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
K	Subscriber Key
PPP	Point To Point Protocol
PPPoE	Point to Point Protocol over Ethernet
ISP	Internet Service Provider
LAN	Local Area Network
SLIP	Serial Line Internet Protocol
LCP	Link Control Protocol
NCP	Network Control Protocol
HDLC	High Level Data Link Control
IANA	Internet Assigned Numbers Authority
PAP	Password Authentication Protocol
CHAP	Challenge Handshake Authentication Protocol
NCP	Network Control Protocol
IPCP	Internet Protocol Control Protocol

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 [1].

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 [2].

[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.

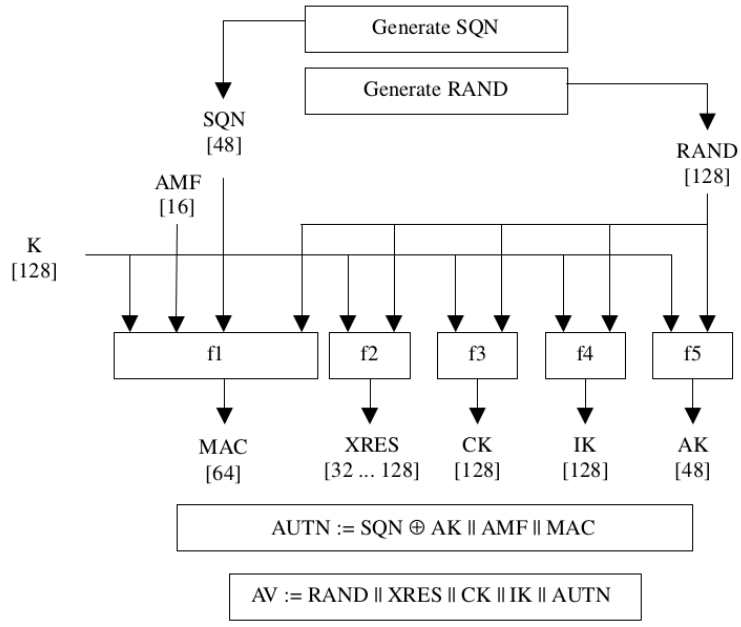

Abbildung 1: Übersicht über die Generierung der Authentifizierungsvektoren

Die Abbildung 1 zeigt, dass zu Beginn die SQN[9] generiert wird. Diese besteht aus den beiden Teilen SEQ und IND, mit SEQ als der die eigentliche Sequenznummer und IND als Arrayindex. Auf der SIM-Karte sind nämlich die letzten SQNs in einem Array gespeichert. Die empfohlene Größe ist 32, was für IND eine Länge von fünf Bits bedeutet. Mit

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

diesem Index kann nachher die Aktualität der SEQ überprüft werden. Für die Bildung der SEQ selbst gibt es drei verschiedene Möglichkeiten:

- teilweise zeitbasiert
- nicht zeitbasiert
- komplett zeitbasiert

2.4.2 AES

2.5 PPP

Zur Bereitstellung einer Punkt-zu-Punkt-Verbindung als Grundlage des Authentifizierungsvorgangs wird von Providern (ISP[10]) Implementierungen eines PPP[11] verwendet. Mit Protokollen dieser Art wurden zum Beispiel schon Modem- oder ISDN-Verbindungen aufgebaut. Heutige Szenarien sind unter anderem auch GPRS- und UMTS-Datenverbindungen - hier hauptsächlich in Form von PPPoE[12]. Auf beide Architekturen wird im folgenden genauer eingegangen.

2.5.1 Architektur PPP

PPP ist Teil der TCP/IP-Protokollsuite und sichert die komplette Funktionalität des Datalink-Layers und wurde speziell für den Betrieb von Modems etc. entwickelt. Jede Maschine, die ein Modem in Betrieb hatte, nutzte bereits PPP um z.B. Internet im lokalen Netzwerk freizuschalten und zu verteilen. Neben der Freischaltung von Internetverbindungen wird PPP von vielen ISP auch dazu verwendet Zugriffe zu monitoren sowie Angriffe durch Intrusion Detection zu vermeiden. In üblichen LAN[13]-Umgebungen ist es notwendig, dass eingesetzte Technologien die Datalink-Layer-Funktion implementieren und darüberhinhaus über einen MAC-Meschanismus verfügen, da verschiedene Quellen/-Ziele das selbe Medium teilen könnten. Dieser Regulierungsmechanismus ist bei PPP nicht notwendig, da es sich um eine Punkt-zu-Punkt bzw. Ende-zu-Ende-Verbindung handelt. In jedem Fall handelt es sich um genau zwei Teilnehmer:

- Quelle
- Ziel

Neben dem Datalink-Layer baut PPP notwendigerweise auch auf der bestehenden Verbindung auf dem Physical-Layer auf.

Motivation Die Architektur ist gezielt sehr simpel gewählt. Es werden lediglich IP-Datagramme zwischen den Endgeräten enkapsuliert. Vergleichbar ist der Aufbau mit dem von Ethernet, jedoch ohne die notwendige Behandlung vieler Probleme die in sonstigen LAN- und Breitbandumgebungen auftreten können. So ist der Header z.B. nur

[10] Internet Service Provider
[11] Point To Point Protocol
[12] Point to Point Protocol over Ethernet
[13] Local Area Network

8 Byte statt 16 Byte lang. Doch dazu später mehr. PPP wurde als Alternative zum bereits bestehenden SLIP[14] implementiert, welches neben den notwendigen Methoden, dem multiplexen verschiedener Netzwerklayer-Protokolle sowie mehrere Authentifizierungsmethoden noch zusätzliche Funktionen ermöglichte, die nicht benötigt werden.

PPP Frame Ein PPP-Frame ist wie folgt aufgebaut:

- flag (1 Byte) - hexadezimal - Funktion des Paketdelimiter
- address (1 Byte) - hexadezimal (FF) - Indikator für 'adressiert an alle Stationen'
- control (1 Byte) - hexadezimal (03) -identifiziert Paket als HDLC[15]
- protocol (2 Byte) - hexadezimal - identifiziert erwünschtes bzw. eingesetztes Protokoll
  - 0xxx bis 3xxx : Netzwerklayer-Protokolle
  - 4xxx bis 7xxx : Low Level Netzwerklayer Protokolle ohne NCP[16]
  - 7xxx bis bxxx : Low Level Netzwerklayer Protokolle mit NCP
  - cxxx bis fxxx : Link Layer Protokoll wie LCP und zusätzliche Authentifizierungsprotokolle
- data and pad (variabel, maximal 1.500 Byte)
- frame check sequence (2 Byte oder 4 Byte)
- flag (1 Byte)

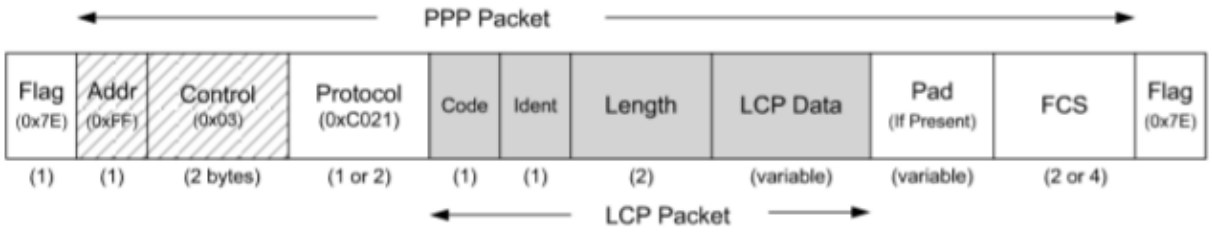

Abbildung 2: Aufbau eines PPP-Frames

[3]

[14]Serial Line Internet Protocol
[15]High Level Data Link Control
[16]Network Control Protocol

Diverse oben genannte Felder können in ihrer Länge variieren, da diese während des Verbindungsaufbaus vom LCP[17] ausgehandelt werden.

LCP Frame Um die effizienteste Verbindungsart zu finden, benutzen PPP-Systeme immer LCP um die korrekten Parameter auszuhandeln. LCP-Nachrichten in ausgetauschten PPP-Frames enthalten somit alle Konfigurationsoptionen für die sich gerade aufbauende Verbindung. Ist eine Konfiguration gefunden, die beide Knoten unterstützen folgt der Link-Establishment-Prozess. Ist dieser erreicht müssen danach keine weiteren redundanten Paketinformationen im Header mitgetragen werden.

Ein LCP-Frame ist wie folgt aufgebaut:

- code (1 Byte) - hexadezimal - enhält den Messagetyp (als Codes spezifiziert)
- identifier (1 Byte) - hexadezimal - mit diesen werden Anfragen bzw. Antworten mit einzelnen LCP-Transaktionen in Verbindung gebracht
- length (2 Byte) - hexadezimal - beinhaltet die Länge der Nachricht (inklusive code, identifier, length, data)
- data (variabel) - hexadezimal - Nutzdaten

LCP ist so entworfen, dass Hersteller ihre eigenen Optionen einsetzen können, ohne selbige explizit über IANA[18] spezifizieren zu müssen. Dokumentiert ist dies in **RFC2153**.

Aufbauphasen Nachfolgend werden die verschiedenen Aufbauphasen des PPP-Protokolls erläutert. Im Anhang 3: Aufbauphasen einer PPP-Verbindung auf S. 13 befindet sich eine Abbildung, die diesen Vorgang illustriert.

Link Dead Phase: Beide Systeme fangen mit dieser Phase an und enden hiermit wieder. Grundlage ist, dass außer (maximal) dem Link auf physischer Ebene, keine Verbindung zwischen beiden Endpunkten besteht. Normalerweise wird nach sicherstellen des physischen Links von einer Seite der Aufbau der Verbindung initiiert. Dies geschieht meist mit einer Form von Modem. Nach Abschluss der Initiierung beginnt die nachfolgende Phase.

Link Establishment Phase: Das initiierende System System sendet eine LCP-Nachricht an das Zielsystem, um Optionen anzufordern, die gesetzt werden sollen. Dazu gehören

[17] Link Control Protocol
[18] Internet Assigned Numbers Authority

Netzwerklayer-Protokoll, Authentifizierungsmethode und andere optionale Funktionen. Sofern das Zielsystem alle angeforderten Optionen beherrscht, kann dieses eine Bestätigung (**ACK**) an das Quellsystem senden. Ist dies nicht der Fall, wird eine Anwort verfasst, die sowohl alle *nicht unterstützten* als auch alle unterstützten Optionen enthält, damit das Quellsystem nach Empfang dieser Information eine Verbindung initiieren kann, die in jedem Fall von beiden Seiten unterstützt wird. Das erfolgreiche Abschließen dieser Phase führt zur nächsten Phase.

Authentication Phase: Diese Phase ist optional. Ausgelöst wird sie durch das vorhandensein einer Authentifizierungsoption in der LCP-Konfigurationsnachricht. Zur Auswahl stehen z.B. PAP[19] oder CHAP[20]. Hierbei greift PAP auf Username und Passwort, CHAP auf eines komplexeren Informationsaustausch mit einem Challenge-Response-Verfahren zurück. Wobei der Erfolg immer zu nächsten Phase führt ist die Reaktion bei Misserfolg des Vorgangs Protokollabhängig.

Link Quality Monitoring: Diese Phase ist wie ihr Vorgänger ebenfalls optional - ebenfalls ausgelöst durch die gewählt Option in der LCP-Nachricht. Hier aus mehreren Protokollen gewählt werden. Eines davon ist standardisiert: das 'Link Quality Report Protocol'. Registriert werden unter anderem der Linktraffic sowie Fehlermeldungen.

Network Layer Protocol Configuration: Wie bereits erwähnt unterstützt PPP das multiplexen von Protokollen auf Netzwerklayerebene. Für jedes einzelne, das eingesetzt wird, führt das System einen separaten Prozess des Verbindungsaufbaus durch. Jedes Netzwerklayerprotokoll verfügt über einen eigenes NCP sowie IPCP[21]. Vergleichbar ist dies mit dem Aufbau von LCP - nur spezifischer.

Link Open Phase: Nachdem alle individuellen Optionen und NCP-Exchanges erfolgreich durchgeführt wurden, ist der Verbindungsaufbau komplett und Protokolldaten können jetzt über den aufgebauten Link in beide Richtungen ausgetauscht werden.

Link Termination Phase: Wird die Verbindung absichtlich (Ablauf der Session, Authentifizierungsfehler) oder durch Fehler o.ä. physikalisch getrennt, wird im Regelfall über LCP eine 'Terminate Request Message' versandt. Diese kann von der Gegenseite angenommen (AKC) werden, sofern die grundlegende Verbindung noch aktiv ist. Beide Systeme sind dann wieder in der ursprünglich genannten 'Link Dead Phase'. Eine

[19] Password Authentication Protocol
[20] Challenge Handshake Authentication Protocol
[21] Internet Protocol Control Protocol

Terminierung der Verbindung ist neben LCP auch auf NCP-Ebene möglich, damit die PPP-Verbindung trotz 'Terminierung' bestehen bleibt.

2.5.2 Architektur PPPoE

2.6 raspberry pi

2.7 pysim

2.8 Die Sprache C

2.9 Projektspecs

3 Tätigkeit

4 Ergebnis

5 Diskussion

6 Appendix sections

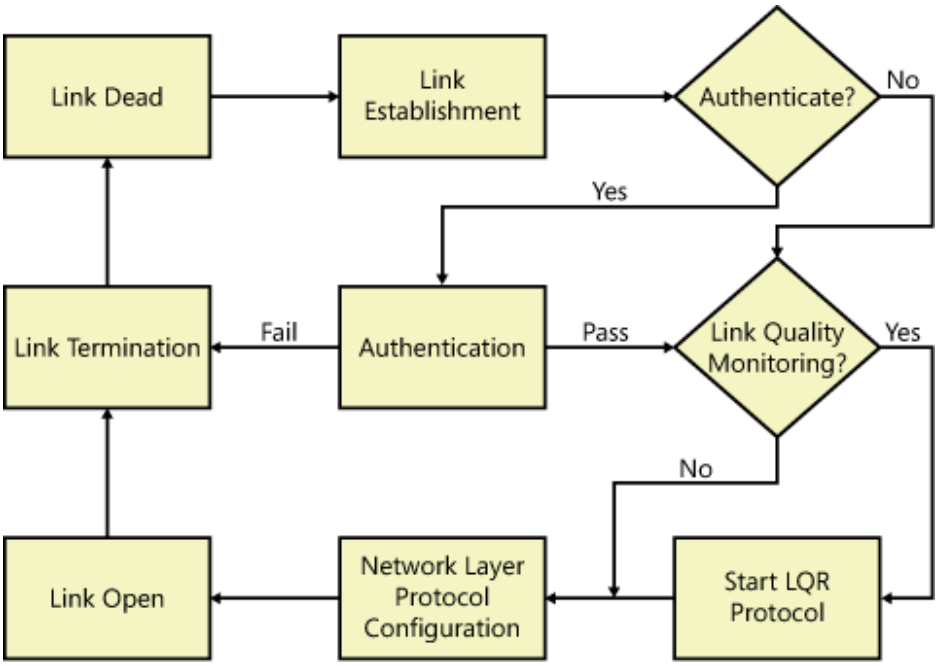

Abbildung 3: Aufbauphasen einer PPP-Verbindung [4]

Literatur

[1] 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.

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

[3] Stevens, R. und Fall, K., *TCP/IP Illustrated, Volume 1: The Protocols*, Addison-Wesley Professional, 2011.

[4] Zacker, C., *CompTIA® Network+® Exam N10-005 Training Kit*, Microsoft Press, 2012.

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.