KRZYSZTOF CABAJ[1,3], WOJCIECH MAZURCZYK[2,3], KRZYSZTOF SZCZYPIORSKI[2,3]

[1] Instytut Informatyki, Politechnika Warszawska, email: *kcabaj@elka.pw.edu.pl*
[2] Instytut Telekomunikacji, Politechnika Warszawska, email: *{wm, ksz}@tele.pw.edu.pl*
[3] SecGroup.PL - Network Security Group, Politechnika Warszawska

Zarządzanie kluczami w sieciach WiMAX

Streszczenie

W artykule przedstawiono protokół PKM (*Privacy Key Management*), który jest wykorzystywany do zarządzania kluczami kryptograficznymi oraz do realizacji usług autoryzacji i uwierzytelnienia w sieciach WiMAX. Za pomocą PKM stacja abonencka przeprowadza proces autoryzacji i uwierzytelnienia w stacji bazowej oraz uzyskuje w bezpieczny sposób dane, z których następnie generowane są klucze kryptograficzne wykorzystywane do zapewnienia usług poufności i integralności. Dodatkowo protokół ten jest odpowiedzialny za synchronizację kluczy pomiędzy komunikującymi się stronami. W artykule scharakteryzowano dwie wersje tego protokołu: pierwszą zdefiniowaną w standardzie definiującym sieci WiMAX (802.16-2004), która z powodu luk została skompromitowana oraz rozszerzoną i poprawioną wersję drugą, która została opublikowana w standardzie 802.16e-2005.

1. System WiMAX i jego ewolucja

WiMAX (*World Interoperability for Microwave Access*) jest jednym z szerokopasmowych, radiowych systemów dostępowych, które uważa się obecnie za najbardziej efektywne w realizacji dostępu do sieci Internet, szczególnie na obszarach słabo zurbanizowanych. WiMAX definiuje interfejs radiowy systemu WMAN (*Wireless Metropolita Area Network*), który jest alternatywą dla sieci kablowych, przy założeniu zasięgu transmisji do kilkunastu kilometrów.

Na standaryzację WiMAX składają się dwa dokumenty organizacji IEEE tj. 802.16-2004 [1] (stacjonarny WiMAX) oraz 802.16e [2] (mobilny WiMAX). W porównaniu z innymi systemami radiowymi (np. *Local Multipoint Distribution*, *LMDS*), WiMax umożliwia szerokopasmowy dostęp zarówno w zasięgu bezpośredniej widoczności (*Line of Sight*, *LOS*), jak i bez niej (*Non Line of Sight, NLOS*). Natomiast głównym celem opracowania drugiego ze standardów WiMAX, IEEE 802.16e, była konieczność wsparcia urządzeń stacjonarnych lub ruchomych, używanych np. w pojazdach. Dokument ten jest właściwie uzupełnioną i poprawioną wersją standardu IEEE 802.16-2004. Uzupełnienia dotyczą przede funkcji umożliwiających obsługę terminali ruchomych przemieszczających się z szybkością pojazdu. Dodano również procedury przenoszenia połączenia pomiędzy obszarami obsługiwanymi przez różne stacje bazowe (procedura *handover*) oraz poprawiono niektóre aspekty bezpieczeństwa (hierarchia i generowanie kluczy, zwiększenie liczby obsługiwanych protokołów uwierzytelniających).

Obecnie potencjalne zastosowania systemów WiMAX to:

- powstawanie nowych sieci lub rozszerzanie zasięgu działania sieci lub dodawanie nowych usług (np. dla operatorów istniejących szerokopasmowych sieci radiowych, lub sieci WLAN),
- powiększenie obszaru działania sieci z zastosowaniem WiMAX jako technologii sieci dostępowej (np. dla operatorów kablowych),
- przesyłanie danych z dużą szybkością w sieci szkieletowej (np. operatorzy sieci radiowych).

2. Architektura bezpieczeństwa w WiMAX

Jedną z istotnych cech sieci budowanych w oparciu o standard 802.16 jest możliwość zrealizowania usług ochrony informacji. W podwarstwie MAC (*Medium Access Control*) wprowadzono podwarstwę bezpieczeństwa (*Security Sublayer*), której rolą jest zapewnianie podstawowych usług ochrony informacji, takich jak: uwierzytelnienie urządzeń korzystających z sieci, zapewnienie poufności i integralności przesyłanych danych oraz zarządzaniem kluczami.

Projektując podwarstwę bezpieczeństwa rozważano przyszłe zastosowania. Sieć WiMAX może służyć nie tylko jako nowa metoda dostępu do sieci danych, ale również m.in. usług multimedialnych (np. wideo na żądanie, *Video on Demand, VoD*). Podwarstwa bezpieczeństwa pozwala na efektywne i bezpieczne wprowadzanie tego rodzaju usług. Każda stacja bazowa jest w stanie prowadzić oddzielnie szyfrowaną komunikację z wieloma pojedynczymi odbiorcami oraz jednocześnie dla wielu grup. Takie podejście do zabezpieczana transmisji

pozwoli w bezpieczny i efektywny sposób zapewnić ochronę różnych rodzajów informacji przesyłanych w sieciach WiMAX.
Jednym z najważniejszych terminów w architekturze bezpieczeństwa sieci WiMAX jest, tworzona dla każdego zidentyfikowanego połączenia, struktura danych zwana **asocjacją bezpieczeństwa** (*Security Association, SA*). Z każdym szyfrowanym kanałem obsługiwanym przez dowolne urządzenie działające w sieci związana jest jedna asocjacja bezpieczeństwa (na każdym urządzeniu, na każdym końcu kanału). W ramach SA przechowuje się wszystkie informacje pozwalające na zaszyfrowanie, odszyfrowanie oraz sprawdzenie integralności przesyłanych danych (aktualnie używane algorytmy kryptograficzne, klucze, czasy ich obowiązywania itp.). Więcej informacji na temat asocjacji bezpieczeństwa oraz architektury bezpieczeństwa sieci WiMAX znajduje się w [8].
Natomiast najważniejszym elementem architektury bezpieczeństwa opracowanym specjalnie na potrzeby sieci WiMAX jest protokół **PKM** (*Privacy Key Management*). Jest on odpowiedzialny za uwierzytelnienie, autoryzację oraz zarządzanie kluczami pomiędzy stacjami abonenckimi (*Subscriber Station, SS*) oraz stacjami bazowymi (Base Stadion, BS). Na podstawie wymienionych w bezpieczny sposób danych pomiędzy SS i BS wykorzystując PKM generowane są klucze służące następnie do zapewnienia usług poufności i integralności przesyłanych danych. Dodatkowo PKM dba o synchronizację kluczy pomiędzy stronami komunikującymi się.

3. Porównanie protokołów PKM w wersji 1 i 2

Pierwsza wersja protokołu PKM, jak wspomniano, została zaproponowana w standardzie 802.16-2004, ale wykazane braki zabezpieczeń zmusiły twórców do opracowania, rozszerzonej (dla zastosowań mobilnych) i poprawionej drugiej wersji, którą opublikowano w standardzie 802.16e-2005. Sposób działania opisywanego protokołu jest oparty na modelu klient-serwer, w którym stroną kliencką jest stacja abonencka, natomiast serwerem stacja bazowa. SS wysyła żądania służące uzyskaniu informacji niezbędnych do wygenerowania kluczy potrzebnych w dalszej części komunikacji. BS odpowiadając na to żądanie dba, żeby klient otrzymał jedynie te dane, do których jest uprawniony.

Wykorzystanie protokołu PKMv1 pozwala na:

a. Jednostronne uwierzytelnienie stacji abonenckiej w stacji bazowej (SS->BS),
b. Dostarczenie uwierzytelnionej stacji abonenckiej właściwych identyfikatorów asocjacji bezpieczeństwa (SAID), do których ma ona uprawienia,
c. Przekazanie uwierzytelnionej stacji abonenckiej klucza **AK** (*Authorization Key*), z którego następnie generowane są: klucz **KEK** (*Key Encryption Key*), który służy do bezpiecznej wymiany klucza szyfrującego **TEK** (*Traffic Encryption Key*) oraz kluczy następnie wykorzystywanych do zapewnienia integralności danych (klucze **HMAC_KEY_D**, **HMAC_KEY_U**) z wykorzystaniem algorytmu HMAC [7].

Jak wspomniano druga wersja protokołu PKM jest rozszerzoną i poprawioną wersją PKMv1. Wprowadzenie nowych mechanizmów uwierzytelnienia (EAP [6]) oraz konieczność obsługi stacji mobilnych wymusiły wprowadzenie zmian do drugiej wersji tego protokołu. Zmieniono również sposób generowania oraz hierarchię kluczy. Generalnie jednak, sposób wymiany wiadomości protokołów PKMv1 i PKMv2 pozostał podobny, choć w przypadku drugiej wersji jest on rozszerzony.

Cechy charakterystyczne obu wersji PKM scharakteryzowano w Tabeli 1.

Wersja PKM	Standard	Najważniejsze cechy
1	802.16-2004	• Możliwe jest jedynie jednostronne uwierzytelnienie SS w BS, • Możliwość wykorzystania do uwierzytelnienia mechanizmu RSA, ale nie EAP (*Extensible Authentication Protocol*), • Nie do końca bezpieczny sposób tworzenia części kluczy
2	802.16e-2005	• Dwustronne uwierzytelnienie SS z BS, • Nowa hierarchia, poprawione tworzenie i dystrybucja kluczy, • Możliwość wykorzystania do uwierzytelnienia mechanizmów rodziny EAP (np. EAP-SIM, EAP-TLS), • Możliwość wykorzystania oddzielnego serwera AAA do sprawdzenia otrzymanych przez BS danych uwierzytelniających, • Nowe mechanizmy m.in. AES-CMAC, czy MBS (*Multicast Broadcast Service*)

Tabela 1: Porównanie dwóch wersji protokołu PKM

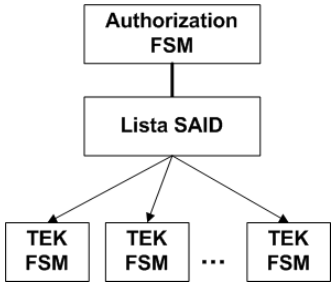
Poniżej przedstawiono w sposób szczegółowy zasadę działania obu wersji protokołu PKM, porównując te wersje w dwóch najważniejszych aspektach: **procesie autoryzacji i uwierzytelnienia**, oraz **zarządzania kluczami**.

4. Przebieg procesu autoryzacji i uwierzytelnienia

Zarządzanie kluczami w sieciach WiMAX jest ściśle powiązane z przebiegiem procesu autoryzacji i uwierzytelnienia, w których są one wykorzystywane. Dlatego też dla zrozumienia celu generowania, dystrybucji i wykorzystania poszczególnych kluczy poniżej opisano ogólny przebieg tego procesu dla obu wersji protokołu PKM.

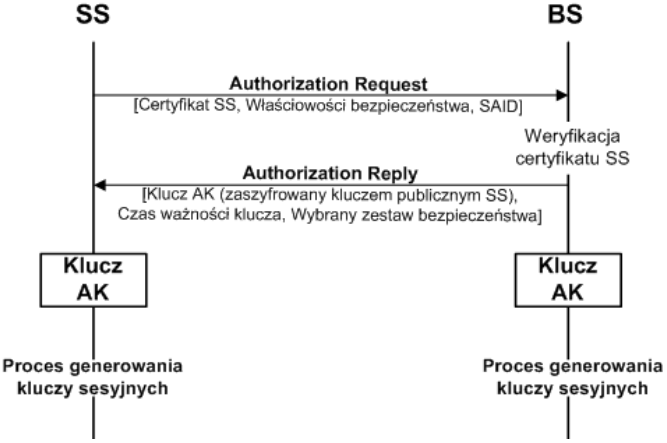
4.1 PKMv1

Przebieg procesu autoryzacji i uwierzytelnienia pomiędzy stacją abonencką, a stacją bazową jest kontrolowany poprzez odpowiedni, zdefiniowany w standardzie, automat stanów i przejść (*Finite State Machine, FSM*) tzw. *Authorization State Machine*. W ramach tego automatu tworzony i uruchamiany jest dla każdego identyfikatora asocjacji bezpieczeństwa (SAID) automat stanów i przejść TEK tzw. TEK FSM. Służy on do zapewniania synchronizacji kluczy sesyjnych pomiędzy BS i SS. Oba automaty pozostają ze sobą w ścisłej zależności tzn. *Authorization State Machine* zarządza podległymi mu TEK FSM i może np. w przypadku braku ponowienia procesu uwierzytelnienia i autoryzacji zablokować wszystkie podległe mu TEK FSM. Na rys. 1 w sposób graficzny przedstawiono zależności pomiędzy opisanymi automatami stanów i przejść.



Rys. 1. Zależności pomiędzy automatami stanów i przejść w WiMAX

Natomiast przebieg procesu autoryzacji i uwierzytelnienia dla PKMv1 przeprowadzany jest tak jak zobrazowano to na poniższym rysunku.



Rys. 2. Przebieg procesu uwierzytelnienia oraz autoryzacji w PKMv1

Przedstawiony na rysunku przebieg procesu autoryzacji i uwierzytelnienia można opisać następująco (w nawiasach podano kierunek komunikacji):

1. (**SS -> BS**) - stacja abonencka rozpoczyna proces uwierzytelnienia poprzez wysłanie wiadomości *Authentication Information* do właściwej sobie stacji bazowej. Zawiera ona certyfikat X.509, właściwy producentowi SS (wydany przez producenta lub jednostkę zewnętrzną). Wiadomość ta w protokole PKM odgrywa rolę jedynie informacyjną i może zostać zignorowana przez BS. Pozwala ona jedynie na uzyskanie informacji o producencie stacji abonenckiej.
2. (**SS -> BS**) - zaraz po wysłaniu wiadomości *Authentication Information* stacja abonencka inicjuje wysłanie wiadomości *Authorization Request* do stacji bazowej. Wiadomość ta jest żądaniem pozyskania

klucza AK oraz identyfikatorów SAID dla asocjacji bezpieczeństwa, do których SS jest uprawniona. Zawiera ona oprócz certyfikatu SS, listę algorytmów kryptograficznych, które wspiera stacja abonencka – ma to na celu prezentację możliwości zabezpieczeń w postaci listy odpowiednich identyfikatorów (pary algorytmów: szyfrowania oraz uwierzytelnienia danych). Wysłanie takiej listy jest niezbędne, aby mogło dojść do negocjacji mechanizmów zabezpieczeń ze stacją bazową.

3. (**BS -> SS**) - po odebraniu wiadomości BS weryfikuje certyfikat SS a następnie dokonuje wyboru odpowiedniego algorytmu kryptograficznego spośród nadesłanych przez stację abonencką. Dodatkowo stacja bazowa aktywuje klucz AK, szyfruje go z wykorzystaniem klucza publicznego SS, a następnie odsyła go w wiadomości *Authorization Reply*. W wiadomości tej zawarty jest również numer sekwencyjny klucza AK (aby można było je odróżnić między sobą) oraz jego czas życia.

Po uwierzytelnieniu stacja abonencka okresowo żąda wykonania ponownej operacji uwierzytelnienia i autoryzacji. SS musi okresowo przechodzić przez ten proces, aby odświeżać klucz sesyjny TEK, którego czas obowiązywania jest skończony. Ponowienie procesu uwierzytelnienia i autoryzacji odbywa się tak samo, jak opisano powyżej, ale z pominięciem kroku 1.

4.2. PKMv2

W drugiej wersji protokołu PKM umożliwiono wykorzystanie do uwierzytelnienia protokołów rodziny EAP, co znacznie ułatwia budowę sieci z dużą ilością stacji bazowych. EAP umożliwia przekazanie odpowiedzialności za przeprowadzenie uwierzytelnienia do oddzielnego serwera AAA. Maszyna taka może odpowiadać na zapytania związane z uwierzytelnianiem stacji abonenckich podłączonych pod dowolną stację bazowa. Tego typu rozwiązanie upraszcza zarządzanie danymi służącymi do uwierzytelnienia i autoryzacji, które nie muszą być trzymane w każdej stacji bazowej, a jedynie w jednym, centralnym miejscu. Inną zaletą tej metody jest zwiększenie funkcjonalności metod uwierzytelniania poprzez możliwość przekazania różnych danych służących identyfikacji klienta (mogą być to np. certyfikaty klienta, czy dane z karty SIM).

W przypadku PKMv2 uwierzytelnienie jest dwustronne i może zostać wykonane na dwa sposoby. W pierwszym przypadku może być to jedynie wzajemne uwierzytelnienie, natomiast w drugim może być dodatkowo poprzedzone uwierzytelnieniem z wykorzystaniem protokołu EAP. W tej drugiej sytuacji wzajemne uwierzytelnienie jest wykonywane podczas inicjującego dołączenia stacji abonenckiej do sieci, a uwierzytelnienie EAP wykonywane jest jedynie, gdy SS ponawia ten proces. Wzajemne uwierzytelnienie oznacza w tym przypadku:

- Uwierzytelnienie SS w BS oraz BS w SS,
- Dostarczenie przez BS uwierzytelnionemu SS klucza pre-AK (*pre-Autorization Key*), z którego następnie zostaną wygenerowane klucze: KEK oraz HMAC/CMAC (do zapewnienia integralności w standardzie IEEE 80216e można wykorzystywać również algorytmy CMAC [5]),
- Dostarczenie przez BS uwierzytelnionemu SS listy asocjacji bezpieczeństwa oraz ich identyfikatorów.

W PKMv1 i PKMv2 wiadomości protokołu wymieniane w procesie autoryzacji i uwierzytelnienia mają takie same nazwy, ale różnią się od tych wykorzystywanych w wersji 1. Różnica polega na tym, że w PKMv2 wiadomości te posiadają dodatkowe pola (przede wszystkim ponieważ wykorzystywana jest inna hierarchia kluczy). Poza tą zmianą sam *Authorization FSM* zarówno w PKMv1 i PKMv2 nie ulega zmianie.

4.2.1 Wykorzystanie protokołu EAP

Przykładowy proces autoryzacji i uwierzytelnienia z wykorzystaniem EAP przebiega następująco - załóżmy, że SS i BS wynegocjowały podwójny tryb działania protokołu EAP (np. *EAP after EAP*). W takim przypadku pomiędzy nimi zachodzi dwufazowa wymiana wiadomości EAP, która odbywa się następująco:

- aby zainicjować pierwszy etap uwierzytelnienia z wykorzystaniem EAP, stacja abonencka wysyła wiadomość *PKMv2 EAP Start*,
- pierwsza faza EAP przebiega z wykorzystaniem wiadomości *PKMv2 EAP Transfer* bez zapewnienia integralności (bez wsparcia mechanizmów HMAC/CMAC),
- w trakcie trwania pierwszej wymiany wiadomości EAP, jeśli BS musi przesłać informację *EAP-Success*, jest ona zawierana w wiadomości *PKMv2 EAP Complete*, którą następnie podpisuje się z wykorzystaniem klucza EIK (*EAP Integrity Key*). Po odebraniu przesłanej wiadomości SS jest w stanie zweryfikować odebrane dane, jeśli jest w posiadaniu klucze EIK oraz PMK (*Pairwise Mater Key*).
- jeśli pierwsza faza EAP zostanie zakończona pomyślnie, to SS przesyła wiadomość *PKMv2 EAP Start* podpisaną kluczem EIK, w celu zainicjowania drugiej fazy uwierzytelnienia. W tym przypadku, gdy w BS wiadomość ta zostanie poprawnie zweryfikowana następuje przesłanie do SS wiadomości *PKMv2 Authenticated EAP*,

- SS i BS przeprowadzają drugą wymianę wiadomości protokołu EAP z wykorzystaniem wiadomości *PKMv2 Authenticated EAP* podpisaną z wykorzystaniem klucza EIK,
- jeśli druga faza uwierzytelnienia EAP kończy się pomyślnie zarówno SS, jak i strona uwierzytelniająca generują klucz AK z kluczy: PMK i PMK2. Następnie SS i BS wykonują procedurą tzw. SA-TEK 3-way handshake (opisany w punkcie 4.2.1).

Po zakończonym sukcesem wykonaniu procedury autoryzacji i uwierzytelnienia stacje: abonencka i bazowa ponawiają ją okresowo zgodnie z czasem obowiązywania kluczy PMK/PMK2. Ponowne wykonanie tego procesu odbywa się z podwójnym wykorzystaniem EAP (tak jak w przypadku uwierzytelnienia inicjującego). W pozostałych przypadkach, SS i BS przeprowadzają tę procedurę z jednokrotnym wykorzystaniem protokołu EAP.

4.2.2 SA-TEK 3-way handshake – wymiana klucza TEK

W PKMv2 po wykonaniu procedury autoryzacji opisanej w poprzednim punktach oraz po tym, jak odpowiednie klucze pomiędzy stronami zostaną wymienione wykonywany jest tzw. *SA-TEK 3-way handshake*. Przebieg tej procedury jest następujący (w nawiasach podano kierunek komunikacji):

- (**BS -> SS**) - stacja bazowa przesyła do SS odpowiednią wiadomość *PKMv2 SA-TEK-Challenge*, zabezpieczoną z wykorzystaniem algorytmu HMAC/CMAC, która zawierająca liczbę losową,
- (**SS -> BS**) - stacja abonencka odpowiada BS wiadomością *PKMv2 SA-TEK-Request* (zabezpieczoną również z wykorzystaniem HMAC/CMAC),
- (**BS -> SS**) - po odebraniu wiadomości *PKMv2 SA-TEK-Request*, BS weryfikuje, czy otrzymany w tej wiadomości **AKID** (identyfikator klucza AK) odnosi się do dostępnego klucza AK oraz integralność wiadomości. Dodatkowo sprawdzana jest wartość liczby losowej przesłana w *PKMv2 SA-TEK-Request* z tą zawartą w wiadomości *PKMv2 SA-TEK-Challenge*. Jeśli wiadomość *PKMv2 SA-TEK-Request* zostanie uznana za prawidłową wtedy BS przesyła *PKMv2 SA-TEK-Response* zawierającą listę asocjacji bezpieczeństwa, do których dana SS jest uprawniona. W wiadomości tej zawarty jest również klucz TEK, jego czas życia, numer sekwencyjny oraz ewentualnie wektor inicjujący dla szyfru w trybie CBC (jeśli jest wymagany),
- po odebraniu wiadomości *PKMv2 SA-TEK-Response* weryfikowana jest wartość wyliczona z wykorzystaniem HMAC/CMAC. Jeśli sprawdzenie przebiegnie pomyślnie, wtedy stacja abonencka staje się posiadaczem kolejnego klucza TEK.

4.2.3 Wykorzystanie algorytmu RSA

Protokół RSA może być opcjonalnie wykorzystany do wymiany klucza pre-PAK. Aby to zrealizować wykorzystuje się następujące wiadomości PKMv2: *PKMv2 RSA-Request, PKMv2 RSA-Reply, PKMv2 RSA-Reject* oraz *PKMv2 RSA-Acknowledgement*. Z wykorzystaniem wymienionych wiadomości odbywa się wymiana w rezultacie której klucz pre-PAK zostaje przesłany z BS do SS zaszyfrowany kluczem publicznym stacji abonenckiej.

5. Zarządzanie kluczami

5.1 PKM v1

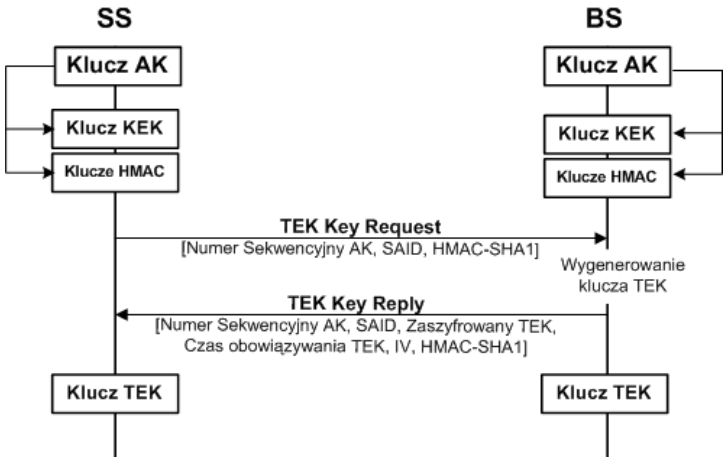
Uzgadnianie kluczy pozwala na prowadzenie bezpiecznej komunikacji. W protokole PKMv1 do bezpiecznej wymiany pierwszego z kluczy wykorzystano kryptografię asymetryczną (patrz punkt 4.1). Zgłaszająca się stacja abonencka wysyła do stacji bazowej prośbę o klucz uwierzytelniający (*Authorization Key, AK*), czyli współdzielony sekret, za pomocą którego będą wytworzone kolejne kluczę używane w dalszej części komunikacji. Na podstawie tego klucza generowane są kolejne:

- klucz **KEK** (Key Encryption Key) służący do szyfrowania kluczy sesyjnych, które będą używane do szyfrowania danych przesyłanych w sieci,
- klucze **HMAC_KEY_D**, **HMAC_KEY_U** służące do zapewnienia integralności danych odpowiednio na łączach „downlink” oraz „uplink”.

Wprowadzenie takiej hierarchii kluczy pozwala na częstszą zmianę kluczy, które używane są do zabezpieczenia większych ilości danych. Nie można zapomnieć także, że podejście takie ma na celu oszczędzanie mocy obliczeniowej w urządzeniach abonenckich, poprzez umiejętne stosowanie kryptografii symetrycznej. Należy pamiętać, że mogą być to urządzenia proste, a szyfry asymetryczne mogą być nawet do 1000 razy wolniejsze niż symetryczne.

Stacja bazowa jest odpowiedzialna za zarządzanie i utrzymywanie kluczy oraz informacji służących do ich generowania, które następnie wykorzystywane są dla wszystkich istniejących asocjacji bezpieczeństwa. Stacja abonencka natomiast dba o okresowe uaktualnianie tych informacji.

Sposób wykorzystania opisanych powyżej kluczy przedstawia poniższy rysunek:

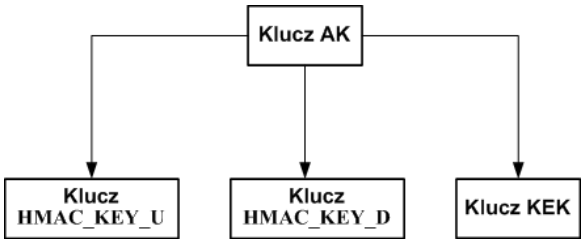


Rys. 3. Przebieg procesu zabezpieczania w sieciach WiMAX dla PKMv1

Na rysunku widać, iż doprowadzenie do przekazania klucza TEK jest możliwe z wykorzystaniem wiadomości *TEK Key Request*, *TEK Key Reply*. W kolejnych podpunktach scharakteryzowano sposób generowania i zarządzanie kluczami AK, KEK oraz TEK.

5.1.1. Klucz AK (Authorization Key)

Klucz AK, jak wspomniano, zarówno w stacji bazowej jak i abonenckiej jest wykorzystywany do wyliczenia kluczy KEK oraz HMAC_KEY_U i HMAC_KEY_D. Jego wartość jest generowana losowo, natomiast jego długość wynosi 128 bitów. Opisaną hierarchię przedstawia rysunek 4.



Rys. 4. Przebieg procesu generowania kluczy KEK oraz HMAC z klucza AK

Inicjacja procesu uwierzytelnienia i autoryzacji przez stację abonencką powoduje uaktywnienie w stacji bazowej nowego klucza AK, który następnie jest przesyłany do SS. Klucz AK pozostaje aktywny przez określony czas. Aby uniknąć potencjalnych zakłóceń w świadczeniu usług podczas ponawiania procesu uwierzytelnienia kolejne okresy ważności kluczy AK zachodzą na siebie. Wynika z tego, że obie strony komunikacji (zarówno SS jak i BS) powinny być w stanie w okresie przejściowym wspierać dwa jednocześnie aktywne klucze AK. Jeśli okres obowiązywania bieżącego klucza AK upłynie, a stacja abonencka nie zdąży wykonać ponownego procesu uwierzytelnienia i autoryzacji, wtedy stacja bazowa nie generuje nowego klucza AK, uznaje SS za nieuprawnioną oraz usuwa z tablicy kluczy wszystkie właściwe danej stacji abonenckiej klucze TEK.

Stacja bazowa wykorzystuje powstałe z AK klucze do:

- weryfikacji skrótów HMAC zawartych w wiadomościach *Key Request* otrzymywanych ze stacji abonenckiej (klucz HMAC_KEY_U),
- obliczenia skrótów HMAC i ich umieszczenia w wiadomościach *Key Reply*, *Key Reject* oraz *TEK Invalid* (klucz HMAC_KEY_D). Po stronie SS są one wykorzystane do uwierzytelnienia odebranych wiadomości,
- szyfrowania klucza TEK, który jest następnie wysyłany w wiadomości *Key Reply* do SS (klucz KEK).

Natomiast stacja abonencka:

- wykorzystuje klucz HMAC_KEY_U do wyliczania skrótu niezbędnego do umieszczenia w wiadomości *Key Request*,

- używa klucza HMAC_KEY_D w celu uwierzytelnienia wiadomości *Key Reply*, *Key Reject* oraz *TEK Reject*,
- szyfruje/odszyfrowuje klucz TEK z wiadomości *Key Reply*.

5.1.2 Klucz KEK (Key Encryption Key)

Klucz KEK jest kluczem generowanym na podstawie wartości klucza AK (por. Rys. 5). Służy do bezpiecznej wymiany klucza TEK. Sposób obliczenia klucza przedstawia poniższe przekształcenie:

$$KEK = \text{Truncate-128}(SHA1(((AK| 0^{44}) \text{ xor } 53^{64})$$

W 802.16-2004 bezpieczna wymiana klucza TEK jest możliwa z zastosowaniem klucza KEK oraz jednego z algorytmów: 3DES, RSA oraz AES. Dodatkowo w standardzie 802.16e-2005 (PKMv2) dodano również możliwość wykorzystania AES w trybie *Key Wrap*.

5.1.3 Klucz TEK (Traffic Encryption Key)

Klucz TEK jest generowany losowo przez stację bazową, a jego bezpieczna wymiana pomiędzy stronami uczestniczącymi w komunikacji jest możliwa poprzez wykorzystanie szyfrowania z użyciem klucza KEK. Stacja bazowa utrzymuje dwa aktywne klucze TEK (wykorzystywane w zależności od kierunku transmisji) dla każdej asocjacji bezpieczeństwa. BS dołącza je do wiadomości *Key Reply* razem z ich czasami obowiązywania. Dodatkowo do tej wiadomości dodawany jest, jeśli jest wymagany dla działania protokołu kryptograficznego, który będzie wykorzystywany w trakcie połączenia, wektor inicjujący CBC.

Stacja abonencka również musi również utrzymywać dwa aktywne klucze TEK dla każdego identyfikatora asocjacji bezpieczeństwa (SAID). Za proces uaktualniania kluczy odpowiedzialna jest stacja abonencka. BS przechodzi na szyfrowanie nowym klucz TEK bez względu na to, czy SS uzyskała jego kopię, czy też nie.

Jak wspomniano wszystkie klucze sesyjne mają swój, określony czas obowiązywania standardowo wynoszący 12 h (możliwa jest zmian parametrów w granicach 30 minut – 7 dni). Po tym czasie klucze przestają być aktualne i nie można za ich pomocą prowadzić szyfrowanej transmisji. W celu umożliwienia płynnej pracy w momencie zmiany kluczy standard przewiduje, iż w każdym momencie urządzania w sieci będą w stanie korzystać z dwóch kluczy TEK.

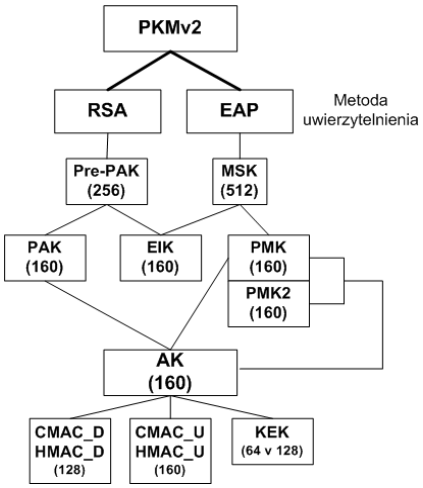
5.2 PKM v2

Jak wspomniano obie wersje protokołu PKM różni między innymi hierarchia kluczy, która w wersji drugiej protokołu została zmieniona. Ponieważ istnieją dwa sposoby uwierzytelnienia (EAP lub opcjonalnie RSA - por. punkty 4.2.1 oraz 4.2.3), dlatego też dostępne są potencjalnie dwa różne źródła informacji wykorzystywane do generowania kluczy.

Klucze, których używa się do zabezpieczenia integralności wiadomości zarządzających oraz szyfrowania danych, generowane są na podstawie informacji dostarczonych w trakcie procesu autoryzacji i uwierzytelnienia. W zależności od tego jaka metoda uwierzytelnienia zostanie wybrana są to:

- dla algorytmu RSA jest to klucz **pre-PAK** (*pre-Primary Authorization Key*),
- dla algorytmu EAP jest to klucz **MSK** (*Master Key*).

Oba wymienione klucze tworzą korzeń w hierarchii kluczy w protokole PKMv2. Nowa, poprawiona hierarchia kluczy, wykorzystywana w PKMv2 została przedstawiona na rysunku 5 (w nawiasach podano długość kluczy w bitach).



Rys. 5. Hierarchia kluczy w PKMv2 w zależności od wykorzystanego algorytmu uwierzytelnienia (w nawiasach podano długość klucza w bitach)

5.2.1 Klucz pre-PAK (pre-Primary Authorization Key)

Klucz pre-PAK wykorzystuje się do wygenerowania klucza **PAK** (*Primary Authorization Key*). Dodatkowo na podstawie klucza pre-PAK powstaje opcjonalny klucz **EIK** (*EAP Integrity Key*), który używany jest do transmisji uwierzytelnionej zawartości mechanizmu EAP. Oba klucze generowane są zgodnie z poniższym przekształceniem:

$$EIK \mid PAK \leftarrow Dot16KDF(\text{pre-PAK}, \text{SS MAC Address} \mid BSID \mid \text{„EIK+PAK”}, 320)$$

Gdzie Dot16KDF jest przekształceniem opisanym następującym pseudokodem (zmienna ALG oznacza algorytm HMAC lub funkcję skrótu SHA-1) :

```
Dot16KDF(key, astring, keylength)
{
 result = null;
 Kin = Truncate (key, 128);
 for (i = 0;i <= int((keylength-1)/128); i++) {
 result = result | ALG (Kin, i | astring | keylength);
}
return Truncate (result, keylength);
}
```

Powstały w ten sposób 160-bitowy klucz PAK służy do wygenerowania klucza AK.

5.2.2 Klucz MSK (Master Session Key)

W sytuacji, gdy przed wymianą protokołu EAP nastąpiło wzajemne uwierzytelnienie pomiędzy BS i SS lub jeśli wykorzystywany jest algorytm EAP w trybie *EAP-in-EAP*, wtedy jego wiadomości mogą być zabezpieczane z wykorzystaniem 160-bitowego klucza EIK (wygenerowanego na podstawie klucza pre-PAK).

Wynikiem wymiany wiadomości protokołu EAP jest klucz **MSK** o długości 512 bitów. Klucz ten musi być znany serwerowi AAA, stronie uwierzytelniającej oraz stacji abonenckiej. Zarówno SS jak i strona uwierzytelniająca tworzą z klucza MSK klucz **PMK** (*Pairwise Master Key*) oraz opcjonalnie klucz EIK. Powstają one poprzez skrócenie klucza MSK do 320 bitów (podczas pierwszej metody EAP) zgodnie ze wzorem:

$$EIK \mid PMK \leftarrow \text{truncate}(MSK, 320)$$

Natomiast podczas drugiej fazy wymiany EAP, powstaje klucz PMK2, utworzony z klucza MSK2 następująco:

$$PMK2 \leftarrow \text{truncate}(MSK2, 160)$$

Jeśli proces autoryzacji i uwierzytelnienia z wykorzystaniem protokołu EAP zakończy się sukcesem oraz gdy protokół EAP działa w trybie *EAP after EAP*, to uwierzytelnione wiadomości tego mechanizmu przenoszą kolejne wiadomości EAP. W ten sposób następuje zatem kryptograficzne powiązanie pomiędzy poprzednią a bieżącą sesją uwierzytelnienia.

5.2.3 Klucza AK(Authorization Key)

Klucz AK w stacji abonenckiej jak i bazowej może być, w zależności od wykorzystanego protokołu uwierzytelniającego, generowany następująco:

- na podstawie kluczy PMK i PMK2 (z procedury autoryzacji opartej na EAP):
  AK ← Dot16KDF (PMK PMK2, SS MAC Address | BSID | "AK", 160)

- na podstawie klucza PAK (z procedury autoryzacji opartej na RSA) oraz PMK (z procedury autoryzacji opartej na EAP):
  AK ← Dot16KDF (PAK PMK, SS MAC Address | BSID | PAK | "AK", 160)

- na podstawie klucza PAK:
  AK ← Dot16KDF (PAK, SS MAC Address | BSID | PAK | "AK", 160)

- na podstawie klucza PMK:
  AK ← Dot16KDF(PMK, SS MAC Address | BSID | "AK", 160)

5.2.4 Klucz KEK oraz klucze HMAC/CMAC

Klucze KEK, CMAC_KEY_U, CMAC_KEY_D, HMAC_KEY_U, HMAC_KEY_D są generowane na podstawie klucza AK (podobnie jak w PKMv1 – również ich rola jest podobna). Generowanie poszczególnych kluczy przebiega zgodnie z poniższymi zależnościami:

- jeśli wykorzystujemy algorytm CMAC wtedy postać generacja kluczy KEK, CMAC_KEY_U, CMAC_KEY_D jest następująca:
  CMAC_KEY_U | CMAC_KEY_D | KEK ← Dot16KDF(AK, SS MAC Address | BSID | "CMAC_KEYS+KEK", 384)
- jeśli wykorzystujemy algorytm HMAC wtedy postać generacja kluczy KEK, HMAC_KEY_U, HMAC_KEY_D jest następująca:
  HMAC_KEY_U | HMAC_KEY_D | KEK ← Dot16KDF(AK, SS MAC Address | BSID | "HMAC_KEYS+KEK", 448)

6 Podsumowanie

W artykule przedstawiono protokół PKM (*Privacy Key Management*), który jest wykorzystywany do zarządzania kluczami kryptograficznymi oraz do realizacji usług uwierzytelnienia i autoryzacji w sieciach WiMAX. Omówiono dwie wersje tego protokołu: pierwszą dla zastosowań stacjonarnych, którą charakteryzuje zastosowanie uproszczonej hierarchii kluczy oraz możliwość jednostronnego uwierzytelnienia (SS w BS), co było poważnym osłabieniem bezpieczeństwa oferowanego przez ten mechanizm. Dostosowując standard WiMAX do wymagań zastosowań mobilnych w wersji drugiej protokołu PKM zlikwidowano niedostatki wersji pierwszej poprzez umożliwienie realizacji procesu autoryzacji i uwierzytelnienia z wykorzystaniem algorytmów EAP i RSA. Znacząco została zmieniona w związku z tym również hierarchia kluczy oraz sposób ich zarządzania. Dlatego też można uznać, iż druga wersja protokołu PKM w sposób zadowalający umożliwia realizację usług: autoryzacji, uwierzytelnienia oraz w sposób bezpieczny dostarcza niezbędnych danych dla realizacji usług poufności i integralności w sieciach WiMAX.

Literatura:

[1] IEEE 802.16-2004: IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems, Czerwiec 2004
[2] IEEE 802.16e-2005: standard: IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands, Grudzień 2005
[3] D. Pareek, The Business of WiMAX, John Willey & Sons Ltd, 2006
[4] A. Fellah, R. Syputa, C. Maynard, WiMAX and Broadband Wireless (Sub-11Ghz) Worldwide Market Analysis and Trends 2005-2010, Kwiecień 2005 (3rd Edition)
[5] M. Dworkin: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, NIST Special Publication 800-38B, maj 2005
[6] B. Aboba, L. Blank, J. Vollbrecht, Extensible Authentication Protocol (EAP), Request for Comments: 3748, Czerwiec 2004
[7] H. Krawczyk, M. Bellare, R. Canetti, HMAC: Keyed-Hashing for Message Authentication, Request for Comments: 2104, Luty 1997
[8] K. Cabaj, W. Mazurczyk, K. Szczypiorski, Bezpieczeństwo bezprzewodowych sieci WiMAX, Materiały: XI Krajowa Konferencja Zastosowań Kryptografii Enigma'2007, 23-25 maja 2007, Warszawa