UMTS를 위한 티켓 기반의 인증과 키 동의 프로토콜

오가경*, 이승현*, 최기현*, 신동렬*
*성균관대학교 전자전기컴퓨터공학과
e-mail:kakyung@skku.edu

Ticket based authentication and key agreement protocol for UMTS

Ka-Kyung Oh*, Seung-Hyun Lee*, Kee-Hyun Choi*,
Dong-Ryeol Shin*
*School of Information and Communication Engineering,
Sungkyunkwan University

요 약

3 Generation Partnership Project(3GPP)에서는 3세대 이동통신 기술 중의 하나인 Universal Mobile Telecommunications System(UMTS)의 무선 구간의 안전한 통신을 위해 인증 및 키 교환 프로토콜인 Authentication and Key Agreement(UMTS AKA) 프로토콜을 제안하였다. 하지만, UMTS AKA는 네트워크 대역폭 소모, 저장 공간의 오버헤드, SQN의 동기화 문제 등이 제기되고 있다. 본 논문에서는 이런 UMTS AKA 프로토콜의 문제점들을 해결하는 티켓 기반의 T-AKA 프로토콜을 제안한다. 제안하는 프로토콜은 프라이버시를 보호하고 상호 인증이 가능하며 전방향 안전성을 제공한다.

1. 서론

3GPP에 의해 표준화된 UMTS는 3세대 이동통신으로서 유럽을 비롯한 많은 나라에서 사용되고 있다[1]. 이러한 무선 이동통신 서비스를 안전하게 제공하려면 보안에 취약한 무선 통신 구간의 보안 기술이 매우 중요하다[2].

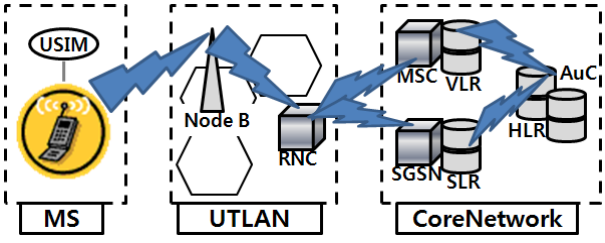
따라서, 3GPP에서는 무선 구간에서의 안전한 통신을 위한 인증과 키 동의 프로토콜인 UMTS AKA 프로토콜을 개발하였다.[3] UMTS AKA 프로토콜은 사용자와 네트워크 사이의 인증을 제공하고, 통신을 하기 위한 암호화 키와 무결성 키를 확립시켜준다. 하지만, UMTS AKA는 네트워크의 대역폭 소모와 저장 공간의 오버헤드 문제, *SQN* 동기화 문제들 이외에도, 특히 비밀키 *K*의 노출 시 발생하는 안전성에 대한 문제가 제기되었다. 이러한 문제점들을 해결하기 위하여 UMTS X-AKA 등[4][5]이 제안되었지만, UMTS X-AKA 역시 비밀키 *K*가 노출됐을 시에 대한 문제점을 해결하지 못했다.

본 논문에서는 UMTS AKA와 UMTS X-AKA의 문제점들을 해결하고 두 프로토콜들의 장점만을 수용하며 프라이버시와 전방향 안전성을 제공하는 T-AKA 프로토콜을 제안한다.

2. 관련 연구

본 장에서는 UMTS 네트워크 구조에 대해 살펴보고 UMTS AKA[3] 프로토콜과 UMTS X-AKA[4] 프로토콜에 대해 살펴본다.

[그림 1]은 UMTS 네트워크 구조이다.



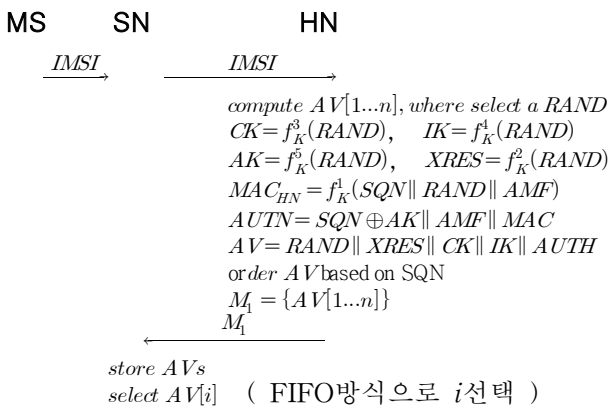
[그림 1] UMTS 네트워크 구조

UMTS 네트워크의 모든 프로토콜에서는 Mobile

Station (MS)의 Universal Subscriber Identity Module(USIM)과 Home Network(HN)의 Authentication Center(AuC)가 비밀키 K 값과 MS의 International Mobile Station Identity(IMSI)값을 공유하고 있고, Service Network(SN)과 HN의 통신구간은 IP Security(IPsec)과 같은 네트워크 도메인 보안을 통해 서로 안전한 통신을 수행한다고 가정한다.

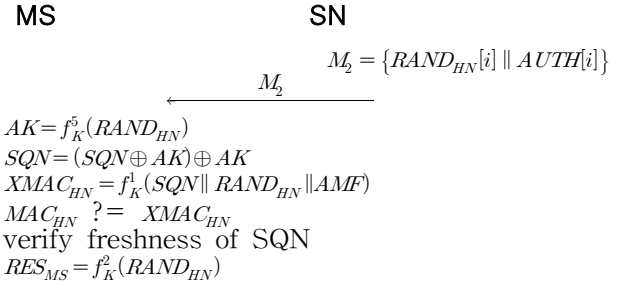
2.1. UMTS AKA

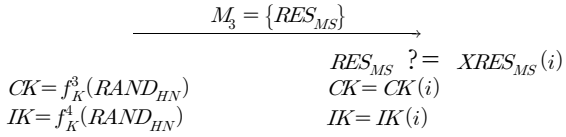
UMTS AKA 프로토콜은 MS가 SN에 서비스를 요청했을 때 신뢰할 수 있는 제 3자인 HN을 통해 인증을 수행하고, MS와 SN 사이에 안전한 통신을 위한 암호화 키와 무결성 키를 확립시켜주는 키 동의 프로토콜이다[3]. [그림 2]는 UMTS AKA 프로토콜의 인증 요청단계를, [그림 3]은 인증 및 키 동의 단계를 보여준다.



[그림 2] UMTS AKA프로토콜 - 인증요청

MS에게 서비스를 요청받은 SN은 HN에게 MS의 인증을 요청하게 되고 HN은 n 개의 *Authentication Vector(AV)*를 생성하여 SN에게 전송한다. SN은 *AVs*를 저장하고 *i*번째 *AV*를 선택하여 M_2를 MS에게 전송한다. MS는 동기화를 위해 *SQN*의 유효성을 검증하고, *SQN*이 유효하다면 M_3를 SN에게 전송하고 SN은 MS를 인증 한 후 *CK*와 *IK*를 확립하여 안전한 통신을 하게 된다.





[그림 3] UMTS AKA프로토콜 - 인증 및 키 동의 단계

UMTS AKA는 *AV*들을 사용하기 때문에 SN과 HN간의 대역폭 소모를 발생시키고, SN의 저장 공간의 오버헤드를 발생시키는 문제점이 있다. 그리고 MS는 HN을 인증하지만 HN은 MS를 인증하지 못하는 단방향 인증만을 제공하며 *SQN*의 동기화 문제점이 발생할 수도 있다. 또한, *IMSI*값이 적어도 한번은 평문 형태로 전송하기 때문에 프라이버시에 대한 문제점도 있으며 비밀키 *K*가 노출되면 이전의 통신 내용이 모두 노출될 수 있다는 문제점이 있다.

2.2. UMTS X-AKA

UMTS X-AKA 프로토콜은 임시 키인 티켓키를 발급함으로써 대역폭 소비문제와 저장 공간 오버헤드 문제를 줄였다[4]. 또한 HN은 MAC_{MS}를, MS는 MAC_{HN}과 MAC_{SN}을 검증함으로써 MS와 HN, MS와 SN간의 상호 인증을 제공한다.

MS가 SN에게 서비스를 요청하기 위해 MAC_{MS}을 계산하고 M_1을 전송하면, SN은 받은 M_1을 HN에게 전송한다. MAC_{MS}를 검증한 HN은 *TK*와 MAC_{HN}를 생성한 다음 $AUTH_{HN}$을 만들고 SN에게 $M_2 = \{AUTH_{HN}\}$를 전송한다.

$$MAC_{MS} = f^1_K(T_{MS})$$
$$M_1 = \{IMSI, T_{MS}, MAC_{MS}\}$$
$$MAC_{HN} = f^1_K(RAND_{HN} \| AMF)$$
$$TK = f^e_K(T_{MS})$$
$$AUTH_{HN} = MAC_{HN} \| RAND_{HN} \| AMF$$

SN은 N_{SN}를 생성하고 MAC_{SN}을 계산한 다음 $AUTH_{SN}$을 만들고 MS에게 $M_3 = \{AUTH_{SN}\}$를 전송한다.

$$MAC_{SN} = f^1_{TK}(MAC_{HN} \| RAND_{SN} + j \times RAND_{HN})$$
$$AUTH_{SN} = MAC_{SN} \| RAND_{SN} \| RAND_{HN} \| AMF \| j$$

MS는 MAC_{HN}과 MAC_{SN}을 검증하고 *j*번째가 맞는지 확인한 후 *RES*를 계산하고 M_4를 SN에게 전송한다. SN은 MS를 인증하고 마지막으로 MS와 SN은 안전한 통신을 위한 *CK*와 *IK*를 생성한다.

$$RES = f^2_{TK}(RAND_{SN})$$
$$M_4 = \{RES\}$$
$$CK = f^3_{TK}(RAND_{SN})$$

$IK = f_{TK}^{4}(RAND_{SN})$

하지만, 여전히 비밀키 K의 노출 시 통신의 안전성에 대한 문제점이 남아있으며 프라이버시 문제도 해결되지 못했다. 특히 AV들 사용의 문제점을 해결하기 위해 사용한 티켓 키인 TK의 노출 시에도 이전의 통신 내용이 모두 노출될 수 있다는 문제점이 있다.

3. 티켓 기반의 T-AKA 프로토콜

본 장에서는 기존의 UMTS AKA 관련 프로토콜들에 대한 문제점들을 해결하고 두 프로토콜들의 장점만을 취하며 프라이버시를 보호하고 전방향 안전성을 제공하는 프로토콜을 제안한다.

[표 1]은 본 논문에서 사용하는 프로토콜의 표기법이다. 제안하는 프로토콜을 위해 다음을 가정한다.

- MS와 HN은 비밀키 K와 프라이버시 보호를 위해 사용하는 임시 ID인 TID_{MS}와 암호알고리즘을 공유하고 있다.
- SN과 HN은 IPsec이나 MACsec과 같은 네트워크 도메인 보안 메커니즘을 통해 안전한 통신을 한다.
- MS는 현재 자신이 속해 있는 SN의 ID인 ID_{SN}을 알 수 있다.

[표 1] 프로토콜 표기법

표기	의미
T_R	R에서 생성한 타임스탬프
ID_R	R의 식별자
f_K^1	*MAC*(*Message Authentication Code*)값과 그에 대응하는 검증 값인 *XMAC* 값을 계산하는 메시지 인증 함수
f_K^2	사용자 인증(*RES*/*XRES*)을 위한 메시지 인증 함수
f_K^3	암호화 키(*CK*)를 생성하는 함수
f_K^4	무결성 키(*IK*)를 생성하는 함수
f_K^5	익명성 키(*AK*)를 생성하는 함수
$cTID_{MS}$ or $pTID_{MS}$	프라이버시 보호를 위해 *IMSI* 대신 사용되는 임시 ID. $cTID_{MS} = f_k^2(pTID_{MS})$:현재 사용되는 임시ID $pTID_{MS}$:이전에 사용된 임시ID
g	$g < p$이고, p와 서로소인 원시근
p	매우 큰 소수
$Ticket$	HN에서 발급하는 티켓
X	MS를 인증하기 위한 비밀 값
SK	Diffie-Hellman 기법으로 생성한 세션키

3.1. 제안하는 프로토콜

제안하는 프로토콜은 티켓을 발급함으로써 AV들을 사용의 문제점을 해결하고 *Diffie-Hellman* 기법을 사용하여 세션 키를 생성하고 도전-응답(*challenge-response*)을 기반으로 함으로써 전방향 안전성을 제공한다. 또한, *MAC*값을 사용하여 MS와 HN, MS와 SN 간의 양방향 인증이 가능하도록 하며 임시 ID를 사용하여 프라이버시를 강화하였다.

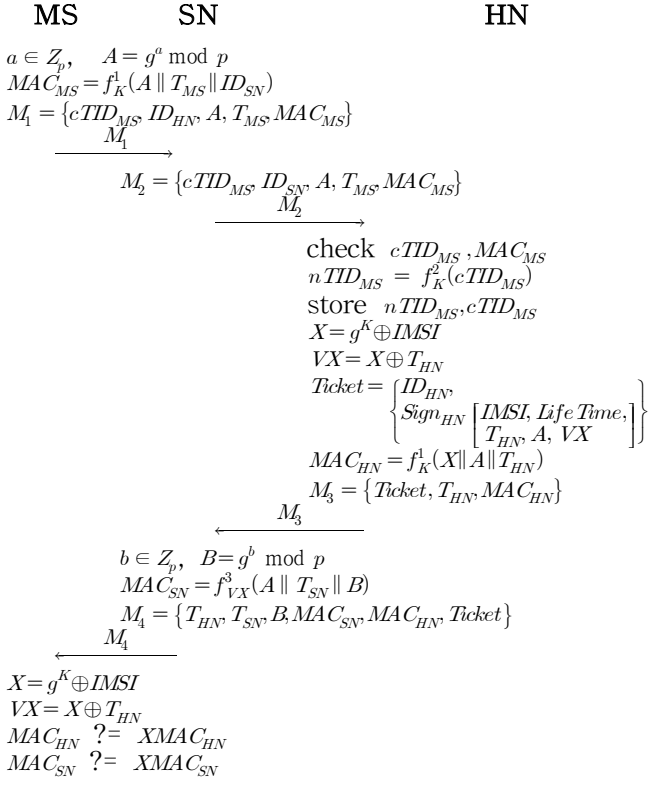
[그림 4]는 T-AKA 프로토콜의 수행과정을 보여준다.

① MS는 랜덤 값 $a \in Z_q$를 선택하고, A와 MAC_{MS}를 계산하여 임시 ID인 $cTID_{MS}$를 포함한 M_1을 SN에게 전송한다. 이때, $cTID_{MS}$는 HN과 미리 공유되어 있고, 세션마다 새롭게 계산된다. 이렇게 세션마다 새로운 임시 ID를 사용함으로써 프라이버시를 강화할 수 있다.

② SN은 ID_{HN}을 확인하여 해당 HN에게 ID_{SN}을 포함한 M_2를 전송함으로써 HN은 올바른 요청이 맞는지를 확인할 수 있다.

③ HN은 T_{MS}의 유효성을 확인하고 나서 데이터베이스에서 수신된 $cTID_{MS}$를 검색하여 MS의 정보를 확인하고 MAC_{MS}값을 통해 MS를 인증하고 ID_{SN}이 올바른지를 확인할 수 있다. 데이터베이스에는 가입자의 $IMSI, K, cTID_{MS}, pTID_{MS}$가 저장되어 있으므로 만약 TID_{MS} 동기화 문제가 발생할 경우, $pTID_{MS}$를 검색해 봄으로써 문제를 해결할 수 있다.

HN은 새로운 $nTID_{MS}$를 생성하여 데이터베이스의 현재 임시 ID 필드에 저장하고 $cTID_{MS}$는 이전 임시 ID 필드에 저장한다. 그 후, 티켓을 생성하고 MAC_{HN}을 계산하여 SN에게 M_3를 전송한다.



$nTID_{MS} = f_K^2(cTID_{MS})$
store $nTID_{MS}, Ticket, VX$
$SK = f_{VX}^5(B^a)$
$MAC'_{MS} = f_{SK}^1(T_{SN} \| B^a)$
$M_5 = \{MAC'_{MS}\}$

M_5 →

$SK = f_{VX}^5(A^b)$
MAC'_{MS} ?= $XMAC'_{MS}$
$CK = f_{SK}^3(T'_{SN})$
$IK = f_{SK}^4(T'_{SN})$
$M_6 = \{(TSMI \| T'_{SN})_{sk}\}$

← M_6

$CK = f_{SK}^3(T'_{SN})$
$IK = f_{SK}^4(T'_{SN})$

[그림 4] 제안하는 프로토콜

④ SN은 랜덤 값 $b \in Z_p$를 선택하고 B를 계산하고 티켓을 확인하여 MAC_{SN}를 계산하여 M_4를 MS에게 전송한다.

⑤ MS는 VX를 생성하고 MAC_{HN}과 MAC_{SN}를 검증하고 나서 다음에 사용할 $nTID_{MS}$를 계산하고 티켓과 VX와 함께 저장한다. MS는 SK를 계산하고 M_5를 SN에게 전송한다.

⑥ SN은 SK를 계산하고 나서 MS를 인증하고 CK와 IK를 생성하고 실제 통신에 사용될 ID인 $TMSI$를 생성해 SK로 암호화시킨 M_6를 MS로 전송한다.

⑦ MS는 M_6를 복호화하고 CK와 IK를 계산한다. 이러한 과정을 통해 MS와 SN은 안전한 통신을 하기 위한 키들을 확립할 수 있다.

4. 안전성 및 효율성 분석

4.1. 안전성 분석

상호인증 : MS와 HN, MS와 SN의 상호 인증이 가능하다. MS와 HN은 서로 공유하는 비밀키 K로 생성한 MAC_{MS}와 MAC_{HN}를 검증함으로써 서로 인증할 수 있다. 또한, MS와 SN은 MS의 정보로 계산한 VX로 생성된 MAC_{SN}를 통해 서로 인증할 수 있다.

재전송 공격 : MAC값에 타임스탬프를 포함함으로써 메시지의 최신성(freshness)을 검증할 수 있다. 만약 허용 시간 안에 메시지를 재전송 한다 하더라도 랜덤 값 a와 b의 쌍을 알지 못하므로 재전송 공격에 강하다.

TID_{MS} 동기화 공격 : 만약 M_1을 차단한다고 가정한다면, 일정 기간에 M_4를 받지 못한 MS는 세션을 종료하고 새로운 세션을 시작하므로 문제가 없다. 만약 공격자가 M_3를 차단했을 때, HN은 M_2를 받았으므로 MS의 새로운 $nTID_{MS}$를 갱신하게 되는데, M_4를 받지 못한 MS는 이전의 $cTID_{MS}$를 유지하게 된다. 하지만, HN은 $cTID_{MS}$와 $nTID_{MS}$를 함께 저장하고 있으므로 MS가 이전의 $cTID_{MS}$를 사용하더라도 동기화가 가능하다.

전방향 안전성 : 2장에서 소개한 프로토콜들은 장기간 비밀키 K 또는 단기간 비밀키 TK가 노출되면 전체 통신에 영향을 끼친다. 제안한 프로토콜은 *Diffie-Hellman*기법을 사용하여 SK를 생성함으로써 전방향 안전성을 제공한다.

4.2. 효율성 분석

[표 2]는 UMTS AKA와 UMTS X-AKA와의 비교분석을 통해 제안하는 프로토콜의 효율성을 보여준다.

제안하는 프로토콜은 티켓기반의 인증을 수행함으로써 UMTS AKA처럼 n개의 인증 데이터를 생성하지 않으므로 SN과 HN의 네트워크 대역폭 소모를 감소시키고, SN의 저장 공간을 낭비하지 않는다. MS와 HN, MS와 SN은 각각 생성한 MAC값을 통해 상호 인증이 가능하며 *Diffie-Hellman* 기법을 사용한 세션 키를 생성함으로써 장기간 비밀키나 단기간 비밀키가 노출되더라도 통신에 영향을 미치지 않는다. 또한, 세션키를 생성하는데 계산상의 부담이 적다.

[표 2] 프로토콜 비교

프로토콜 / 문제점 & 속성	UMTS AKA	UMTS X-AKA	제안하는 프로토콜
AVs 사용 여부	○	Ticket key	Ticket
MS와 SN의 동기화	*SQN*	Time stamp	Time stamp
SN과 HN간의 네트워크 대역폭 소모	○	×	×
SN의 저장공간 오버헤드 여부	○	×	×
MS와 HN의 상호 인증	×*	○	○
MS와 SN의 상호 인증	○	○	○
장기간 비밀키 노출의 영향	○	○	×
단기간 비밀키 노출의 영향	×	○	×

* MS만 HN을 인증 할 수 있음

5. 결론

본 논문에서는 전방향 안전성을 제공하는 티켓기반의 T-AKA프로토콜을 제안하였다. T-AKA 프로토콜은 티켓 기반의 인증을 수행함으로써 네트워크 대역폭 소모와 저장 공간의 낭비 문제를 해결하였으며, 임시 ID를 사용함으로써 프라이버시를 강화시켰다. 또한, 상호인증이 가능하며 *Diffie-Hellman* 기법을 사용한 키 동의 프로토콜로서 전방향 안전성을 제공하였다.

감사의 글

본 연구는 국토해양부 첨단도시개발사업의 연구비지원(07첨단도시 A01)에 의해 수행되었습니다.

참고문헌

[1] 3GPP TS 23. 101 v7.0.0 (2007-06)

[2] G. Horn, K. M. Martin and C. J. Mitchell, "Authentication protocols for mobil network environment value-added services", IEEE Trans. on Vchi. Tech., 51, pp. 383-392, 2002.

[3] 3GPP TS 33. 102 (v7.0.0), Security architecture, Release 7, 2005.

[4] C. Huang and J. Li, "Authentication and Key Agreement Protocol for UMTS whit Low Bandwidth", Proc. of the 19th IEEE Conf. on AINA, pp. 392-397, 2005.

[5] M. Zhang and Y. Fang, "Security Analysis and Enhancement of 3GPP Authentication and Key Agreement Protocol," IEEE Trans. on Wireless Communications, Vol. 4, No. 2, pp. 734-742, 2005.