3GPP-WLAN 연동을 위한 EAP-AKA에서의 키 생성에 관한 연구

박미애*, 김용희*, 김창범*, 이옥연*

*국민대학교 수학과

Study on key generation in EAP-AKA for 3GPP-WLAN interworking

Mi-Ae Park*, Yong-Hee Kim*, Chang-Bum Kim*, Ok-Yeon Yi

*Department of Math. Kookmin Univ.

요 약

본 논문에서는 3GPP-WLAN 연동 보안에 필수인 EAP-AKA를 기반으로 한 인증/재인증의 개요와 인증 진행 부분에서 협상되는 마스터 세션 키 생성과 EAP AKA 패킷을 보호하기 위해 사용되는 키 생성에 관하여 설명하고, EAP-AKA 과정에서 생성되는 키의 안전성을 분석하고, EAP-AKA를 사용하는 3GPP-WLAN 연동의 효율성 및 고려사항에 대하여 고찰하였다.

I. 서론

최근 무선 인터넷 서비스 및 WLAN 활성화 흐름과 함께 3G 이동 통신망과 WLAN 망간의 서비스 연동 이슈가 큰 관심을 일으키고 있다. 현재 3GPP-WLAN 연동 표준화는 ETSI BRAN 및 3GPP 등에서 수행되고 있으며, 연동을 위해서 크게 단말에서의 dual-mode 기능 지원, 각 망에서의 로밍과 이동성 지원, 과금 및 연동 보안과 같은 기술적인 이슈들이 고려되고 있다.

3GPP-WLAN 연동시, 보안 요구사항으로 인증 기법은 Challenge/Response 프로토콜을 기반으로 상호 인증되어야 한다는 것과 가입자나 네트워크 인증에 사용되는 모든 long-term security credentials은 UICC나 SIM 카드에 저장되어야 하고, 여기에 저장된 내용들은 밖으로 유출될 수 없다는 것이 제시되었다.

이러한 요구사항을 만족시키기 위해서 연동 시스템에서는 인증과 키 일치를 위한 보안 메커니즘으로 USIM을 기반으로 한 AKA 인증 메커니즘을 EAP에 적용하였다.

EAP(Extensible Authentication Protocol)는 WLAN UE와 3GPP AAA 서버 사이에 end-to-end 인증을 허가하는 것으로 이것은 전송을 위해 다른 인증 메커니즘(EAP-SIM, EAP-AKA, EAP-TLS 등)을 허가하는 일반적인 프로토콜이다.

본 논문에서는 WLAN-3GPP 연동 보안에 필수인 EAP-AKA의 개요와 이 프로토콜의 진행 과정을 서술하고 선택적인 재인증과 인증 진행 부분에서 협상되는 마스터 세션 키 생성과 EAP AKA 패킷을 보호하기 위해 사용되는 키 생성에 관하여 논하였다. 마지막으로 키를 중심으로 EAP-AKA의 안전성을 분석하였다.

II. 본문

1. EAP-AKA 인증 기법

1) 개요

EAP AKA 인증 기법은 UMTS 인증과 AKA(Authentication & Key Agreement) 메커니

증을 사용하는 EAP(Extensible Authentication Protocol)이다. UMTS AKA는 대칭키를 기반으로 Challenge-Response 메커니즘을 사용하며, 상호 인증을 제공한다. 이 인증 기법을 위해 USIM은 필수적이다. EAP AKA 인증 기법은 다른 EAP 방법들과는 달리 무선랜 환경에서 3GPP 인증 인프라를 사용할 수 있으므로 접근 제어/과금이 요구되는 3GPP-WLAN 연동 시나리오 2[5] 이상의 조건을 만족할 수 있다.

2) 프로토콜의 진행

EAP AKA는 다음과 같은 순서로 진행된다.

1. UE와 WLAN 사이에 연결이 성립된다.

2. WLAN은 UE에게 EAP-Request/Identity를 송신하여 UE의 신원을 요구한다.

3. UE는 자신의 신원을 포함한 EAP-Response /Identity로 응답하고, WLAN은 신원을 확인하여 3GPP AAA 서버를 선택한 후, UE로부터 수신한 패킷을 재전송 한다.

4. 3GPP AAA 서버는 사용자에 대한 AV가 존재하는지를 확인하고, 존재하지 않는다면 HSS /HLR에서 검색한다.

5. 3GPP AAA 서버는 사용자가 WLAN 서비스를 사용할 권한이 있는지를 검증한다.

6. 3GPP AAA 서버는 RAND, AUTN, 익명 신원이 포함된 EAP-Request/AKA-Challenge를 UE에게 송신한다.

7. UE는 AUTN 검증으로 네트워크를 인증하고, SQN으로 동기를 확인한다. SQN, AUTH 검증에 성공하면 RES, CK, IK를 계산한다. (만일 SQN 검증에 실패하면, 재동기화를 실행하고, AUTN 검증에 실패하면 프로토콜을 종료한다.)

8. UE는 계산한 RES를 포함한 EAP-Response/ AKA-Challenge를 3GPP AAA 서버에게 송신한다.

9. 3GPP AAA 서버는 XRES와 RES를 비교하여 UE를 인증한다. 인증에 실패하면 3GPP AAA 서버는 UE에게 EAP-Failure를 송신하고 프로토콜을 종료한다.

10. 인증에 성공하면, 3GPP AAA 서버는 WLAN에게 EAP-Success 메시지와 함께 key material을 송신한다.

11. WLAN은 key material을 저장하고, UE에게 EAP-Success 메시지를 송신한다.

그림 1은 프로토콜의 전체 흐름을 보여준다.

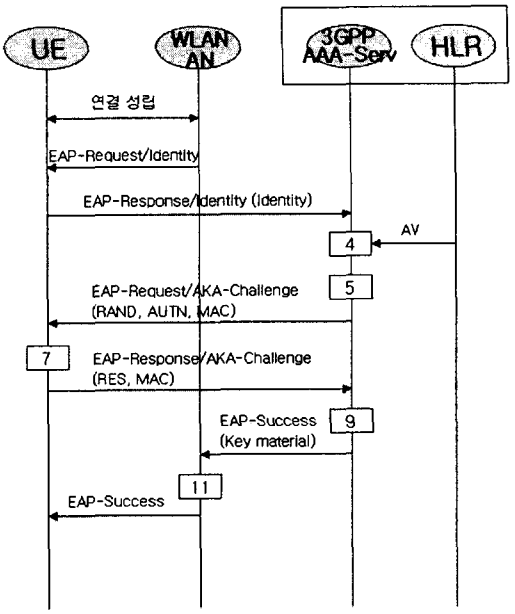

그림 1: EAP-AKA 인증 진행과정

3) 신원 관리

UMTS는 영구 신원 IMSI(International Mobile Subscriber Identity)으로 사용자를 식별한다. 인터넷 AAA 프로토콜은 username@realm 형식으로 표현되는 NAI(Network Access Identifier)로 사용자를 식별한다. 운영자는 EAP-AKA를 사용하는 사용자를 위해 특별한 realm을 지정할 수 있다.

EAP/AKA는 영구, 익명, 재인증의 username 형식을 갖는 각각의 세 가지 신원을 갖는다. 영구 신원은 "0" ‖ IMSI 형태로 유도될 수 있으며, 익명 신원은 UMTS의 TMSI와 같은 역할을 한다. 영구 신원이 암호화되지 않은 상태에서 전송되기 때문에, 익명 신원은 수동적인 청취 공격으로 사용자의 영구 신원을 보호하기 위하여 옵션으로 사용될 수 있다. 주어진 서버와 첫 번째로 연결되는 경우를 제외하고, UE는 성공적인 이전의 인증 과정을 통해 수신한 익명을 저장함으로써 영구 신원을 요구하는 능동 공격으로부터 부분적으로 보호될 수 있다. (UE는 서버가 익명을 정확하게 유지하고 있다고 간주할 수 있을 경우에만 영구 신원 요구에 대해 거부할 수 있다.) 서버는 익명 신원을 디코드할 수 없을 경우에 UE에게 영구 신원을 요구하며, 영구 신원을 인지할 수 없는 경우에는 인증을 종료한다.

영구, 익명 신원은 그림 1에서 표현된 full 인증을 위해서 요구되며, 재인증 신원은 오직 재인증 시에만 유효하다.

4) 재인증

그림 1에서 나타난 full 인증은 매번 새로운 AV(인증 벡터)를 요구하고 있다. 따라서 인증이 빈번히 발생할 경우에는 네트워크 운영에 부하를 초래하므로, EAP AKA는 HLR/HSS로부터 새로운 AV를 요구하지 않는 재인증 과정을 옵션으로 포함한다.

재인증은 옵션이므로 인증 개체 중 어느 하나라도 full 인증을 요구하면, full 인증을 수행해야만 한다.

재인증에 관한 좀 더 자세한 내용은 3장에서 다룬다.

2. EAP-AKA에서의 키 생성

1) EAP-AKA에서의 키 생성

EAP-AKA full 인증에서, MK(Master Key)는 다음과 같이 유도된다.

MK(160비트) = SHA1(Identity ‖ IK ‖ CK)

MK는 TEKs(Transient EAP Keys)를 생성하는 PRF (Pseduo-Random number Function)로 공급된다. 2-2) 참조. TEKs는 EAP-AKA 패킷과 MSK (Master Session Key)를 보호하기 위해 요구된다. EAP-AKA는 메시지 인증 키 K_aut와 암호화 키 K_encr의 두 개의 TEKs를 갖는다. MK와 TEKs가 full 인증에서만 업데이트 되는데 비해, MSK와 IV는 재-인증에서도 업데이트 된다.

PRF로의 초기 설정 및 입력 값을 정리하면 다음과 같다.

- b = 160 (160 비트의 XKEY와 XVAL 사용)
- XKEY = MK
- XSEED_j = 0 ,∀j

EAP-AKA를 위해 필요한 각 키들은 다음과 같이 얻어진다.

- PRF의 결과물 $x_0, x_1, \ldots, x_{m-1}$을 연접하고, 이것을 x라고 하자.
- K_encr(128비트) : x의 0~127비트
- K_aut(128비트) : x의 128~255비트
- MSK(128바이트) : x의 256~1279비트
- IV(64비트) : x의 1280~1344비트
- 만일 복수의 키가 요구된다면, x를 위와 동일한 순서로 반복하여 잘라내어 원하는 만큼의 키를 생성한다.

* IV는 MSK와 함께 EAP-AKA 패킷의 암호화에 사용된다.

재인증에서는 동일한 PRF와 MK로 XKEY'을 다음과 같이 계산하여 새로운 TEKs를 계산한다.

XKEY'=SHA1(Identity ‖ counter ‖ NONSE_S ‖ MK)

NONSE_S(16비트)와 counter는 EAP-Response /AKA-Reauthentication 패킷에 사용된 속성이다.

MSK의 첫 번째 32바이트는 802.11i의 PMK로 사용될 수 있다.

2) PRF

Step 1: Seed-Key(XKEY)를 선택

Step 2: t = 67452301 EFCDAB89 98BADCFE 10325476 C3D2E1F0라고 하자. 이것은 FIPS SHS -SHA1[1] 내의 H0 ‖ H1 ‖ H2 ‖ H3 ‖ H4에 대한 IV 이다.

Step 3: For j=0 to m-1 do

3.1 XSEED_j → 사용자 입력(필수)

3.2 For i=0 to 1 do

a. XVAL=(XKEY+XSEED_j) mod 2^b

b. w_i = G(t, XVAL)

c. XKEY=(1+XKEY+w_i) mod 2^b

3.3 x_j = w_0 ‖ w_1

* G : [1]의 부록 3.3참고.

3. 재인증

2장에서 설명된 바와 같이, 재인증은 이전에 선행된 full 인증에서 유도된 키에 기반한다. 이전 full 인증에서 생성된 TEKs 즉, K_aut와 K_encr 키가 EAP-AKA 패킷과 속성들을 보호하기 위해 사용되고, 이전의 MK가 새로운 MSK를 생성하기 위해 사용된다.

UE는 unsigned 16비트 counter로 재생에 대비한다. Counter 속성은 full 인증에서 양쪽 모두에서 1로 초기화되며, 첫 번째 재인증에서 최소 1이상의 counter가 사용된다. 그 후의 재인증에서는 이전 재인증의 counter 보다 큰 counter 값을 사용해야 한다. counter 속성은 암호화 된다.

서버는 Challenge/Response 인증 기법을 사용하기 위해 NONCE_S를 이용한다. 즉 서버는 암호화된 NONCE_S를 UE에게 송신하고, UE는 NONCE_S를 복호화한 후에, 이것에 대한 MAC 값을 송신한다. 이 NONCE_S가 새로운 MSK를 생성하기 위해 사용되기 때문에(2-1 참조), NONCE_S를 cleartext로 송신하지 않고, 단지 MAC 값만을 계산하여 Response에 포함시켜 송신한다.

서버는 다음 번 인증에서 재인증을 지원하기 위해 UE에게 새로운 재인증 신원을 암호화하여 송신할 수 있다. 재인증 신원은 1회용이기 때문에, 만약 UE가 새로운 재인증 신원을 받지 못했다면, 다음번 인증에서는 이전에 수신했던 재인증 신원을 재사용하거나, 또는 full 인증을 해야만 한다.

따라서 재인증을 사용하기 위해서는 UE와 서버는 다음과 같은 값들을 각각 저장해야만 한다.

- full 인증을 통하여 얻은 MK, K_aut, K_encr
- 마지막 counter 값
- 다음번에 사용할 재-인증 신원

4. 안전성 및 효율성

1) EAP-AKA에서 생성된 키는 공격에 안전하다.

EAP AKA 패킷과 MSK을 보호하기 위해 사용되는 TEKs(K_aut, K_encr)와 MSK는 암호학적으로 연관이 없으므로 MSK를 공격하는 공격자는 K_encr 또는 K_aut로부터 어떠한 정보를 가지고도 이 키를 유추해 낼 수 없다. 또한 공격자는 UMTS AKA IK, UMTS AKA CK, EAP AKA K_encr, EAP AKA K_aut , MSK로부터 미리 공유된 비밀키를 계산할 수 없다.

또한 EAP-AKA는 128비트 효력을 갖는 키유도를 지원한다. 따라서 계산적으로 실행 가능한 brute-force 공격은 존재하지 않는다. 또한 UMTS AKA가 패스워드 프로토콜이 아니기 때문에 dictionary 공격에도 취약하지 않다.

2) 3GPP-WLAN 연동시 효율적인 운영을 할 수 있다.

EAP-AKA는 3GPP UMTS AKA를 기반으로 하기 때문에 WLAN-3GPP 연동시 동일한 메커니즘을 사용할 수 있다. 또한 USIM과 3GPP AAA 서버는 인증 횟수에 따라서 SQN을 동일하게 증가시키므로 3GPP AAA 서버는 USIM이 WLAN 또는 셀룰러 망 중 어느 것으로 접속해도 동일한 메커니즘으로 공통 접속 제어가 가능하다. 이는 [5]에서 제안된 시나리오 2이상의 연동 시나리오를 가능하게 할 수 있다.

더구나 WLAN 환경에서는 3GPP보다 더 빈번한 인증 과정이 발생할 것이다. 따라서 재인증이 옵션으로 제공되고, 재인증을 사용할 경우에는 full 인증을 하는 경우보다 빠르며, 1회용 재인증 신원을 사용하므로 더 안전하게 운영될 수 있다.

3) USIM을 기반으로 한 EAP-AKA는 안전한 운영을 실행할 수 있다.

[8]에서 제시된 USIM의 특징은 WLAN 망을 통해서 접속하는 경우에도 그대로 유지된다. 즉, EAP-AKA를 위한 USIM의 변경사항은 존재하지 않는다. USIM은 셀룰러 망을 통해 인증했을 경우와 마찬가지 방법으로 UE에게 CK와 IK를 전달하며, MK는 UE에서 생성되고, UE는 이것을 통해 인증을 완료할 수 있다. 즉, USIM이 UE와 여전히 분리되어 있으므로 USIM안에 존재하는 사용자의 중요한 정보가 유출될 수 없다는 장점을 가지고 있다. 따라서 3GPP-WLAN 연동은 UMTS 보안 레벨과 동일한 보안 레벨 안에서 운영될 수 있다. 그러므로 가입자는 WLAN 접근에 적어도 현재 셀룰러 접근과 동일한 보안 레벨을 갖게 된다.

Ⅲ. 결론

본 논문에서는 3GPP-WLAN 연동시, 인증과 키 일치를 위한 보안메커니즘으로 USIM을 기반으로 한 EAP-AKA에 대해 고찰하고 연동시 효율성에 대해 논하였다.

하지만 3GPP USIM의 안전성에도 불구하고 WLAN의 모든 개체들 사이에 보안상 취약점이 존재할 수 있다는 것을 배제할 수 없다. 따라서 개체들과 그 사이의 안전한 채널 형성에 관한 연구가 좀 더 이루어져야 한다.

참고문헌

[1] Federal Information Processing Standards

(FIPS) Publication 186-2 (with change notice), "Digital Signature Standard (DSS)",
[2] Internet Draft draft-arkko-pppext-eap-aka, "draft-arkko-pppext-eap-aka-09.txt",
[3] RFC2284 "PPP Extensible Authentication Protocol(EAP)".
[4] IEEE Std 802.11i D6.0 "Medium Access Control(MAC) Security Enhancement".
[5] 3GPP TR 22.934 v6.1.0, "Feasibility study on 3GPP system to Wireless Local Area Network(WLAN) interworking (Release 6)".
[6] 3GPP TS 23.934 "3GPP systems to Wireless Local Area Network(WLAN) Interworking; Functional and Architectural definition".
[7] 3GPP TS 33.105 v4.1.0, "Cryptographic Algorithm Requirements (Release 4)".
[8] 3GPP TS 33.102 v6.0.0 "3G Security; Security Architecture"
[9] 3GPP TS 33.234 v 0.6.0 "Wireless Local Area Network(WLAN) Interworking Security(Release 6)".