IoD 환경에서의 키 교환 프로토콜 분석

임준호*, 정재열*, 정익래*
*고려대학교 정보보호대학원
pcrjh@korea.ac.kr, blue7angels@korea.ac.kr, irjeong@korea.ac.kr

Analysis of Key Exchange Protocol in IoD Environment

Joon Ho Lim*, Jae Yeol Jeong*, Ik Rae Jeong*
*School of Cybersecurity, Korea University

요 약

드론 기술의 발전은 우리 실생활에서 IoD를 활용한 다양한 서비스 제공을 가능하게 해주었다. IoD 환경을 실질적으로 활용하기 위해서는 드론과 사용자 간의 인증 및 프라이버시 보호가 보장되어야 한다. 본 논문에서는 안전한 드론 운용환경 제공을 위하여 발표된 키 교환 프로토콜들을 분석하여 문제점을 도출하였다. 두 프로토콜 모두 IoD 환경에서 사용자와 드론 간 안전한 키 교환을 목적으로 제안되었으나, 세션키 또는 사용자의 익명 아이디가 노출되어 사용자의 프라이버시가 노출될 수 있다는 문제점과 키 교환 간 사용자의 참여가 요구되어 재해·재난 환경 등 자동화된 프로토콜을 요구하는 실제 환경에 활용되기 부적합하다는 한계를 보여주었다. 따라서 향후 IoD 환경에 적합한 키 교환 프로토콜은 강한 익명성을 동반한 프라이버시 보장뿐 아니라, 드론이 직접 키 교환 요청을 할 수 있어야 한다.

1. 서론

최근 무인 항공 기술의 급속한 발전으로 인해 IoD (Internet of Drones)[1]를 기반으로 한 다양한 서비스가 실생활에서 상용화되고 있다. 오늘날 드론은 고성능 센서 기반 하드웨어를 장착할 수 있으므로 사회의 다양한 분야 (예 : 물류, 군사, 교통, 재난 지역 등)에서 긴급한 정보를 수집, 전송, 경고하고 실제 제품을 배송하는 데 사용된다. 현재 무인 드론은 COVID-19 관리 시스템에서 중요하게 사용되고 있으며 중국에서는 드론이 특정 지역 (또는 검역소)을 비행한 후 마스크를 쓰지 않는 사람들에게 경보 알림 서비스를 제공한다.

그림 1과 같이 IoD 환경은 다수의 드론이 서버가 관리하는 구역 곳곳에 존재하고 사용자가 요청하거나 특별한 정보를 획득하면 서버에게 전송한다. 사용자가 실시간으로 드론과의 통신을 원하면 서버는 사용자와 드론 간의 세션키를 생성하는 것을 지원하고 생성된 키를 이용하여 사용자와 드론은 안전하게 통신한다. 하지만 드론은 무인 제어 기계이기 때문에 서버와 사용자가 보낸 메시지를 어떻게 신뢰하거나 숨길 수 있는지 등에 대한 보안 문제를 해결해야 한다. 이러한 점을 해결하기 위해서 IoD 환경에서의 안전한 키 교환 프로토콜이 많은 문헌에서 연구되고 있다.[2-5].

(그림 1) IoD 환경

본 논문에서는 최근에 연구된 IoD 환경에서의 사용자와 드론 간의 키 교환 프로토콜 중에서 2가지 프로토콜을 분석한다.

2. Wazid 프로토콜[3]

Wazid 프로토콜[3]은 사전준비 단계, 사용자 등록 단계, 사용자 모바일기기 접속 단계, 키 교환 단계로 구성되어 있다.

사전준비 단계는 서버에 드론을 등록하는 단계이다. 서버는 드론의 아이디와 서버의 비밀 키를 이용하여 드론의 익명 아이디와 인증값을 생성한다. 그리고 서버와 드론의 메모리에 생성한 값을 저장한다.

사용자 등록 단계는 서버에 사용자를 등록하는 단계이다. 사용자는 모바일기기에 자신의 아이디를 입력하고 아이디를 서버에게 전송한다. 서버는 사용자의 아이디와 자신의 비밀 키를 이용하여 사용자의 익명 아이디와 서버의 익명 아이디, 인증값을 생성하고 생성한 값들과 드론의 익명 아이디를 사용자에게 보내준다. 모바일기기는 타인이 접근할 수 있으므로 데이터 보호를 위해서 비밀번호, 생체 정보를 입력하여 생체 정보의 퍼지를 추출하여 비밀번호와 함께 사용한다. 사용하는 방법은 사용자의 아이디와 사용자의 익명 아이디, 비밀번호, 생체 정보의 퍼지 값을 다양하게 결합하고 해시 한 뒤 서버가 보낸 값들과 XOR 연산한다. 즉, 서버가 보낸 값들을 모바일기기에서 사용하려면 정확한 아이디와 비밀번호, 생체 정보를 입력해야 한다. 서버는 사용자의 아이디와 사용자의 익명 아이디, 서버의 익명 아이디, 인증값을 저장하고 모바일기기는 서버가 보내준 값을 변환한 값들과 그 값의 복원할 때 사용할 값들을 저장한다.

사용자 모바일기기 접속 단계는 사용자가 모바일기기에 접속하여 서버에게 키 교환 요청을 하는 단계이다. 사용자는 자신의 모바일기기에 아이디와 비밀번호, 생체 정보를 입력한다. 입력한 정보들이 사용자 등록 단계에서 입력한 정보들과 같다면 모바일기기는 입력받은 정보를 이용하여 퍼지 값을 추출하고 아이디와 비밀번호와 함께 사용하여 저장된 정보를 서버가 보낸 값으로 복원한다. 난수를 생성하고 난수와 복원한 값을 이용하여 4개의 메시지를 생성하고 현재시간 정보와 함께 서버에게 보낸다.

키 교환 단계는 사용자가 보낸 값을 서버가 받아서 데이터베이스에 저장된 사용자가 맞는지 확인하고 맞으면 사용자가 요청한 드론과의 키 교환을 하기 위해서 데이터베이스에 저장된 드론의 정보를 가져온다. 서버도 난수를 생성하고 생성한 난수와 드론의 정보를 이용하여 3개의 메시지를 생성하고 현재시간 정보와 함께 드론에게 보낸다. 드론은 서버가 보낸 값을 이용하여 정당한 서버가 맞는지 확인한다. 맞으면 난수를 생성하고 난수와 서버가 보낸 값들을 이용하여 3개의 메시지를 생성하고 현재시간 정보와 함께 사용자의 모바일기기에게 보낸다. 이때 드론은 서버가 보낸 값을 사용하여 세션키를 생성한다. 모바일기기는 드론이 보낸 값을 이용하여 정당한 드론이 보낸 게 맞는지 확인한 뒤 세션키를 생성한다. 생성된 세션키는 사용자, 드론, 서버의 익명 아이디와 생성한 난수들을 결합하여 해시 한 것이다. 이 키를 사용자와 서버가 안전한 통신을 위해서 사용한다.

3. Zhang 프로토콜[5]

Wazid 프로토콜[3]은 큰 문제점을 가지고 있다. 그것은 서버에 등록된 다른 사용자들이 생성된 세션키를 알게 되는 것이다. 그 이유는 세션키가 서버와 드론, 사용자의 익명 아이디와 생성한 난수로 구성되어 있는데 생성한 난수들을 익명 아이디들로 보호한다. 그런데 서버의 익명 아이디를 이용하여 사용자의 익명 아이디를 보호하는데 서버에 등록된 다른 사용자들은 서버의 익명 아이디를 알고 있기에 사용자의 익명 아이디를 알 수 있다. 또한, 다른 사용자들은 드론의 익명 아이디도 알고 있기에 모든 익명 아이디들을 알 수 있고 그것을 토대로 난수들도 알 수 있으며 결국 세션키도 알 수 있다.

$$
\begin{aligned}
M_1 &= RID_i \oplus h(RID_S \| T_1) \\
M_8 &= h(RID_S \| r_1 \| r_2) \\
M_{10} &= h(RID_{DR_j} \| RID_i \| T_3) \oplus r_3 \\
M_{11} &= h(RID_i \| RID_{DR_j} \| r_3) \oplus M_8 \\
SK &= h(M_8 \| r_3 \| RID_i \| RID_{DR_j}) \\
&= h(h(RID_S \| r_1 \| r_2) \| r_3 \| RID_i \| RID_{DR_j})
\end{aligned}
$$

(그림 2) 세션키 노출 과정

그림 2는 세션키의 노출 과정을 자세히 나타낸 것이다. $M_1, M_{10}, M_{11}, T_1, T_3$는 공개되어 있어서 모든 사람이 알 수 있고 M_8, SK는 숨겨져 있다. RID_i는 사용자의 익명 아이디, RID_{DR_j}는 드론의 익명 아이디, RID_S는 서버의 익명 아이디이고, r_1은 사용자가 선택한 난수, r_2는 서버가 선택한 난수, r_3는 드론이 선택한 난수이다. M_1, T_1은 공개되어 있으므로

RID_S를 알고 있으면 RID_i를 알 수 있다. 그러면 공개된 M_{10}, T_3과 알고 있는 RID_i, RID_{DR_j}를 이용하면 r_3를 알 수 있다. M_{11}과 RID_i, RID_{DR_j}, r_3를 통해서 M_8을 알 수 있고 결국 SK를 알게 된다.

Zhang 프로토콜[5]는 이러한 단점을 발견하고 개선한 것으로 프로토콜은 Wazid 프로토콜[3]과 거의 유사하다. 차이점은 생체 정보를 사용하지 않고 오직 사용자의 아이디와 비밀번호만을 사용하는 것과 세션키를 생성할 때 사용하는 정보들을 보호하는 방법을 조금 변경한 것이다. 그 방법은 앞에서 생성한 난수를 이용하여 뒤에 생성한 난수를 보호하는 것이다.

$$
\begin{aligned}
M_1 &= RID_i \oplus h(RID_S \| T_1) \\
M_2 &= h(RID_i \| RID_S \| \alpha_i) \oplus r_1 \\
M_5 &= h(RID_{DR_j} \| \alpha_j) \oplus r_1 \\
M_8 &= h(RID_{DR_j} \| RID_i \| r_1) \oplus r_2 \\
M_9 &= h(r_1 \| r_2) \\
SK &= h(RID_i \| RID_{DR_j} \| RID_S \| M_9) \\
&= h(RID_i \| RID_{DR_j} \| RID_S \| h(r_1 \| r_2))
\end{aligned}
$$

(그림 3) 세션키 보호 방법

그림 3은 Zhang 프로토콜[5]에서 세션키의 노출의 방지하는 방법을 나타낸 것이다. M_1, M_2, M_5, M_8, T_1은 공개되어 있어서 모든 사람이 알 수 있고 M_9, SK는 숨겨져 있다. RID들은 Wazid 프로토콜[3]과 같으며, r_1은 사용자가 선택한 난수, r_2는 드론이 선택한 난수이다. a_i는 사용자의 인증값이고 a_j는 드론의 인증값이다. a_i는 사용자와 서버만 알고 있고 a_j는 드론과 서버만 알고 있다. Wazid 프로토콜[3]과 마찬가지로 M_1, T_1이 공개되어 있고 RID_S를 알고 있으면 RID_i를 알 수 있다. 그러면 SK를 알기 위해서는 r_1, r_2를 알아야 한다. 하지만 M_8을 통해서 r_2를 알려면 r_1를 알아야 하고 M_2, M_5를 통해서 r_1을 알려면 a_i 또는 a_j를 알아야 하는데 이 값은 사용자와 서버 또는 드론과 서버만 아는 값으로 다른 사용자뿐만 아니라 사용자와 서버, 드론을 제외한 모든 사람은 SK를 알 수 없다.

4. 기존 프로토콜들의 문제점

Wazid 프로토콜[3]과 Zhang 프로토콜[5]의 세션키 노출, 보호 방법을 보면 두 프로토콜 모두 사용자의 익명 아이디가 노출되는 것을 알 수 있다. 두 방법 모두 사용자와 드론의 프라이버시를 보호하기 위해 사용자와 드론의 익명 아이디를 사용한다. 그러나 익명 아이디가 노출되거나 연결 해제가 없는 익명 아이디로는 강력한 익명성이 제공되지 않는다. 왜냐하면, 공격자가 사용자 (또는 드론)의 같은 익명 아이디를 관찰할 수 있다면 통신 빈도나 사용자 (또는 드론)의 움직임을 추적할 수 있다. 이것은 결국 사용자 (또는 드론)의 실제 신원으로 이어질 수 있으며, 따라서 사용자 (또는 드론)의 프라이버시를 침해할 수 있다. 따라서 익명 아이디의 노출을 막거나 익명 아이디를 계속해서 변경해줘야 한다.

또한, 드론은 공중에서 넓은 지역의 정보를 수집하기 때문에 특정 유형의 이벤트를 감지하거나 발견할 수 있다. 이 경우 사용자가 특정 이벤트를 발견하고 정보를 요청하기 전에 드론이 먼저 정보를 제공할 수 있어야 한다. 이는, 현재 코로나로 인해서 확진자가 발생할 때마다 재난문자를 제공하는 것과 같은 것이다. 그래서 드론이 서버에게 인증 프로토콜의 시작을 요청해야 한다. 즉, 사용자와 서버 간의 보안 채널을 만들기 위해 드론이 먼저 인증 프로토콜을 시작하도록 요청하는 경우 (예 : 긴급 재난, 교통 경보 서비스, 적군 공습 서비스)에 대한 프로토콜이 필요하다. 이런 상황에도 사용자 (드론 소유자)의 개인 정보보호도 마찬가지로 보장돼야 한다. 하지만, 기존 프로토콜들은 오직 사용자의 요청에 따라서만 키 교환 프로토콜을 수행할 수 있다.

5. 결론

Marketsandmarkets의 2017년 보고서에 따르면 드론 서비스 분야의 글로벌 시장규모는 2016년 7억530만 달러에서 연평균 71.62%의 고성장을 지속하여 2022년에는 180억2,270만 달러의 거대시장을 형성할 것으로 예상한다. 즉, 드론을 이용한 서비스 제공의 보편화는 가까운 시일에 이뤄질 것으로 보인다. 하지만 드론을 사용하는 모든 분야에서 드론과 사용자 간의 인증 및 프라이버시 보호는 중요한 이슈이다.

우리가 분석한 두 프로토콜 모두 IoD 환경에서의 사용자와 드론 간의 안전한 키 교환을 목적으로 제안되었다. 하지만 Wazid 프로토콜[3]은 세션키가 노출된다는 단점이 존재하고 이를 개선한 Zhang 프로

토콜[5]도 사용자의 익명 아이디가 노출된다는 단점이 있다. 그리고 기존에 제안된 프로토콜 모두 사용자의 요청에 따라서만 키 교환을 수행할 수 있다. 이것으로는 사건·사고가 잦고 자연재해가 자주 발생하는 현재에는 부족한 것으로 보인다.

앞으로는 기존에 제안된 프로토콜의 분석을 토대로 사용자와 드론의 프라이버시를 보호하는 강한 익명성을 만족하고 사용자뿐만 아니라 드론이 키 교환 요청을 할 수 있는 프로토콜을 개발할 계획이다.

Acknowledgement

본 연구는 고려대 암호기술 특화연구센터(UD170109ED)를 통한 방위사업청과 국방과학연구소의 연구비 지원으로 수행되었습니다.

참고문헌

[1] M. Gharibi, R. Boutaba, S. L. Waslander “Internet of drones”, IEEE Access, vol. 4, pp. 1148-1162, 2016.

[2] D. He, S. Chan, and M. Guizani “Drone-assisted public safety networks: The security aspect” IEEE Communications Magazine, vol. 55, no. 8, pp. 218-223, 2017.

[3] M. Wazid, A.K. Das, N. Kumar, A.V. Vasilakos, and J.J.P.C. Rodrigues, “Design and Analysis of Secure Lightweight Remote User Authentication and Key Agreement Scheme in Internet of Drones Deployment” IEEE Internet of Things Journal, vol. 6, no. 2, pp. 3572-3584, 2019.

[4] J. Srinivas, A.K. Das, N. Kumar, and J.J.P.C. Rodrigues. “TCALAS: Temporal Credential-Based Anonymous Lightweight Authentication Scheme for Internet of Drones Environment” IEEE Transactions on Vehicular Technology, vol. 68, no. 7, pp. 6903-6916, 2019.

[5] Y. Zhang, D. He, L. Li, and B. Chen “A Lightweight Authentication and Key Agreement Scheme for Internet of Drones” Computer Communications, vol. 154, pp. 455-464, 2020.