Grundlagen der Verschlüsselung und Authentifizierung (1): symmetrische Verschlüsselung und Authentifizierung

Ausarbeitung zum gleichnamigen Vortrag im Seminar „Konzepte von Betriebssystemkomponenten“, SS 2010

Michael Fiedler <michael.fiedler@informatik.stud.uni-erlangen.de>

21. Juli 2010

Inhaltsverzeichnis

1 Einleitung

Der Einsatz von Kryptographie ist heute nicht mehr nur auf den Austausch vertraulicher Nachrichten mittels Verschlüsselung beschränkt, sondern vermag, neben der besagten Vertraulichkeit, noch weitere Sicherheitsziele zu erfüllen, namentlich Authentizität, Integrität und Zurechenbarkeit, welche dem Bereich der Authentifizierung zuzuordnen sind2.[2] Diese Ausarbeitung soll dabei einen grundlegenden Überblick über die kryptographischen Bereiche der symmetrischen Verschlüsselung sowie der Authentifizierung geben. Hierzu werden zunächst einige Verfahren der symmetrischen Verschlüsselung vorgestellt; anschließend soll auf den *Data Encryption Standard* (DES) als bekanntes symmetrisches Verschlüsselungsverfahren eingegangen werden. Es folgt ein Abschnitt zu grundlegenden Authentifizierungsverfahren sowie abschließend eine Zusammenfassung.

2 Symmetrische Verschlüsselung

2.1 Definition und Einordnung

Um eine Einordnung der symmetrischen Verschlüsselungsverfahren zu ermöglichen, muss man sich zunächst klar machen, was generell unter einem Verschlüsselungsverfahren zu verstehen ist. Hierzu sei folgende Definition einiger kryptographischer Grundbegriffe vorgenommen, welche auch als Definition 4.1.1 in [2] zu finden ist:

Definition 1 (Verschlüsselungsverfahren, Kryptosystem) *Ein* Verschlüsselungsverfahren *oder* Kryptosystem *ist ein Fünftupel* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *mit folgenden Eigenschaften:*

1. $\mathcal{P}$ *ist eine Menge. Sie heißt* Klartextraum*. Ihre Elemente heißen* Klartexte*.*
2. $\mathcal{C}$ *ist eine Menge. Sie heißt* Chiffretextraum*. Ihre Elemente heißen* Chiffretexte *oder* Schlüsseltexte*.*
3. $\mathcal{K}$ *ist eine Menge. Sie heißt* Schlüsselraum*. Ihre Elemente heißen* Schlüssel*.*
4. $\mathcal{E} = \{E_k : k \in \mathcal{K}\}$ *ist eine Familie von Funktionen* $E_k : \mathcal{P} \to \mathcal{C}$*. Ihre Elemente heißen* Verschlüsselungsfunktionen*.*
5. $\mathcal{D} = \{D_k : k \in \mathcal{K}\}$ *ist eine Familie von Funktionen* $D_k : \mathcal{C} \to \mathcal{P}$*. Ihre Elemente heißen* Entschlüsselungsfunktionen*.*
6. *Für jedes* $e \in \mathcal{K}$ *gibt es ein* $d \in \mathcal{K}$*, so dass für alle* $p \in \mathcal{P}$ *die Gleichung* $D_d(E_e(p)) = p$ *gilt.*

Ein *symmetrisches* Verschlüsselungsverfahren stellt nun genau denjenigen Spezialfall dar, in dem $d = e$ gilt, also der Schlüssel zum Verschlüsseln und der Schlüssel zum Entschlüsseln identisch sind. Dieser Fall soll durch Beispiel 1 noch einmal veranschaulicht werden. Das „Gegenstück“, die *asymmetrische* Verschlüsselung, ist durch $d \neq e$, also ein nicht-identisches Schlüsselpaar, gekennzeichnet.

Beispiel 1 (symmetrische Verschlüsselung) *Alice möchte an Bob eine vertrauliche Nachricht $p \in \mathcal{P}$ übertragen. Beide besitzen denselben Schlüssel $k \in \mathcal{K}$. Hierzu verschlüsselt Alice ihre Nachricht mit der Verschlüsselungsfunktion $E_k \in \mathcal{E}$ und versendet $E_k(p)$. Nach Erhalt kann Bob die empfangene Botschaft mit der Entschlüsselungsfunktion $D_k \in \mathcal{D}$ entschlüsseln und erhält damit wieder den Inhalt der vertraulichen Nachricht $D_k(E_k(p)) = p$.*

2.2 Symmetrische Chiffren und Verschlüsselungsmodi

Im Folgenden wird auf einige grundlegende symmetrische Verschlüsselungsverfahren eingegangen, wobei sich stark an [4] orientiert wird. Zum Verständnis sind jedoch zunächst noch zwei Definitionen erforderlich:

Definition 2 (Blockchiffre) *Unter einer* Blockchiffre *versteht man ein Verschlüsselungsverfahren, in dem Klartext- und Schlüsseltextraum die Menge Σ^n aller Wörter der Länge $n \in \mathbb{N}$ über einem Alphabet Σ sind: $\mathcal{P} = \mathcal{C} = \Sigma^n$.[2]*

Ein Block entspricht dabei also einem Wort der Länge n. Sind die zu verschlüsselnden Blöcke so kurz, dass der Klartext der Verschlüsselungsfunktion ungepuffert zugeführt werden kann, so ist in diesem Zusammenhang statt von Blöcken und Blockchiffren auch von *Zeichen* und *Stromchiffren* die Rede.[5]

Definition 3 (Stromchiffre) *Sei Σ ein beliebiges Alphabet und sei $\mathcal{P} = \mathcal{C} = \Sigma^n$ für eine natürliche Zahl $n \in \mathbb{N}$, $n \geq 1$. Weiterhin seien $\mathcal{K}$ und $\hat{\mathcal{K}}$ Schlüsselräume. Eine* Stromchiffre *wird durch eine Verschlüsselungsfunktion $E : \hat{\mathcal{K}} \times \mathcal{P} \to \mathcal{C}$ und einen Schlüsselstromgenerator $g : \mathcal{K} \times \Sigma^* \to \mathcal{K}$ beschrieben. Der Generator g erzeugt aus einem externen Schlüssel $k \in \mathcal{K}$ für einen Klartext $p = p_0 \dots p_{l-1}$, $p_i \in \mathcal{P}$, eine Folge $\hat{k}_0, \dots, \hat{k}_{l-1}$ von internen Schlüsseln $\hat{k}_i = g(k, p_0 \dots p_{i-1}) \in \hat{\mathcal{K}}$, unter denen p in den Chiffretext*

$$E_g(k, p) = E(\hat{k}_0, p_0) \dots E(\hat{k}_{l-1}, p_{l-1})$$

überführt wird.[5]

2.2.1 Electronic Code Book Mode (ECB)

Wie in Abbildung 1 dargestellt, werden Klartextblöcke p_i der Länge b voneinander unabhängig verschlüsselt. Dies hat zur Folge, dass ein Bitfehler in einem Block i nur lokale Auswirkungen, nämlich im i-ten Block des Chiffretextes bzw. wiederhergestellten Klartextblocks hat, welcher dann jedoch nicht mehr lesbar ist. Des Weiteren bleibt ein Synchronisationsverlust für den Fall, dass ganzzahlige Vielfache der Blocklänge verloren gehen, ohne Auswirkung; andernfalls ist eine Resynchronisation vonnöten. Beachtung finden sollte des Weiteren, dass gleiche Klartextblöcke in gleichen Chiffretextblöcken resultieren. Dies begünstigt sogenannte *known-plaintext*-Angriffe, bei denen der Angreifer Paare von Klartext und Chiffretext kennt und daraus auf den Schlüssel rückzuschließen versucht.

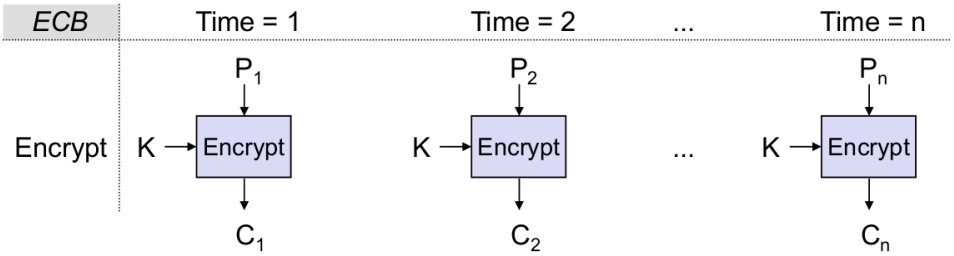

Abbildung 1: *Electronic Code Book Mode* (ECB): voneinander unabhängige Verschlüsselung der Blöcke p_i; Quelle: [4]

2.2.2 Cipher Block Chaining Mode (CBC)

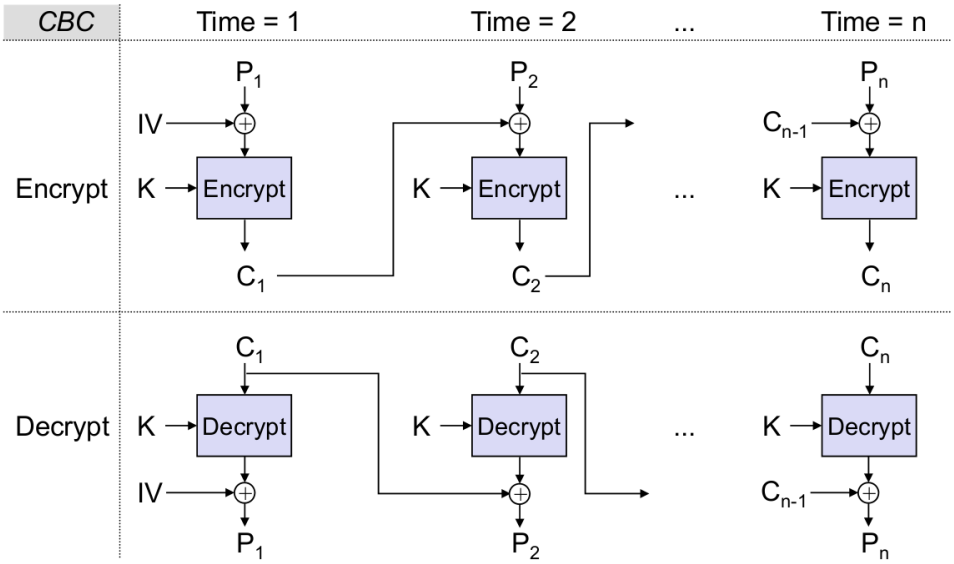

Abbildung 2: *Cipher Block Chaining Mode* (CBC): Durch XOR-Verknüpfung von p_i mit c_{i-1} entsteht eine Abhängigkeit des Verschlüsselungsvorgangs von der Vorgeschichte der Veschlüsselung der einzelnen Blöcke; Quelle: [4]

Wie in Abbildung 2 dargestellt, funktioniert der *Cipher Block Chaining Mode* (CBC) im Prinzip ähnlich zum *Electronic Code Book Mode* mit der Abweichung, dass vor jeder Verschlüsselung eine zusätzliche XOR-Verknüpfung zwischen dem Klartextblock an Stelle i und dem Chiffretextblock an Stelle $i-1$ vorgenommen wird. Beim Block 1 ist daher ein Initialisierungsvektor für c_0 notwendig, welcher auch bei der Entschlüsselung bekannt sein muss. Diese Rückkopplung resultiert im Allgemeinen in verschiedenen Chiffretextblöcken für die gleichen Klartextblöcke und wirkt so der Anfälligkeit des *Electronic Code Book Mode* gegenüber *known-plaintext*-Angriffen entgegen.

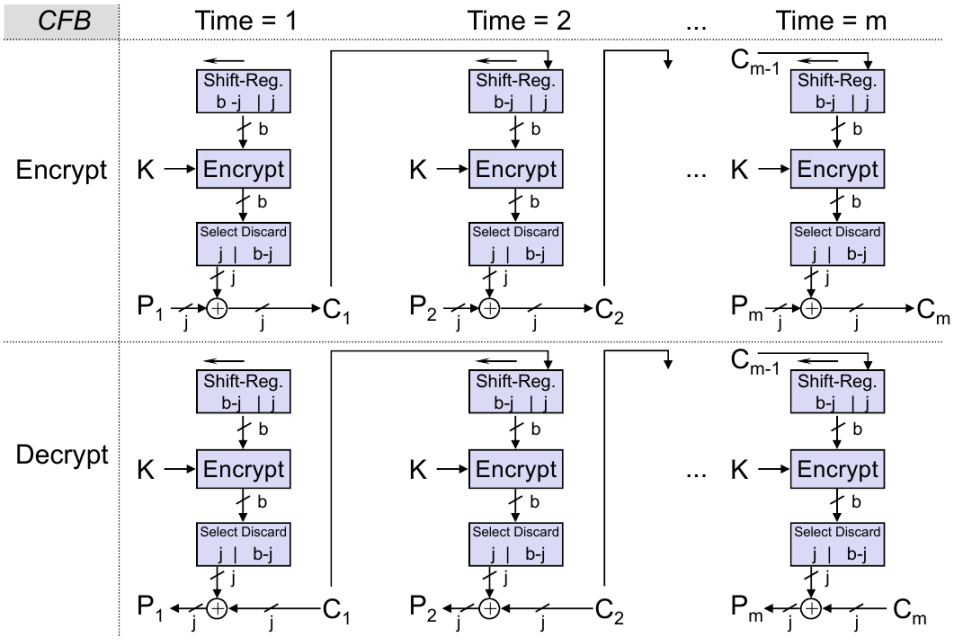

Abbildung 3: *Ciphertext Feedback Mode* (CFB): Rückkopplung durch Chiffretextblock des vorherigen Verschlüsselungsschrittes; Quelle: [4]

2.2.3 Ciphertext Feedback Mode (CFB)

Ein Blockverschlüsselungsverfahren mit Blocklänge b kann nun so modifiziert werden, dass er auf einer neuen Blockgröße $j < b$ arbeitet; beispielsweise kann man eine neue Blocklänge $j = 8$ (entspricht dann einem Zeichen) wählen. Durch diese Modifikation kann man dann aus einer Blockchiffre eine Stromchiffre erzeugen. Im *Ciphertext Feedback Mode* können so jeweils j Bits des Schlüsselstroms mit j Bits des Klartextes XOR-verknüpft werden; die Rückkopplung im Schlüsselstrom hängt dabei, wie der Name schon sagt, vom erzeugten Chiffretextblock ab. Abbildung 3 soll dies veranschaulichen. Der Schritt zum Schlüsselstrom funktioniert wie folgt:

$S(j, x) :=$ die j höherwertigen Bits von x
$P_i, C_i :=$ i-ter Klartext-/Chiffretextblock
$IV :=$ Initialisierungsvektor für Ver-/Entschlüsselung
$R_1 := IV$
$R_n := (R_{n-1} \cdot 2^j \bmod 2^b) \oplus C_{n-1}$ // Linksshift um j Bits, XOR-Verknüpfung mit altem Chiffretext
$C_n := S(j, E_K(R_n)) \oplus P_n$
$S(j, E_K(R_n)) \oplus C_n = S(j, E_K(R_n)) \oplus S(j, E_K(R_n)) \oplus P_n$
$S(j, E_K(R_n)) \oplus C_n = P_n$

Durch das schrittweise Schieben der Chiffretextblöcke durch das Register werden durch einen fehlerhaften Block c_i der wiederhergestellte p_i und die $\lceil \frac{b}{j} \rceil$ folgenden beeinträchtigt. Des Weiteren wird die Synchronisation bei Verlust eines Vielfachen von j Bits nach oben

besagten $\lceil \frac{b}{j} \rceil$ Blöcken wiederhergestellt; sonst ist eine explizite erneute Synchronisation erforderlich. Als problematisch ist bei CFB der erhöhte Rechenaufwand für die Verschlüsselungsfunktion anzusehen (um j Bits Klartext zu verschlüsseln, müssen b Bits verschlüsselt werden).

2.2.4 Output Feedback Mode (OFB)

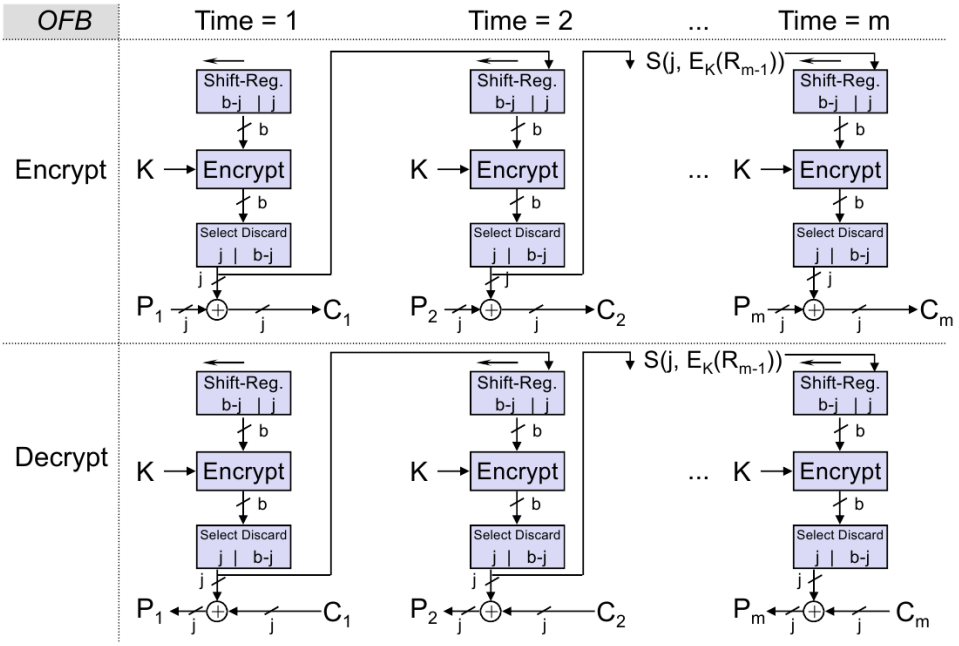

Abbildung 4: used in the second argume*Output Feedback Mode* (OFB): Rückkopplung durch Teil des verschlüsselten Schieberegisterinhalts; Quelle: [4]

Der *Output Feedback Mode* (OFB) (vgl. Abbildung 4) funktioniert nun wiederum ähnlich zum *Ciphertext Feedback Mode*. Wie der Name wiederum erahnen lässt, findet die Rückkopplung diesmal jedoch nicht über den erzeugten Chiffretextblock statt, sondern über den Vektor der ersten j Bits des verschlüsselten Schieberegisterinhalts, welcher mit dem Klartext verknüpft wird.

$S(j, x) :=$ die j höherwertigen Bits von x
$P_i, C_i :=$ i-ter Klartext-/Chiffretextblock
$IV :=$ Initialisierungsvektor für Ver-/Entschlüsselung
$R_1 := IV$
$R_n := (R_{n-1} \cdot 2^j \bmod 2^b) \oplus S(j, E_K(R_{n-1}))$ // Linksshift um j Bits, Verknüpfung mit altem Wert
$C_n := S(j, E_K(R_n)) \oplus P_n$
$S(j, E_K(R_n)) \oplus C_n = S(j, E_K(R_n)) \oplus S(j, E_K(R_n)) \oplus P_n$
$S(j, E_K(R_n)) \oplus C_n = P_n$

Beim OFB tritt dabei keine Vervielfältigung von Fehlern auf (1 Bitfehler $\Rightarrow$ 1 Bitfehler), bei Bitverlust ist stets Resynchronisation erforderlich. Weiterhin ist vorteilhaft, dass die Zufallszahlenfolge vorab berechnet werden kann, wodurch der Einfluss des Verschlüsselungsvorgang auf die Verzögerung zwischen zwei Kommunikationspartnern gering bleibt. Jedoch besteht wie bei CFB auch bei OFB der Mehraufwand bei der Berechnung der Verschlüsselungsfunktion. Außerdem kann ein Angreifer einzelne Klartextbits manipulieren.

2.3 Der Data Encryption Standard (DES)

Der *Data Encryption Standard* (DES) wurde, abgeleitet vom einem als *Lucifer* bezeichneten Algorithmus von IBM, 1977 zum US-amerikanischen Regierungsstandard gemacht und fand in den Folgejahren starke Verbreitung[4]. Auch wenn das Verfahren nach heutigem Stand nicht mehr als sicher gelten kann und inzwischen in Form des *Advanced Encryption Standard* (AES) (nach seinen Entwicklern Daemen und Rijmen auch als Rijndael-Algorithmus bekannt) als US-amerikanischer Standard abgelöst wurde, findet es in abgewandelter Form dennoch Anwendung. So kann die Sicherheit durch Mehrfachanwendung erhöht werden; es resultiert der sogenannte Triple-DES oder auch 3DES mit einer effektiven Schlüssellänge von 112 Bit.
Der DES ist eine sogenannte Feistel-Chiffre (eine Blockchiffre) mit dem Alphabet $\{0, 1\}$ und einer Blocklänge von 64. Hiervon wird je Byte ein Bit als Prüfbit (ungerade Parität) verwendet, woraus ein frei wählbarer Schlüssel der Länge 56 Bit resultiert (64 Bit inklusive Paritätsbits). Dies führt zu insgesamt $2^{56} \approx 7,2 \cdot 10^{16}$ verschiedenen Schlüsseln. Der Verschlüsselungsvorgang lässt sich in drei grobe Teilschritte gliedern, namentlich die initialen Permutationen, die Anwendung einer internen Blockchiffre[1] sowie die sogenannten S-Boxen (spezielle Funktionen). Die Entschlüsselung erfolgt durch Anwendung des DES mit umgekehrter Schlüsselfolge. Für eine detailliertere Funktionsweise sei an dieser Stelle jedoch auf [2] und [5] verwiesen.
Wie oben bereits erwähnt, ist der DES nach heutigen Ansprüchen jedoch nicht mehr als hinreichend sicher zu erachten. Bereits 1998 wurde mit einer mit dem Namen „DES Cracker“ versehenen Maschine der *Electronic Frontier Foundation* (EFF) eine vollständige Schlüsselsuche in 56 Stunden durchgeführt[5]. Des Weiteren ist ein 56-Bit-Schlüssel bei $10^6 \frac{\text{Verschlüsselungen}}{\mu s}$ in etwa 10 Stunden auffindbar[4]. Das 3DES-Verfahren bietet aber durch die größere Schlüssellänge immer noch Sicherheit.

[1] Nach [5] wurden alle vier oben dargestellten Blockchiffren (ECB, CBC, CFB, OFB) zur Verwendung vorgeschlagen; in [2] wird die Blockchiffre nicht weiter spezifiziert.

3 Authentifizierung

3.1 Teilnehmerauthentifizierung und Nachrichtenauthentifizierung

Neben der Verschlüsselung ist, wie bereits in der Einleitung angedeutet, auch die Authentifizierung[2] ein wichtiges Gebiet der Kryptographie. Man geht dabei von folgenden Fragen aus:[3]

- Wie kann ich sicherstellen, dass mein Kommunikationspartner derjenige ist, für den er sich ausgibt?
- Wie kann ich sicherstellen, dass die Nachricht meines Kommunikationspartners während des Transports über den Kommunikationskanal nicht verändert wurde?

Im ersten Fall möchte man einen Nachweis der Identität des Kommunikationspartners, im zweiten Fall einen Nachweis der Authentizität der Nachricht erhalten. Nach [1] kann die Problematik so in die beiden Teilgebiete der *Teilnehmerauthentifizierung* (*peer entity authentication*), „deren Ziel es ist, die Identität einer Person oder Instanz nachzuweisen“, sowie die *Nachrichtenauthentifizierung* (*message authentication*), „bei der er sowohl darum geht, den Ursprung einer Nachricht zweifelsfrei zu belegen als auch Veränderungen der Nachricht zu erkennen“, aufgeteilt werden.
Eindeutige Merkmale, auf die bei der Teilnehmerauthentifikation aufgebaut werden kann, sind nun biologische Eigenschaften (Fingerabdruck, Gesichtserkennung), der Besitz eine einzigartigen Objekts (Schlüssel, RFID) oder auch einzigartiges Wissen, abstrakt als „Geheimnis“ bezeichnet (Passwort o. Ä.). Bei der Nachrichtenauthentifizierung kann ähnlich auf charakteristische Informationen oder Fähigkeiten des Empfängers zurückgegriffen werden. Aus dem Alltag sollten die Beispiele Unterschrift wie auch die Echtheitsmerkmale von Banknoten bekannt sein. Generell kann durch die Verknüpfung eines Geheimnisses mit einem Dokument dessen Authentizität überprüft werden. Auf digitale Signaturverfahren, die auch in diesem Kontext zu sehen sind, soll hier jedoch nicht weiter eingegangen werden.

3.2 Kryptographische Protokolle

Kryptographische Protokolle, also festgelegte Kommunikationsabläufe unter Verwendung von kryptographischen Methoden, sollen bei der Authentifizierung behilflich sein. Um sinnvoll eingesetzt zu werden, sollten diese Verfahrensweisen jedoch bestimmten Kriterien genügen. Hier wäre zum einen die *Durchführbarkeit* zu nennen: Solange sich alle Kommunikationspartner an die Spezifikation des Protokolls halten, soll auch das gewünschte

[2] Wie man zum Beispiel [3] entnehmen kann, wird im Deutschen teilweise eine begriffliche Unterscheidung dafür vorgenommen, was im Englischen *authentication* genannt wird: Die Begriffe Authentisierung (Vorlage eines Nachweises zur Identifikation), Authentifizierung (Überprüfung des Nachweises), und auch Authentikation finden sich. Der Einfachheit halber soll in dieser Ausarbeitung stets von Authentifizierung die Rede sein.

[3] Der nachfolgende Abschnitt beruht im Wesentlichen auf [1].

Ergebnis resultieren. Zum anderen sollte die Wahrscheinlichkeit, dass es einem Angreifer gelingt, „zu betrügen“, vernachlässigbar klein sein; man spricht hierbei von *Korrektheit*. Im Folgenden sollen einige grundlegende Konzepte vorgestellt werden; beide Kriterien sind dabei erfüllt.

3.2.1 Passwortverfahren

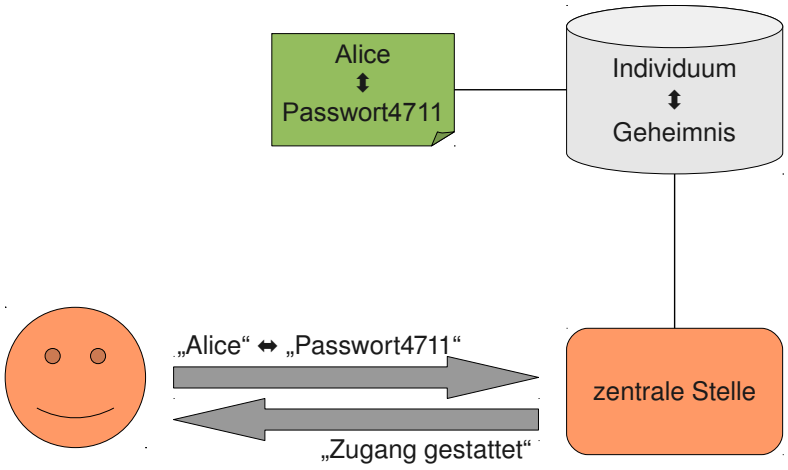

Abbildung 5: Passwortverfahren; in Anlehnung an [1]

Das Passwortverfahren, wie in Abbildung 5 veranschaulicht, sollte unmittelbar einleuchtend sein. Möchte sich Teilnehmerin Alice bei der zentralen Stelle, beispielsweise einem Bankautomaten, authentifizieren, so übermittelt sie zusammen mit ihrer Identität (Bankkarte) ihr Passwort (PIN). Dieses Paar muss auch der Zentrale bekannt sein; stimmen gespeichertes Datenpaar und von Alice übermittelte Information überein, so ist die Authentifikation gelungen.
Die Probleme bei diesem Verfahren sind jedoch offensichtlich:

- Wörterbuchangriffe können durch „schlechte“ Passwortwahl ermöglicht werden.
- Die Datenanhäufung in der zentralen Stelle kann bei einer Kompromittierung derselben zum Risiko werden; durch Speichern von Hashwerten der Passwörter (zum Beispiel unter Verwendung des *message-digest algorithm* 5 (MD5)) statt der Passwörter im Klartext und Hashwertbildung des übermittelten Passworts zur Authentifizierung kann dies jedoch behoben werden.
- Die offene Übertragung des Passworts kann Angriffsfläche bieten.

3.2.2 Wechselcodeverfahren

Durch Wechselcodeverfahren wird die Problematik der Passwortverfahren angegangen, dass die Authentifikationsnachricht stets identisch ist. Um diese also variieren zu können,

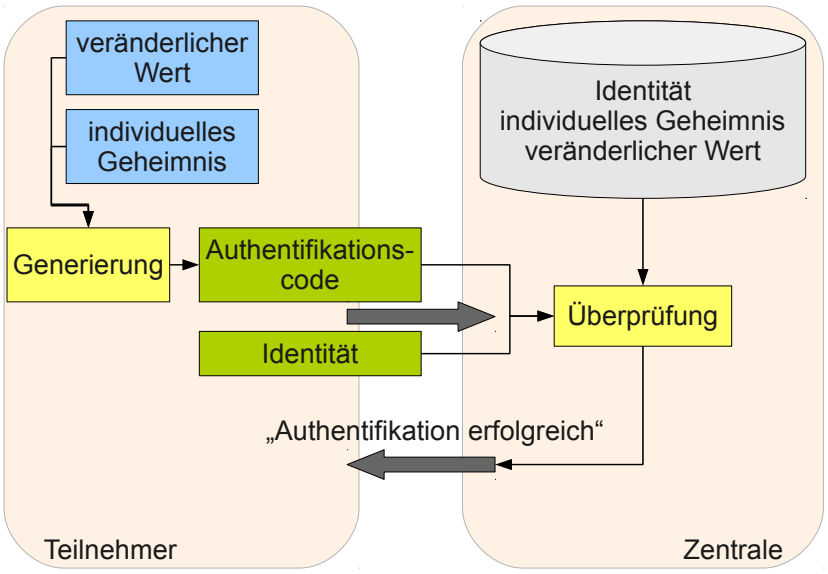

Abbildung 6: Wechselcodeverfahren; in Anlehnung an [1]

haben (vergleiche auch Abbildung 6) Teilnehmer und Zentrale Kenntnis von einem konstanten Geheimnis (im Beispiel Bankautomat wieder die PIN), jedoch auch von einem veränderlichen Wert (Transaktionsnummer, TAN). Beim Authentifikationsvorgang berechnet nun der Teilnehmer aus konstantem Geheimnis und veränderlichem Wert einen Authentifikationscode, den er zusammen mit seinder Identität an die Zentrale übermittelt. Diese kann die Berechnung wiederholen und authentifiziert den Teilnehmer bei Übereinstimmung der Ergebnisse erfolgreich.
Trotz erhöhter Sicherheit im Vergleich zum Passwortverfahren, gibt es Angriffsmöglichkeiten. Kann ein Angreifer auf den Datenverkehr Einfluss nehmen (*man-in-the-middle*-Angriff), so könnte er sich als Zentrale ausgeben und vom Teilnehmer einen Authentifikationscode berechnen lassen, mit dem er sich wiederum gegenüber der wirklichen Zentrale als A ausgeben kann.

3.2.3 Challange-and-Response-Verfahren

Durch *Challenge-and-Response*-Verfahren kann dem “Vorproduzieren“ von Nachrichten durch einen Angreifer, wie es beim Wechselcodeverfahren möglich wäre, verhindert werden, indem das zur Authentifizierung notwendige Wissen nicht vorab bekannt oder vorhersehbar ist. Will Teilnehmer A sich also bei Teilnehmer B authentifizieren, so sendet B eine unvorhersehbare Frage, welche A korrekt beantworten muss. Im Gegensatz zu den beiden vorherigen Verfahren fließt hier also die zur Authentifizierung notwendige Information nicht nur in eine Richtung durch den Kommunikationskanal, sondern in zwei Richtungen, sodass man in diesem Fall von *bidirektional* im Gegensatz zu *unidirektional* bei den vorhergehenden Verfahren spricht.
Zur Veranschaulichung sei nun auf eine Möglichkeit zur Authentifizierung beim *Post Office Protocol* Version 3 (POP3) eingegangen (vgl. Abbildung 7). Der Client möchte

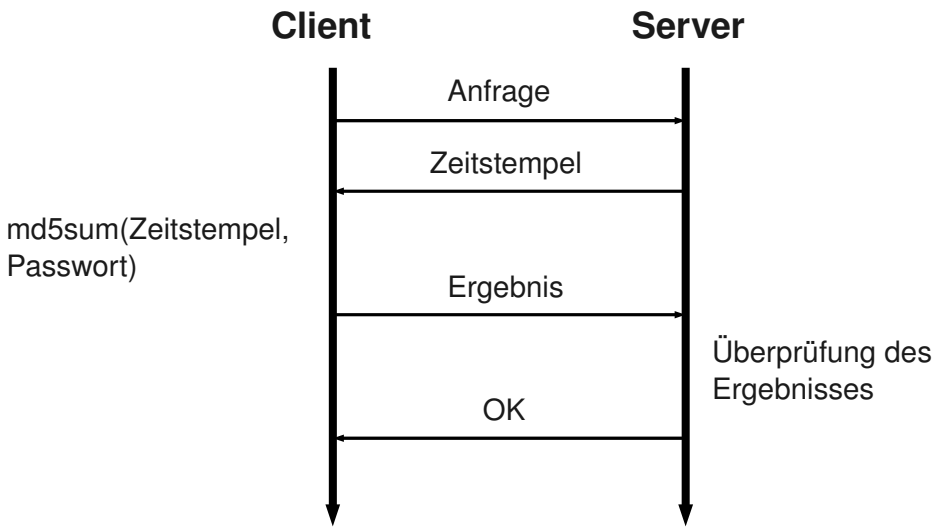

Abbildung 7: Möglichkeit zur Authentifizierung bei POP3, vgl. [6]

sich also beim Server authentifizieren, beide Teilnehmer haben Kenntnis vom geheimen Passwort. Die unvorhersehbare Frage, die der Server nun stellt, ist hier ein Zeitstempel. Auf Clientseite wird daraufhin aus vom Server erhaltenem Zeitstempel und dem Passwort die MD5-Summe berechnet und an den Server übertragen. Stimmt das Ergebnis mit seiner eigenen Überprüfungsrechnung überein, so ist die Authentifizierung gelungen.

4 Zusammenfassung

Im Rahmen dieser Ausarbeitung wurde zunächst auf symmetrische Verschlüsselung eingegangen. Nach einer Abgrenzung von symmetrischer und asymmetrischer Verschlüsselung wurden einige bekannte Verschlüsselungsverfahren detaillierter dargestellt. Durch Eingehen auf den *Data Encryption Standard* wurde der Zusammenhang zu einem langjährigen Verschlüsselungsstandard hergestellt. Abschließend wurden anhand von ausgewählten kryptographischen Protokollen grundlegende Aspekte der Authentifizierung veranschaulicht.

Abbildungsverzeichnis

Literatur

[1] Beutelspacher, A. ; Schwenk, J. ; Wolfenstetter, K.-D. : *Moderne Verfahren der Kryptographie*. 2., verbesserte Auflage. Brauschweig/Wiesbaden : Vieweg, 1998. – ISBN 3–528–16590–1

[2] Buchmann, J. : *Einführung in die Kryptographie*. 4., erweiterte Auflage. Berlin/Heidelberg : Springer, 2008. – ISBN 3–540–74451–7

[3] Bundesamt für Sicherheit in der Informationstechnik: *IT-Grundschutz-Kataloge, 11. Ergänzungslieferung*. `http://www.bsi.bund.de/grundschutz`. Version: November 2009. – zuletzt abgerufen am 21. Juli 2010

[4] Dressler, F. ; Kleinöder, J. : *Vorlesungen Netzwerksicherheit/Systemsicherheit*. Website zur Vorlesung. `http://www7.informatik.uni-erlangen.de/~dressler/lectures/netzwerksicherheit-ws0708/`. Version: WS 2007/08. – zuletzt abgerufen am 21. Juli 2010

[5] Köbler, J. : *Vorlesungsskript Kryptologie 1*. Website zur Vorlesung. `http://www.informatik.hu-berlin.de/forschung/gebiete/algorithmenII/Lehre/ws09/krypto1/`. Version: WS 2009/10. – zuletzt abgerufen am 21. Juli 2010

[6] Rose, M. : *Post Office Protocol - Version 3*. RFC 1460 (Draft Standard). `http://www.ietf.org/rfc/rfc1460.txt`. Version: Jun. 1993 (Request for Comments). – Obsoleted by RFC 1725