Duale Hochschule Baden-Württemberg Mannheim

Studienarbeit

Programmierung einer SIM-Authentifizierung

Irgend ein Untertitel

Name:	**Marco Heumann**
Matrikelnummer:	4188528
Kurs:	TINF13AI-BI
Studiengang:	Angewandte Informatik
Studiengangsleiter:	Prof. Dr. C. Bürgy
Betreuer:	
Semester:	5. - 6. Semester
Datum:	

Ehrenwörtliche Erklärung

Gemäß § 5 Abs. 3 der Studien- und Prüfungsordnung DHBW Technik vom 22.09.2011 versichere ich hiermit, die vorliegende Arbeit selbstständig und nur mit den angegebenen Quellen und Hilfsmitteln verfasst zu haben.

Datum	Marco Heumann

Inhaltsverzeichnis

Abkürzungsverzeichnis

3GPP	3rd Generation Partnership Project
SIM	Subscriber Identity Module
USIM	Universal Subscriber Identity Module
IMSI	International Mobile Subscriber Identity
GSM	Global System for Mobile communications
UMTS	Universal Mobile Telecommunications System -
PIN	Personal Identification Number
EF	Elementary File
AuC	Authentication Center
UE	User Equipment
RES	Response to Challenge
AMF	Authentication Management Field
SQN	Sequence Number
AES	Advanced Encryption Standard
OP	Operator Variant Algorithm Configuration Field
OPc	Operator Variant Algorithm Configuration Field encrypted
K	Subscriber Key
RAND	Random Challenge

Abbildungsverzeichnis

Tabellenverzeichnis

Listings

1 Einleitung

1.1 Idee zur Arbeit

1.2 Geschichte der USIM

2 Theorie

2.1 Mobilfunkstandards

2.2 SIM-Karten

2.3 Authentifizierungsvorgang

2.4 Milenage Algorithmus

Zwischen SIM[1]-Karte und Netzprovider muss eine sichere Authentifizierung und Kommunikation gewährleistet werden können. Dies war wie in Kapitel 1.2 bereits beschrieben mit dem ersten entwickelten Algorithmus des 3GPP[2] nicht mehr gewährleistet, weshalb mit der Entwicklung des neuen Netzstandards auch ein neuer Algorithmus entwickelt wurde, namentlich der Milenage Algorithmus.
Dieser verfügt über die sieben Funktionen *f1*, *f1**, *f2*, *f3*, *f4*, *f5*, *f5** mit Hilfe derer eine sichere Authentifizierung und Schlüsselgenerierung ermöglicht wird. 3GPP hat allerdings wie auch beim Vorgänger diese Funktionen nicht näher spezifiziert und ermöglicht den Netzprovidern eigenen Lösungen zu implementieren. Stattdessen beschrieben sie den Kontext in dem diese Funktionen Anwendung finden und definieren generelle Anforderungen an diese Algorithmen [2].

Der Milenage Algorithmus hat wie erwähnt zwei Hauptaufgaben, nämlich einerseits die Authentifizierung, als auch die Generierung eines Schlüssel, um die versendeten Nachrichten zu ver- und entschlüsseln. Wenn es um die Authentifizierung geht muss sich einerseits die SIM-Karte, bzw. das UE[3], gegenüber dem Netzprovider authentifizieren, aber andererseits muss sich auch das Netzwerk gegenüber der SIM-Karte authentifizieren. Damit soll die Möglichkeit der Man-in-the-Middle Attacken reduziert werden, die es einem Außenstehenden erlauben die Kommunikation mitzulesen. Auch so genannte Replay-Attacken, bei denen zuvor aufgezeichnete Daten genutzt werden, sind nicht möglich, auf Grund der Sequence Number [3].

[1] Subscriber Identity Module
[2] 3rd Generation Partnership Project
[3] User Equipment

In den nachfolgenden Unterkapiteln wird die Funktionsweise des Algorithmus, sowie die Funktionsweise der eingesetzten Blockschiffrierung AES[4] erläutert.

2.4.1 Funktionsweise

In Kapitel 2.3 wurde beschrieben, welche Daten zwischen AuC[5] und UE verschickt werden, jedoch nicht wie diese Daten generiert werden. Es gibt einige Werte, die auf der USIM[6] und der Datenbank des AuC fest eingespeichert sind. Diese sind der OP[7] und K[8], sowie jeweils fünf Rotations- und XOR-Konstanten (r1, ..., r5 und c1, ... c5). Welche Funktion welche Werte benötigt und generiert zeigt dabei Abbildung 1.

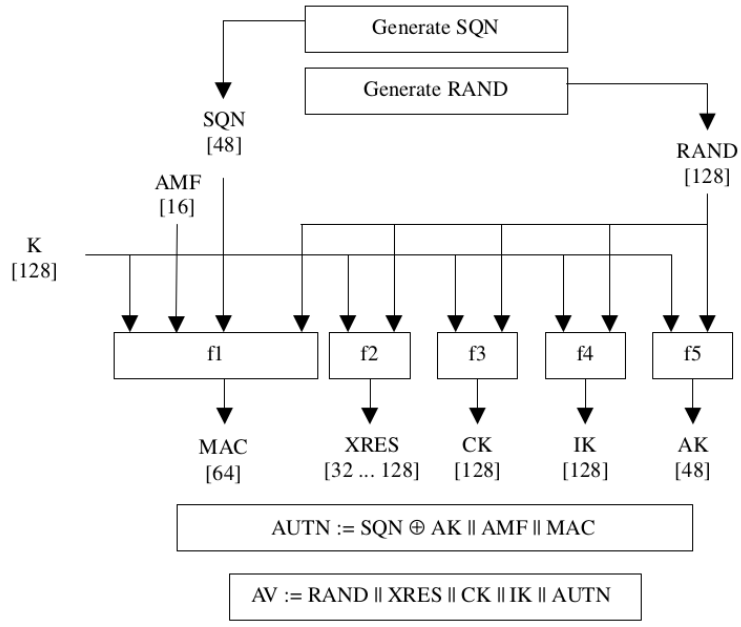

Abbildung 1: Übersicht über die Generierung der Authentifizierungsvektoren [1]

In Abbildung 1 ist zu sehen, dass zu Beginn die SQN[9] generiert wird. Diese ist insgesamt 48 Bits lang und besteht aus den beiden Teilen SEQ und IND, mit SEQ als der die eigentliche Sequenznummer und IND als Arrayindex. Dieser Index wird benötigt, da auf der SIM-Karte sind die letzten SQNs in einem Array gespeichert. Die empfohlene

[4]Advanced Encryption Standard
[5]Authentication Center
[6]Universal Subscriber Identity Module
[7]Operator Variant Algorithm Configuration Field
[8]Subscriber Key
[9]Sequence Number

Arraygröße ist 32, was für IND eine Länge von fünf Bits bedeutet. Mit diesem Index kann nachher die Aktualität der SEQ überprüft werden.[1]
Für die Bildung der SEQ selbst gibt es drei verschiedene Möglichkeiten:

- teilweise zeitbasiert
- nicht zeitbasiert
- komplett zeitbasiert

Die einfachste Variante ist die nicht zeitbasierte Lösung, bei der lediglich ein Zähler hochgezählt wird mit jeder Authentifizierungsanfrage. Die SEQ ist initial also 0 und wird hochgezählt und gleichzeitig vom AuC in einer Datenbank gespeichert [1]. Auf die anderen Möglichkeiten wird hier nicht näher eingegangen, da sie in dieser Arbeit keine Anwendung fanden.

Als nächstes wird die RAND[10] gebildet. Das Verfahren, wie der Netzprovider diese RAND generiert darf nicht offen gelegt werden, da dies die Sicherheit stark beeinflussen würde. Generell handelt es sich dabei um eine 128-bit lange Zufallszahl, die für jede Funktion benötigt wird.

Abbildung 1 zeigt zwar, welche Variablen in die Funktionen einfließen und welche Werte sie zurückgeben, aber sie zeigt nicht näher wie diese Werte nun verarbeitet werden. Dies zeigt Abbildung 2 besser. Dort ist zu erkennen, dass *f2* bis *f5** nach dem selben Schema berechnet werden können und *f1*, sowie *f1** noch einige zusätzliche Parameter haben.

Zunächst die Erklärung der Symbole sowie einiger weiterer Abkürzungen. OPc[11] wird durch folgende Formel generiert:

$$OP_C = OP \oplus E(OP)_K$$

$E()$ ist die Blockschiffrierung. In diesem Falle wird also OP mit dem Schlüssel K verschlüsselt. Welche Verschlüsselung gewählt wird, wird von 3GPP nicht vorgegeben. In dieser Arbeit wurde AES verwendet, welche im Kapitel 2.4.2 näher beschrieben wird.
Der verschlüsselte OP wird dann im zweiten Schritt über XOR ($\oplus$) mit dem ursprünglichen OP verknüpft.

In Abbildung 2 ist weiterhin der Funktionsblock “rotate by r” zu lesen. Beim rotieren wird der Eingabewert um die Anzahl an Bits des Wertes von r rechts rotiert und die

[10] Random Challenge
[11] Operator Variant Algorithm Configuration Field encrypted

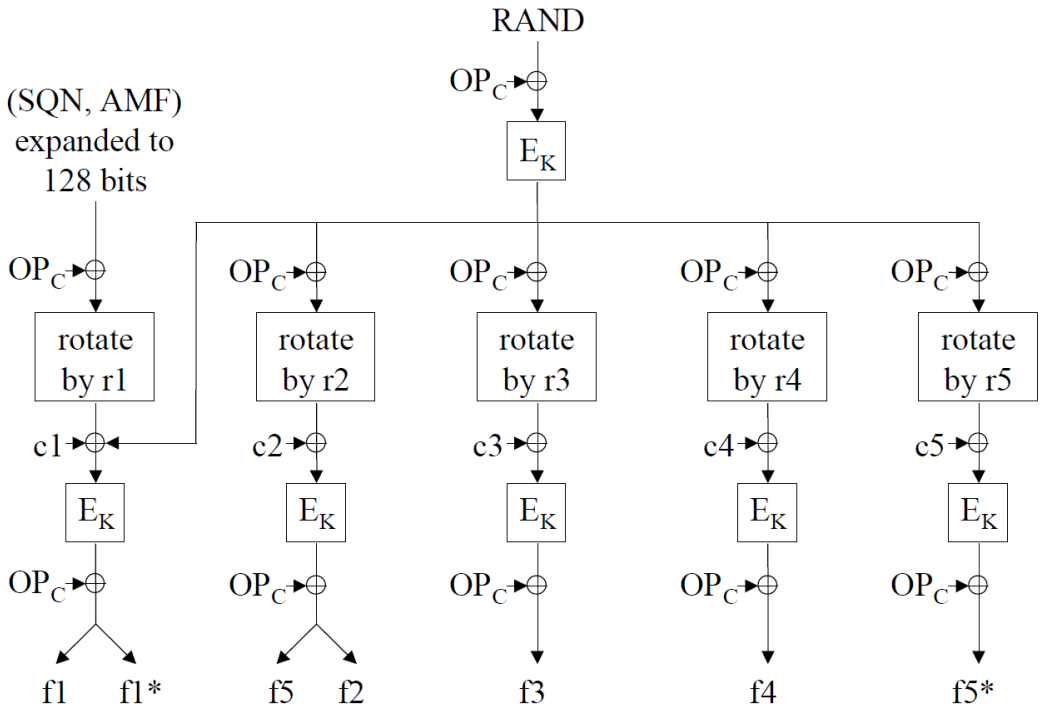

Abbildung 2: Schematische Darstellung zur Berechnung der Authentifizierungsvektoren [1]

Bits die herausfallen links wieder eingefügt. Beispielsweise wird aus 110101 bei einem Rotationswert r von 2: 011101.

2.4.2 AES

2.5 PPPoE

2.6 raspberry pi

2.7 pysim

2.8 Die Sprache C

2.9 Projektspecs

3 Tätigkeit

4 Ergebnis

5 Diskussion

Literatur

[1] Horn, G., 3G security; Security architecturel, TS 33.102, 3rd Generation Partnership Project (3GPP), 2015.

[2] Walker, M., 3G Security; Specification of the MILENAGE algorithm set: An example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 1: General, TS 35.205, 3rd Generation Partnership Project (3GPP), 2015.

[3] Spitz, S., Pramateftakis, M. und Swoboda, J., *Kryptographie und IT-Sicherheit: Grundlagen und Anwendungen*, Studium : IT-Sicherheit und Datenschutz, Vieweg+Teubner Verlag, 2011.

Studienarbeit

Titel:	Programmierung einer SIM-Authentifizierung
Subtitel:	Irgend ein Untertitel
Autor:	Marco Heumann
Hochschule:	Duale Hochschule Baden-Württemberg Mannheim
Datum:	
Bearbeitungszeitraum:	
Studiengang:	Angewandte Informatik
Matrikelnummer, Kurs:	4188528, TINF13AI-BI
Betreuer:	
Gutachter:	Prof. Dr. C. Bürgy

Abstract

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Studienarbeit

Title:	Programmierung einer SIM-Authentifizierung
Subtitle:	Irgend ein Untertitel
Author:	Marco Heumann
University:	Duale Hochschule Baden-Württemberg Mannheim
Date:	
Time of Project:	
Study Course:	Angewandte Informatik
Student ID, Course:	4188528, TINF13AI-BI
Supervisor in the Company:	
Reviewer:	Prof. Dr. C. Bürgy

Abstract

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.