3GPP MTC 환경에서 그룹에 효율적인 보안 및 인증 기법

홍성대, 강성용, 최대성, 김승룡, 최형기
성균관대학교
e-mail:{sdhong, sykang, dschoi, srkim, hkchoi}@hit.skku.edu

An efficient security protocol for group in 3GPP MTC

Sung-Dae Hong, Seong-Yong Kang, Dae-Sung Choi, Seung-Ryong Kim,
Hyoung-Kee Choi
Sungkyunkwan University

요 약

Machine to Machine (M2M) 통신은 수 많은 장비들이 우리 주변환경의 정보를 감지하여 상호간에 주고받는 방법을 활용한다. Third generation partnership project (3GPP)는 이러한 M2M 통신을 활용하여 유비쿼터스 환경과 자가인식형 서비스를 제공하려 하고 있다. 본 논문은 이러한 추세에 맞추어 새로운 개념의 통신인 3GPP M2M 통신의 시스템 구조, 특징들과 이에 대한 3GPP의 연구방향 등에 대한 튜토리얼로 사용될 수 있도록 하기 위해 작성되었다. 특히 M2M 통신에서의 특징들을 구체화하는 과정에서의 보안상 위협과 이에 대처하기 위한 요구사항들에 대해서 강조하였다. 이러한 보안상 위협을 제거하는 요구사항을 만족하면서 네트워크에 과부하를 가하지 않도록 하기 위한 그룹 기반의 최적화된 인증 프로토콜을 제안한다. 본 논문에서는 그룹을 기반으로 bilinear pairing을 사용하여 인증을 진행하며, 통신 딜레이를 줄이고 네트워크의 중심 장비에 과부하를 줄이기 위해 HSS(Home Subscriber Server)의 참여를 제한한 인증 및 키 동의 프로토콜을 제안하고, 이에 따른 연산 및 통신 오버헤드를 분석한다.

1. 서론

사람의 제어나 관여 없이 디바이스 스스로 필요한 정보를 수집하고 서로 간의 통신을 통해 정보 공유가 가능한 통신을 M2M(Machine to Machine)이라 한다. 즉, 디바이스를 이용하여 네트워크를 구성하고 수집한 정보를 공유하는 개념 및 기술을 M2M 통신이라고 한다[1].

M2M 통신은 H2H(Human to Human)통신과 비교하여 이종간(Heterogeneous) 많은 수의 디바이스, 낮은 주기의 트래픽 패턴, 저속의 이동성 지원, 다양한 서비스 지원, 저전력 디바이스를 위한 향상된 전력 관리 기법 [1]등의 특성을 가지고 있기 때문에, M2M 통신을 3GPP에 적용하는 방안은 아직 초기 단계에 머물러 있다[2][3][4][5][6].

또한 많은 수의 디바이스가 가지는 특징을 효과적으로 처리할 수 있는 그룹 기반 최적화와 보안 관련 부분에 대한 표준화 진행사항 역시 아직 초기 단계이다. [7]에서는 LTE-A(Long Term Evolution-Advanced)에서 데이터율(data rate)은 빨라졌고 코어 시스템 구조도 대폭 변경하였지만, 통신지연(communication delay)은 기존에 비해 크게 향상되지 않아 아직도 문제점으로 남아 있음을 지적하였다. 위에서 지적한 문제점들로 인하여 HSS(Home Subscriber Server)는 많은 인증 요청을 처리함에 있어서 과부하에 걸리게 되고, 그로 인한 처리지연(processing delay) 역시 클 것이다. 따라서 많은 디바이스를 그룹 단위로 인증할 수 있는 그룹 기반 AKA(Authentication and Key Agreement)가 필요하고, 통신 지연을 줄이기 위한 기법이 필요하다.

본 논문은 디바이스가 많을 경우 인증과정에서 발생하는 문제점을 해결하기 위해 단일 디바이스 별로 인증과정을 진행하지 않는 그룹 기반 AKA를 수행하고, 통신 지연을 줄이기 위해 AKA 과정에서 HSS가 참여하지 않는 방법을 제안한다.

2. 기존 M2M 통신 구조

M2M 환경에서 다수의 디바이스의 인증요청으로 인한 HSS의 오버헤드를 줄이고 전체 인증과정을 간략화 하기 위해 Chen et al.은 UMTS(Universal Mobile Telecommunications System) 환경에서의 그룹 기반 AKA를 제안하였고[8], Han et al.은 LTE-A 기반의 M2M 환경에서의 그룹 기반 AKA를 제안하였다[9]. [8][9]에서는 디바이스의 리더만 초기 AKA 과정 전체를 진행하여 그룹을 인증 받고, 나머지 멤버들은 간소화된 인증과정을 거치도록 하였다.

Han et al.의 구조는 MME(Mobility Management Entity)가 HSS로부터 그룹 맴버들의 값을 미리 저장함으로써 리더를 제외한 나머지 멤버들의 인증과정에 HSS의 참여를 없앴다. 또한 MME가 각 멤버에게 보내는 인증토큰을 하나만 생성하여 브로드캐스트 채널로 전송하도록 함으로써 MME의 통신 오버헤드를 줄였다. 하지만 Han

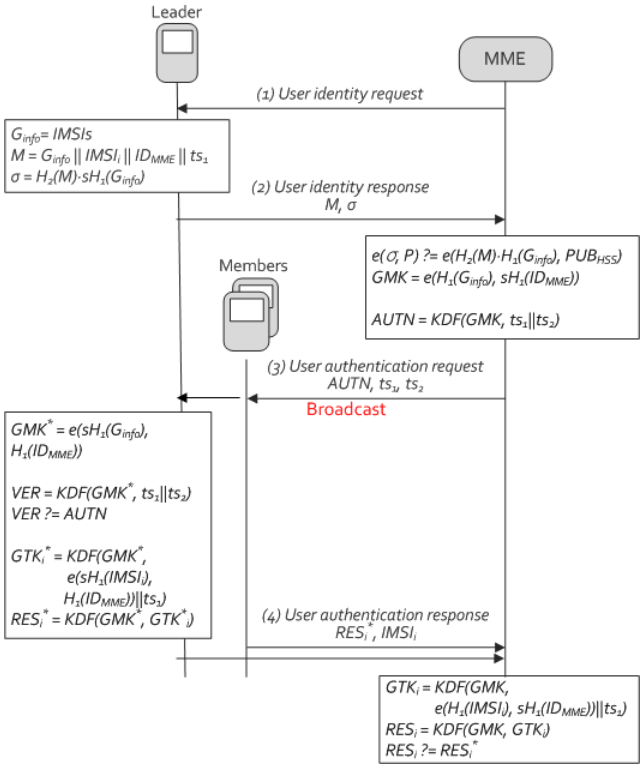
et al.의 구조는 리더의 인증을 위해 아직 HSS의 참여를 필요로 함으로 인해 MME-HSS간 통신지연만큼 기다려야 하는 단점이 있다. 또한 그룹 맴버의 인증에 HSS가 필요치 않게 하였지만, M2M 환경에서 많은 노드들이 지속적으로 인증을 요청함으로 인해 생기는 HSS의 오버헤드를 줄일 필요가 있다.

AKA 과정에서 HSS의 참여를 완전히 없애기 위한 방법으로 프록시 시그니처 구조를 사용할 수 있다. Mambo et al.은 기존 시그널을 대신하여 지정된 시그널로 서명을 할 수 있도록 하는 프록시 시그니처 기법을 제안하였다[10]. 프록시 시그니처 기법은 기존 시그널의 참여를 없앨 수 있는 강력한 기법이지만 RSA와 같은 비대칭 암호는 모듈러 지수승(exponentiation)을 사용하므로 AES와 같은 대칭 암호화를 사용하는 기존 기법들에 비해 연산비용이 크다는 단점이 있다. Boneh et al.은 비교적 연산비용이 적은 타원 곡선을 기반으로 하는 bilinear pairing을 사용하여 시그니처를 생성하는 BLS 시그니처 구조를 제안하였다[11]. He et al.은 BLS 구조를 응용하여 MN과 AP가 인증서버의 도움 없이 상호인증 및 키생성을 하는 프로토콜을 제안하였다[12].

3. 제안기법

위에서 언급한 bilinear pairing 기법을 사용하여 비교적 연산비용 증가를 최소화 하면서 HSS의 AKA 참여를 제한하여 전체적인 인증 과정의 오버헤드를 줄일 수 있는 기법을 제안한다.

제안하는 구조는 Han et al.'s 구조처럼 그룹기반 AKA



[그림 1] 제안한 기법

이다. 여기서 리더는 고정적으로 지정할 수 있지만, 그런 경우 리더가 통신 불능 상태에 있거나 악의적인 공격자에 의해 물리적인 공격을 받은 경우 그룹 전체에 영향을 미칠 수 있다. 따라서 유동적으로 리더를 선출하는 기법들이 제안되었다[13]. 본 논문에서는 그룹 멤버 중 가장 먼저 인증과정을 진행하는 디바이스를 MME가 리더로 인식하여 전체 프로토콜을 진행한다고 가정한다.

EPS-AKA(Evolved Packet Service-AKA) 과정과 같이 MME가 인증을 [그림 1]의 (1)과 같이 요청하면 리더는 메시지(M)와 시그니처(σ)를 (2)로 MME에게 전송한다. 여기서 M은 그룹 멤버들의 IMSI 값의 세트인 Ginfo, 본인의 IMSI, 인증을 요청하는 MME의 SN-ID (IDMME), 그리고 타임스탬프(ts1)으로 구성된다. σ은 (2)를 통해 보내는 M값의 H2 해쉬값과 초기에 HSS로부터 전달 받은 그룹의 개인키인 sH1(Ginfo)의 곱으로 구성된다. MME는 리더로부터 전송 받은 (2)에서 ts1을 우선 확인하여 리플레이 여부를 검사한 후, 식(1)과 같이 서명값을 확인하고, 이를 통해 그룹의 정당성을 검증한다.

$$e(\delta, P) = e(H_2(M) \cdot H_1(G_{info}), PUB_{HSS}) \quad \text{식(1)}$$
$$= e(H_2(M) \cdot H_1(G_{info}), sP) = e(H_2(M) \cdot sH_1(G_{info}), P)$$

또한 M에 들어있는 Ginfo를 통해 연산한 그룹의 공용키와 자신의 개인키의 bilinear pairing 연산 값을 GMK로 갖는다. MME는 GMK와 리더가 보낸 ts1, 그리고 현재 시각인 ts2를 사용하여 키유도 함수(KDF)로 인증토큰(ATUN) 값을 생성한다. 따라서, MME는 전체 그룹의 인증과정 중 단 한 개의 인증토큰을 생성하고, 이 값을 그룹 멤버 모두에게 공통적으로 사용한다. MME는 생성한 인증토큰과 타임스탬프들을 (3)에 담아 LTE의 물리하향채널(i.e., 브로드캐스트 채널 [BCH])을 이용하여 브로드캐스트한다. (3)을 받은 각 멤버는 ts2를 확인한 후, group의 개인키와 MME의 공용키를 사용하여 GMK*를 연산하고 KDF를 사용하여 VER을 생성한 후 AUTN과 비교하여 MME를 인증한다. 각 멤버는 자신의 IMSI(International Mobile Station Identity)를 ID로 갖는 개인키와 MME의 공용키를 bilinear pairing 연산한 값에 ts1을 연결연산자(concatenation)한 값과 GMK*를 인자로 GTKi*를 생성하고 이를 이용하여 MME에게 보낼 응답(RESi*)을 생성한다. 각 멤버는 최종적으로 생성한 RESi*와 IMSIi를 (4)에 담아서 전송한다. MME는 전송된 IMSI 값을 통해 해당 유저가 그룹의 멤버임을 확인하고, 식(2)와 같이 GTKi를 생성하여 각 멤버가 보내온 RESi*와 RESi를 비교하여 인증을 완료한다. 멤버마다 생성된 GTK는 기존 EPS-AKA에서의 KASME의 역할을 하여 NAS 및 AS 구간의 암호화와 무결성을 위해 사용된다.

$$GTK_i = KDF(GMK, e(H_1(IMSI_i), sH_1(ID_{MME}))\|ts_1) \quad \text{식(2)}$$
$$= KDF(GMK^*, e(sH_1(IMSI_i), H_1(ID_{MME}))\|ts_1) = GTK^*$$

4. 제안기법 분석

4.1 보안성 분석

• **상호인증(Mutual authentication)** : MME는 리더가 초기에 보낸 서명 값에 포함된 HSS의 마스터키를 확인함으로써 그룹의 정당성을 인증 할 수 있고, 이후 인증을 수행하는 멤버들이 같은 그룹의 멤버임을 확인 할 수 있다. 이후 MME가 보내는 인증 토큰 값(AUTN)을 생성하기 위해서는 정당한 GMK를 유도할 수 있어야 하므로, 각 멤버는 MME가 보내는 인증 토큰 값을 확인함으로써 MME를 인증할 수 있다. 또한 MME는 각 멤버가 보내는 RES 값을 확인함으로써 각 멤버를 인증할 수 있다.

• **비밀키 유도(Secure key derivation)** : 악의적인 공격자는 MME와 나눠 갖은 GMK와 GTK를 생성하려고 시도할 수 있다. 하지만 공격자는 HSS의 마스터키 s를 알 수 없으므로 각각의 개인키를 유도할 수 없고, GMK와 GTK 역시 생성할 수 없다.

• **재전송 공격 방어(Replay attack resistance)** : 공격자는 정당한 리더로 위장하여 [그림 1]의 (2)와 (4)를 재전송 공격을 할 수 있다. 하지만 MME는 (1)에 포함된 ts1을 확인함으로써 인증요청을 무시할 수 있다. 만일 공격자가 매우 짧은 시간 안에 재전송 공격을 수행 하더라도 공격자는 GMK와 GTK를 생성할 수 없으므로 공격이 불가능 하다. 공격자는 정당한 MME로 위장하여 (3)을 재전송하는 공격을 시도할 수 있다. 하지만 MME가 보내는 인증토큰에는 리더가 선택한 ts1과 MME가 (3)을 전송 할 때의 ts2가 있으므로 이를 통해 재전송 공격을 거부할 수 있으며, 마찬가지로 매우 짧은 시간 안에 공격을 수행 하더라도 키를 생성할 수 없으므로 공격이 불가능 하다.

• **내부공격(Inside attack)** : 공격자는 그룹의 멤버 중 하나의 디바이스를 탈취하여 공격을 수행할 수 있다. 공격자는 해당그룹과 MME 사이의 GMK는 얻어낼 수 있지만, GTK는 디바이스 고유의 개인키를 사용하여 생성하므로 다른 멤버의 GTK를 알 수 없다. 따라서 공격자가 하나의 디바이스를 탈취하여 내부공격을 시도하더라도 그 피해범위는 탈취한 디바이스로 제한된다.

4.2 성능 분석

제안하는 프로토콜의 성능을 기존의 EPS-AKA, Han et al.'s G-AKA와 연산과 통신 오버헤드 측면에서 비교하여 제안하는 프로토콜의 타당성을 입증하려 한다.

• **통신 오버헤드**

[표 1]은 인증과정에 참여하는 그룹에 속한 디바이스가 n일 때, 필요한 통신 횟수를 나타낸다. 간략화를 위하여 UE-MME 구간의 메시지 전송 비용과 MME-HSS 구간의 메시지 전송 비용의 기대값을 각각 α와 β로 표현하였다. 일반적으로 α에 비해 β가 충분히 크므로 제안하는 프로토콜의 통신 오버헤드가 충분히 작다.

더욱이 n의 수가 증가하면 EPS-AKA의 경우 HSS에 걸리는 통신 오버헤드가 커져서 전체 인증 대기시간이 증가할 것을 예상할 수 있는 반면 제안하는 프로토콜은 HSS가 인증 과정에 참여하지 않으므로 HSS의 부담을 크게 줄여 줄 수 있을 것으로 예상한다. Han et al.'s G-AKA의 경우 n이 증가하여도 HSS와의 통신은 한번만 수행하지만, HSS가 모든 멤버의 IMSI와 SQN을 전송해 주어야 하므로 제안하는 프로토콜이 더 우수하다고 할 수 있다.

[표 1] 통신 오버헤드 비교

	통신 오버헤드
EPS-AKA	$4n\alpha + 2n\beta$
Han *et al.*'s G-AKA	$(2n + 2)\alpha + 2\beta$
제안기법	$(n + 3)\alpha$

[표 2] 연산 오버헤드 비교

	암호학적 연산 (MME & HSS)	암호학적 연산 (UE)
EPS-AKA	6n KDF	6 KDF
Han et al.'s G-AKA	(2n+2) KDF	4 KDF
제안기법	1 ECSM + (n+3) pairing + (2n+1) KDF	2 pairing + 3 KDF (+ 1 ECSM)

• **연산 오버헤드**

[표 2]는 인증과정에서 코어 유닛(MME and HSS)과 UE에게 각각 요구되는 연산 오버헤드를 암호학적 연산의 단위로 나타내고 있다. 제안하는 프로토콜에서 그룹 멤버의 수가 n이라 할 때, MME는 1회의 타원곡선 스칼라 곱셈(ECSM), n+3회의 bilinear pairing, 그리고 2n+1회의 KDF 연산이 필요하다. 또한 UE의 경우 2회의 paring과 3회의 KDF 연산이 필요하며, 리더의 경우에는 1회의 ECSM이 추가로 필요하다.

각각의 암호학적 연산에 대해 최적화가 잘 된 유명한 암호학적 연산을 사용하여 각 프로토콜을 수행하는데 걸리는 시간을 자세히 알아보았다. ECSM 연산은 MIRACL 라이브러리를 사용하였으며, Bilinear 연산은 PBC 라이브러리를 사용하였다. 여기서 160bits를 order로 갖고 embedding degree k = 6인 MNT 커브를 사용하였다. 이 커브에서 elements in Zq*, G and GT는 160, 161, 960 bit로 표현되었다. KDF는 [14]에 따라 HMAC with SHA-256으로 간주하고 Crypto++ 라이브러리를 사용하였다.

테스트베드는 우분투 12.10 (linux kernel 3.5.0-17) 환경에서 intel core 2 Duo Processor E6300 (2M Cache, 1.86 GHz, 1066 MHz FSB)와 2GB Random Access Memory (RAM)을 사용하였으며, 연산 오버헤드는 [표 3]과 같다.

테스트 환경에서 코어 유닛이 하나의 디바이스에 대한 인증과정을 수행하는데 걸리는 연산 시간은 5.488ms와 같으며, 디바이스가 증가함에 따라 각각 걸리는 시간은 [표 4]와 같다.

[표 3] 암호학적 연산 시간

HMAC with SHA-256	ECSM	Bilinear pairing
0.166 ms	0.454 ms	1.134 ms

[표 4] 각 AKA 프로토콜의 연산 시간

	연산 시간 (n = 1)	연산 시간 (n = 10)	연산 시간 (n = 100)
EPS-AKA	0.996 ms	9.96 ms	99.6 ms
Han et al.'s G-AKA	0.664 ms	3.652 ms	33.532 ms
제안기법	5.488 ms	18.682 ms	150.622 ms

전체적인 성능을 평가 하였을 때, 제안하는 프로토콜은 연산비용이 비교적 높게 측정 되지만, 코어 유닛의 경우 테스트베드 보다 월등히 강력한 성능을 가진 머신으로 구성되어 있으므로 통신지연의 절대 수치는 더욱 줄어들 것이다. 반면 제안하는 프로토콜이 코어 유닛간의 통신을 수행하지 않으므로 생기는 통신지연은 줄이기 쉽지 않으므로 전체 인증과정에 소요되는 시간은 충분히 납득할만한 수준이라고 평가할 수 있다.

4.3 핸드오버 고려사항

EPS-AKA는 기본적으로 여러 개의 AV를 HSS로부터 받아 올 수 있고, 핸드오버가 발생한 경우 매번 HSS에게 인증 및 AV를 요청하지 않고 저장되어 있는 AV를 활용하여 핸드오버 지연을 줄일 수 있도록 고안되었다. 따라서 핸드오버가 빈번한 경우 AV 메커니즘을 사용하지 않는 제안한 프로토콜의 장점이 크지 않을 수 있다고 예상할 수 있다. 하지만 LTE-Advanced에서는 강화된 키 체계(hierarchy)로 인해 KASME 및 KeNB의 재사용이 가능하므로, 표준에서도 AV를 1개만 생성해서 MME에게 보내기를 추천하고 있다. 따라서 제안하는 프로토콜은 AV 메커니즘을 사용하지 않지만 키 재사용 메커니즘을 사용하면, 핸드오버가 빈번한 시나리오의 경우에도 충분히 효과적이라고 할 수 있다.

5. 결론

M2M환경에서 많은 디바이스 특성으로 인해 기존의 EPS-AKA는 적합하지 않으며, 이로 인해 HSS의 오버헤드가 심각할 것임을 지적하였다. 이를 해결하기 위하여 프록시 구조를 사용한 그룹기반 AKA를 제안하였다. 제안한 프로토콜은 보안상 안전하고, 퍼포먼스 측면에서도 충분히 효과적이다.

참고문헌

[1] Geng Wu, Talwar, S., Johnsson, K., Himayat, N.,Johnson, K.D., "M2M: From mobile to embedded internet,"Communications Magazine, IEEE , Vol. 49, No. 4, pp. 36-43, April 2011.

[2] S.-Y. Lien, et al., "Toward Ubiquitous Massive Accesses in 3GPP Machine-to-Machine Communications," IEEE Communications Magazine, Vol. 49, No. 4, pp. 66-74, ,Apr. 2011.

[3] 3GPP TR 23.888, "System Improvements for Machine-Type Communications," Apr. 2011.

[4] R. Lu, et al., "GRS: The Green, Reliability, and Security of Emerging Machine to Machine Communications," IEEE Communications Magazine, Vol. 49, No. 4, pp. 28-35, Apr. 2011.

[5] Y. Zhang, et al., "Home M2M Networks: Architectures, Standards, and QoS Improvement," IEEE Communications Magazine, Vol. 49, No. 4, pp. 44-52, Apr. 2011.

[6] Z. Md. Fadlullah, et al., "Toward Intelligent Machine-to-Machine Communications in Smart Grid," IEEE Communications Magazine, Vol. 49, No. 4, pp. 60-65, Apr. 2011.

[7] Y. Shah, et al., "Trust in M2M communication," IEEE Vehicular Technology Magazine, Vol. 4, No. 3, pp. 69-75, Sep. 2009.

[8] Y. W. Chen, et al., "Group-Based authentication and key agreement," Wireless Personal Communications, Vol. 62, No. 4. pp. 965-979, 2012.

[9] C. K. Han, et al. "Security Analysis and Enhancements in LTE-Advanced Networks," Ph.D. Dissertation in SungKyunKwan university, 2011.

[10] M. Mambo, et al. "Proxy signatures for delegating signing operation," Proceeding of the 3rd ACM conference on Computer and Communications Security, PP. 48-57, 1996.

[11] D. Boneh, et al., "Short Signature from the Weil Pairing," Journal of Cryptography, Vol. 17, No. 4, pp. 297-319, Sep. 2004.

[12] D. He, et al., "Secure and Efficient Handover Authentication based on Bilinear Pairing Functions," IEEE Transactions on Wireless Communications, Vol. 11, No. 1, pp. 48-53, Jan. 2012.

[13] A. Richa, et al., "Self-stabilizing leader election for single-hop wireless networks depite jamming," Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing, No. 15, 2011.

[14] 3GPP TS 33.401, "3GPP System Architecture Evolution (SAE); Security Architecture", Mar. 2011.