스마트홈 환경에서 무인증서 기반 디바이스 인증 기법

이재승*, 석상기**, 전문석*
*숭실대학교 컴퓨터학과
*서울과학기술학교 컴퓨터공학과
e-mail : ljs0322@ssu.ac.kr

Certificateless-based Device Authentication Scheme in Smart Home Environment

Jaeseung Lee*, Sangkee Suk*, Moon-Soeg Jun*
*Dept of Computer Science & Engineering, Soongsil University
**Dept of Seoul National University of Science and Technology

요 약

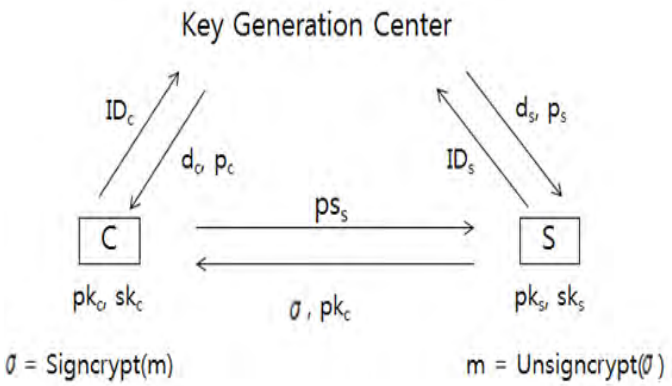
최근 무선 통신 기술과 센서 디바이스들의 발달로 인터넷을 기반으로 모든 사물을 연결하여 사람과 사물, 사물과 사물 간의 정보를 상호 소통 가능한 센서 기반 IoT 환경이 다양한 분야에 활용되고 있다. 이러한 사물인터넷 환경은 지능형 서비스를 위해 다양하고 방대한 양의 디바이스 정보를 수집하며, 사용자 정보를 기반으로 서비스를 제공받고 디바이스를 제어해야 하며, 이기종 간의 디바이스를 활용함으로 올바른 표준을 기반으로 통신이 이루어져야 한다. 하지만, 사물인터넷 환경에서의 기존 연구나 표준 정의를 살펴보면, 현재 IoT 서비스에서는 이미 취약점이 들어난 커버로스 및 센서 노드의 수를 고려하지 않은 PKI 기반 보안 기술이 활용되고 있다. 따라서, 본 논문에서는 안전한 디바이스 인증을 위한 무인증서 기반 상호 인증 기법을 제안한다.

1. 서론

사물인터넷(Internet of Things)은 인터넷을 기반으로 모든 사물을 연결하여 사람과 사물, 사물과 사물 간의 정보를 상호 소통하는 지능형 기술 및 서비스를 말한다. IoT환경에서는 지능형 서비스를 위해 다양(variety)하고 방대한 양(volume)의 디바이스 정보를 수집하며, 사용자를 정보를 기반으로 서비스를 제공받고 디바이스를 제어한다. 이러한 IoT 서비스의 핵심 포인트는 언제 어디서나 지속적으로 서비스가 제공되어야 한다는 점이다. 또한, 다양한 디바이스를 통해 정보가 전송되고 제어되기 때문에 올바른 표준을 기반으로 통신이 이루어져야 한다. 또한, 이 때 정보를 전송하는 디바이스 별 보안방안을 제공하여 디바이스 하드웨어 제한에 따른 각기 다른 보안이 제공되어야 한다.

하지만, 현재까지의 연구 결과는 IoT 표준에서 정의하고 있는 프레임워크와 통신 절차에 적합하지 않다. 또한, 지속적인 서비스를 위해서는 경량화되고 안전한 통신 규약이 필요하지만, 기존 연구들은 공개키 통신 등 에너지 효율이 떨어지는 보안 통신을 하고 있다. 마지막으로, 현재 IoT 서비스에서는 이미 취약점이 들어난 커버로스를 보안기술로서 활용하고 있으며, IoT 환경에 적합하지 않은 PKI를 활용하고 있다. 따라서, 본 논문에서는 안전한 디바이스 인증을 위한 무인증서 기반의 상호 인증 기법을 제안한다.

2. 무인증서 인증 기법



(그림 1) 무인증서 인증 기법

무인증서 인증 기법은 Barbosa와 Farshim이 처음으로 제안하였으며, Xie 등은 쌍선형지도를 통한 서명과 복호화 과정의 두 단계에서 2개의 페어링연산만을 사용한 무인증서 서명 기법을 제안했으나, 계산 비용이 높은 페어링 연산에 의한 한계를 가지고 있다. 이에 Xie와 Zhang은 페어링을 사용하지 않는 효율적인 무인증서 암호화 기법을 제

안하였다.

2.1 무인증서 서명 암호화 기법

▪ **Setup** : 임의의 k로부터 시스템 파라미터 params와 마스터 비밀키 Master-Key를 생성한다.

▪ **Partial-Key-Extract** : 알고리즘에서 얻어진 params, Master-Key, 및 사용자 식별 값 ID로부터 부분 개인키 D_{ID}와 부분 공개키 P_{ID}를 생성한다.

▪ **Set-Secret-Value** : parmas와 사용자 식별 값 ID로부터 비밀 값 S_{ID}를 생성한다.

▪ **Set-Public-Key** : params, 사용자의 ID_A와 비밀 값 SID로부터 사용자의 공개키 PK_{ID}를 생성한다.

▪ **Set-Private-Key** : params, 사용자의 부분 개인키 D_{ID}와 비밀 값 S_{ID}로부터 사용자의 개인키 SK_{ID}를 생성한다.

▪ **Encrypt** : params, 송신자의 개인키 sk_ID_s, 메시지 m으로부터 암호문 를 생성, 즉 =signcrypt(params, SK_{ID}, ID_R, PK_{ID},, m)이다.

▪ **Decrypt** : params, 송신자의 식별 값 IDS와 공개키 PK_{ID},수신자의 개인키 SK_{ID}와 암호문 로 부터 메시지 m을 복원, 암호문 가 올바르다고 증명되면 메시지 을 복원하고, 그렇지 않으면 복호화에 실패, 즉, p=unsigncrypt(params, IDs,PK_{ID},SK_{ID}, 이고, P가 정상적으로 복호화 되면 메시지 m을, 그렇지 않으면 에러를 출력한다.

Setup과 Partial-Key-Extract 알고리즘은 KGC에 의해 수행된다. 부분 개인키 DID와 부분 공개키 P_{ID}는 사용자에게 비밀채널을 통해 전달되고, 사용자의 공개키 및 개인키 쌍을 생성하는 알고리즘과 Set-Secret-Value 알고리즘은 사용자에 의해 수행된다.

3. 디바이스 초기인증 및 검증

▪ **초기 단계** : 보안 파라미터 k를 입력 받은 후 p,q를 생성한다. 이때, p와 q는 q>2k 와 q|(p-1)를 만족 해야 한다. 이후, g를 선택한 후, Master Key x를 랜덤 하게 선택하여 $y = g^x$를 계산한다. 다음으로 해시 함수 H_1, H_2, H_3, H_4, H_5 를 계산하고 시스템 파라미터 params = <p,q,n,g,y, H_1, H_2, H_3, H_4, H_5)를 공개 한다.

▪ **부분키 생성** : 파라미터와 마스터키 x, 사용자 식별 값 ID∈{0,1}*를 획득한 후 r_{ID}, $r`_{ID}$를 선택한다. 이후, $w_{ID}$ = $gr_{ID}$, $d_{ID}$ = $r_{ID}$ + $xH_1$(ID,$w_{ID}$), vID = $gr`_{ID}$, σ_{ID} = $r`_{ID}$ + xH_2(ID, w_{ID}, v_{ID})를 계산하여 부분 비밀 키 D_{ID}와 P_{ID} = (w_{ID}, v_{ID}, σ_{ID})를 반환한다.

▪ **비밀 값 생성** : 사용자는 params 값과 ID를 입력 받고, z ∈R Z*q를 선택한 후, s_{ID}=z를 생성 한다.

▪ **개인 키 생성** : 사용자의 부분 공개 키 p_{ID}와 비밀 값 s_{ID}를 이용하여, 사용자 개인키 sk_{ID}=(d_{ID},s_{ID})를 생성 한다.

▪ **공개 키 생성** : params, 사용자의 부분 공개 키 P_{ID}와 비밀 값 S_{ID}를 입력 받아 μ_{ID} = gs_{ID}를 계산하고 공개키 Pk_{ID}=(μ_{ID}, w_{ID},v_{ID},σ_{ID})를 생성 한다.

▪ **서명** : 수신자의 식별 값과 공개 키를 가지는 서비스 제공자에게 메시지를 보내기 위해 다음 인증 과정을 수행한다.

먼저, $g^{\sigma R} = v^B y^{H_2(B, Wb, vB)}$를 검증한다. 이후, 난수 r을 생성하여

$h = H_3(m,t,\mu^r,B,w^r,B,y^{H_1(B,wB)}r,\mu_A,w_A,\mu_B,w_B)$,

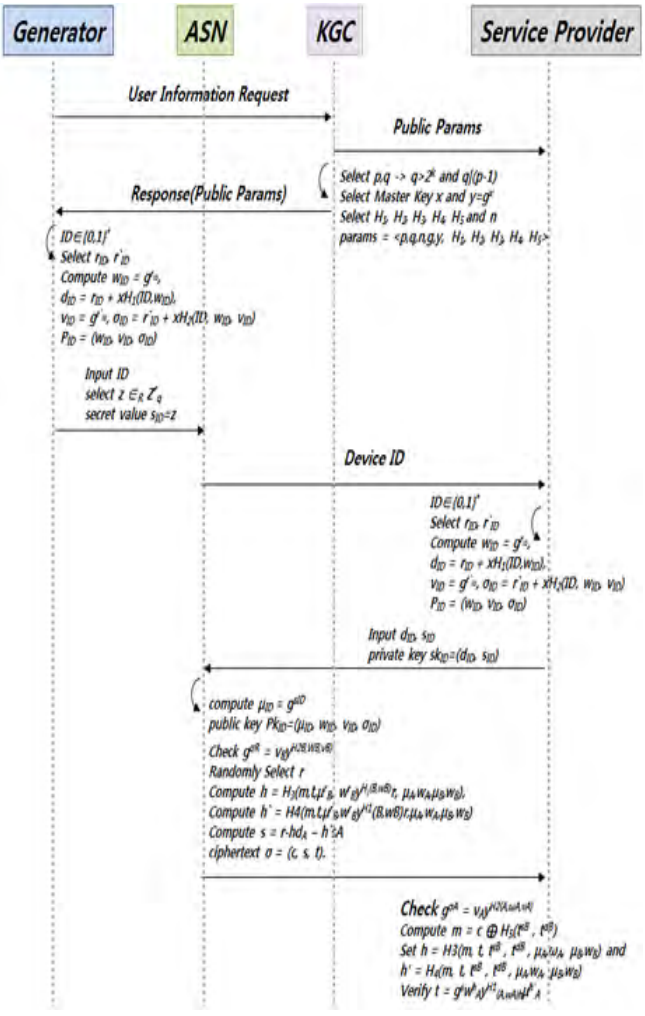
$h' = H_4(m,t,\mu^r B,w^r,B,y^{H_1(B,wB)}r,\mu_A,w_A,\mu_B,w_B)$,

$s = r = hd_A - h\,s_A$를 계산하고 암호문 σ = (c, s, t)를 전송 한다.

▪ **검증** : 서비스 제공자는 $g^{\sigma A} = v^A y^{H_2(A, W_A, v_A)}$를 검증한다. 먼저, $m = c \oplus H_5(t^{sB}, t^{dB})$를 계산한다. 이후

$h = H_3(m,t,t^{sB},t^{dB},\mu_A,\omega_A,\mu_B,\omega_B)$,

$h' = H_3(m,t,t^{sB},t^{dB},\mu_A,\omega_A,\mu_B,\omega_B)$가 만족하면 인증이 완료 된다.



(그림 2) 스마트 홈 환경에서 무인증서 인증 기법

4. 성능평가

스마트홈 ASN에서 랜덤 오라클 모델에서의 보안 IND-CCA2에 대한 보안 성능 평가이다. 먼저, 임의의 공격자가 IND-CCA2 공격을 이용하여 암호 키를 획득하려고 할 경우 (수식 x) 에서 ϵ 이상 가져야 하는 문제를 해결해야 한다. 이때, (수식 x)에서 q_{pak}는 partial key 생성 쿼리, q_{sk}는 private key 생성 쿼리, q_{pk}는 public key 요청을 의미한다. 또한, q_{pkr}은 public key 대체 쿼리, qs는 암호화 쿼리를 의미하며, q_u는 복호화 쿼리를 의미한다.

$$\epsilon' \geq \frac{\epsilon}{q_3+q_5+q_s}\frac{(1-\epsilon')^{q_{pkr}}}{q_{pk}}\left(1-q_s\frac{2q_3+q_4+q_5+3q_s}{2^k}\right)\left(1-\frac{q_u}{2^k}\right)$$

다음은 보안 문제 EUF-CMA-1에 대한 보안 성능 평가이다. 공격자는 선택한 평문에 대한 암호문을 가지고 공격을 시도하여 암호 키를 획득하려고 할 경우 (수식 x)의 문제를 해결해야 한다. 이때, q_{pk}는 public key 요청 과정, qpkr은 public key replacement 쿼리를 의미하며, 공격자는 매칭되는 평문 암호문에 대해 ϵ 이상 가져야 하는 문제를 해결해야 한다.

$$\epsilon' \geq \frac{1}{9q_{pk}}(1-\epsilon')^{q_{pkr}}$$

5. 결론

제안하는 인증 프레임워크에서는 다양한 하드웨어 연산 능력을 가진 IoT 디바이스를 지원하기 위해, 무인증서를 기반으로 디바이스의 타입에 따라 인증 프로토콜을 제안하였다.

제안하는 IoT 환경에서 무인증서 기반 인증 및 권한관리 프레임워크는 국제 표준을 기반으로 설계하였으며, 디바이스의 물리적 특성 및 정보 활용 특성을 고려하여 각 디바이스 군별로 설계되었다. 또한, IoT 환경의 상황인식 컴퓨팅의 목적을 지원하도록 클라우드 컴퓨팅과의 통합 환경을 고려하여 설계하여, IoT 환경에 즉시 실용화가 가능하다.

참고문헌

[1] BARBOSA, Manuel; FARSHIM, Pooya. Certificateless signcryption. In: Proceedings of the 2008 ACM symposium on Information, computer and communications security. ACM, 2008. p. 369-372

[2] WU, Chenhuang; CHEN, Zhixiong. A new efficient certificateless signcryption scheme. In: 2008 International Symposium on Information Science and Engineering. IEEE, 2008. p. 661-664.

[3] XIE, Wenjian; ZHANG, Zhang. Efficient and provably secure certificateless signcryption from bilinear maps. In: Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on. IEEE, 2010. p. 558-562.

[4] XIE, Wenjian; ZHANG, Zhang. Certificateless Signcryption without Pairing.IACR Cryptology ePrint Archive, 2010, 2010: 187.