CSN-341 Computer Networks
Project Report

Enhancing Authentication and Identity Management Protocols for Secure User Access and Services in 5G Networks.

Submitted By: Group 7

Name	Enrollment Number	E-Mail Id	Mobile Number
Ashutosh Kumar	21114021	a_kumar1@cs.iitr.ac.in	7061339843
Kirtankumar Vijaykumar Patel	21114051	k_vpatel@cs.iitr.ac.in	6395134351
Mudit Gupta	21114061	m_gupta@cs.iitr.ac.in	9302875744
Pise Ashutosh Kalidas	21114073	p_akalidas@cs.iitr.ac.in	7262851200
Raiwat Narendra Bapat	21114078	r_nbapat@cs.iitr.ac.in	7666191528
Rishi Kejriwal	21114081	r_kejriwal@cs.iitr.ac.in	9123814293
Rohan Kalra	21114083	r_kalra@cs.iitr.ac.in	9718007248
Sahil Safy	21114087	s_safy@cs.iitr.ac.in	8949994859
Sanidhya Bhatia	21114090	s_bhatia@cs.iitr.ac.in	8817471350
Tanmay Mohit Shrivastav	21114103	t_mshrivastav@cs.iitr.ac.in	9574560001

1. Introduction

The primary aim of our project was to enhance the protocols used in the 5G Authentication. We engaged in comprehensive research, thoroughly reviewing a variety of research papers. We also studied tools like Proverif and Tamarin Prover which were used in the research papers to present a formal analysis of the 5G-AKA Protocol. We have used Java as the primary language in the implementation of the protocol, and the suggested enhancements have been presented in the report below.

The project aims to implement, analyze, and assess the security aspects of a 5G authentication protocol 5G AKA (Authentication and Key Agreement). The protocol involves key components such as User Equipment (UE), Security Anchor Function (SEAF), Authentication Server Function (AUSF), and Unified Data Management (UDM). The project is implemented in Java Programming Language, and it has two run modes, with a UE and an Evil UE.

2. Implementation Overview

2.1. System Architecture

The implemented system architecture consists of the following key components:

UE (User Equipment):

Represents end-user devices participating in the 5G network. E.g., Mobile Phone.

SEAF (Security Anchor Function):

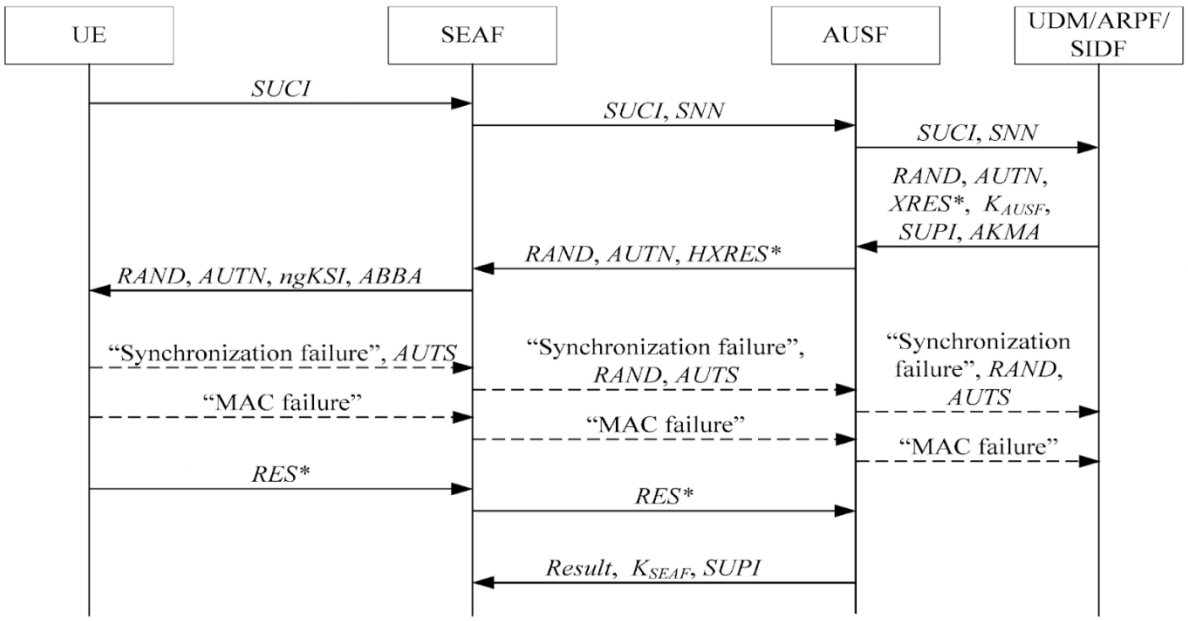
It is implemented within the Serving Network and manages security functions at the network edge.

AUSF (Authentication Server Function):

It is implemented within the Home Network and is responsible for authenticating users and generating security keys.

UDM (Unified Data Management):

It is present within the Home network and is implemented alongside SIDF (Subscriber Identity De-Concealing Function) and manages subscriber data, including authentication credentials.



2.2. Key Generation and Initialization

The project utilizes strong cryptographic keys to enhance the security of the implemented protocol. Key generation includes parameters such as Key (K), Subscription Permanent Identifier (SUPI), Security Name (SN_NAME), and Authentication Management Field (AMF). Additionally, RSA public/private key pairs are generated for secure communication.

2.3. Authentication Protocol

The authentication protocol involves a multi-step process, including the initiation of authentication, key exchange, and validation. The detailed steps include:

Initiation: UE initiates authentication with SEAF.

Key Exchange: Secure key exchange between UE, SEAF, AUSF, and UDM.

Validation: AUSF validates the user's identity, and secure keys are generated.

3. Results and Analysis

In the terminal output, we have logged all the messages that have been sent by one module to another. These include authentication requests and responses, confirmation messages, etc. In the end, if the authentication is successful, K_{seaf} values (256-bit encodings) that are present in UE and SEAF match. If authentication is unsuccessful, received and calculated K_{seaf} values in SEAF do not match, thus showing NULL.

3.1. Standard Protocol Run

Output on standard protocol run mode: -

```
Starting protocol...
RunMode: Protocol

UE -> SEAF : N1 Registration Request
SEAF -> AUSF : Nausf_UEAuthentication_Authenticate Request
AUSF -> UDM : Nudm_UEAuthentication_Get Request
Authentication vector created in the UDM
UDM -> AUSF : Nudm_Authentication_Get Response
AUSF -> SEAF : Nausf_UEAuthentication_Authenticate Response
SEAF -> UE : Authentication Request
UE -> SEAF : Authentication Response
  SEAF is considering the authentication as successful.
SEAF -> AUSF : Nausf_UEAuthentication Confirmation Request
  AUSF is considering the authentication as successful.
AUSF -> SEAF : Nausf_UEAuthentication Confirmation Response
AUSF -> UDM : Authentication Information
  UDM is considering the authentication as successful.
Authentication was successful.
 UE:   Kseaf: 8BF59BF330D2B6F1C7712CD4458E24022D6C2C45E377A686CC9933622C9C9C54
 SEAF: Kseaf: 8BF59BF330D2B6F1C7712CD4458E24022D6C2C45E377A686CC9933622C9C9C54

Exiting...
```

3.2. Vulnerability Run

Output on vulnerability run mode: -

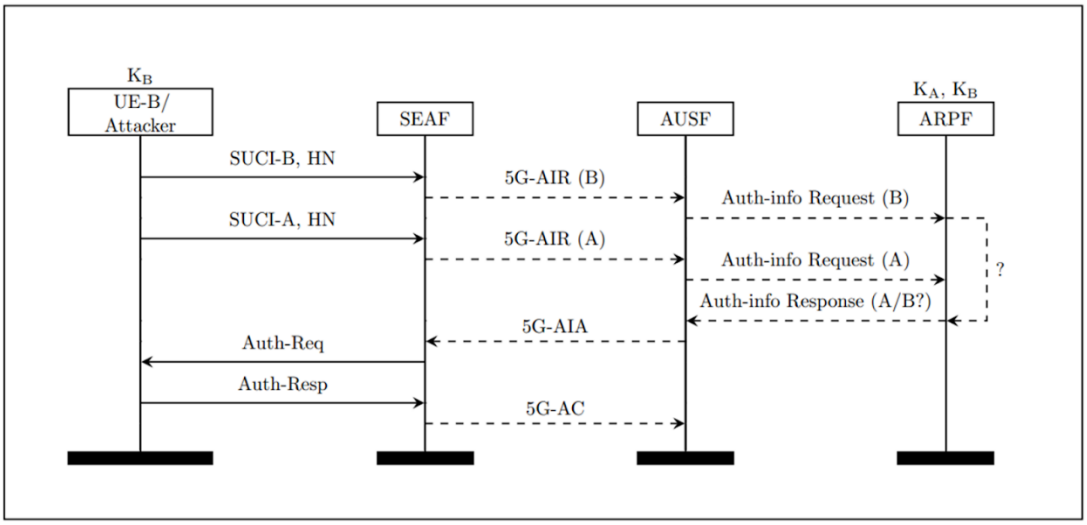
```
Starting protocol...
RunMode: Vulnerability

UE -> SEAF : N1 Registration Request
UE -> SEAF : N1 Registration Request
SEAF -> AUSF : Nausf_UEAuthentication_Authenticate Request
SEAF -> AUSF : Nausf_UEAuthentication_Authenticate Request
AUSF -> UDM : Nudm_UEAuthentication_Get Request
AUSF -> UDM : Nudm_UEAuthentication_Get Request
Authentication vector created in the UDM
Authentication vector created in the UDM
UDM -> AUSF : Nudm_Authentication_Get Response
UDM -> AUSF : Nudm_Authentication_Get Response
AUSF -> SEAF : Nausf_UEAuthentication_Authenticate Response
AUSF -> SEAF : Nausf_UEAuthentication_Authenticate Response
SEAF -> UE : Authentication Request
UE: Ignoring all further authenticate messages.
UE -> SEAF : Authentication Response
  SEAF is considering the authentication as unsuccessful.
SEAF -> AUSF : Nausf_UEAuthentication Confirmation Request
  AUSF is considering the authentication as successful.
AUSF -> UDM : Authentication Information
AUSF -> SEAF : Nausf_UEAuthentication Confirmation Response
  UDM is considering the authentication as successful.
Authentication failed.
SEAF -> UE : Authentication Reject
 UE:   Kseaf: 369DFEB1EDBBB08425322099DD88FE09389C6D5497FB753C20143C36791B486C
 SEAF: Kseaf: null
Vulnerability failed.

Exiting...
```

4. Enhancements

4.1. Solution to the Replay Attack



The replay attack occurs when attacker B is aware of the SUCI of an honest User A, by eavesdropping means. The protocol draft lacks a crucial containment property: an attacker that can compromise the long-term key of a user will be able to impersonate any user to the SEAF and the AUSF because it knows the K_{SEAF} of sessions that the SEAF and AUSF assume to be from 'A'. This attacker could then bill services, airtime, or access charges to another user account, rather than its own; this is not the intended behaviour or level of security required within 5G networks.

Solution 1: Explicit identity binding

They are adding SUPI and SUCI in the message sent from ARPF to the AUSF and Adding SUCI to the message sent from AUSF to SEAF. These minor changes now successfully bind the correct parties to the messages throughout the full flow of the protocol, preventing this (and other) identity mis-binding attacks from working. The proposed modifications have a negligible impact on the computational efficiency of the protocol.

Solution 2: Tighter session binding

Emulating individual sessions within a long-lived TLS or DIAMETER session between AUSF and ARPF by an intermediate layer. Initiating an entirely new TLS or DIAMETER session for each Auth-info Request and expecting the Auth-info Response within that session.

4.2. Introduction of random numbers and SN-Name for MitM and Redirection attacks

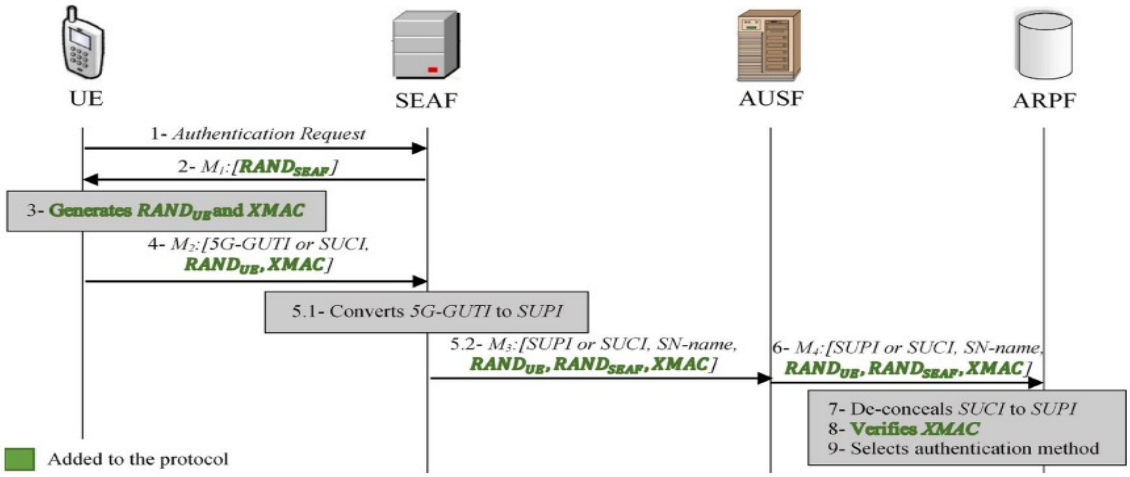


Fig. 8. Initiation procedure of the improved protocols.

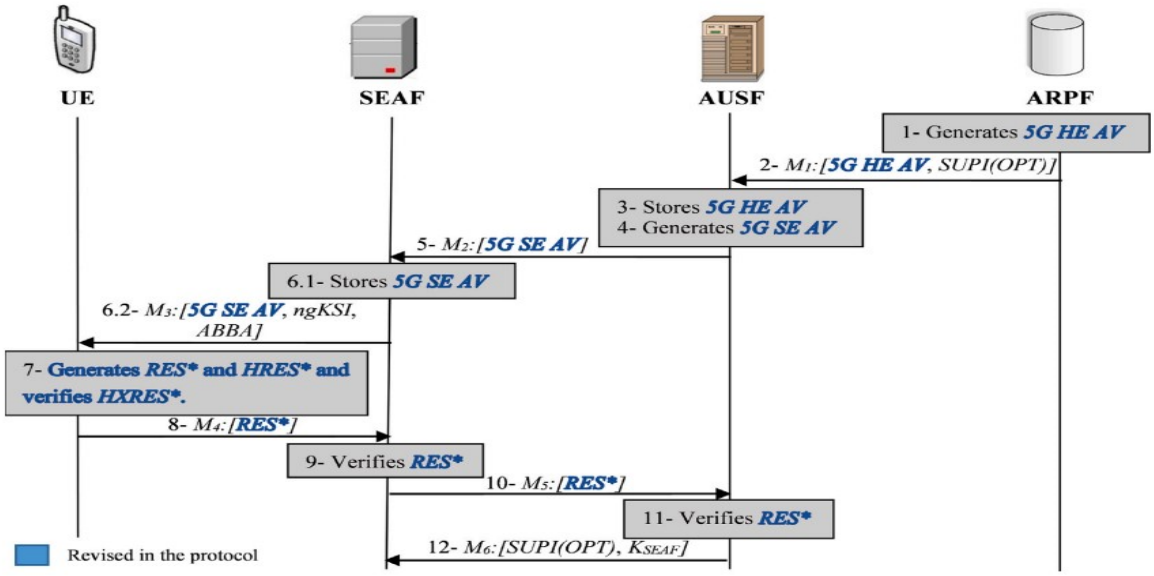


Fig. 9. Authentication procedure for the improved 5G AKA protocol.

In original 5G-AKA protocol, the initial authentication request sent by UE contained a SQN field. MitM attacks and replay attacks exploit this vulnerability to logically guess correct next SQNs and thus getting false authentication. This can be prevented by incorporating a random -number pair by UE and SEAF in requests and responses.

Redirection attacks are based on setting false base stations and making users connect to those stations rather than desired ones. In our enhancement, we consider SN-Name in MAC generation functions. If such an attack occurs, the authentication procedure can be aborted by checking the MACs.

5. Conclusion

In conclusion, our project aimed at enhancing the protocols in 5G authentication, with a particular focus on the 5G AKA protocol. Extensive research, including the study of formal analysis tools like ProVerif and Tamarin Prover, informed our approach. Java was chosen as the primary language for implementation, allowing the presentation of suggested enhancements. The system architecture was meticulously designed, involving key components like UE, SEAF, AUSF, and UDM. The key generation, authentication protocol, and results analysis demonstrated the project's technical depth. Notably, vulnerabilities such as replay attacks were addressed through innovative solutions, including explicit identity binding and tighter session binding. The introduction of random numbers and SN-Name further fortified the protocol against MitM and redirection attacks. The results and analysis sections provided insights into standard, and vulnerability run modes, showcasing the project's effectiveness. In summary, our project not only implemented enhancements to the 5G AKA protocol but also addressed identified vulnerabilities, contributing to the overall security and robustness of 5G authentication protocols.