

sBS . . . . . Serving Base Station

SEAF . . . . . Security Anchor Function

SFN . . . . . System Frame Number

SIB . . . . . Secondary Information Block

SUCI . . . . . Subscription Concealed Identifier

SUPI . . . . . Subscriber Permanent Identifier

tBS . . . . . Target Base Station

UE . . . . . User Equipment

UDM . . . . . Unified Data Management

USIM . . . . . Universal Subscriber Identity Module

1 Introduction

The increased demand for services on mobile devices that require high throughput and low latency (*e.g.*, video calling) has guided the rapid evolution of cellular networks. The current state-of-the-art in cellular communications is the *Fifth Generation* (5G) technology [2], which provides a significantly improved throughput over previous technologies, such as the Fourth Generation Long-Term Evolution (4G-LTE). From the physical layer perspective, the 5G technology employs also the millimeter-wave (mmWave) spectrum. While on one side this increases the available bandwidth and reduce the transmission latency, on the other, the mmWave spectrum offers a lower penetration ability compared to frequencies already in use, and consequently, it offers a shorter transmission range. As a result, 5G networks require a more dense Base Station (BS) deployment [3].

Although many countries have already deployed functioning 5G networks, the technology is still under continuous development in order to meet higher standards for performance and security. Besides, in recent years the cellular communication technologies are not only confined for mobile communication purposes anymore, rather they have also been increasingly adopted in other fields of applications such as vehicular networks [4], real-time medical procedures [5] and industrial automation [6, 7]. These new applications, however, may create novel and additional attack surfaces vulnerable to exploitation by an adversary [8]. Therefore, it is essential to develop solutions for cellular communications that can ensure high levels of security, confidentiality, and reliability before they become ubiquitous in other applications. Furthermore, due to the ultra-low transmission latency requirement in 5G networks, security solutions must be

efficient so to introduce minimum computational and time overhead.

Mobility management is one of the most critical aspects of cellular communications [9], as it deals with moving users ensuring the stability of their connection and access to services. It refers to the set of all those procedures that are needed to manage the cellular service for users that move from one location to another. Such a connection handling is implemented through a set of protocols used to signal the movement of the user between nodes of the network [9]. Specifically, the *handover procedure* ensures that users' mobile devices have the ability to switch between BSs with (almost) no interruption in the connection and service [10]. Although the dense BS deployment in 5G networks allows to reduce transmission times between the BSs and users, it also results in more frequent handovers compared to previous communication technologies, including 4G-LTE. Consequently, ensuring the security and efficiency of the handover procedure must be a major concern for developers and researchers in order to meet the desired maturity of the technology.

Ever since the introduction of the first standard for cellular networks, the technology has been shown to be vulnerable to *Fake Base Station* (FBS) attacks [11]. An adversary carrying out an FBS attack sets up a *rogue Base Station* (rBS) that emulates a legitimate BS. This can deceive a User Equipment (UE) (*e.g.*, mobile phone) into connecting with the rBS, while believing it to be legitimate. Following connection with the rBS, the UE does not have the ability to restore connection to a genuine BS without rebooting the device or going out of the transmission range of the rBS. As a result of a successful FBS attack, the adversary controls the user connection and might use this as a first-step towards carrying out more severe attacks, including Denial-of-Service (DoS), Man-in-the-Middle (MitM), and bidding-down [11, 12] attacks, thus affecting the reliability of the network [13].

Despite awareness of the vulnerability described above, defenses against FBS attacks have primarily focused on mechanisms and tools to detect FBSs [12]. These, however, do not impede an adversary from successfully carrying out an FBS attack as they do not prevent the UE from connecting to the rBS, neither allow the UE to detect online the non-legitimacy of the BS it is connecting to. In fact, FBS detection mechanisms use and correlate the collected data from the network to detect possible inconsistencies due to the presence of rBSs. Such an analysis is carried out offline, meaning that is not part of the mobility management framework and does not apply during connection to a BS or before/during handover. For what concerns solutions that are specifically designed to address the FBS attack vulnerability, the most common approach consists on protecting the broadcast messages that are transmitted by the BSs. These messages contain specific information related to the transmitting BS, and are eventually used

by the UEs to connect with the BS. Protecting these broadcast messages can be achieved by implementing message integrity protection or digital signature schemes. The latter approach represents, at present, the state-of-the-art defence approach against FBS attacks. In a sequence of research papers, authors of [12, 14] propose two digital signature-based mechanisms to protect messages broadcast by BSs. Results in [12], in particular, manage to reduce the introduced computation overhead up to 31% compared to other related works ([14, 15, 16]) that make use of asymmetric cryptography. However, both proposals in [12, 14] may be vulnerable to replay attacks, making them ineffective and preventing their deployment. Besides, they also require to introduce a Public Key Infrastructure (PKI) or a similar entity in the core network for key management, which may result in increased manufacturing and set-up costs.

The severe impact that FBS attacks may have in a scenario with frequent handovers due to dense BS deployment, the lack of an extensive research in solutions specifically designed to address such an attack vulnerability, as well as the possible issues related to replay attacks vulnerability and manufacturing costs for the state-of-the-art approach, are all motivations that moved us to design a new defence mechanism following a different approach. Hence, in this thesis we present BARON[1], a new BS authentication methodology for secure initial access and handover in 5G networks specifically designed to defend against FBS attacks. BARON enables the UE to (i) determine whether the BS it is connecting to is legitimate or not, and (ii) efficiently recover a legitimate connection when subject to an FBS attack. We carry out extensive experiments to evaluate the performance of BARON in terms of time overhead introduced during handover, and efficiency in recovering a legitimate connection in case of an FBS attack. Our experiments reveal that the additional overhead induced by BARON is less than 1% of the total time required for handover completion, and is 10000× lower than the additional overhead reported recently in [12]. Further, during an FBS attack, BARON is able to effectively recover connection to a legitimate BS in a time that is of the same order of magnitude as the time required for handover completion.

The rest of this thesis is organized as follows. Chapter 2 gives background on those aspects of 5G networks that are necessary for a complete understanding of the FBS attack and BARON frameworks. Chapter 3 defines the threat model considered, and describes the steps through which an FBS attack develops. Chapter 4 presents the details of BARON, also giving a formal proof of its security against the considered adversary model. Chapter 5 provides the details of the setup of our experiments, and presents the corresponding results. Chapter 6 discusses more in detail the related works, and Chapter 7 concludes the thesis.

[1] BARON: Base-station Authentication thRough cOre Network

2

5G Networks Background

In this chapter, we provide the necessary background for a complete understanding of the FBS attack and BARON authentication logic. We will then present the architecture and the main components of 5G networks as defined by the *Third Generation Partnership Project* (3GPP), a collaboration between multiple telecommunications standards organizations that develop and maintain standards for mobile telecommunications. Afterwards, we discuss how the current 5G standard provides security in the communications. Then, we introduce the key concepts for the Random Access Channel (RACH) procedure, which allows the UE and BSs to agree on the physical parameters of the transmission to begin the effective communication. Finally, we provide an overview of the initial access and the different possibilities for the handover process.

2.1 5G Network Architecture

Standard 5G networks, as defined by the 3GPP, consist of multiple entities with dedicated functions and hierarchically organized. Entities of upper layers provide then functions and services for entities located in the lower layers, and vice-versa [17]. We can identify two main blocks that compose a 5G network:

- **Radio Access Network (RAN)**. This is the part of a 5G network responsible for connecting the UE to the core network through BSs using a wireless interface. The RAN also comprises of other secondary components such as antennas or equipment to man-

age the radio frequency signals [17, 18]. The RAN is a critical component of 5G networks, both from the functional and security point of view, as it provides the wireless communication and ultra-low latency with high-speed data rates.

- **Core Network (CN)**. The CN comprises of all entities excluding the RAN, and it is responsible for data and connection management [17]. The CN provides security and storage services, as well as advanced applications such as multimedia streaming and augmented reality. As the CN infrastructure of a 5G network is designed for being scalable, it is usually defined thanks to a Network Function Virtualization (NFV) interface, meaning that the architecture and many functions of the network can be software implemented and defined [19]. Such a network design approach allows to adapt the network topology and resources according to the current needs, while at the same time it also simplifies the introduction of new services [19].

Fig. 2.1 shows an example of a simplified 3GPP 5G network architecture, with the main entities involved. In what follows, we describe the roles and functions of these major entities [1, 17, 20]:

- **User Equipment (UE)**. The UE refers to the device used by the end-users in order to access the services that are provided by the network. The UE may be a smartphone or an Internet-of-Things (IoT) device with a mobile broadband chip conforming to the 5G Standard.
  The UE is provided with an integrated module consisting of the Universal Subscriber Identity Module (USIM) associated to a 15-digit unique Subscriber Permanent Identifier (SUPI). The latter is used for authentication of the UE when it initiates a connection with the CN. The USIM stores the security keys (symmetric and asymmetric) for authentication and encryption, used to protect communications.

- **Base Station (BS)**. A BS, technically called as *gNB* in the 5G context, is responsible for establishing and maintaining wireless communications with the UEs and connect them to the CN. As shown in Fig. 2.1, each BS covers a certain area within its range. Ranges of two neighboring BSs may overlap with each other. A handover procedure will be required when the UE moves out of the range of one BS and falls within the range of another one. A BS communicates with entities in the CN through a secure channel, *i.e.*, authenticated and encrypted, using the "*N2 interface*". Besides, BSs may also directly communicate with each other using a secure "*Xn interface*".

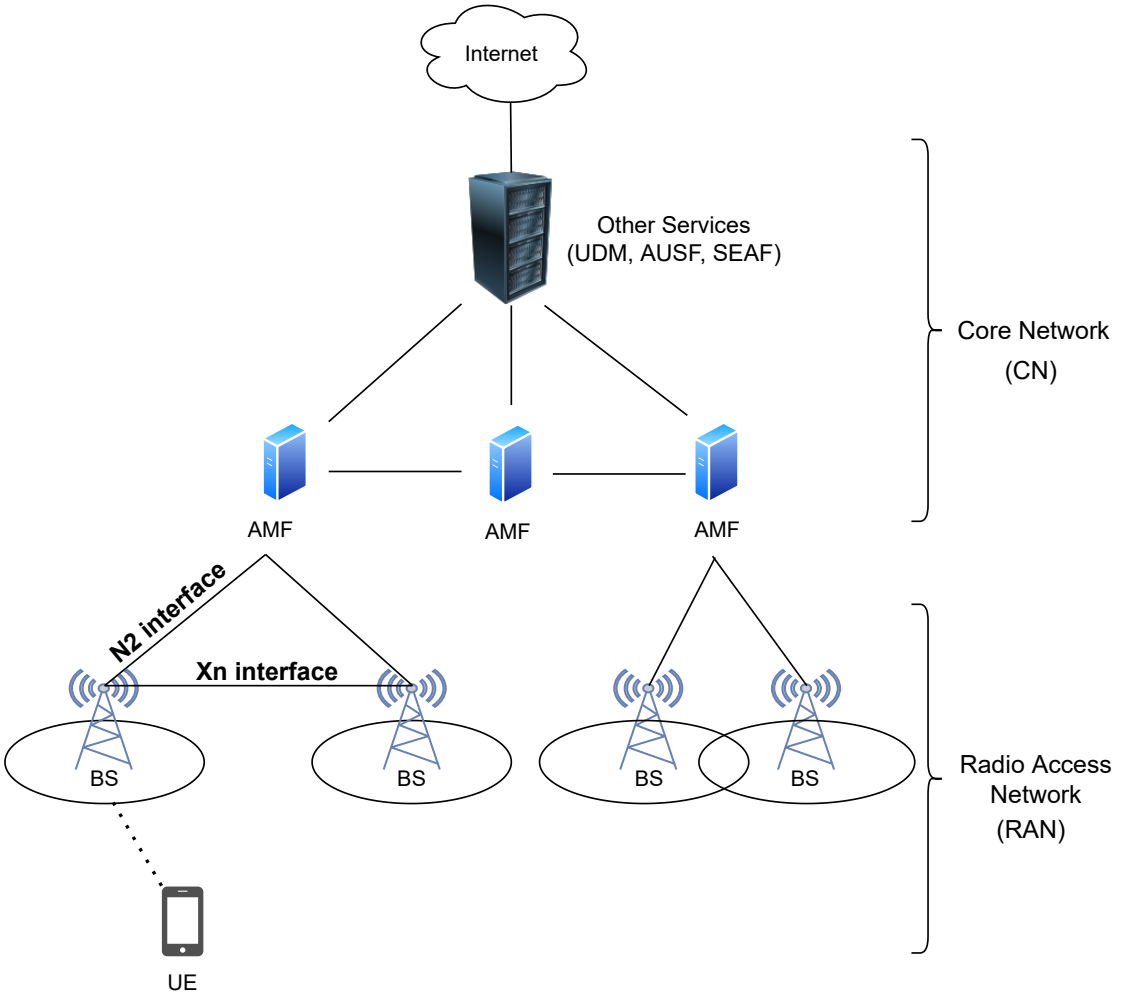


Figure 2.1: Example of simplified 3GPP 5G Network architecture.

The BSs broadcast the *Master Information Block* (MIB) and *Secondary Information Block* (SIB) messages multiple times a second. These special messages, often referred as beacons, contain the information needed to the UEs to facilitate the access to the BS itself. The MIB message contains the cell Identifier (ID), the Physical Cell Identity (PCI) and the System Frame Number (SFN). This information is used to identify the specific BS. The SIB messages contain information both related to the physical properties of the signal (*i.e.* frequency and time-stamp), and to mobility management purposes.

- **Access and Mobility Function (AMF).** The AMF is primarily responsible for mobility, location, connection and session, and security context management. Besides, it is

responsible to enforce the policy related to the network resources usage and quality of service maintenance.
As shown in Fig. 2, the AMF controls a set of neighboring BSs. The AMF can communicate with neighbouring AMFs and other CN entities.

- **Security Anchor Function (SEAF)**. The SEAF is usually located next to the AMF and acts as a proxy by offering security services. In particular, it is dedicated to ensure secure connections with external networks by enforcing access control and data filtering procedures defined according to the security policies.

- **Unified Data Management (UDM)**. The UDM is responsible for the access rights and authorization functions, thus it manages the process of user initial registration to the network. The UDM also plays a role in the manage of the policy and actions to enforce for maintaining the target quality of service.

- **Authentication Server Function (AUSF)**. The AUSF provides authentication and security support for 5G services. Thus, it is responsible to manage the encryption keys used for secure communications between the UEs and entities of the network.

To simplify the notation, we will now refer to the SEAF, UDM, and AUSF as a single entity called CN. We have chosen not to include the AMF in this aggregation due to its central role in the development of BARON.

2.2 Securing Communications in 5G Networks

We can use encryption and authentication mechanisms to protect messages exchanged between entities in the 5G network in order to achieve secure communication [1]. The 3GPP standard for 5G networks defines the *Authentication & Key Agreement* (AKA) protocol, and the *Extensible Authentication Protocol - AKA* (EAP-AKA). These two protocols follow a challenge-response procedure between the UE and the CN, and define a set of security procedures to provide [1, 20]:

- Mutual authentication between the UE and entities of the serving network;

- Message integrity and confidentiality;

- Computation of security parameters that will be used for subsequent procedures (*e.g.*, horizontal/vertical key derivation).

Starting from a (symmetric) master key shared between the UDM and the USIM, the UE goes through a set of challenge-response authentication and hierarchical key derivation processes. Upon completion of the AKA (or EAP-AKA) protocol, the UE builds a *chain of trust* with the serving network, and derives one or more keys for each entity in the CN, and for the current serving Base Station (sBS). The set of secret keys, security parameters, and employed encryption and authentication algorithms define the *UE security context*.

Thanks to the AKA and EAP-AKA procedures, it is possible to open multiple security contexts with just a single authentication procedure, thus allowing for movement from 3GPP networks to non-3GPP networks without the need to go through a new authentication process [20]. The overall authentication process can be divided into two phases [1, 20, 21]: *Initialization and User Identification*, and the proper *5G-AKA Protocol*.

2.2.1 Initialization and User Identification

As previously mentioned, the USIM stores the cryptographic keys for encryption and authentication purposes. In particular, the two most relevant keys are the PK_{CN}, which is the public asymmetric key of the serving network, and K_{CN}, a long-term symmetric key used as shared secret between the UE and CN. Besides, a sequence number SQN is stored to provide freshness for the exchanged messages [1, 21]. The CN, in turn, stores the same information associated with the SUPI of the UE.

The initialization protocol aims to identify the UE in order define which security primitives adopt for the authentication process. Once the serving network triggers authentication initialization, the UE sends a message containing the Subscription Concealed Identifier (SUCI) together with the identifier of the network ID_{CN}. The SUCI is the encrypted version of the SUPI, and is derived via asymmetric encryption by using PK_{CN} [1, 21]. The SUCI has security and privacy purposes, in order to avoid transmitting the SUPI in the clear, which in fact may lead to impersonation and location tracking threats. Upon reception of the message, the CN retrieves the SUPI that identifies the subscriber and its corresponding security parameters, thus chooses an authentication method, namely the AKA or EAP-AKA protocol.

2.2.2 The 5G-AKA Protocol

The AKA and EAP-AKA protocols are very similar to each other: they rely on the same challenge-response mechanisms and have the same usage of the secret keys. The main difference between the two is their flow and the key derivation functions, which have minor differences [21]. Therefore, we will only introduce the main aspects of the AKA protocol only. More detailed information about the specific implementations and flow of the two protocols can be found in the "3GPP Technical Specification for Security Architecture and Procedure" [1]. In the following chapters, we will assume the AKA protocol being completely secure, since providing proof of its security is out of the scope of this work.

Upon authentication request, the UDM generates an *Authentication Vector* (AV) defined as:

$$AV_{UDM} = (R, XRES^*, SUCI, AUTN, K_{SAUF}) \ ,$$

where:

- R is a random number that represents the challenge to the UE;
- $XRES^*$ is the expected response to the challenge;
- $AUTN = (C, MAC)$, where C is an encrypted version of SQN, and MAC (Message Authentication Code) is computed from a one-way keyed cryptographic hash function receiving as input the concatenation of R and the counter SQN;
- K_{AUSF} is a secret key directly computed by the UDM itself, and used for the secure channel that will be eventually established.

The UDM transmits the AV_{UDM} to the AUSF, which stores the $XRES^*$ and K_{AUSF} associated with the SUCI received. The AUSF builds, in turn, a new authentication vector:

$$AV_{AUSF} = (R, HXRES^*, AUTN) \ ,$$

and forwards it to the SEAF. The $HXRES^*$ is the hash value computed from $XRES^*$ using the SHA-256 hashing algorithm [1]. Besides, the AUSF computes the key K_{SEAF} from K_{AUSF}. The SEAF will then forward to the UE the following authentication vector:

$$AV_{SEAF} = (R, AUTN, ngKSI) \ ,$$

where *ngKSI* is a security parameter used to derive K_{AMF} for secure communication between the UE and the AMF in case of successful authentication. Upon reception of the authentication vector, the UE verifies the $AUTN$ value and checks for message freshness. In case of positive verification, the UE computes the keys K_{AUSF} and K_{SEAF}, and the challenge response RES^*. The UE returns the challenge response back to the SEAF, which in turn computes $HRES^*$ and compares it to $HXRES^*$. In case the two values match, the SEAF considers the authentication as successful, and forwards the challenge response RES^* to the AUSF. This latter verifies in turn the received value: it compares RES^* with the $XRES^*$ previously stored, and considers the authentication successful if the two values are the same. At this point, the AUSF first inform the UDM about successful authentication, then transmits the K_{SEAF} to the SEAF. This key becomes then the anchor key from which to compute the K_{AMF} [1, 21].

The overall AKA procedure proceeds according to the following steps (Fig. 2.2):

5G-AKA protocol (Fig. 2.2):

① UE → SEAF: (SUCI, ID_{CN})

- SUCI = $AE_{PK_{CN}}(SUPI)$ | $AE_k(\cdot)$ = asymmetric encryption, key k
- ID_{CN} = CN identifier

② SEAF → AUSF: (`UE_Authentication_Request`, SUCI)

③ AUSF → UDM: (`UE_Authentication_Request`, SUCI)

④ UDM:

- SUPI = $AE^{-1}_{PRK_{CN}}(SUCI)$ | PRK_{CN} = private asymmetric key, $AE^{-1}_k(\cdot)$ = asymmetric decryption, key k
- Authentication method selection

⑤ UDM → AUSF: $AV_{UDM} = (R, XRES^*, SUCI, AUTN, K_{AUSF})$

- R = random number
- $XRES^*$ = challenge expected response
- $AUTN = (C, MAC)$ | C = encryption of SQN, MAC = Message Authentication Code

- K_{AUSF} = secret key

⑥ AUSF: computes K_{SEAF} from K_{AUSF}

⑦ AUSF → SEAF: $AV_{AUSF} = (R, HXRES^*, AUTN)$

- $HXRES^* = \text{SHA-256}\,(XRES^*)$

⑧ SEAF → UE: $AV_{SEAF} = (R, AUTN, ngKSI)$

- $ngKSI$ = security parameter for computation of K_{AMF}

⑨ UE: verifies $AUTN$, and computes k_{AUSF}, k_{SEAF} if positive check

⑩ UE → SEAF: (RES^*)

- RES^* = challenge response

⑪ SEAF: verifies if $\text{SHA-256}\,(RES^*) = HXRES^*$

⑫ SEAF → AUSF: (RES^*)

⑬ AUSF: verifies if $RES^* = XRES^*$

⑭ AUSF → UDM: (`Successful Authentication`)

⑮ AUSF → SEAF: (K_{SEAF})

- K_{SEAF} = secret key computed from K_{AUSF}

In addition to the initial authentication procedure, the AKA protocol also defines the details for vertical and/or horizontal keys derivation. Fig. 2.3 presents a graphical representation of the hierarchy and dependencies of the secret keys that are derived through the proper procedures.

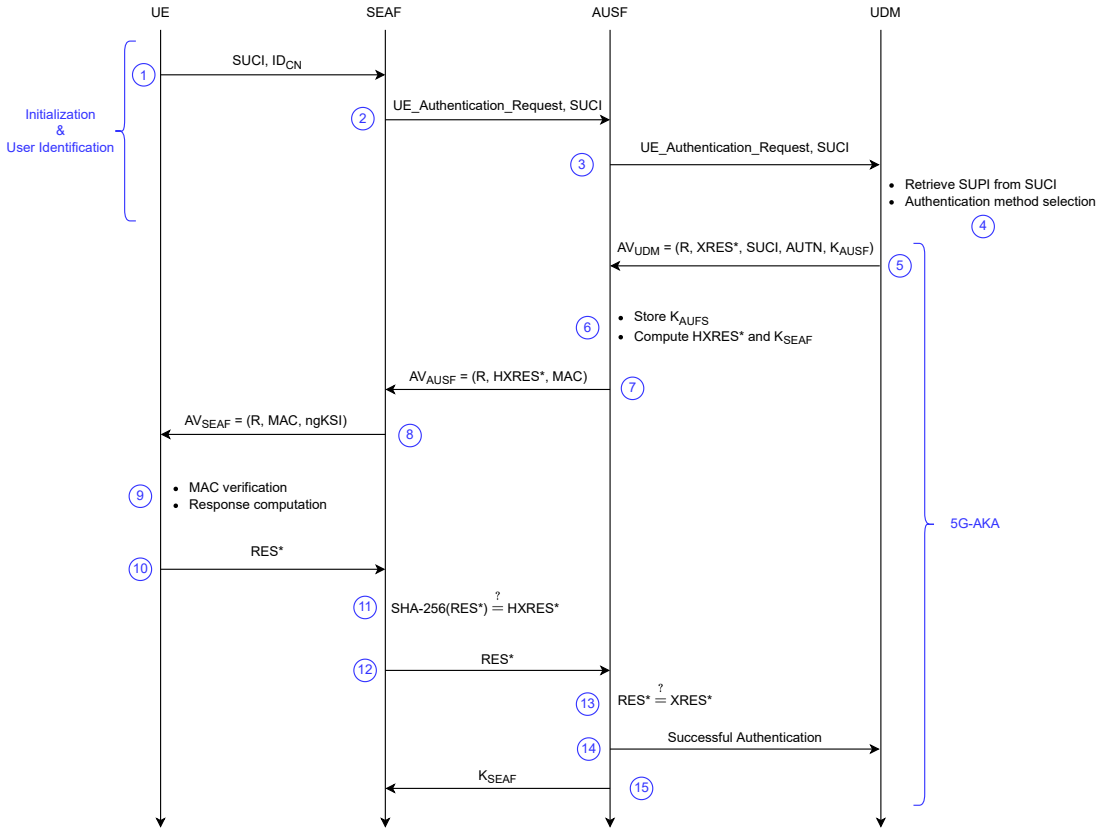


Figure 2.2: Call flow of the 5G-AKA authentication procedure.

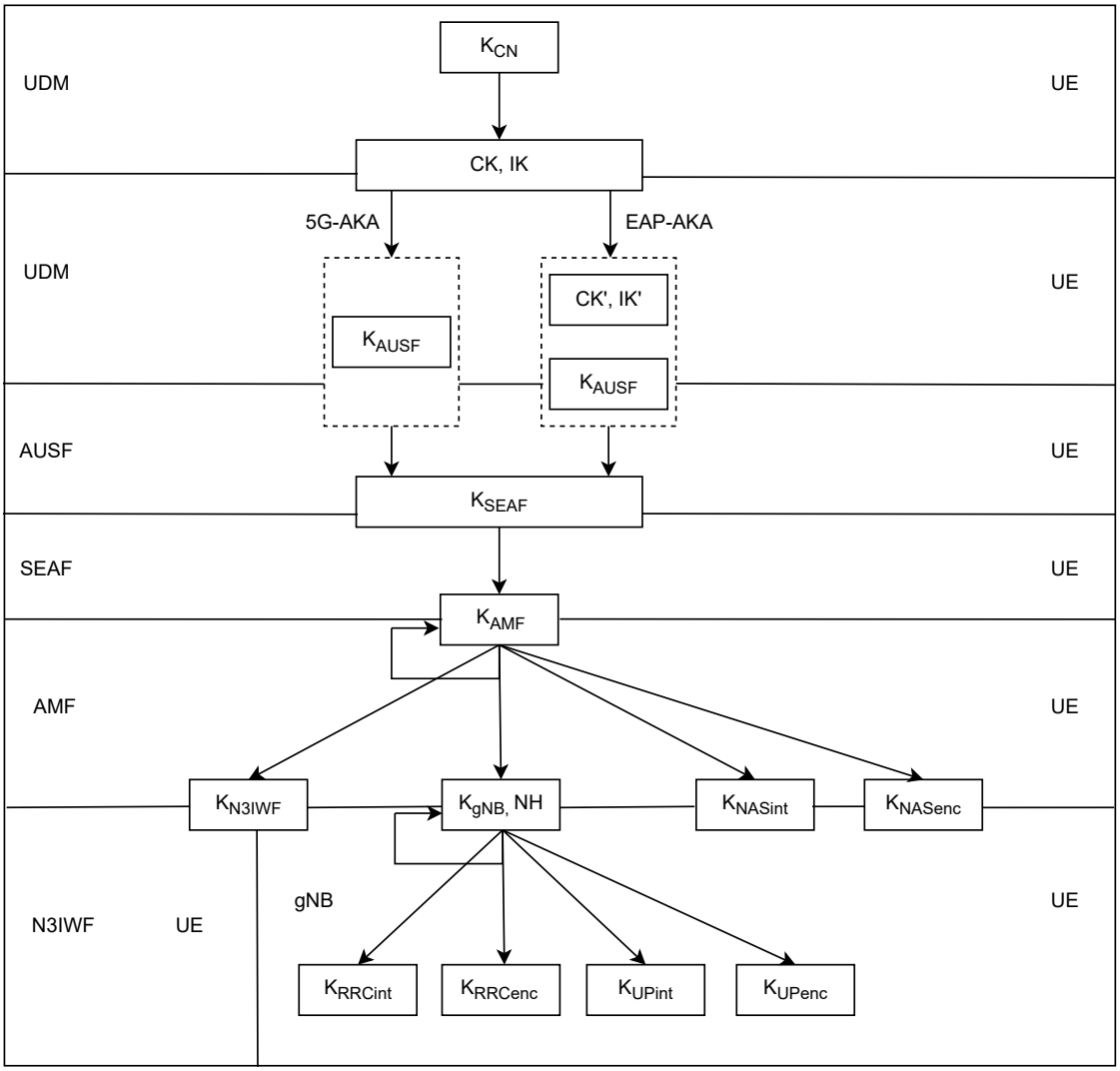


Figure 2.3: Authentication and encryption keys hierarchy in 3GPP 5G networks [1].

2.3 The Random Access Channel Procedure

The *Random Access Channel* (RACH) procedure is the protocol that in 5G and 4G-LTE networks allows for time synchronization between a transmitter and a receiver when they need to establish a connection. Through the RACH procedure, UEs establish a connection with the network even if they are not already synchronized or registered [22]. Besides, the procedure is designed in such a way that it ensures that multiple UEs can access the network simultaneously.

The RACH procedure is needed in order to establish an up-link channel, where the UE is the transmitter and the BS is the receiver [22, 23]. Indeed, while for the down-link channel establishment (the BS is the transmitter) we can adopt a broadcast strategy for which the BS periodically transmits a synchronization signal, the same solution would not be efficient for the up-link case. In this latter case, in fact, the synchronization process should happen only when necessary, and should be dedicated for a specific UE. Besides, a broadcast strategy would cause a waste of energy and increase the interference in the wireless channel. For the following scenarios, a RACH procedure is necessary in order to establish a connection [22, 23]:

- **Initial access from the `RRC_IDLE` status**. When being in the `RRC_IDLE` status, the UE is not assigned to any radio resources, thus it has no ongoing connections. *Initial access* refers then to the scenario in which the UE first connect to the network for the current session. It must not be confused with *initial registration*, in which the UE is registered to the network for the first time. An example of initial access is when the user switches on its device.

- **Radio Resource Control (RRC) Connection Re-establishment procedure**. In this case the UE loses an ongoing connection due to various reasons, and need to re-establish a connection. Since the connection must be re-established, time and frequency synchronization must firstly agreed.

- **Handover**. In this case the UE connects to a different BS or cell, which may have different time-stamp and frequency parameters. Thus a proper synchronization is required before establishing a valid connection.

- **Up-link and down-link data arrival in the `RRC_CONNECTED` status is not synchronized**. In this case there could be issues in the ongoing connection causing delayed acknowledgment, buffer overflow/underflow, reduced throughput, low quality of service.

It is then necessary to re-synchronize the connection by re-establishing the up-link and down-link.

- **Beam failure recovery**. In this case the UE asks the network for different resources as the previously dedicated one seems not to offer a good quality of connection. Since new resources are needed, a new synchronization must be established between sender and receiver.

Generally, the RACH procedure develops according the following four steps [18, 23]:

1. **Random Access (RA) preamble**. The UE selects one of the 64 available *Random Access Channel Preamble* (PRACH) signatures in the cell, and transmits it to the BS. This signals the request for connection. The PRACH is a short sequence of bits that is used to temporarily identify the UE.

2. **Contention resolution**. This phase happens when multiple UEs send the same PRACH at the same time, causing a contention for the same resource. Since only one UE will be granted access to that resource, in this phase the BS resolves the contention by informing which of these UEs will be accepted. The other UEs will need to restart the RACH procedure selecting a new PRACH.

3. **Random access response**. The BS informs the (selected) UE at which time-slot it has to transmit its data. This phase allows for synchronization between the UE and the BS.

4. **Data transmission**. The UE transmits its data to the BS using the time-slot assigned in the *Random Access Response*. If the data is correctly received, the BS replies with an acknowledgement to confirm the successful transmission.

As we have seen, there is a possibility of PRACH collision which causes a contention for the same resource. Although such a contention is resolved during the procedure, there are some cases in which it is not acceptable because of timing restrictions. In these time-sensitive applications, we need some mechanisms that allows to avoid multiple UEs to select and transmit the same PRACH. We can address such a requirement by modulating the PRACH selection strategy. We thus distinguish between [18, 23]:

- **Contention-based RACH**. The UE randomly selects one of the possible preambles. Doing so, there is the possibility of collision at PRACH transmission, which must be resolved in a subsequent stage with a contention resolution procedure.

- **Contention-free RACH**. In this case it is the network itself that informs the UEs which preamble they have to use for the PRACH transmission. Clearly, this approach can only be adopted when the UE is already connected with the network (during handover, for example).

2.3.1 Contention-Based RACH

In this section, we present a detailed overview of the Contention-based RACH procedure as defined by the 3GPP organization. Consider two UEs, UE_A and UE_B, that initiate the RACH procedure with the same BS and exactly at the same time. Assume also that both UEs pick the same preamble creating then a contention, and that at contention resolution it is UE_A that will be granted access to the BS. In such a scenario, the contention-based RACH procedure evolves as follows [10, 23, 24] (Fig.2.4):

(1) **PRACH transmission (Msg_1)**. UE_A randomly selects the preamble from the set of possible sequences available for the target cell. The transmission contains the following information:

- *Preamble index*, which identifies the selected preamble.
- *Random Access - Radio Network Temporary Identifier* (RA-RNTI), an identification number that identifies one specific radio channel and one specific user. In this case it is implicitly determined by the timing of the transmission.

Assume that both UE_A and UE_B select: PRACH = [preamble: 1, RA-RNTI :1].
Assume also that UE_B's transmission is lost due to collision with the transmission from UE_A.

(2) **PRACH reception**. The BS detects the preamble and estimates the corresponding timing of the up-link transmission. In this way, the BS derives the corresponding RA-RNTI associated with the detected preamble. The BS assigns a Cell-RNTI (C-RNTI), which is a UE identifier used for future transmissions to each detected PRACH message. In this case, the PRACH detected is only the one coming from UE_A, since transmission from UE_B was lost.

(3) **Random Access Response (Msg_2)**. The BS assigns resources to the detected UEs and transmits the `Random Access Response` message on the shared channel. Together with the C-RNTI assigned, this response also contains the following information:

- *Timing Advance*, which provides time synchronization between the UE and the BS to adjust the timing for the up-link transmission.
- *Up-link grant*, that specifies the time slot, frequency resource, and power level that the UE should use for the up-link transmission.
- *Back-off indicator*, a parameter that tells the UE how many time slots it will need to wait before attempting a new data transmission in case of contention.

④ **Processing of `Random Access Response` (Msg_3).** Both UE_A and UE_B receive the `Random Access response`, and save the assigned temporary C-RNTI. Notice that, since UE_B chose the same PRACH as UE_A, it will believe being the receiver of the detected `Random Access Response`. Then, since at this stage the UEs can not understand whether they are in a contention scenario or not, and UE_B has no mean to understand that its previous transmission was lost, both the UEs will transmit the `RRC Connection Request` message using the up-link resources reported in the up-link grant. In this message, the UE attaches a new RNTI value which re-identifies the UE. Unlike the PRACH transmission, in which the RNTI was determined by the transmission time, its value is now randomly selected. Once the message is transmitted, the UE starts the T300 timer awaiting for BS response.
Assume that due to collision, the transmission from UE_B is lost again.

⑤ **Contention resolution (Msg_4).** The BS accepts the received `Random Access Response` and transmits the `Random Access Contention Resolution & RRC Connection Setup` message. This message signals the down-link resource assignment using the RA-RNTI associated with the new RNTI just received.
In this specific case, since the BS only received Msg_1 and Msg_3 coming from UE_A, the message will report [RA-RNTI: 1, RNTI_A]. As a consequence, when UE_B receives the contention resolution message, it realizes that RNTI_A does not correspond to the one it had previously selected, thus it understands it has lost contention in favor of another UE. UE_B will need then to restart the RACH procedure from step ①.

⑥ **Connection setup**. UE_A has won the contention and transmits the `RRC Connection Setup` message to initiate further signaling and data transmission. UE_B instead initiates the procedure from the beginning.

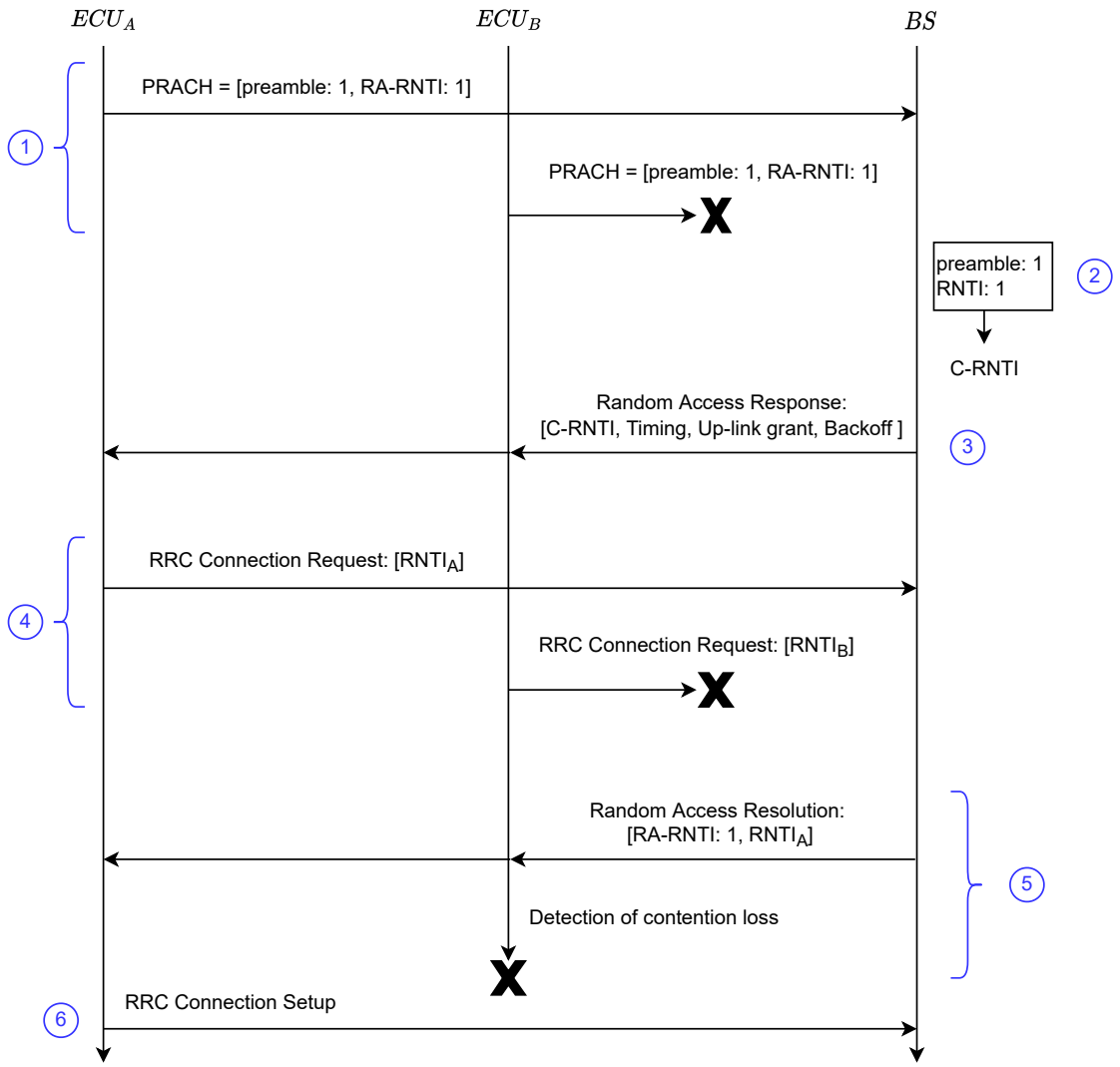


Figure 2.4: Example of call flow of a contention-based RACH procedure.

2.3.2 Two-Steps RACH Procedure

In order to further reduce delays when trying to re-establish a connection, the 3GPP organization also defines a variation of the standard RACH procedure, moving the RACH procedure from a four to a two-steps process. The idea is to combine into a single transmission the information in *Msg*1 and *Msg*3, and the same for *Msg*2 and *Msg*4 [18, 25]. Such an information aggregation not only allows to reduce the delay in the establishment of the connection, but also improves the ability of the network to handle scenarios in which a large number of UEs are try-

ing to establish a connection simultaneously. This improves network efficiency and quality of service being offered. The two-steps RACH procedure, however, can only be adopted (i) when the UE is in the `RRC_CONNECTED` mode, (ii) in handover scenario, or (iii) transitioning from the `RRC_CONNECTED` *active* to the *inactive* mode [25]. In all the other cases, the standard four-steps RACH procedure must be adopted.

2.3.3 RACH Improvements: 5G *vs* 4G-LTE

Due to the requirements for 5G networks design, *i.e.*, improved throughput and reduced transmission delays, the RACH procedure plays a crucial role in achieving such goals. In particular, there is a need for a more efficient and flexible procedure compared to 4G-LTE. To this purpose, when developing the 5G technology, the 3GPP organization introduced the following changes and improvements for the RACH procedure [23, 26]:

- **Preamble format**. While in 4G-LTE the preamble is a sequence of cyclic shifted Zadoff-Chu sequences, in 5G the preamble is a Zadoff-Chu sequence but with different root indices. Such a new format allows for more UEs to use the same RACH channel, thus having more UEs connecting to the network at the same time.

- **Slot format**. The RACH slots in 5G are modified to be flexible both in size and duration, while for the 4G-LTE standard these parameters are fixed. This change enables a more efficient network resources utilization.

- **Resource allocation**. In LTE, the RACH procedure uses a predefined set of resources for random access, whereas in 5G the RACH procedure uses dynamic allocation of resources based on the network load and the number of UEs requesting access.

- **Multiple access techniques**. In 5G, RACH procedures use both Orthogonal Frequency-Division Multiple Access and Non-Orthogonal Multiple Access technologies, where in 4G-LTE only the former is used. The introduction of the Non-Orthogonal Multiple Access strategy allows for an improved utilization of network resources.

2.4 Cellular Handover

The process by which the transfer of user information from the sBS to a new target Base Station (tBS) is carried out, is termed as *handover*. Handover is critical to ensuring continuity of cellular

services, and is typically triggered when the UE senses stronger reception from a BS other than the sBS. This is likely to happen when the UE is approaching the limit of the transmission range of the sBS.

There are many taxonomies to classify handover procedures [27]. For example, we can have a classification based on (i) the transmission frequency, (ii) the Radio Access Technology (RAT) used (from 5G to 4G-LTE, for instance), (iii) if the target and serving network are 3GPP networks or not, and others. For our purposes, we will consider a classification based on the position of the sBS and tBS. According to the 3GPP specifications [23], we can then give the following classification of cellular handovers:

- **Inter/Intra-cell handover**. Each BS is equipped with several cells, each of which has a specific identifier. Cells belonging to the same BS may differ for the coverage area and transmission frequency. The classification is based on whether the serving and target cells belong to the same (intra) or different (inter) BSs. In case of intra-cell handover, there is no need for coordination with neighbouring BSs and the whole process is managed by the sBS.

- **Intra-AMF handover**. This type of handover occurs when the sBS and the tBS belongs to the same AMF.

- **Inter-AMF handover**. This type of handover occurs when the sBS belongs to a different AMF than the tBS. Here, the sAMF interacts with the tAMF to provide the necessary information for the transfer of the connection.

Further, for 5G networks, we distinguish between two handover scenarios [17, 28]:

- **N2-handover**. It occurs when the CN, and therefore the serving AMF (sAMF), is involved in the handover process.

- **Xn-handover**. It occurs when there is a dedicated and direct secure communication channel between the sBS and tBS. In this case, the time required for handover completion is lower since there is no need to interact with the sAMF during handover preparation (in which the sBS asks the tBS for handover availability).

Table 2.1 summarizes the classification just discussed, while Fig. 2.5 shows a comparison between intra-AMF and inter-AMF handover scenarios.

Independently on the specific classification, all the aforementioned cellular handover procedures develop into three phases [11, 29]:

Table 2.1: Handover procedure classification.

Handover classification	Handover scenario
Intra-cell handover	Serving and target cells belong to the same BS
Inter-cell handover	Serving and target cells belong to different BSs
Intra-AMF handover	sBS and tBS belong to the same AMF
Inter-AMF handover	sBS and tBS belong to different AMF
N2-handover	The AMF is directly involved in the handover preparation process
Xn-handover	sBS and tBS directly communicate each other with no intervention of the AMF in the handover process

1. **Handover Preparation**. The sBS decides to proceeds for handover, and there is the verification of available resources to handle the handover on the tBS side.

2. **Handover Execution**. The UE is instructed to proceed for handover and to connect with the tBS.

3. **Handover Completion**. The CN records the transfer of connection management for the considered UE to the new sBS. Also, the previous sBS is instructed to release the old resources dedicated to that UE.

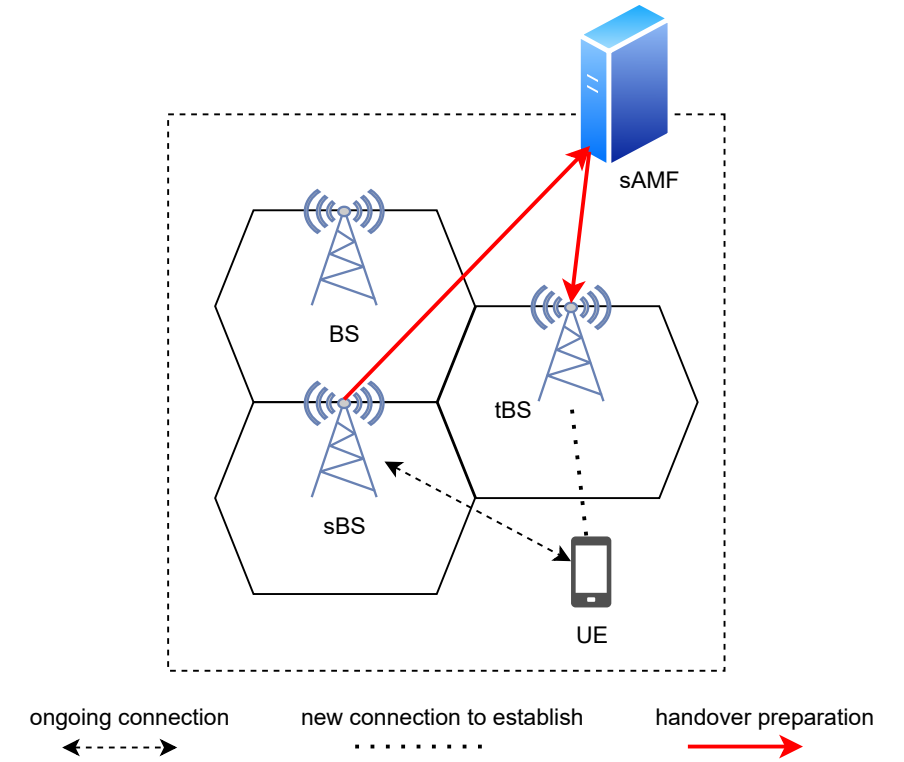
As discussed in Sec.2.2, upon completion of the AKA procedure, the CN, the BS, and UE share the necessary security parameters and keys for secure communication [30]. Hence, before initiating a handover process, the following symmetric keys are established:

- K_{SEAF}, (long term) key shared between UE and CN.

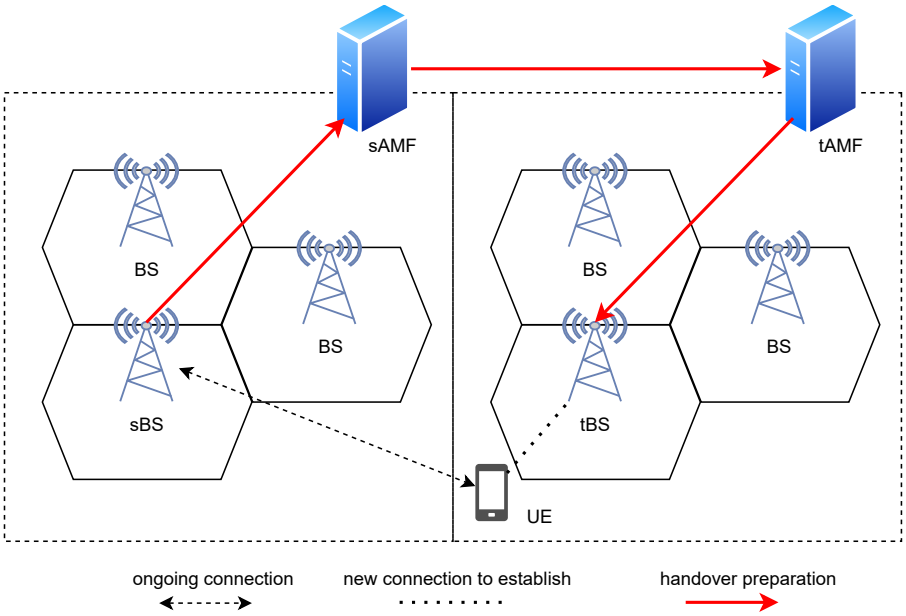
- K_{AMF}, (long term) key shared between UE and AMF. It is obtained from K_{SEAF} through a key derivation process [1].

- K_{gNB}, (short term) session key shared between UE and sBS. This can be derived either from K_{AMF} or from a previous K_{gNB} [1].

Once handover is triggered, the UE derives a new session key K_{gNB} to establish secure communication with the tBS. We assume that most communication between any pair of entities in the network is secure. The exceptions are (i) the RACH procedure, and (ii) the `RRC Reconfiguration` message. These exceptions occur due to the fact that according to 3GPP specifications, the UE security context is activated only after that the RRC Connection is established [10].



(a) Intra-AMF N2-handover.



(b) Inter-AMF N2-handover.

Figure 2.5: Comparison between inter/intra-AMF N2-handover.

2.4.1 N2-Handover

As anticipated, in case of an N2-handover the CN is involved in the procedure. Depending on whether we have an intra-AMF or inter-AMF N2-handover, we have different degrees of CN involvement. For example, in case of intra-AMF N2-handover only the sAMF is involved, as both the sBS and tBS are under the control of the same AMF. On the other hand, in case of an inter-AMF N2-handover we have a change of handler AMF. Thus, there will be the need to pass through both sAMF and tAMF, which increases the handover completion required time. However, both the mentioned scenarios follow the same execution pattern previously presented.

Consider a scenario in which the UE is connected to a sBS and is approaching a tBS that has a greater signal strength. The intra-AMF N2-handover procedure from sBS to tBS develops according to the following steps [17, 31] (Fig. 2.6):

① The UE periodically senses the MIB and SIB messages broadcast from neighbouring BSs, and transmits a `Measurement Report` (`MR`) message. This message contains information about the strength of the received signal from the sBS, as well as signals from surrounding BSs.

② Based on the content of the `MR`, the sBS decides whether there is a need to hand over the UE to another BS. In the case where a handover is deemed necessary, the sBS selects the tBS. The handover decision is generally threshold based: if the signal strength from another BS exceeds a certain threshold compared to the signal from sBS, then handover is triggered. At handover decision, sBS transmits the `Handover Required` message to the AMF. This message contains information about the choice of tBS and the protocol data unit sessions that need to be handed over.

③ The AMF identifies the tBS and forwards to it a `Handover Request` message, providing information such as UE security context, capabilities and session information.

④ Based on the information received and available resources, the tBS decides whether to admit the UE. In the case of handover acceptance, the tBS replies to the AMF with a `Handover Acknowledge (ACK)` message, which specifies which sessions it can accept.

⑤ Upon receipt of handover confirmation from the tBS, the AMF sends a `Handover Command` message to the sBS. This message contains information included in `Handover ACK` that the UE needs in order to obtain access to the target.

⑥ The sBS triggers the handover procedure by forwarding to the UE the information received through a `RRC Reconfiguration` message, received from the AMF.

⑦ The UE interrupts the connection with sBS and performs a RACH procedure with the tBS [10]. After successful RACH, the UE considers the handover as completed, and transmits a `RRC Reconfiguration Completed` message to the tBS.

⑧ The tBS considers the handover completed and sends a `Handover Notify` message to the AMF to inform it about the change of connection handler for the UE.

⑨ The AMF transmits a `UE Context Release` message to the sBS, instructing it to release resources that were dedicated to the UE.

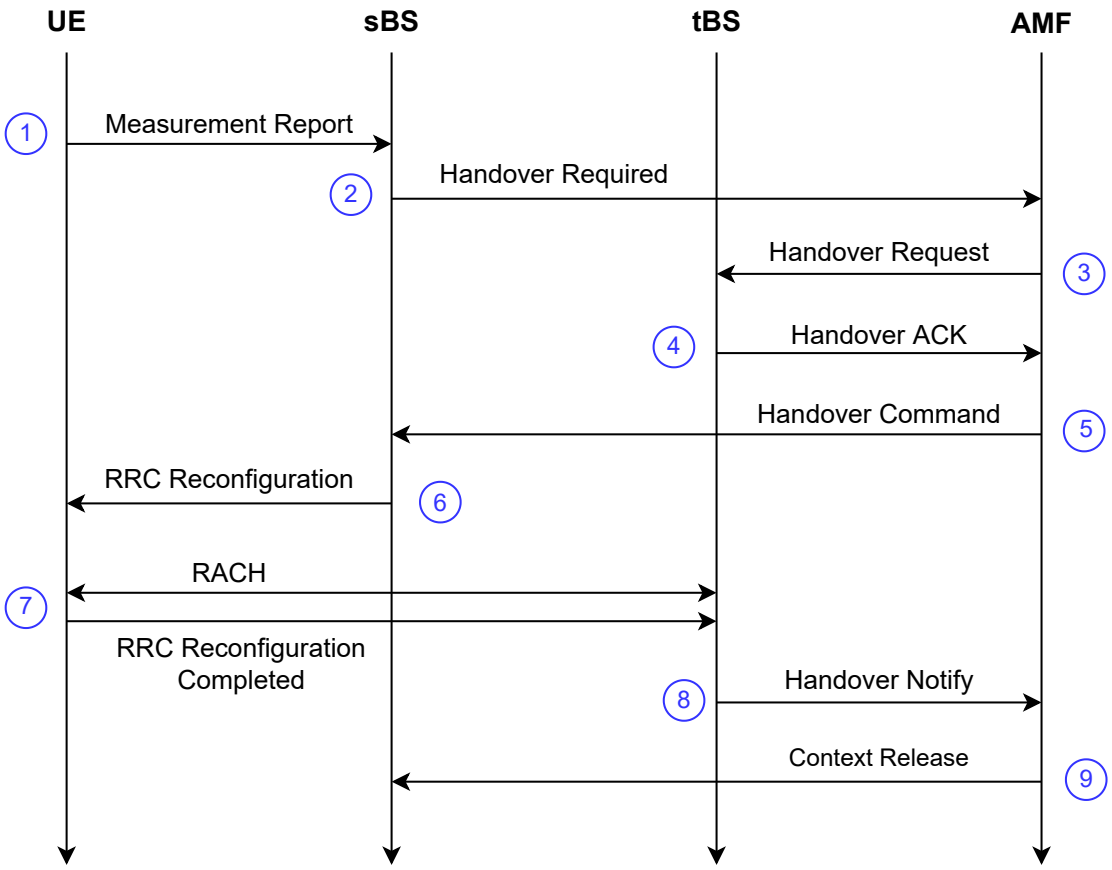


Figure 2.6: Example of the call-flow for a 3GPP intra-AMF N2-handover.

In case of inter-AMF N2-handover, the procedure would have been very similar, with the only difference being the additional steps needed for the communication between the sAMF and tAMF, as shown in Fig. 2.5b.

2.4.2 Xn-Handover

In the Xn-handover, we have a very similar call-flow as for the N2-handover. The difference between the two is that now we do not need to pass through the sAMF for the resource allocation request: it is the sBS that directly communicates through the Xn-interface with the tBS, and vice versa. The same applies to the `Context Release` message. In an Xn-handover, the sAMF is involved only at the end of the handover procedure in order to be informed of the change of connection handler for the given UE. Fig. 2.7 shows the call-flow for an Xn-handover scenario.

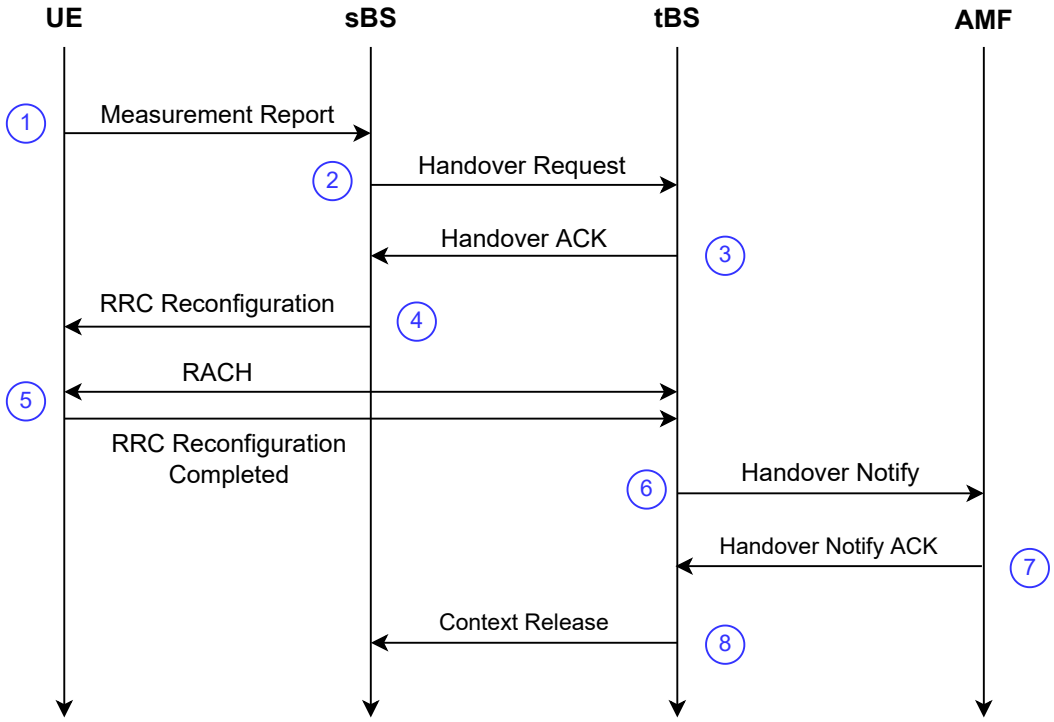


Figure 2.7: Example of the call-flow for a 3GPP intra-AMF Xn-handover.

2.4.3 DAPS Handover

Generally, once the `RRC Reconfiguration` message is received, the UE first closes the connection with sBS, then proceeds with the RACH procedure toward the tBS. This causes an interruption of the ongoing connection for few tens of milliseconds [10, 23]. Although the interruption time is minimal, this can be very critical for applications that are extremely time-sensitive. To overcome this issue, the 3GPP organization proposes a solution thanks to which the UE can maintain, for a short period of time, a connection with both sBS and tBS. Then, the connection with sBS is released only after an explicit release command from the tBS. This special case of handover is called as *Dual Active Protocol Stack* (DAPS) handover [17, 32]. The DAPS handover presents the following characteristics [32]:

- The UE continues transmission and reception with sBS after receiving order to proceed for handover.
- The UE receives user data from sBS and tBS simultaneously.
- The UE switches the up-link transmission from sBS to tBS only after completion of the RACH procedure.
- Connection interruption during handover is reduced close to 0ms.
- DAPS handover is possible over both Xn and N2 interfaces.
- Whether to use standard or DAPS handover generally depends on the UE capabilities and tBS availability.

Due to the double connection, in order to secure in-sequence delivery of user data, messages must contain a sequence number throughout the all handover procedure. Besides, the UE has to maintain a separate security context with sBS and tBS [17, 32].

3

Adversary Model and Fake Base Station Attack

In this chapter we define our threat model, thus the assumptions that we make over the actions that an attacker carrying out an FBS attack can and cannot do. Furthermore, we discuss the several steps through which the attacker prepares and carries out an FBS attack, and analyse in details the call flow of an handover scenario in which the UE is victim of an FBS attack. Finally, we give a discussion about the possible consequences and impact such an attack may have both to the UE and network sides.

3.1 Threat Model

An adversary carrying out an FBS attack aims to stealthily make the UEs connect to a rogue BS (rBS) instead of a legitimate BS. Such an attack enables the adversary to gain control over the UE connection, possibly leading to other types of attacks such as DoS, MitM or bidding-down attacks [33]. We adopt a threat model similar as what defined in [11] and [12]. In defining our threat model we make then the following assumptions:

- The attacker can drop, modify, inject and eavesdrop messages exchanged between legitimate parties. Specifically, the attacker is able to collect the MIB and SIB messages broadcast by BSs.

- The attacker can set up a rBS that has the same capabilities as a legitimate BS.
- The attacker cannot tamper with the USIM card, BSs, and CN. Specifically, the attacker cannot learn the keys derived during the AKA protocol other than by exploiting vulnerabilities of the AKA protocol itself.
- The attacker can successfully complete a standard RACH procedure with the victim UE [18, 34].

As we will discuss better in Sec. 3.2.2, the last assumption is not strictly necessary for an attacker aiming to carry out an FBS attack with the purpose of bringing the UE in a DoS status. However, if instead the attacker's goal is to use the FBS attack as a starting point for subsequent exploitation, the ability to successfully complete RACH with the victim will be required.

3.2 The Fake Base Station Attack

This section provides the details of the preparation steps for an FBS attack, as well as the corresponding call flow for a successful FBS attack scenario. Lastly, we discuss about the impact and consequences such an attack may have both to the UEs and to the network.

3.2.1 Attack Preparation Steps

In order to carry out a successful FBS attack, the attacker needs to go through several preparation steps with the purpose of gathering sufficient data of the surrounding environment and networks. The collected data will then be used to properly set up the rBS. For the attack preparation steps we refer to what presented in [11]:

1. **Initial Reconnaissance**. The attacker wants to gather enough knowledge of the target cellular network environment. To this purpose, the attacker needs to capture the broadcast messages, as well as the UEs traffic. In particular, the attacker must sniff the MIB and SIB messages that the surrounding BSs broadcast. The collection of such data can be done quite easily. For example, the attacker could simply use a smartphone device with a legitimate subscription with the target network.
2. **Determination of the network structure**. The attacker aims to accurately determine the network composition and properties. For instance, the attacker will try to determine

BSs locations, the cell identifiers, the radio frequency cell numbers, and other related information. Having an accurate representation of the network structure will indeed allow the attacker to better determine the most suitable target BS to attack. Besides, the attacker should continuously scan the network so to maintain the rBS updated and consistent with network parameters and structure, thus lowering the probability of being detected.

3. **Target selection**. At this point, the attacker should have all the information to select the best target cell and BS for the attack. Clearly, the best option is to emulate a BS that is closer to the location of the rBS. Doing so, in fact, would allow the attacker to set up the rBS using the correct parameters. Besides, this approach will make harder for FBS detection mechanisms to identify the presence of rBS.

4. **Fake base station configuration**. After selection of the target BS for the attack, the following step is to properly configure the rBS. In particular, the rBS installed shall not only replay MIB and SIB messages broadcast by the target BS, but it should also have the same cell identifier, PRACH root sequence, tracking area identifier, and type of service. Besides, in order to make the attack more effective, the rBS should be configured in such a way that it can properly respond to RACH, RRC and NAS (Non-Access Stratum [23]) messages. It is possible to accomplish this latter objective by using open or closed software available on the market.

5. **Exploitation**. Once the rBS is properly settled up, the attacker can launch the attack. The attacker will then progressively increase the transmission power until the UE is attracted by the signal generated by the rBS rather than the legitimate BS. This should trigger handover and force the UE into trying to connect with rBS since it is getting the best signal reception from rBS. If this happens and the UE connects to rBS, the FBS attack is considered successful and the attacker can then control the UE's connection. Controlling the user connection enables the attacker for further attacks and exploitation.

3.2.2 Attack Flow

From the attack preparation steps just presented, we can observe that an FBS attack is feasible not only during the handover, but during initial access as well. Indeed, the key point of such an attack is the emulation of the target BS and the replay of its MIB and SIB messages. These are in fact transmitted in a broadcast fashion, and are therefore used by both UEs that are in the

RRC_CONNECTED (UEs that already have a connection established) or RRC_IDLE mode (UEs that do not have an ongoing connection yet, and that need establish one through the initial access procedure presented in Sec.2.3). In what follows, we describe step by step what happens when a UE in the RRC_CONNECTED status is victim of an FBS attack. This is a more interesting case to analyze compared to initial access because the UE already has an ongoing connection, thus handover toward rBS is triggered. In this way we can then have a more complete understanding of how the attack works and its implications.

Consider a scenario in which an attacker sets up a rBS to imitate a tBS. First, the attacker sniffs the SIB and MIB messages broadcast by the tBS, and replays them without modification. The general principle that 5G networks follow for selecting the best BS is based on the power of the received signal. The BS providing the highest signal strength is (commonly) chosen as the best BS, and thus as the tBS for handover [11, 12]. Consequently, the attacker transmits the replayed messages with a higher transmission power compared to the surrounding BSs (Fig. 3.1, ①). Doing so, even if the UE would receive SIB and MIB messages from both rBS and tBS (which are the same because rBS replays those of tBS), because the signal coming from rBS dominates over the one from tBS, the UE will prefer to establish a connection with rBS [13]. Hence, once a UE falls within the transmission range of the rBS, it will read the replayed SIB and MIB messages and transmits the MR message containing information about surrounding BSs. In particular, the MR message will contain data of the legitimate tBS (*e.g*, BS and cell identifiers), but the corresponding received signal strength will be the one coming from rBS, which we recall being higher than any other surrounding BS. Furthermore, as a consequence of replayed messages reception, the UE steers its antenna towards the direction the messages are coming from, thus towards rBS, believing it is pointing to the legitimate tBS. After receiving the MR message, the sBS triggers handover towards the legitimate tBS because from the MR content it is the BS that offers the best signal reception (Fig. 3.1 ②). However, upon receiving the RRC Reconfiguration message, the UE connects to the rBS instead of the tBS. This happens because the UE had previously steered its antenna in the direction of the rBS. At this point, the attack is considered successful and the adversary can control the UE's connection (Fig. 3.1 ③). Finally, since the tBS does not receive any connection from the UE, it does not send the Handover Notify to the AMF, which in turn does not send the Context Release command to sBS (Fig. 3.1 ④,⑤).

Let us consider now what would happen in case the attacker does not successfully complete RACH with the victim. In this case, the UE experiences a handover failure and generates a Radio Link Failure (RLF) report containing information of rBS, which however are the same

as the tBS since the UE believes it was trying to connect with the legitimate BS [10, 13]. Afterwards, the UE tries to re-establish a connection sending a `RRC Connection Re-Establishment Request` message [13]. However, since rBS has not been detected as a non-legitimate BS, it will still remain the best BS to connect with, and the UE will then transmit the re-connection request to rBS. On the other hand, due to the fact rBS can not offer a legitimate service, the only option the attacker has is to bring the UE to a DoS status by rejecting any incoming message. Thus, the rBS replies with a `RRC Connection Re-Establishment Reject` message. This causes the UE to close the ongoing procedure, *e.g.*, call or data exchange, and to remain connected to rBS in a DoS status until moving out of the transmission range of the attacker [13].

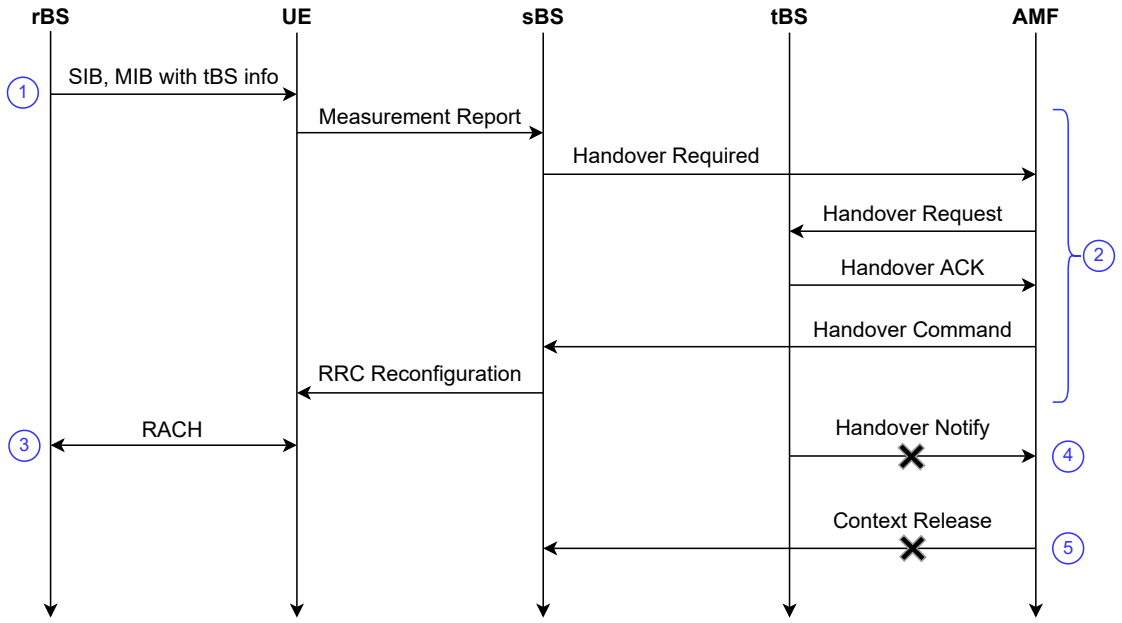


Figure 3.1: Example of call flow in a successful Fake Base Station attack scenario.

3.2.3 Attack Consequences

An FBS attack has an impact on the victim user, as well as the CN. Besides, it can be exploited as a first step to obtain connection with a UE so to launch subsequent, and often more severe, attacks. In what follows, we discuss the possible consequences and impacts an FBS attack may have on the UEs and networks [11, 12, 13].

Impact on the network.

- *Resource wastage*: If the handover fails, then all resources used during handover preparation are wasted. However, this issue is not limited to handover preparation only. From the perspective of the CN, the UE has disappeared, and thus the AMF initiates a *paging* procedure [17] to locate the UE. This clearly results in an additional resource utilization.

- *BS disconnection*: A BS that has a handover success rate below a desired threshold (usually set at 95% [13]) may be removed from the list of possible targets until it is recovered. Hence, the attacker may have the goal to force several handover failures so to disconnect a selected BS from the network, thus affecting the network reliability.

Impact on the UE.

- *DoS attacks*: The attacker rejects all the incoming messages resulting in complete DoS to the UE.

- *Bidding-down attacks*: The attacker forces the UE into adopting older cellular standards (*e.g.*, 2G or 3G). Older standards usually provide lower service quality and security. Therefore, the attacker may exploit these protocols' vulnerabilities to carry out subsequent attacks.

- *Location tracking*: In the case of 4G networks, the attacker can exploit the lack of authentication and integrity protection of the `Identity Request` message. This forces the UE to transmit its permanent or temporal subscriber identifier in plain-text. As a result, the attacker can track UE's movements by exploiting vulnerabilities in the paging protocol [35]. The 5G standard overcomes this vulnerability by requiring the UE to encrypt its identifiers, and to periodically refresh the temporal identifier [10]. However, an adversary may still be able to perform location tracking by carrying out first a bidding-down attack.

4
BARON

In this chapter, we describe the working of BARON authentication methodology, and detail how it mitigates the impact of an FBS attack. We start our discussion by identifying the FBS vulnerability cause we want to address with BARON, and proceed by defining the concept of the *Closest Trusted Entity*, which plays a central role in BARON. Then, we give an overview of our proposed defense methodology by presenting two possible mechanisms that implement it. Lastly, we present an efficient and secure connection recovery mechanism the UE will follow when understanding being victim of an FBS attack.

The adversary is assumed to be as defined in Sec. 3.1, and all exchanged messages, except for those between the UE and tBS (or rBS), are assumed to be authenticated and encrypted according to the AKA procedure (Sec. 2.2).

4.1 FBS Attack: Reasons for Vulnerability

As identified in [11], there are three major reasons that make 5G networks vulnerable to an FBS attack. These are:

- **Insecure transmission of broadcast messages**. The UE completely trusts the content and provenance of the MIB / SIB messages, which are indeed transmitted and accepted without any authentication;

- **Unverified measurement reports**. The sBS completely trusts the content of the `MR` message without verification;
- **Missing cross-validation**. There is no cross-verification to check if the content of the `MR` message reports data values that correspond to those expected for the real tBS.

In works [14, 12], authors identify the *insecure transmission of broadcast messages* as the primary cause for 5G networks being vulnerable to FBS attacks. In order to mitigate this vulnerability, the authors developed a sequence of digital signature schemes to authenticate the MIB and/or SIB messages. Such an approach, however, might be vulnerable to replay attacks [14, 12]. In fact, the broadcast nature of the MIB/SIB messages allows users to gather information about the BS. As a result, all users, including those that are not yet in the `RRC_CONNECTED` [10] status, need to be able to verify the authenticity of MIB/SIB messages. Hence, in the absence of a pre-distributed public key, the transmitting BS itself must provide the public key for signature verification together with the authenticated message. This means that the attacker can simply follow the attack strategy described in Ch. 3 (replay the beacons without modification), without the need to reverse engineer the authentication key. Since the authors were aware of this replay attack vulnerability, they introduced a time-based mechanism for the acceptance of the signature. However, this may still be not enough from preventing the attacker to succeed in the attack. We discuss more in detail these works and related issues in Ch. 6.

In order to design a completely secure defense mechanism against the threat model described in Sec. 3.1, we analyze the vulnerability to FBS attacks from a different perspective. Consider the handover procedure presented in Sec. 2.4: according to the 3GPP specifications, the UE considers a handover to be complete at the conclusion of the RACH procedure [17]. We must observe that the handover success condition is *"the UE successfully concludes RACH"*, and not *"the UE successfully concludes RACH with the tBS"*. As a result, in the absence of an active adversary, the UE connects to the tBS correctly; however, when subject to an FBS attack, the UE will connect to the rBS while believing the handover to be completed successfully. We also observe that the UE is *"left alone"* during handover execution, meaning that once it receives instruction to proceed for handover, there is no feedback from the CN or sBS to verify whether a legitimate tBS has actually been reached or not.

In conclusion, in our analysis we observe that in current standard 3GPP 5G networks, **the UE has no means to corroborate whether the RACH procedure has been executed with the legitimate tBS**. BARON, by design, addresses this vulnerable aspect of the BS connection procedure, both in case of handover or initial access. Moreover, since BARON does not pre-

vent connection with the rBS, it is integrated with a fast and efficient mechanism to allow the victim UE to recover connection with a legitimate BS when subject to an FBS attack.

4.2 BARON Methodology Overview

BARON authentication methodology to defend against FBS attacks relies on the *Chain of Trust* built by the UE through the AKA protocol. Before diving into the details of the proposed solution, we need to introduce the notion of *Closest Trusted Entity* (CTE). The CTE will act as a guarantor for the authenticity of the tBS with which the UE is establishing a connection.

***Definition* 4.2.1 (Closest Trusted Entity (CTE))**
The Closest Trusted Entity in a 5G network is that closest node to the UE that can ensure (i) trust and (ii) security on behalf of the core network, and for which the UE has a valid and active security context. During a handover, these two conditions must hold for both, serving and target base stations.

We provide two examples to better illustrate the introduced concept of CTE and how it applies depending on the specific scenario considered.

Example 4.2.1 *Consider an intra-AMF N2-handover as shown in Fig. 2.6 (both the sBS and tBS are under the control of the same AMF). In this case, the AMF acts as the CTE since it is the last node of the network (the closest to UE) that is common to both sBS and tBS, and for which the UE has a valid and active security context. We must notice also that in the case of an inter-AMF N2-handover (where the sBS and tBS do not belong to the same AMF), the serving AMF (sAMF) will act as the CTE as well. Indeed, the sAMF can reach the target AMF (tAMF), which in turn reaches the tBS, both with secure communications.*

Example 4.2.2 *Consider an intra-AMF Xn-handover. In this case, since sBS and tBS can directly communicate without the need to pass through the AMF, the sBS acts as the CTE. The sBS is clearly the closest node of the network to the UE, and it securely communicates with both the UE and tBS.*

The objective of BARON is to allow a UE to be cognizant of whether a reached BS is legitimate or not. We can accomplish this objective by requiring the tBS to prove that is has communicated with the CTE. Hence, at completion of the RACH procedure, the UE expects to receive within a certain time interval an *Authentication Token (AT)* that only the

CTE could have correctly computed. In case of missed reception of the AT within the time interval, or in case the AT does not match the expected value, the UE aborts the handover or initial access procedure, initiating then a connection recovery process. We leave the specific determination of such a time interval for acceptance to the manufacturers, depending on their specific needs and constraints.

Let us analyze what would happen as a consequence of a FBS attack during handover, in a scenario where BARON methodology is active. Since the rBS is not legitimate, it can not offer any legitimate service. For the same reason it is not able to establish a connection with the CTE. Moreover, encryption of the AT with the symmetric secret key K_{CTE}, shared between the UE and CTE (described in Sec. 4.3) ensures that an attacker sniffing a message during wireless transmission will not be able to use that before decrypting it. In such a scenario, the attacker has two options: (i) not transmit anything, or (ii) randomly guess the AT. For case (i), the absence of a transmitted message will make the timer to expire, ensuring that the UE considers the handover (or initial access as well) as failed, and initiates the re-connection procedure. In the case of (ii), if n bits are used for AT, the probability of a correct guess is 2^{-n}, which decreases to 0 as n increases. As a result, the UE will reject the connection with high probability (for large enough values of n), initiating the re-connection procedure.

Overall, **BARON leverages the above insight and uses reception of the correct AT from the tBS as proof of communication with the CTE, thereby guaranteeing the legitimacy of the tBS with high probability**. We note, however, that BARON does not prevent the UE from connecting to the rBS. Rather, it provides a means to verify if a reached BS is legitimate. In what follows, we propose two possible mechanisms in which the BARON defense methodology can be implemented.

4.3 BARON: Authentication Mechanism 1

The first approach we propose is a *challenge-response* mechanism, wherein the UE challenges the tBS by transmitting a random value, and expects to receive a response that could have been correctly computed only by the CTE. Such a mechanism is suitable for both initial access and handover, with minor differences between the two cases. We assume that the UE has already performed *Initial Registration* to the CN, implying it has already a valid security context.

Let the sAMF be the CTE. Together with the `MR` message, the UE transmits the AT, which is the encryption of a random number R. The encryption is performed by using the symmetric key K_{CTE}, shared between the UE and CTE. The sBS then forwards the AT to the sAMF,

which retrieves R and computes $R' = H(R)$. The function $H(\cdot)$ can be any deterministic or randomized function. The sAMF encrypts R' to obtain the AT' value, and forwards it to the tBS with the `Handover Required` message. After completing the RACH procedure, the UE starts an internal timer for both handover and initial access. If the timer expires, the UE considers the connection attempt as failed. In the meantime, the UE also computes AT' and expects to receive a message from the tBS containing $\tilde{AT}'$. In case of reception, the UE suspends the timer and verifies whether $AT' = \tilde{AT}'$. If the two values match, the UE deems the tBS to be legitimate. Otherwise, it initiates a connection recovery procedure in case of handover, or selects a new tBS in case of initial access.

The underlying working principle in the case of initial access is very similar to that for handover, with the main difference being that there is no sBS. Therefore, the UE transmits the AT to tBS. For example, the N2-handover (Fig. 4.1) and initial access (Fig. 4.2) procedures with BARON develop according to the following steps:

N2-Handover with BARON (Fig. 4.1):

① UE → sBS: (MR, AT)

- MR = Measurement Report
- $AT = E_{K_{CTE}}(R)$ | R = random number, $E_K(\cdot)$ = encryption, key K

② sBS → AMF: (`Handover Required`, AT)

③ AMF → tBS: (`Handover Request`, AT')

- $R = D_{K_{CTE}}(AT)$ | $D_K(\cdot)$ = decryption, key K
- $R' = H(R)$ | $H(\cdot)$ = any algorithm
- $AT' = E_{K_{CTE}}(R')$

④ tBS → AMF: `Handover ACK`

⑤ AMF → sBS: `Handover Command`

⑥ sBS → UE: `RRC Reconfiguration`

⑦ UE ↔ tBS: RACH procedure

⑧ tBS → UE: $\tilde{AT}'$

⑨ UE: verifies whether $AT' = \tilde{AT}'$

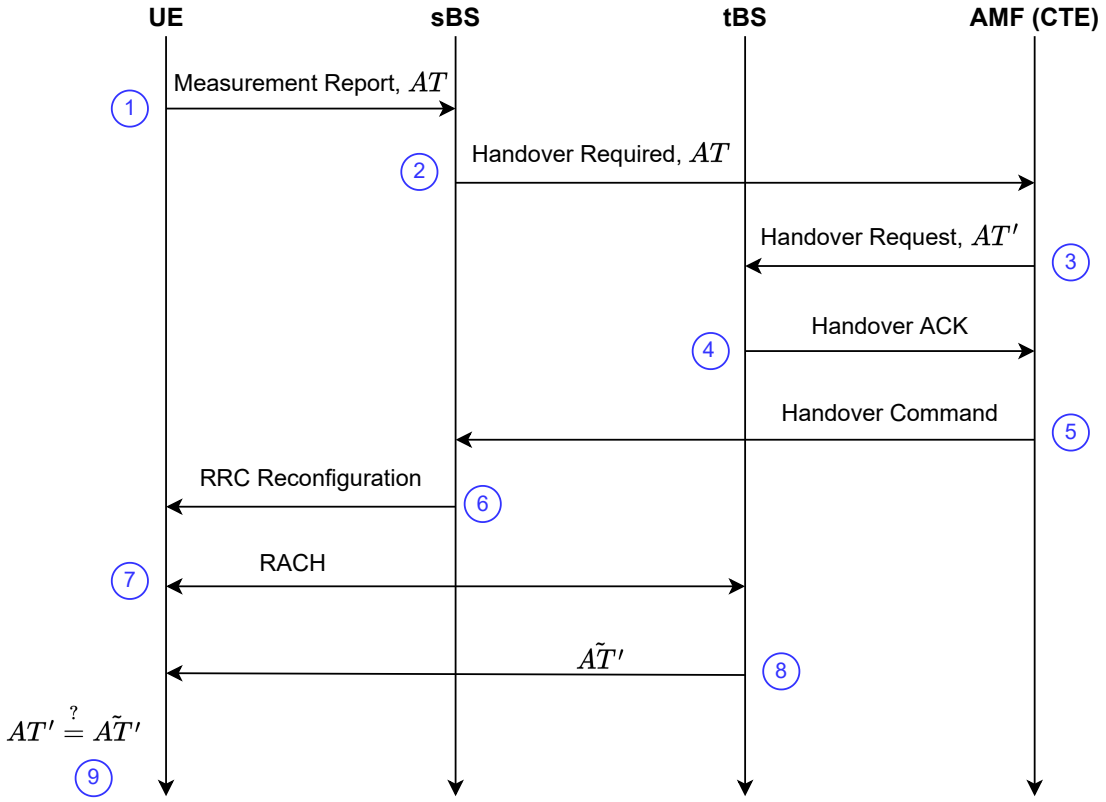


Figure 4.1: Example of an N2-handover using BARON with *Authentication Mechanism 1*. Receipt of the correct AT' from the tBS proves it has communicated with the CTE, thereby establishing its legitimacy.

Initial Access with BARON (Fig. 4.2):

① UE → tBS: $msg = \left(UE_{info}, ID_{CTE}, AT, MAC_{UE}\right)$

- UE_{info} = user information
- ID_{CTE} = identifier of the CTE
- $AT = E_{K_{CTE}}(R) \quad | \quad R$ = random number, $E_K(\cdot)$ = encryption, key K
- MAC_{UE} = Message Authentication Code of the UE, computed using K_{CTE}

② tBS → AMF: msg

③ AMF: verifies the MAC_{UE} and computes AT'

- $R = D_{K_{CTE}}(AT) \quad | \quad D_K(\cdot) =$ decryption, key K
- $R' = H(R) \quad | \quad H(\cdot) =$ any algorithm
- $AT' = E_{K_{CTE}}(R')$

④ AMF → tBS: $(VerificationOK, AT')$

⑤ tBS → UE: $\left(VerificationOK, \tilde{AT'}\right)$

⑥ UE: verifies for $AT' = \tilde{AT'}$

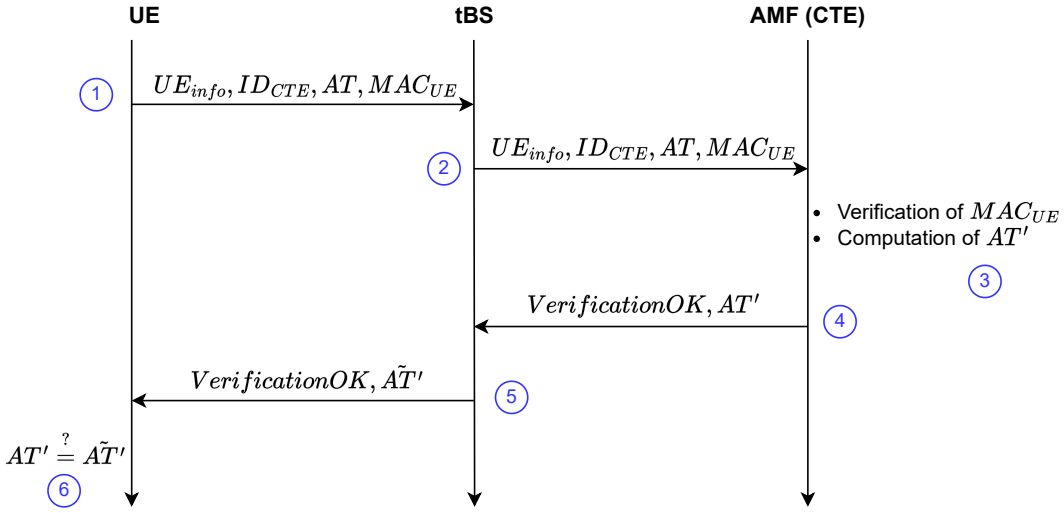


Figure 4.2: Example of initial access procedure using BARON with *Authentication Mechanism 1*. Receipt of the correct MAC_{UE} establishes legitimacy of the UE to the AMF. Receipt of the correct AT' from the tBS proves that it has communicated with the CTE, thereby establishing its legitimacy.

We make the following remarks about this mechanism:

- It is always true that the sAMF is the CTE for an N2-handover. However, this may not hold for initial access. Thus, the ID_{CTE} and K_{CTE} need be properly selected. We provide additional details on CTE selection at initial access in Sec. 4.3.1.
- During the initial access procedure, UE_{info} is the set of all information that the UE transmits according to the 3GPP standard for initial access [10], and the MAC_{UE} is used to provide UE authentication to the CTE. Indeed, although we want to challenge the tBS, at the same time we want the AMF and tBS to respond if and only if the challenge comes from a legitimate user.

- We leave the specific implementations of the $E(\cdot)$, $D(\cdot)$ and $H(\cdot)$ algorithms to the service provider, based on their needs and constraints, as long as these are reasonably fast algorithms. The same applies to the MAC_{UE} computation.

Fig. 4.3 shows the flow for the case of an Xn-handover with the authentication mechanism just described. The underlying procedure is the same as that for the N2-handover, with the only difference being that the sBS is the CTE for the Xn-handover.

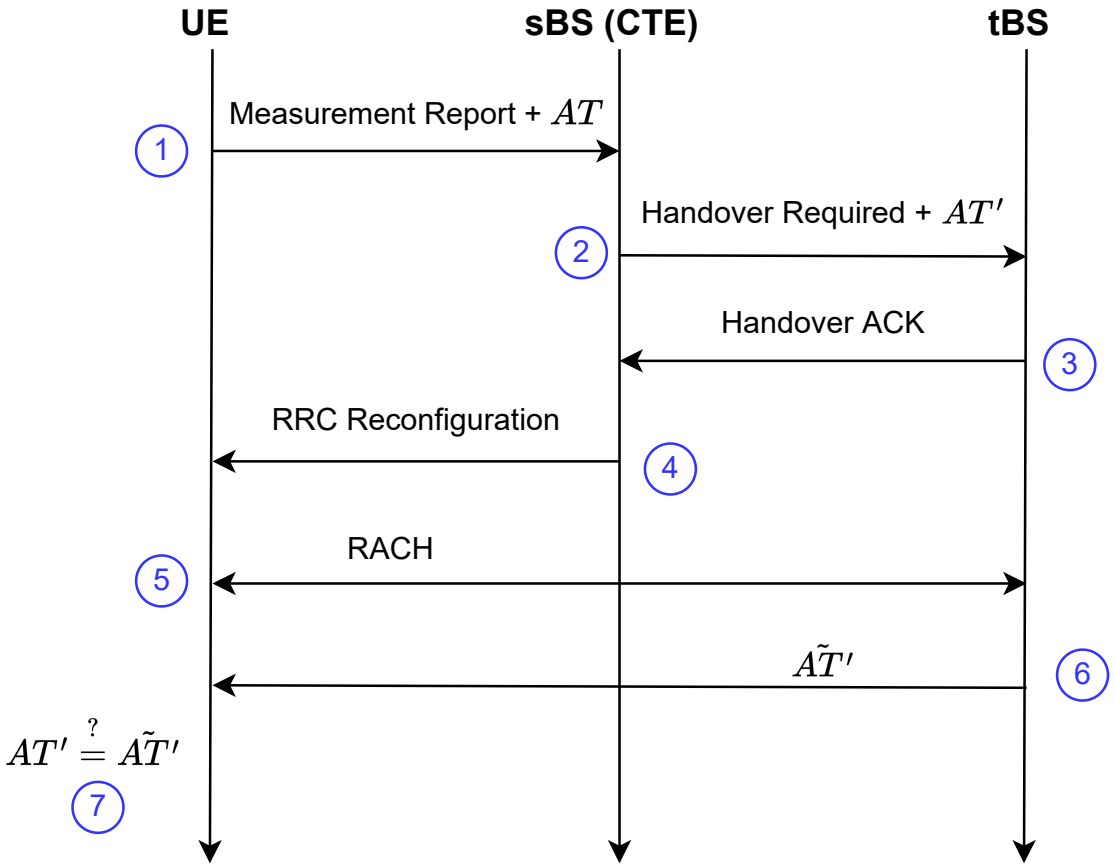


Figure 4.3: Example of Xn-handover using BARON with *Authentication Mechanism 1*. Here, the sBS is the CTE due to the direct communication between sBS and tBS. Receipt of the correct value of AT' from the tBS is proof that the tBS has communicated with the CTE, thereby establishing its legitimacy.

4.3.1 CTE Selection During Initial Access

The most important and critical aspect of the BARON *Authentication Mechanism 1* in the initial access scenario is the selection of the CTE. In practice, it is not always guaranteed that the

tBS will belong to the same AMF cluster with which the UE was most recently connected. Further, it might be the case that the encryption key K_{AMF} is no longer valid and must be updated. We propose then two possible methods for CTE selection for initial access:

1. The UE simply reports the identity of the AMF stored, and the task of resolving for that AMF is left to the CN, or

2. The UE is notified that the reported AMF is not a valid CTE for the tBS in question, thus it need to be changed.

In the second scenario, the CTE is changed with an entity of the network in a higher layer than the AMF (as long as this is a valid choice). In other words, we move the CTE to be the node in the layer above the AMF; if this entity is still not a valid CTE, we move to the upper layer again. We repeat this procedure until finding a valid entity, which in the worst case will be the root entity of the network.

We recognize that both these methods results in an additional effort for the CN, and consequently in an increased resource consumption and connection delays. However, we believe that the increased connection delay in the initial access scenario might be acceptable in order to accomplish improved security. On the other hand, increased resource utilization might impact the performance of the CN. Analyzing trade-offs between performance and resource utilization is an interesting avenue for future research.

4.4 BARON: Authentication Mechanism 2

We can apply this second authentication mechanism to implement BARON only to handover because it requires that the UE already has an ongoing trusted connection. In this case, the UE does not challenge the tBS but receives the AT along with the `RRC Reconfiguration` message from sBS (which is a trusted node). The AT is computed then by the CTE (sAMF for the N2-handover, and sBS for Xn-handover). Following this, similar to *Authentication Mechanism 1* (Sec. 4.3), the UE expects to receive the $\tilde{AT}$ from tBS, and compares it with the previously received AT. If the two values match, the UE can conclude that the tBS is legitimate. Here, the AT can take any arbitrary value, as the only condition that needs to be satisfied is $AT = \tilde{AT}$.

The *Authentication Mechanism 2* is better suited for resource-constrained devices, *e.g.*, IoT devices, since the UE does not need to compute any cryptographic value, but only has to compare the received values. Fig. 4.4 shows the steps of an N2-handover using *Authentication Mechanism 2*.

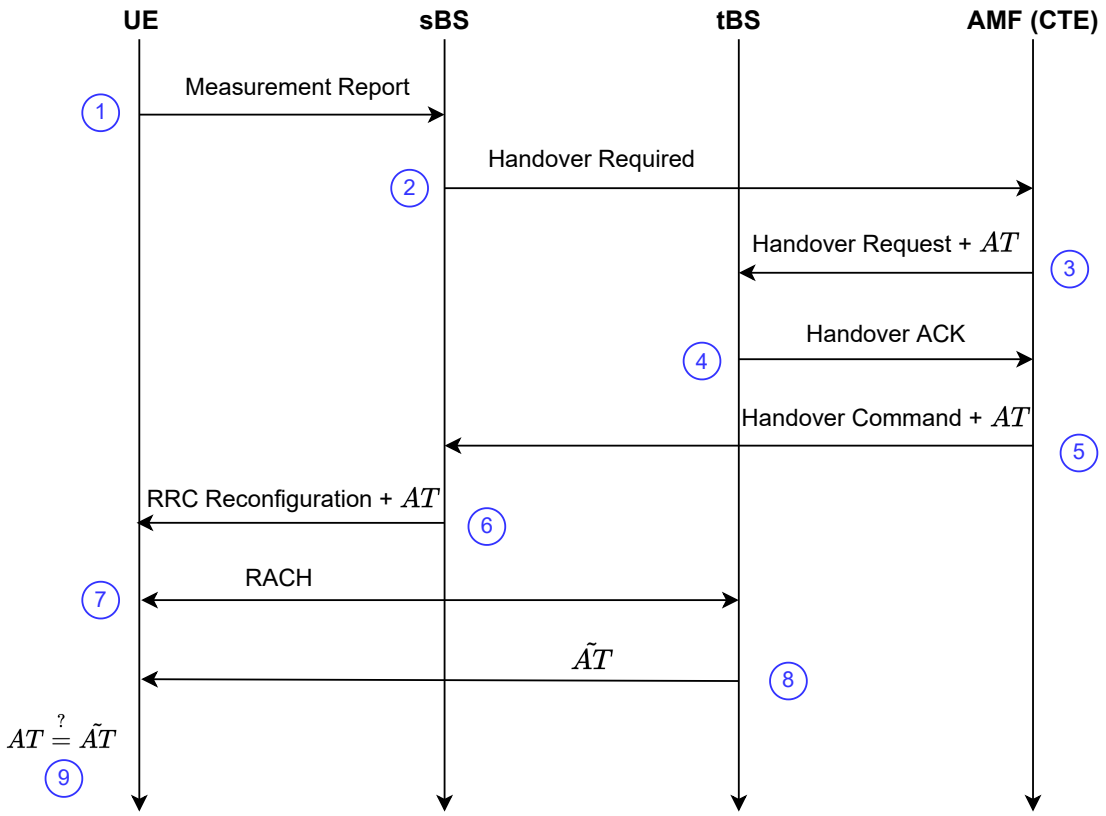


Figure 4.4: Example of an Xn-handover using BARON with *Authentication Mechanism 2*. Here, the UE does not challenge the network but receives the AT from the sBS, which is trusted. Receipt of the correct AT from the tBS proves it has communicated with the CTE, thereby establishing its legitimacy.

4.5 Security Discussion

In this section we formally prove the security of BARON against the threat model previously defined. In particular, we need to prove that BARON is not vulnerable to replay attacks, which would then make the proposed mechanism completely ineffective. Indeed, if BARON was vulnerable to such an attack, the UE would deem the rBS as legitimate, which clearly goes against the initial objective.

Consider a N2-handover scenario using BARON with *Authentication Mechanism 1* active, as presented in the previous sections, and an attacker running a FBS attack. We also recall that since the rBS is not legitimate, it cannot establish a valid connection with legitimate network entities, and with the CTE in particular. In such a scenario, upon reception of the RRC `Reconfiguration` message, which instructs to proceed for handover, the UE connects to the

rBS instead of tBS. As a consequence, since the tBS is never reached by the expected user, it will never transmit the AT. Hence, there is no message to be replayed by the attacker, proving that BARON is resistant against replay attacks. Furthermore, the only other opportunities in which the AT is transmitted are (i) during the transmission between the UE and the sBS (Fig. 4.1 step ①), and (ii) during the transmissions between the sBS and sAMF, and between the sAMF and the tBS (Fig 4.1 steps ② and ③, respectively). While for (i) we have a wireless communication, thus being easier for the attacker to sniff the transmission, in cases (ii) the message transmission is wired, requiring that attacker wiretaps the communication between involved nodes. However, even assuming the attacker manages to gather the messages carrying the AT, in both cases the AT is encrypted with K_{CTE} first, and with the shared key between the communicating parties afterwards. Such a double encryption provides a double layer of security, not giving the attacker the possibility to read the AT value during the two transmissions. This means that in order to be able to retrieve the value R to correctly reply the challenge, the attacker must learn the corresponding encryption keys, which is not possible by assumption. Indeed, we assumed the AKA protocol being secure, and the attacker is assumed not being able to tamper with the network nodes. Hence, the attacker is only left with a random guess or giving no answer. In the latter case, we already discussed that the UE considers the handover being unsuccessful, and it initiates then a connection recovery procedure; in the first case, instead, the correct guess probability decreases as the length (in bit) of AT increases. For sufficiently large values of AT the attack success probability is then close to 0.

The exact same reasoning applies to BARON with *Authentication Mechanism 2* and Xn-handover as well, thus proving the overall security of BARON authentication methodology.

4.6 BARON: Recovering Connection to a Legitimate Base Station

The BARON authentication mechanisms described in previous sections allow the UE to determine if a tBS with which it has established a connection is legitimate. However, since BARON does not prevent the UE from connecting with a rBS, it will be paramount to provide a fast and efficient mechanism to recover connection to a legitimate BS.

The recovery mechanism that we propose follows similarly to the BARON Authentication Mechanisms. The objective is to allow a UE that is the victim of an FBS attack to efficiently and securely recover connection with a legitimate BS. When instructed to proceed for handover,

the UE receives an additional token termed as *Recovery Token* (RT). This token will be used to quickly recover a connection in case of handover failure, while at the same time ensuring legitimacy of both, the UE and the new tBS. In case of handover failure, the UE initiates the recovery procedure by selecting a new tBS, but different from the previous one. As already discussed, this is usually the BS with the highest (in this case the second highest) signal strength. Hence, the UE transmits a `Connection Recovery` message containing UE_{info}, ID_{CTE} and RT' (computed from RT). Then, as well as for handover or initial access, it starts an internal timer, computes RT'', and expects to receive the same value from the tBS, which only the CTE could have correctly computed. If the received value $\tilde{RT}''$ matches RT'', the UE deems the new tBS as legitimate. Depending on the new tBS we can then have three possible scenarios:

1. **The new tBS coincides with the previous sBS**. In this case, since the sBS did not receive the `Context Release` command due to handover failure, the security context between the UE and sBS can be considered still valid. In this case, in order to speed up the re-connection process we make the sBS being the CTE.

2. **The new tBS does not coincide with previous sBS, but belongs to the sAMF**. In such a scenario, the sAMF will be the CTE. We need to highlight that in this case it must be the sAMF being the CTE although there could be an Xn interface between the sBS and the new tBS. Indeed, the presence of such a direct communication between the BSs may be transparent to the UE, which would then be unable to recognize whether the sBS can be a valid CTE, thus increasing the re-connection delay if this was not the case. Besides, making the sAMF being the CTE would also speed up the process of recognisance from the CN of the handover failure, avoiding the transmission of an additional and dedicated transmission to the AMF.

3. **The new tBS does not belong to the sAMF**. In this case also, the sAMF is the CTE since, as well as for the handover case, it can reach the tAMF which in turn can reach the tBS.

We can notice that the described procedure reduces the recovery of a legitimate connection to that of initial access, where the RT' simultaneously serves as the MAC_{UE} first, and AT later. Indeed, the reception of the correct RT' value signals to the CTE the legitimacy of the UE. Afterwards, receiving the correct RT'' from the tBS proves its legitimacy to the UE.

Consider an intra-AMF N2-handover scenario using BARON *Authentication Mechanism 1* where the UE is victim of an FBS attack, and assume that the UE reconnects to the sBS. The

connection recovery mechanism develops as follows (Fig. 4.5):

Legitimate Connection Recovery (Fig. 4.5):

① AMF → sBS: `Handover Command`

② sBS → UE: (`RRC Reconfiguration`, RT)

- $RT = E_{K_{CTE}}(M)$ | M = random number, $E_K(\cdot)$ = encryption, key K

③ After RACH, rBS → UE: $\tilde{AT}'$

④ UE: verifies for $AT' = \tilde{AT}'$.
For an n-bit message, with probability $1 - 2^{-n}$: $\tilde{AT}' \neq AT'$

⑤ UE → sBS: (`Connection Recovering`, UE_{info}, ID_{CTE}, $\tilde{RT}'$)

- UE_{info} = user information
- ID_{CTE} = identifier of the CTE
- $M = D_{K_{CTE}}(RT)$ | $D_K(\cdot)$ = decryption, key K
- $M' = H(M)$ | $H(\cdot)$ = any algorithm
- $\tilde{RT}' = E_{K_{CTE}}(M')$

⑥ sBS: verifies for $\tilde{RT}' = RT'$

⑦ sBS → UE: (`Re-connection Accepted`, $\tilde{RT}''$)

- $M'' = H(M')$
- $\tilde{RT}'' = E_{K_{CTE}}(M'')$

⑧ UE: verifies for $\tilde{RT}'' = RT''$

4.6.1 Re-connection Token Computation

We need now to define which entity between the sBS and sAMF should compute the RT value, as in the above discussion we concluded that in the three possible scenarios these can be the CTE. Let us analyze the advantages and disadvantages for each of the two options.

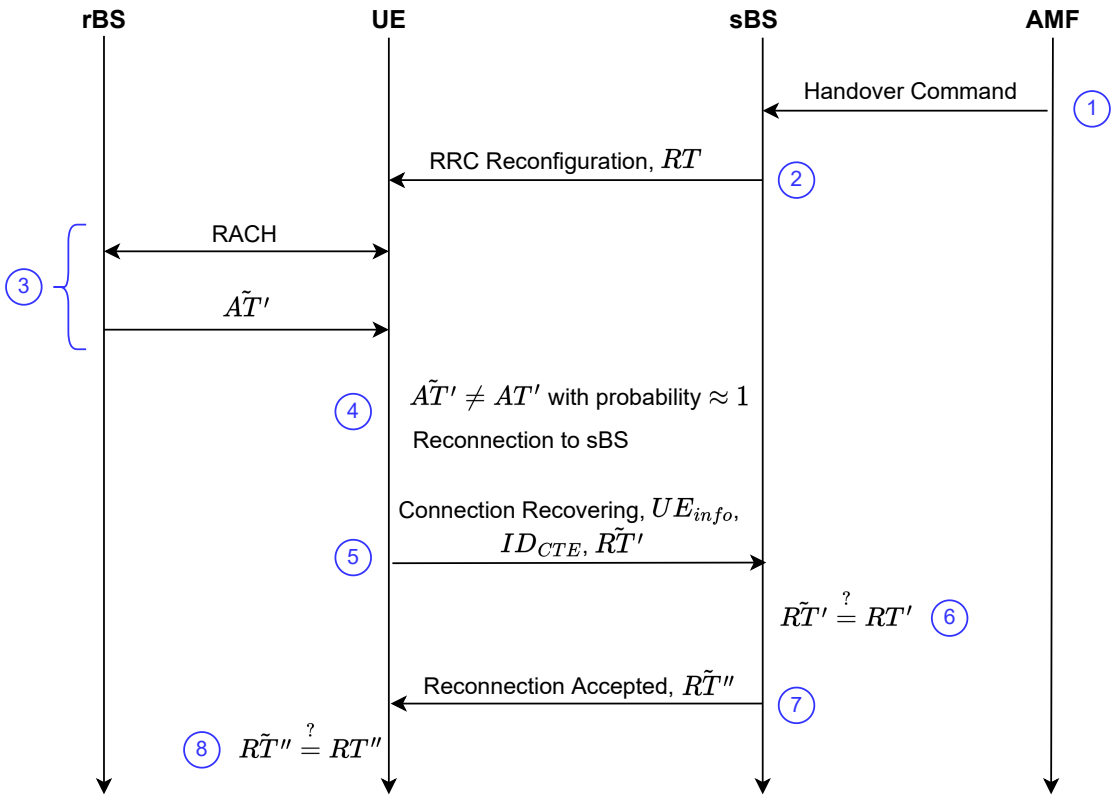


Figure 4.5: Example of BARON legitimate connection recovery procedure after an FBS attack. At the end of the procedure, the UE reconnects with the sBS.

Following the very similar reasoning for which we previously decided to let the sBS being the CTE in the first scenario, if we let the sBS to compute the RT we clearly have the advantage of a faster re-connection since we do not need to pass through the AMF. Besides, we do not introduce any communication overhead between the sAMF and sBS for the transmission of the RT (which indeed would need to be delivered to the UE through a trusted node, the sBS). However, the drawback of this solution is that the UE could efficiently reconnect only to sBS.

Consider now the case in which it is the sAMF to compute the RT. While it is true that this would introduce a (minimal) transmission overhead between the sAMF and sBS, and it would require a longer re-connection time since we need to involve the AMF, on the other hand the UE could efficiently reconnect to any other BS under the control of that AMF. Lastly, for the third scenario presented, we can see that the solution of letting the AMF to compute the RT is better, as this would avoid two rounds of transmission (between the previous sAMF and sBS and the corresponding reply of the sBS) which would increase the re-connection required time.

Table 4.1: Comparison of advantages and drawbacks for *Re-connection Token* (RT) computation (i) by sBS and (ii) by AMF.

RT computation method	Advantages (✓) and Disadvantages (×)
sBS computes RT	✓ Faster re-connection since there is no need to pass through the AMF ✓ No transmission overhead introduced between sBS and AMF × Can reconnect only to sBS if no direct communication between BSs (Xn-configuration)
AMF computes RT	✓ Can reconnect to any reachable BS under that AMF × Introduces transmission overhead between sBS and AMF × Longer re-connection time

Table 4.1 summarizes the pros and cons just discussed.

In order to provide the higher efficiency for the recovery mechanism described, we propose to adopt a hybrid strategy to overcome the limitations of each of the two solutions. Hence, it is the sAMF that initially generates the RT. However, if the UE tries to reconnect with the sBS, the RT value is treated as if it was the random number M, thus $RT' = H(RT)$. This will allow an immediate re-connection to the sBS, without requiring an intervention from the sAMF. If instead, the new tBS is not the same as sBS, we will need to pass through the sAMF. Such a hybrid solution allows selection of the best re-connection strategy depending on the specific situation, being flexible to the specific re-connection scenario. Furthermore, in the case of connection recovery with the sBS when a deterministic algorithm $H(\cdot)$ is used, an attacker sniffing the communication channel may be able to easily compute the value of RT' $(= H(RT))$ and thus increasing the probability of success for a FBS attack. To prevent this possibility, we shall transform the value of RT before passing it as an input to $H(\cdot)$. One very simple way to carry out this transformation, while also encrypting RT, is through an XOR operation between RT and (a portion of) the key K_{gNB} that was previously shared between the UE and sBS. This process will ensure that only the legitimate UE could have computed the correct value of RT'.

The security discussion for the defined re-connection procedure follows the very similar reasoning as for the authentication part, thus proving the security of this procedure as well.

Lastly, we observe that such a recovery mechanism is required only for standard handover, since it will be sufficient to change the tBS in the case of initial access. The DAPS handover scenario [10] also does not require a dedicated connection recovery mechanism. In fact, in this setting, since the UE does not drop connection with the sBS until successful RACH with tBS, or until reception of the AT in case of using BARON, when a tBS is identified as not legitimate the UE can simply fall back to the connection with the sBS.

5

BARON Performance Evaluation

In this chapter we discuss the performance evaluation of BARON. All numbers we report are computed as the average over 10 turns (each of 1000 runs) of a self-contained software simulation, fully written in C++. The source code is publicly available[1]. The performance of BARON is quantified in terms of two metrics:

1. Overhead induced due to the computation, transmission, and evaluation of the AT and RT values, and

2. Time required for connection recovery after an FBS attack, compared to the handover required time in case of no FBS attack.

We summarize our main findings in Tab. 5.1, which indicates that the induced overhead when using BARON is minimal. Tab. 5.1 also shows that the proposed connection recovery mechanism is fast and efficient, as the time required for re-connection following an FBS attack is a fraction of the time required for handover completion.

In what follows, we first discuss about the experiment setup giving details of how we implemented our experiments, defining the specific network topology scenario and functions implementation. Then, we present BARON performances and compare them to the state-of-the-art method from [12].

[1] https://github.com/aleLtt/BARON_simulation.git

Table 5.1: Performance of BARON in terms of induced overhead and time required for re-connection following an FBS attack.

	tBS is sBS	**tBS in sAMF**	**tBS not in sAMF**
BARON overhead*	N.A.	$< 1\%$	$< 1\%$
BARON re-connection time	0.25 - 0.30 T_1^{**}	0.65 - 0.70 T_1	0.25 - 0.30 T_2^{**}

*Overhead is given comparing the BARON handover time and 3GPP standard handover time in case of no FBS attack.

** T_1 = BARON handover time with tBS in sAMF; T_2 = BARON handover time with tBS not in sAMF.

5.1 Simulation Setup

Our experiments to evaluate the performance of BARON consist on a software simulation of a N2-handover scenario. Since our analysis is focused on the induced overhead for AT, RT computation and transmission, we decided to reduce to the minimum the implementation of the common procedures between the standard 3GPP handover and the modified procedure using BARON. This implies that we implemented a high-level simulation in which we abstract all the physical procedures that would be performed in a real-life scenario. For example, we avoid implementing the filtering and recognisance of the ID of the BSs, as well as the beam swapping procedure performed by the UE in order to steer the antenna in the direction of the transmitter. Such an abstraction also includes the data that is transmitted in the exchanged messages. Since our evaluation is focused on the overhead, this abstraction does not affect the correctness and reliability of the experiment results. Indeed, the abstracted procedures are common for both standard and with BARON handovers, thus not giving any contribution in the overhead analysis. Moreover, this high-level simulation allows to reduce the complexity of the implementation of our experiments.

The simulation of the communication between nodes is implemented as follows. We build a data structure, called as `msg`, which is an array with as many elements as the number of the entities in the simulation, adapting whether we consider an attack scenario or not (Fig. 5.1). Each entity of the simulation is assigned to an element of this array. The value of each element of the `msg` data structure is a pointer to the memory allocated for the `message` data type. The latter is a data type we defined to represent the exchanged message between parties, and that is pro-

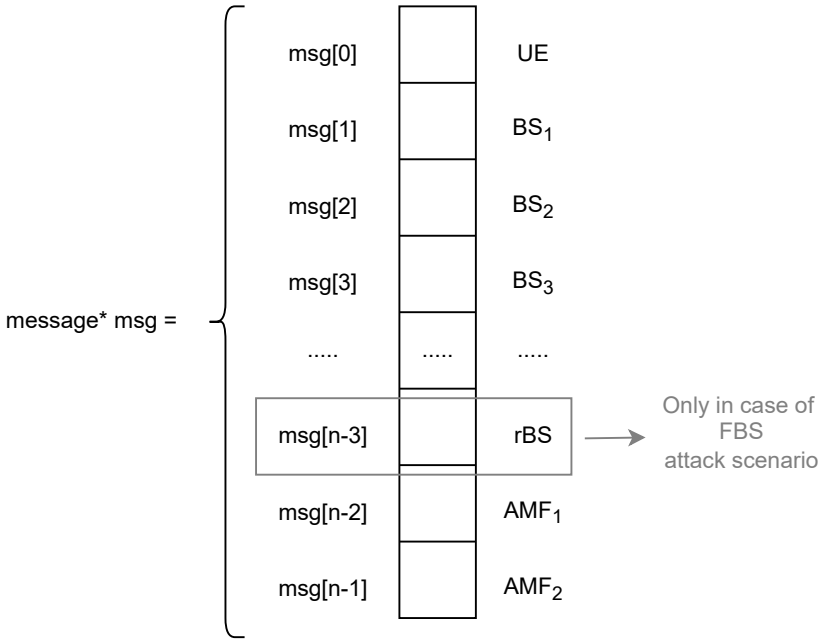
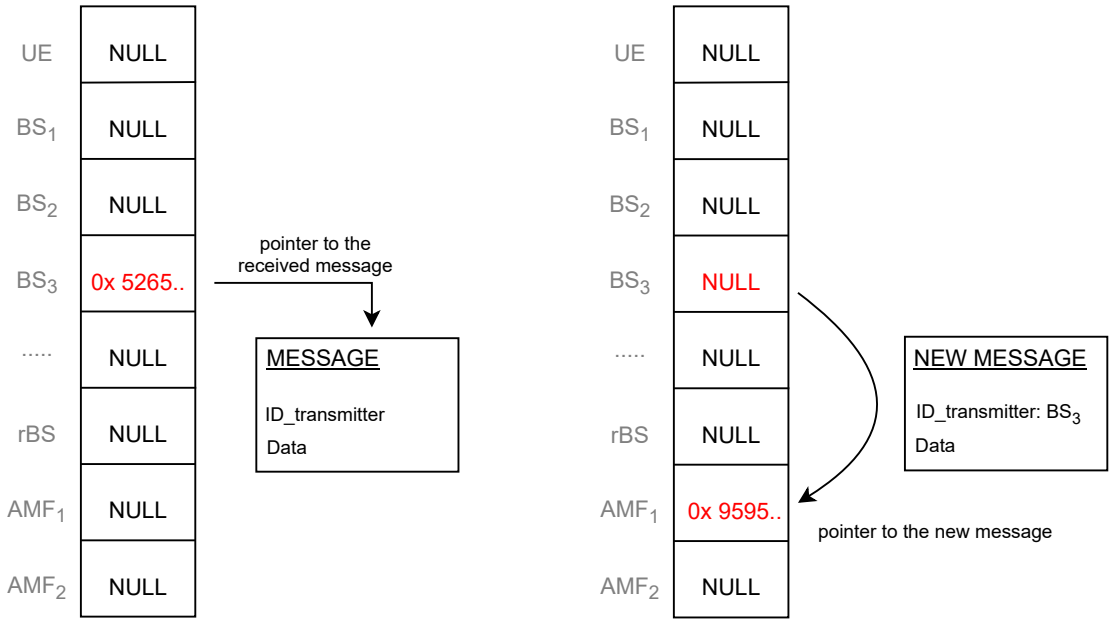


Figure 5.1: Implementation and abstraction of the communication channel for the simulation.

vided with the proper attributes that represent the message content. Through the pointer, it is possible to access to the information contained in the referenced message. All the elements of the array are set to `NULL`, with the only exception being that one element corresponding to the network node that has received the message (Fig. 5.2a). Upon reception of the message, namely having the pointer different to `NULL`, the node handles the message, extracts and processes the needed information, builds and transmits the proper response message. The simulation of the transmission consists on setting to `NULL` value its element of the `msg` data structure, and copying the pointer to the new message into the entry corresponding the receiver (Fig. 5.2b).

5.1.1 Simulation Scenario

We carry out extensive experiments to evaluate the performance of BARON by simulating an N2-handover scenario. Fig. 5.3 gives a graphical representation of the network BSs deployment we considered for our experiments. In a 2-dimensional plane, we have two AMFs controlling a set of BSs each. The dashed line divides the plane in such a way that BSs belonging to the same region are under the control of the corresponding AMF. While the BSs and AMFs location is fixed, the UE is randomly placed in the plane at the start of each iteration of the simulation. Besides, we assume that the UE has a connection to a legitimate BS at the start of the run. We



(a) Message reception. **(b)** Message transmission.

Figure 5.2: Message reception and transmission simulation.

choose the sBS to be the second-nearest BS to the UE. As a result, the nearest BS will be selected as the tBS for handover. The received signal power (PR) from base station i (BS_i) is modelled according to standard signal power propagation:

$$PR_i = \frac{PT_i}{d\left(UE, BS_i\right)^2}, \tag{5.1}$$

where $d\left(A, B\right)$ is the distance between A and B, and PT_i is the transmission power of BS_i. All the BSs are assumed to have the same transmission power. To simulate the computation of the signal power received from each BS, we used the same solution approach as for message transmission. Thus, we created a matrix `channel` in which elements in the first column are filled with the power level received by the UE from the associated to that row element, while elements in the second column report the ID of the BS. This simulates the transmission and reception of the beacons.

In the presence of an attacker carrying out an FBS attack, we randomly place the rBS within a range of 150 m from the UE's position. The rBS uses a BS identifier assigned at random, but different from that of the sBS. We additionally ensure that the rBS has a higher transmission power in order to maximize the probability of being under an FBS attack scenario.

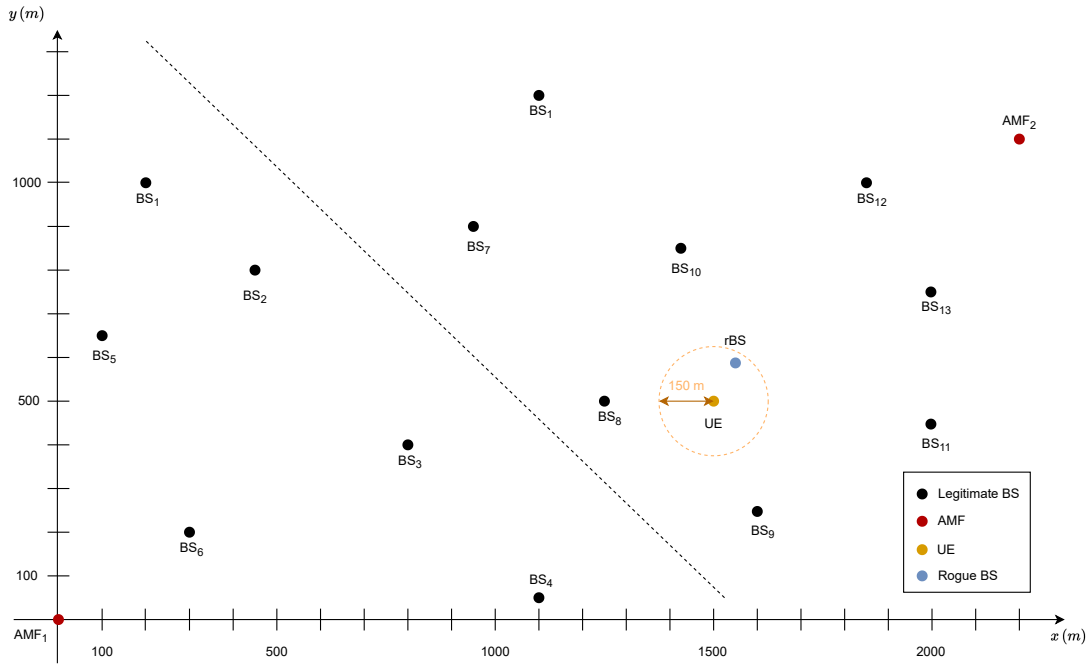


Figure 5.3: Simulation scenario for BARON performance evaluation.

The specificity of the network structure considered does not influence the results for the overhead analysis. Indeed, for the considered context, the network topology mostly affects the signal propagation time between nodes. However, this aspect is the same for both the standard and with BARON handover versions, hence is a common factor that cancels out during the overhead analysis. On the other hand, this does not hold for the connection recovery time analysis, which requires the exchange of more messages compared to the only handover procedure. Being aware of this, we decide not to give the absolute times of the re-connection required time, but to present them in relation with the time it takes for handover completion with the same network structure (with no FBS attack). This allows to make an objective evaluation of the efficiency of the proposed mechanism.

5.1.2 Handover Completion Time Computation

In our experiments, we assume that all communications between the UE and BSs are wireless. On the other hand, communication between BSs and AMF, and between two AMFs are wired. The wireless and wired media have different speeds of light: $3 \times 10^8\ m/s$ and $2 \times 10^8\ m/s$, respectively. The total time required to complete a handover will depend on times associated

with transmission delays and message handling. The former represents the time required for a message to reach the destination, while the latter quantifies the time required to generate a response message after receiving an incoming message. We compute the transmission delay (TD) between nodes A and B as:

$$TD = \frac{d(A,B)}{c}\,, \tag{5.2}$$

where c is the speed of light, set according to the transmission medium. For computing the handling time we used the C++ standard "*chrono*" library, which allows for a time precision of around the nanosecond.

To evaluate BARON, we use the following procedure:

1. Run the simulation to obtain 1000 samples for each scenario considered (Table 5.1);
2. For each scenario, determine the median of the 1000 samples;
3. Repeat steps (1)-(2) 10 times;
4. Compute the arithmetic mean (average) over the collected median values.

The median is used to eliminate outlier samples. In fact, as shown in Fig. 5.4, in our experiments we observed that the magnitude of a very small number of outliers was very large compared to all the other samples. In such a scenario, using the arithmetic mean would have resulted in misleading values of overhead and connection recovery times. Using the median, instead, allows to limit but not removing the influence of such outliers.

5.1.3 Functions Implementation

In this section we give the details of the security parameters and functions we employed for our experiments. With regard to the security parameters, in our implementation we use 32-bit random numbers R, M, which represent a balanced trade-off between security and memory overhead. The length of the random numbers can be suitably adjusted, based on the needs of service providers. For the security functions, we used the following in our experiments:

- For encryption and decryption, functions $E(\cdot), D(\cdot)$, respectively, we use a custom implementation of AES-128 algorithm. With *"custom"* we mean that we implemented the crypto functions writing the code without using any third-party libraries except the C/C++ standard libraries.

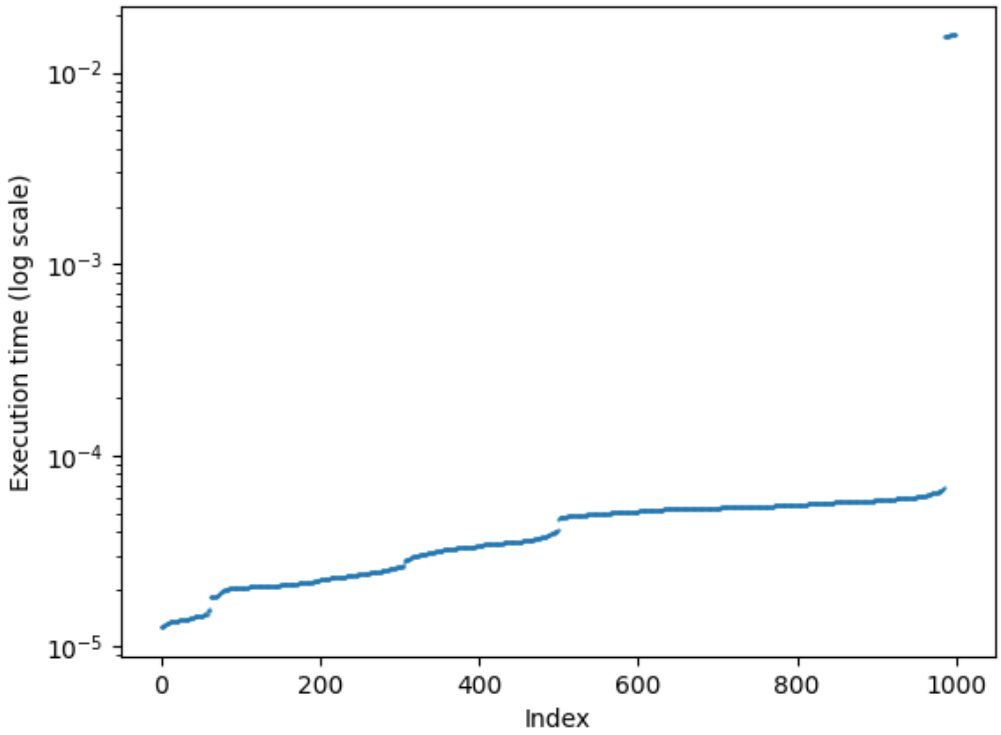


Figure 5.4: Example of simulation results with outliers empathizing. On the right corner we can notice few outliers that have an order of magnitude much higher than the rest of the data. Values reported are given in a logarithmic scale so to emphasize the outliers.

- We use a simple, but effective deterministic function $H(x) = x+1$ in order to process the response for AT. This choice of $H(\cdot)$ introduces minimal overhead. At the same time, the security of BARON is not compromised since AT will be encrypted at a subsequent stage.

- $H'(x) := x \oplus K_{gNB}$, where $\oplus$ is the binary XOR operation. This function is applied to the RT during re-connection with sBS in order to compute the value of RT' in the `Connection Recovery` message (Fig. 4.5 ④).

5.2 BARON: Induced Overhead

We evaluate the time overhead induced by BARON by comparing the time required for a handover using a standard 3GPP procedure with the time required to complete a handover when using BARON. In this case, we assume that there is no FBS attack. This allows us to examine

the additional time that will be required to manage and transmit values of AT and RT when using BARON. We separately evaluate the cases wherein the tBS is under the control of the sAMF and when it is not in the sAMF. Fig. 5.5 shows the comparison between the actual times taken to complete a handover using the standard 3GPP procedure (orange bars) and when using BARON (blue bars). We observe that the time taken to complete the handover when using BARON is almost equal to the time taken when following the standard 3GPP procedure. **The overhead induced by BARON is** ~ 43 *ns*, **which is about** $10000\times$ **lower** than the overhead reported in [12] (0.53 *ms*).

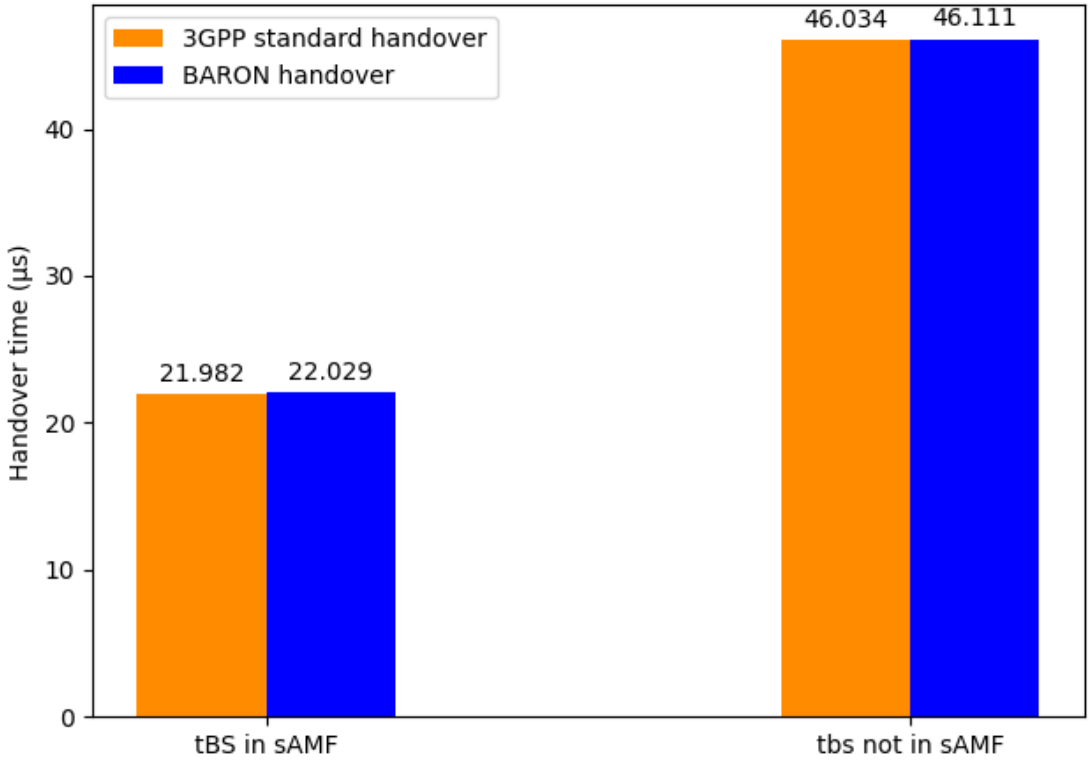


Figure 5.5: BARON overhead evaluation. Comparison of times for handover completion between the standard 3GPP procedure and BARON with *Authentication Mechanism 1* when there is no FBS attack. The additional overhead introduced by BARON in both cases is negligible.

5.3 BARON: Connection Recovery Time

To evaluate the connection recovery time, we consider a scenario where an adversary is carrying out an FBS attack. In this setting, we measure the time required to recover connection to a

legitimate BS when using BARON. Since the absolute value of the time needed for connection recovery strictly depends on the specific network topology, we present our results as a fraction of the time required for BARON handover completion in the case of no FBS attack. Let:

- T_1 be the time for handover completion using BARON when tBS is under the control of the same AMF as sBS (sAMF).
- T_2 be the time for handover completion using BARON when tBS in not under the same AMF as sBS.

We evaluate the time required for re-connection when the tBS is (i) the same as the sBS, (ii) in the same AMF cluster as the sBS, and (iii) not in the same AMF cluster as the sBS. Further, we implement an active attacker that randomly guesses the value of AT'.

Fig. 5.6 compares the time for handover completion when using BARON in the absence of an FBS attack (orange bars) and the (total) time for handover completion and connection

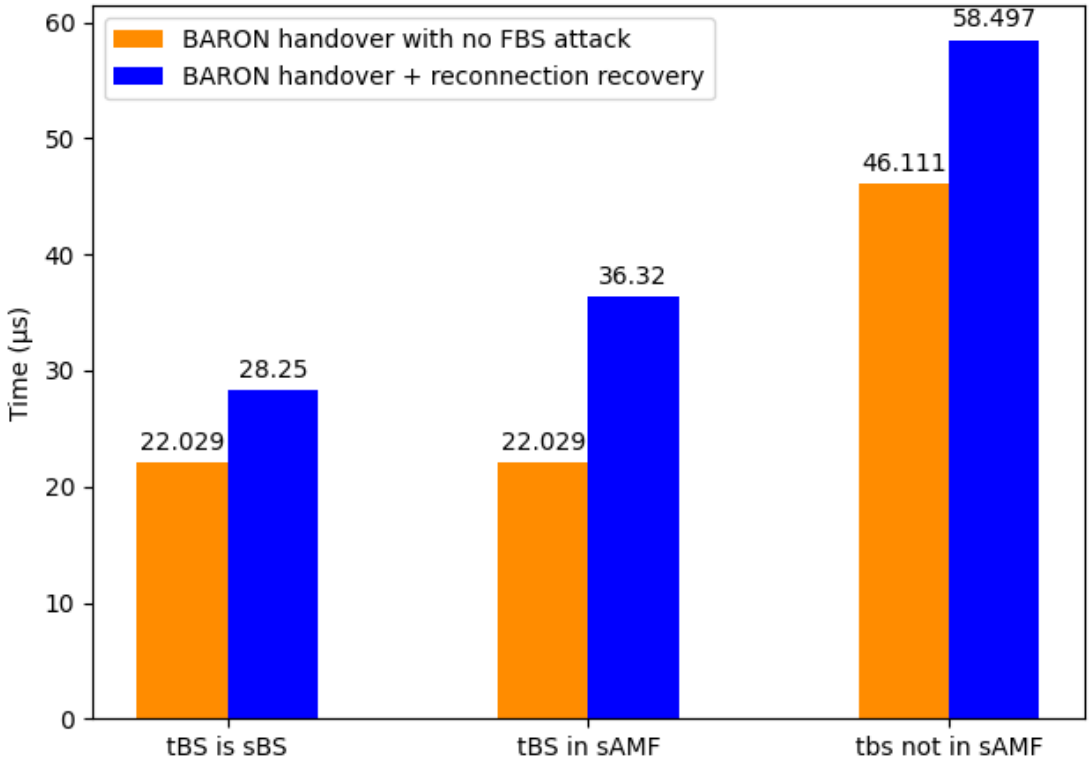


Figure 5.6: BARON connection recovery evaluation. Comparison of times between handover completion using BARON with *Authentication Mechanism 1* in absence of an FBS attack, and total time for handover and connection recovery completion in case of an FBS attack.

recovery when using BARON in the presence of an FBS attack (blue bars) for the scenarios (i) - (iii). We observe that the additional time for connection recovery is of a similar order of magnitude as the time required for handover completion. Based on our results from Sec. 5.2, we can conclude that the additional time required to re-establish connection to a legitimate BS is almost entirely associated with transmission delays rather than the computation and verification of values of RT. Moreover, we notice that the hybrid strategy defined in Sec. 4.6 allows for a significant reduction in the re-connection time.

Our results in this section reveal that **the total time for handover completion and re-connection to a legitimate BS using BARON is still lower than the** 0.53 *ms* **overhead** presented in [12]. Reporting the re-connection time as a fraction of the time required for handover completion will also allow our experiments to be extended for arbitrary network topology and UE location.

6
Related Works

A large part of the existing research in the security of 4G and 5G cellular networks focuses on the identification of weaknesses and the design of countermeasures to overcome these weaknesses [36, 37, 38, 39, 40, 41]. These solutions, however, are not adequate or designed to defend against an FBS attack. Solution that focus on misbehaving BSs can be grouped depending into three categories according to the implemented defense approach: (i) addition of integrity protection values to the broadcast messages, (ii) use of FBS detection tools, and (iii) use of digital signatures for the broadcast messages.

Integrity protection to broadcast messages. These kind of solutions aim to ensure that the received message has not been altered by an attacker. A cryptographic value is then computed using a shared key between the BS and UE, and attached to the message itself. The UE then verifies if the crypto value corresponds to the expected one, and in case of positive check it accepts the message. Examples of proposals following such a defense approach can be found in [13, 34, 42]. However, such an approach is not effective against FBS attacks as described in Sec. 3.2. The attacker can in fact simply replay the beacons without any modification, and the message will be accepted anyway as it has not been modified. Besides, an attacker might be able to deceive such a defense mechanism by carrying out a bidding-down attack, as noted in [12].

FBS detection. Solutions following this defense approach are designed to identify inconsistencies in the content of `MR` messages and deployment information (*e.g.*, BS identifier, operation

frequency) of a legitimate BS [1, 43, 44]. Furthermore, the integration of FBS detection mechanisms with machine learning techniques is becoming increasingly popular [45, 46].

Particularly relevant mechanisms in this field are proposals in [43] and [44]. In the former, authors propose FBS-Radar, a FBS detector based on the analysis of crowd-sourced data and spam messages to accurately geolocating FBSs, while maintaining users' privacy at the same time [43]. In the second mechanism, authors propose FBSleuth, which instead defines a FBS detector for forensic analysis that makes use of radio frequency fingerprinting to characterize the message transmitter, thus proving the misbehaviour of a BS [44].

Despite being great solutions to detect malicious and non-legitimate BSs, FBS detection tools do not alleviate the FBS vulnerability cause pointed out in [12, 14], neither the one identified by us in the design of BARON. Therefore, they are not adequate to defend against FBS attacks in an online scenario, *i.e.*, during initial access or handover, as the detection process is carried out offline. Furthermore, recent works in [13, 47] has demonstrated that such solutions can be ineffective.

Digital signatures to broadcast messages. In this kind of approach, beacons transmitted by BSs are authenticated through digital signatures using asymmetric cryptography. This defense approach represents the current state-of-the-art defense methodology against FBS attacks since it better addresses the vulnerability root cause previously discussed. Digital signature schemes to authenticate broadcast messages, either using a PKI infrastructure or a certificate authority, have been proposed in [12, 14, 15, 16].

In [14] authors propose an optimized PKI infrastructure-based solution to authenticate SIB1 and SIB2 messages. Challenges that were identified in [14] are related to the management of a PKI infrastructure, and they include the size of the certificate, vulnerabilities to replay attacks, and public key revocation. These challenges were faced by the use of a custom encoding to limit the size of the certificate, a location-dependent parameter to mitigate replay attacks, and a time-based expiration mechanism for public keys, respectively. Several signature implementation schemes are evaluated, with the smallest overhead reported in [14] was $\sim$ 176*ms*. However, this solution is not fully secure against replay attacks [14].

The same research group proposes in [12] a Schnorr-HIBS digital signature-based scheme with hierarchical key derivation. This work does not rely on a PKI, rather it introduces a Private Key Generator (PKG) node to generate private keys from a master secret. The PKG is embedded within the CN and distributes the generated keys to participating entities. These entities then generate private keys for lower-layer entities. The hierarchical key derivation proceeds up

to the AMF generating private keys for the BSs. The BSs authenticate the SIB1 messages with their private key, and attach to it the corresponding public key for verification. The security of this mechanism is guaranteed by the hierarchical key derivation process, which binds the BSs' private and public keys to those of the CN entities. The proposed scheme, thanks to optimizations, managed to reduce communication overhead by 31% compared to [14, 15, 16], and incurred a fixed end-to-end delay of $0.53ms$ [12].

In order to address the replay attack vulnerability, both proposals in [12, 14] make use of a time-dependence signature validity. Together with the authenticated message(s), a signature expiration time is attached aiming to harden the success in replaying the message(s) from an attacker and being accepted from victim UEs. However, this solution introduce some computation overhead, and since the expiration time must be short enough to ensure replay attacks resistance, this increases the number of broadcast messages transmitted. Such a signature lifetime must be carefully and precisely computed. However, this may not be enough, and these solutions may still be vulnerable to replay attacks.

7 Conclusion

In this thesis, we have emphasized the vulnerability of the cellular network technology to **Fake Base Station (FBS) attacks**, which enable the adversary to control the UE connection, and proposed BARON, a secure framework for initial access and handover in 5G networks to mitigate these attacks. Consequences of the FBS attack not only may result into harm and privacy threats to the users, but can also affect the network reliability and induce a waste of resources. We have then first provided all the needed background to understand how cellular networks, and in particular 5G networks, are defined and provide connection establishment and management. Then, the FBS attack evolution steps are presented, together with the possible consequences and impact it may have. We therefore highlighted the importance of developing secure and efficient mechanisms to possibly prevent or limit the impact of such an attack. These solutions shall also introduce minimum overhead, both in terms of time and computational resources, and infrastructure changes, thus being backward compatible as much as possible. Such requirements are particularly strict for 5G networks, in which the use of mmWave requires a dense BS deployment, leading then to very frequent handovers.

To the purpose of addressing the above vulnerability and meeting the necessary security requirements for cellular communications, BARON aims to provide a mean to the UE to verify the legitimacy of the BS it is connecting to, thus defending against FBS attacks. In order to accomplish this objective, BARON relies on the **Chain of Trust** that is established by the UE with the serving network through the AKA procedure. Hence, in developing BARON we introduced the concept of **Closed Trusted Entity (CTE)**, which is that trusted entity of the

network for which the UE has a valid security context and that acts as a guarantor for legitimacy of the BS the UE is connecting to. After connection, the UE expects then to receive from the BS an **Authentication Token** that only the CTE could have computed correctly. Therefore, reception of the correct token proves the interaction of the BS with the CTE, thus the legitimacy of the BS itself since it shows it belongs to the serving network. We have then proposed and detailed two possible mechanisms to implement BARON authentication logic. The proposed methodology, however, does not prevent a UE to connect to a rBS. Therefore, BARON must also come with a **legitimate connection recovery procedure**, which, thanks to a **Re-connection Token** received by the UE at handover command, enables a victim UE to efficiently and securely establish connection with a legitimate BS. The logic of the procedure follows the same of the authentication framework. Overall, BARON enables the UE to (i) determine whether a BS it is connecting to is legitimate or not, and (ii) efficiently recover a legitimate connection when subject to a FBS attack.

We must further highlights that BARON is the authentication logic, thus the idea of relying on the chain of trust and use an already trusted entity to authenticate a new one. BARON then abstracts from the specific implementation mechanism, which can be adapted and designed according to the needs and constraints of the specific service provider.

In order to evaluate BARON performance, we carried out extensive **experiments** by implementing a software simulation which replicates a handover scenario where an adversary is running a FBS attack. The performance evaluation is based in terms of the **time overhead** introduced during handover, and **effectiveness in recovering a legitimate connection** in case of an FBS attack. Our experimental results revealed that BARON introduces an overhead that is less than 1% of the time required for standard 3GPP handover completion. This result is 10000× lower than the overhead reported in [12]. Besides, in case of being victim of a FBS attack, the time taken by a UE to recover connection to a legitimate BS using BARON is of the same order of magnitude as the time required for handover completion in the absence of an attack. This proves that BARON fully meets the security and performance requirements for security solutions in the context of 5G networks.

Compared to existing current state-of-the-art approach discussed the literature, BARON offers several advantages. Firstly, it does not require any infrastructure modifications or new introduction, but rather only necessitates a software update to network entities and UEs, ensuring backward compatibility and minimal adoption efforts. Additionally, BARON is immune to replay attacks within the considered threat model, and exhibits **negligible overhead** and resource utilization. These qualities position BARON methodology as a secure, efficient, and

practical authentication methodology suitable for real-life networks.

In conclusion, this thesis highlights the significance of addressing the vulnerability of cellular networks to FBS attacks and presents BARON as a promising solution. It provides a concise summary of the research findings, showcases the efficacy of the proposed framework, and underscores its superiority over existing approaches.

References

[1] 3GPP, "Security architecture and procedures for 5G system," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 33.501, 09 2022, version 17.7.0. [Online]. Available: https://portal.3gpp.org/ChangeRequests.aspx?q=1&specnumber=33.501

[2] D. Kombate and Wanglina, "The internet of vehicles based on 5g communications," in *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2016, pp. 445–448.

[3] A. Gupta and R. K. Jha, "A survey of 5g network: Architecture and emerging technologies," vol. 3, 2015, pp. 1206–1232.

[4] Y. Yang and K. Hua, "Emerging technologies for 5g-enabled vehicular networks," vol. 7, 2019, pp. 181 117–181 141.

[5] S. Dananjayan and G. M. Raj, "5g in healthcare: how fast will be the transformation?" vol. 190, no. 2. Springer, 2021, pp. 497–501.

[6] I. Ahmad, S. Shahabuddin, T. Kumar, J. Okwuibe, A. Gurtov, and M. Ylianttila, "Security for 5g and beyond," vol. 21, no. 4, 2019, pp. 3682–3722.

[7] A. Aijaz, "Private 5g: The future of industrial wireless," vol. 14, no. 4, 2020, pp. 136–145.

[8] S. Sullivan, A. Brighente, S. A. P. Kumar, and M. Conti, "5g security challenges and solutions: A review by osi layers," vol. 9, 2021, pp. 116 294–116 314.

[9] N. Akkari and N. Dimitriou, "Mobility management solutions for 5g networks: Architecture and services," vol. 169, 2020, p. 107082. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128619306346

[10] 3GPP, "5G; NR; Radio Resource control (RCC); Protocol specification," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.331, 07 2019, version 15.6.0. [Online]. Available: https://portal.3gpp.org/ChangeRequests.aspx?q=1&specnumber=38.331

[11] E. Bitsikas and C. Pöpper, "Don't hand it over: Vulnerabilities in the handover procedure of cellular telecommunications," in *Annual Computer Security Applications Conference*, ser. ACSAC '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 900–915. [Online]. Available: https://doi.org/10.1145/3485832.3485914

[12] A. Singla, R. Behnia, S. R. Hussain, A. Yavuz, and E. Bertino, "Look before you leap: Secure connection bootstrapping for 5g networks to defend against fake base-stations," in *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, ser. ASIA CCS '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 501–515. [Online]. Available: https://doi.org/10.1145/3433210.3453082

[13] A. Shaik, R. Borgaonkar, S. Park, and J.-P. Seifert, "On the impact of rogue base stations in 4g/lte self organizing networks," in *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, ser. WiSec '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 75–86. [Online]. Available: https://doi.org/10.1145/3212480.3212497

[14] S. R. Hussain, M. Echeverria, A. Singla, O. Chowdhury, and E. Bertino, "Insecure connection bootstrapping in cellular networks: The root of all evil," in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1–11. [Online]. Available: https://doi.org/10.1145/3317549.3323402

[15] C.-C. Lee, I.-E. Liao, and M.-S. Hwang, "An extended certificate-based authentication and security protocol for mobile networks," vol. 38, no. 1, 2009.

[16] X. Yi, E. Okamoto, and K. Y. Lam, "An optimized protocol for mobile network authentication and security," vol. 2, no. 3. New York, NY, USA: Association for Computing Machinery, jul 1998, p. 37–39. [Online]. Available: https://doi.org/10.1145/1321387.1321391

[17] 3GPP, "5g; nr; nr and ng-ran overall description; stage-2," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.300, 07 2020, version 16.2.0. [Online]. Available: https://portal.3gpp.org/ChangeRequests.aspx?q=1&specnumber=38.300

[18] 3GPP , "Physical layer procedures for control," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.213, 07 2020, version 16.2.0. [Online]. Available: https://portal.3gpp.org/ChangeRequests.aspx?q=1&specnumber=38.213

[19] S. Abdelwahab, B. Hamdaoui, M. Guizani, and T. Znati, "Network function virtualization in 5g," vol. 54, no. 4, 2016, pp. 84–91.

[20] Techplayton.com, "5g authentication and key management 5g-aka," March 2021. [Online]. Available: https://www.techplayon.com/5g-authentication-and-key-management-aka-procedure/

[21] D. Basin, J. Dreier, L. Hirschi, S. Radomirovic, R. Sasse, and V. Stettler, "A formal analysis of 5g authentication," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1383–1396. [Online]. Available: https://doi.org/10.1145/3243734.3243846

[22] ShareTechnote.com, "5g/nr - initial access/rach." [Online]. Available: https://www.sharetechnote.com/html/5G/5G_RACH.html#:~:text=5G%20RACH%20in%20Details,to%20be%20'RACH%20process'.

[23] 3GPP, "5g; procedures for the 5g system (5gs)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 33.809, 1 2021, version 16.7.0. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/123500_123599/123502/16.07.00_60/ts_123502v160700p.pdf

[24] eventhelix.com, "Lte random access procedure," 2015. [Online]. Available: https://www.eventhelix.com/lte/random-access-procedure/lte-random-access-procedure.pdf

[25] E. Peralta, T. Levanen, F. Frederiksen, and M. Valkama, "Two-step random access in 5g new radio: Channel structure design and performance," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, 2021, pp. 1–7.

[26] W. T. Toor, A. Basit, N. Maroof, S. A. Khan, and M. Saadi, "Evolution of random access process: From legacy networks to 5g and beyond," vol. 33, no. 6, 2022, p. e3776, e3776 ett.3776. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.3776

[27] M. Tayyab, X. Gelabert, and R. Jäntti, "A survey on handover management: From lte to nr," vol. 7, 2019, pp. 118 907–118 930.

[28] Techplayton.com, "5g sa inter gnb handover – xn handover," October 2021. [Online]. Available: https://www.techplayon.com/5g-sa-inter-gnb-hanodver-xn-handover/

[29] Techplayton.com , "5g mobility scenarios – handovers," July 2022. [Online]. Available: https://www.techplayon.com/5g-mobility-scenarios-handovers/

[30] A. Peltonen, R. Sasse, and D. Basin, "A comprehensive formal analysis of 5g handover," in *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 1–12. [Online]. Available: https://doi.org/10.1145/3448300.3467823

[31] Techplayton.com, "5g sa inter gnb handover – n2 handover," October 2021. [Online]. Available: https://www.techplayon.com/5g-sa-inter-gnb-handover-n2-or-ngap-handover/

[32] Techplayon.com, "5g nr dual active protocol stack (daps) handover," October 2020. [Online]. Available: https://www.techplayon.com/5g-nr-dual-active-protocol-stack-daps-handover-3gpp-release-16/

[33] D. Zhao, Z. Yan, M. Wang, P. Zhang, and B. Song, "Is 5g handover secure and private? a survey," vol. 8, no. 16, 2021, pp. 12 855–12 879.

[34] 3GPP, "Technical Specification Group Services and System Aspects Study on 5G Security Enhancement against False Base Stations (FBS) (Release 17)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 33.809, 12 2020, version 0.12.1. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3539

[35] S. R. Hussain, M. Echeverria, O. Chowdhury, N. Li, and E. Bertino, "Privacy attacks to the 4g and 5g cellular paging protocols using side channel information," 2019.

[36] N. Golde, K. Redon, and J.-P. Seifert, "Let me answer that for you: Exploiting broadcast information in cellular networks," in *22nd USENIX Security Symposium (USENIX Security 13)*. Washington, D.C.: USENIX Association, Aug. 2013, pp. 33–48. [Online]. Available: https://www.usenix.org/conference/usenixsecurity13/technical-sessions/presentation/golde

[37] Y. Zeng, H. Guang, and G. Li, "Attribute-based anonymous handover authentication protocol for wireless networks," vol. 2018. Hindawi, 2018.

[38] Y. Qiu, M. Ma, and X. Wang, "A proxy signature-based handover authentication scheme for lte wireless networks," vol. 83. Elsevier, 2017, pp. 63–71.

[39] R. Piqueras Jover, "Security attacks against the availability of lte mobility networks: Overview and research directions," in *2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2013, pp. 1–9.

[40] B. Reaves, L. Blue, H. Abdullah, L. Vargas, P. Traynor, and T. Shrimpton, "Authenticall: Efficient identity and content authentication for phone calls," in *26th USENIX Security Symposium (USENIX Security 17)*. Vancouver, BC: USENIX Association, Aug. 2017, pp. 575–592. [Online]. Available: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/reaves

[41] P. Traynor, M. Lin, M. Ongtang, V. Rao, T. Jaeger, P. McDaniel, and T. La Porta, "On cellular botnets: Measuring the impact of malicious devices on a cellular network core," ser. CCS '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 223–234. [Online]. Available: https://doi.org/10.1145/1653662.1653690

[42] A. Shaik, R. Borgaonkar, S. Park, and J.-P. Seifert, "New vulnerabilities in 4g and 5g cellular access network protocols: Exposing device capabilities," in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 221–231. [Online]. Available: https://doi.org/10.1145/3317549.3319728

[43] Z. Li, W. Wang, C. Wilson, J. Chen, C. Qian, T. Jung, L. Zhang, K. Liu, X. Li, and Y. Liu, "Fbs-radar: Uncovering fake base stations at scale in the wild," in *NDSS*, 2017.

[44] Z. Zhuang, X. Ji, T. Zhang, J. Zhang, W. Xu, Z. Li, and Y. Liu, "Fbsleuth: Fake base station forensics via radio frequency fingerprinting," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, ser. ASIACCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 261–272. [Online]. Available: https://doi.org/10.1145/3196494.3196521

[45] V. T. Do, P. Engelstad, B. Feng, and T. van Do, "Strengthening mobile network security using machine learning," in *Mobile Web and Intelligent Information Systems*, M. Younas, I. Awan, N. Kryvinska, C. Strauss, and D. v. Thanh, Eds. Cham: Springer International Publishing, 2016, pp. 173–183.

[46] J. Jin, C. Lian, and M. Xu, "Rogue base station detection using a machine learning approach," in *2019 28th Wireless and Optical Communications Conference (WOCC)*, 2019, pp. 1–5.

[47] H. Yang, S. Bae, M. Son, H. Kim, S. M. Kim, and Y. Kim, "Hiding in plain signal: Physical signal overshadowing attack on lte," in *28th USENIX Security Symposium (USENIX Security 19)*. Santa Clara, CA: USENIX Association, Aug. 2019, pp. 55–72. [Online]. Available: https://www.usenix.org/conference/usenixsecurity19/presentation/yang-hojoon

Acknowledgments

I am deeply grateful to many persons who contributed to the success of this thesis and research work. Their support in various capacities, whether direct or indirect, has been invaluable.

First and foremost, I extend my appreciation and gratitude to my supervisor, Prof. Mauro Conti. His guidance, support, and motivation have played a crucial role in accomplishing the results of this work. I am truly grateful for his inspiration, invaluable suggestions, and constructive feedback, which aided my personal and professional growth.

I would like to express my sincere thanks to Alessandro Brighente, who has supervised me throughout the entire development of this project, offering guidance and support that extended beyond its scope. His mentorship has fostered my research thinking, and his availability and helpfulness have been precious in our interactions.

I am extremely grate to Prof. Radha Poovendran for his warm welcome and the exceptional opportunity to work under his supervision at the University of Washington. This experience has been truly remarkable, and I am grateful for his guidance. I would also like to acknowledge Prof. Bhaskar Ramasubramanian for his invaluable contributions to the technical realization of this research project.

I wish to extend a special thanks to my friends, whose unwavering support and shared experiences have been paramount throughout my academic journey and life. Without their encouragement and the time spent together, I would not have been able to achieve the goals and results I have attained. A special mention goes to Stefano and Manuel, you know you are like brothers to me. I would like to express my gratitude to Sara, who has stood by my side through the best and most challenging times with love and patience.

Thanks to my mother Stefania, there are not enough words to express my gratitude for everything you have done in all these years. Among all, thanks for your help in hard times and for giving me the opportunity to undertake this great experience with my Master.

Thanks to my sister Noemi, your dedication has always been a strong motivation to me. A special thanks also for your patience and precious advice in correcting my English and writes.

Lastly, I want to dedicate a heartfelt and special thanks to my father, Paolo. Although I cannot express it in person, I am sure you can feel the depth of my gratitude to both you and mom for your unconditional support and love.