이동 환경에서의 안전한 티켓팅 서비스

홍준우*, 황성민*, 김순자*

*경북대학교, 전자공학과

Secure Ticketing Service for Mobile Environment

Jun-woo Hong*, Sung-min, Hwang*, Soon-Ja Kim*

*Department of Electronics Engineering Kyungpook National Univ.

요 약

이동 환경과 관련된 기술이 발달함에 따라 모바일 티켓팅 서비스에 대한 관심이 높아지고 있다. 그러나 지금까지 제안된 모바일 티켓팅 시스템은 만족할 만한 확장성과 안전성을 제공하지 못하고 있다. 본 논문에서는 보다 안전한 모바일 티켓팅 시스템을 제안한다. 기존의 모바일 티켓팅 시스템의 문제점을 살펴보고 모바일 환경에 적합한 프레임 워크를 바탕으로 티켓의 복사, 수정을 방지할 수 있는 시스템을 제시한다. 기존의 시스템에서 티켓의 수정이 가능한 것을 서명을 사용함으로서 해결하고 그에 대한 안전성을 분석한다.

I. 서론

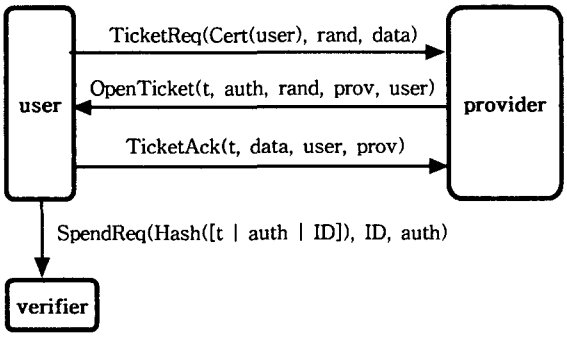
최근 이동 환경과 관련된 기술이 발달함에 따라 이동 환경 상에서의 다양한 서비스에 대한 관심이 높아지고 있다. 그 중에서 모바일 티켓팅 서비스는 그 유용성과 편리함으로 인해 매력적인 서비스 중 하나가 되었으며 휴대전화, PDA 뿐만 아니라 스마트 카드에서도 사용이 가능하다. 전통적인 오프라인 티켓팅과 모바일 티켓팅이 다른 점은 오프라인 티켓팅은 단지 하나의 서비스에 국한되고 티켓을 구입하기 위해 특정 장소에 가야하지만 모바일 티켓팅은 다양한 서비스에서 사용될 수 있고 장소의 제약을 많이 받지 않고 티켓 구입이 가능한 점이다. 모바일 티켓은 복사나 위조가 어려워야 하고 다양한 서비스에 사용하기 위한 확장성이 좋아야 하며 위임이 가능해야 한다. 그러나 현존하는 티켓팅 서비스는 만족할 만한 확장성과 안전성을 제공하지 못하고 있다. K. Fujimura 등이 범용 디지털 티켓에 관한 프레임 워크를 제안하였고 [1,2] A. Maña 등이 이동 환경에서의 티켓팅 서비스를 제안하였지만 [3] 충분한 안전성을 제공하지 못하였다.

본 논문에서는 안전성을 강화한 모바일 티켓팅 시스템에 관하여 논의할 것이다. 2장에서는 기존 시스템을 소개하고 문제점을 살펴본다. 3장에서는 모바일 티켓팅 시스템이 가져야할 요건과 사용할 언어를 살펴본 뒤 안전성이 강화된 모바일 티켓팅 시스템을 제시한다. 4장에서는 제시한 시스템의 안전성을 분석하고, 끝으로 5장에서 결론을 맺는다.

II. 기존 연구

A. Maña 등이 제안한 GSM-Ticket 시스템은 [3] 사용자(user), 제공자(provider), 검증자(verifier)로 구성되어 있으며 사용자는 티켓 구매자, 제공자는 티켓 판매자, 검증자는 티켓이 올바른지 확인하는 객체이다. 사용자가 티켓 요청을 하면 제공자는 티켓을 보내고 티켓을 받은 사용자는 응답을 보내며 티켓을 사용하기 위해 사용자는 티켓을 검증자에게 보내는 구조이다. 이 시스템은 모바일 환경에서 효과적인 프로토콜이지만 충분한 안전성을 제공하지 못한다. 다음 그림 1에 나타난 GSM- Ticket 시스템에서 사용자는 제공자로부터 t와 auth가 포함된 티켓을 받게 된다. t와 auth는 티켓고유 아이디와 티켓 제한 사

항에 관한 파라미터이다. 제공자는 이 파라미터를 자신의 개인키로 서명하여 보낸다. 티켓을 사용하는 과정에서 사용자는 이 파라미터를 그대로 사용하기 때문에 쉽게 수정 가능하다. 즉, 사용자가 티켓을 받은 다음 t나 auth 값을 수정하여 검증자에게 보내더라도 검증자 측에서는 이 파라미터들이 수정되었는지 알지 못한다.



BCert(data, signer, recipient) ::= Encipher(Sign(data, PrivKey(signer)), PubKey(recipeient))
OpenTicket(ticketSecretID, auth, rand, Provider, User) ::= BCert([ticketSecretID | auth | rand], Provider, User)
TicketAck(ticketSecretID, ticketData, User, Provider) ::= BCert([ticketSecretID | ticketData], User, Provider)
auth : data, Cert(Provider)
data : 가격, 상태, PublicID 등
t : ticketSecretID
rand : random number

그림 1: GSM-Ticket 시스템

III. 제안 방식

본 장에서는 제안하는 모바일 티켓팅 시스템에 대해 설명하고 특징을 알아본다.

1. 요구 요건

모바일 티켓팅 시스템은 다음과 같은 요건을 갖춰야 한다. 첫째, 티켓은 간단하게 표현되어야 한다. 둘째, 시스템 구성을 위해 별도의 인프라가 필요없어야 한다. 셋째, 티켓 구매자만 티켓을 사용하는 것이 아니라 다른 사람에게 위임이 가능해야 한다. 넷째, 데이터 전송량이 작아야 한다. 다섯째, 티켓의 복사나 위조가 어려워야 한다.

2. 티켓 표현 언어

디지털 티켓을 표현하는 언어는 여러 가지가 있다. 대표적으로 XML을 이용한 티켓 [4,5], RDF를 사용한 티켓 [1,7], EDDF를 사용한 티켓 [5] 등이 있지만 본 논문에서는 복잡한 데이터 구조를 간단하게 표현할 수 있는 S-expression을 사용한다. [8] 에서 소개된 S-expression은 텍스트로 표현하거나 베이스 64로 코드화가 가능하기 때문에 오프라인 상에서의 티켓 확인과 온라인 상에서 전송이 용이하다. 그림 2는 S-expression을 사용한 영화 티켓의 예이다.

```
(authorization
    (not-before "2003-07-04_18:00:00")
    (not-after "2003-07-04_20:00:00")
    (cinema
        (price "6500" "WON"
        (film "Friend")
        (site "Megabox.Daegu.Korea")
        (place
            (theater "3"
                (seats "C" "12")
            )
        )
    )
)
```

그림 2: S-expression으로 표현한 영화 티켓

3. 시스템 계수

다음 표 1은 제안 방식에서 사용하는 시스템 계수에 대한 설명이다.

표 1: 시스템 계수

계 수	의 미
user	사용자. 티켓 구매 및 사용
provider	제공자. 티켓 판매
verifier	검증자. 티켓 검증
rand	임의의 난수
Cert(A)	객체 A의 인증서
$Encipher_A()$	객체 A의 공개키로 암호화하는 함수
$Sign_A()$	객체 A의 개인키로 서명하는 함수
Hash[]	안전한 일방향 해쉬함수

4. 프로토콜

제안 방식은 티켓의 구매와 사용의 단계로 나누어진다. 다음은 각 단계에 대한 설명이다.

1) 티켓 구매

그림 3은 기본적인 티켓 구매 과정을 나타내고 있다.

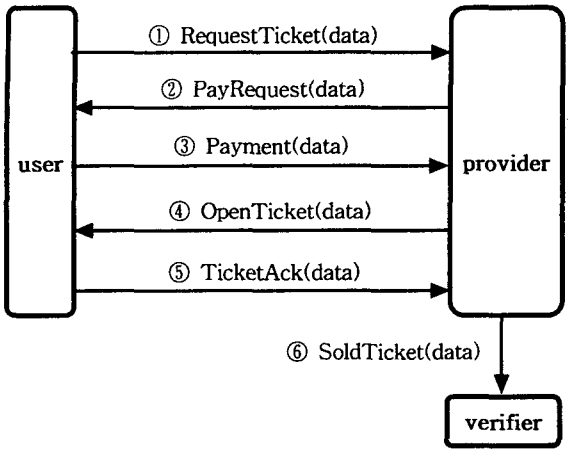


그림 3: 티켓 구매 과정

① 사용자는 원하는 종류의 서비스와 날짜 등이 포함된 data와 난수 rand, 자신의 인증서 Cert(user)를 제공자에게 보낸다. 이 때 난수는 반복 공격을 방지하기 위해 사용된다.

② PayRequest(data) ::= $Encipher_{user}(Sign_{provider}$(data | rand), Cert(provider))

사용자가 티켓을 요청하면 제공자는 지불 요청을 한다. 지불 요청은 제공자가 data와 rand를 서명하고 자신의 인증서를 첨부하여 사용자의 공개키로 암호화하여 보낸다.

③ Payment(data) ::= $Encipher_{provider}$(card, data, rand)

지불 요청을 받은 사용자는 자신의 신용카드 정보 card와 data, rand를 제공자의 공개키로 암호화하여 보낸다.

④ OpenTicket(data) ::= $Encipher_{user}(Sign_{provider}$(ticket | data), rand, Cert(provider))

제공자는 사용자의 인증서를 확인한 후 이상이 없으면 티켓에 관한 정보인 ticket과 data를 서명한 후 티켓 요청 때 받은 rand와 자신의 인증서를 포함하여 사용자의 공개키로 암호화하여 전송한다. 이 때 티켓은 어느 누구에게도 소속되어 있지 않아서 open ticket이라 한다. 기존의 모바일 티켓팅 시스템의 문제점을 해결하기 위해 티켓에 관한 정보인 ticket과 data를 서명한다. open ticket에 서명한 ticket 값을 사용하므로 ticket의 복사와 위조가 불가능하다.

⑤ TicketAck(data) ::= $Encipher_{provider}$(Hash[ticket | data | rand])

사용자는 티켓을 받은 후 ticket, data, rand를 해쉬한 후 제공자의 공개키로 암호화하여 응답을 보낸다.

⑥ SoldTicket(data) ::= $Encipher_{verifier}$($Sign_{provider}$(ticket | data))

응답을 받은 제공자는 검증자에게 티켓이 판매되었음을 알린다.

2) 티켓 사용 및 위임

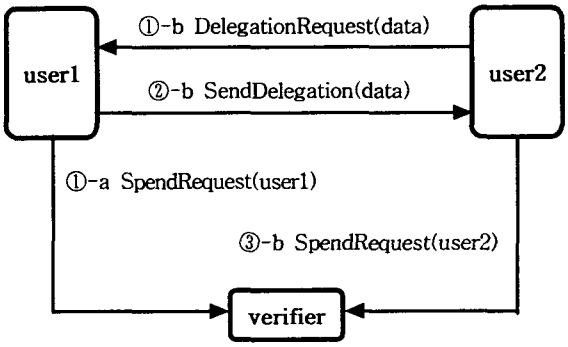


그림 4: 티켓 사용 및 검증 과정

그림 4는 티켓의 사용 및 위임 과정을 나타내고 있다. open ticket을 받은 사용자는 티켓을 사용하기 위해 검증자에게 자신의 아이디를 포함한 필요한 정보를 보낸다. open ticket은 티켓을 구입한 사람뿐만 아니라 다른 사람에게 위임할 수 있다. 티켓을 사용할 때 자신의 아이디를 포함하여 티켓을 보내는데 이것을 close ticket이라 한다. 기존의 시스템과는 달리 close ticket에 ticket 파라미터를 그대로 사용하지 않고 제공자가 서명한 ticket 파라미터를 포함했기 때문에 사용자는 티켓의 임의 수정이 불가능하다. 티켓 사용 과정은 다음과 같다.

①-a SpendRequest(user1) ::= (Hash[$Sign_{provider}$(ticket | data) | userID1], userID1)

티켓을 구입한 사용자1은 티켓을 사용하기 위해 open ticket에서 $Sign_{provider}$(ticket | data) 값을 추출하여 자신의 아이디 userID1과 함께 해쉬하여 close ticket을 만든 후 자신의 아이디와 함께 검증자에게 보낸다. 티켓 사용 요청을 받은 검증자는 제공자에게서 받은 $Sign_{provider}$(ticket | data) 값과 userID1을 해쉬하여 결과값을 close ticket과 비교하여 검증한다.

사용자1이 구입한 티켓을 사용자2에게 위임하여 사용하는 과정은 다음과 같다.

①-b DelegationRequest(data) ::= ($Sign_{user2}$(data), Cert(user2))

사용자2는 위임을 원하는 티켓의 data 값을 서명하여 자신의 인증서와 함께 사용자1에게 전송한다.

②-b SendDelegation(data) ::= Hash[$Sign_{provider}$(ticket | data) | userID2]

위임 요청을 받은 사용자1은 사용자2의 아이디 userID2를 포함하여 close ticket을 만들어서 사용자2에게 전송한다.

③-b SpendRequest(user2) ::= Hash([$Sign_{provider}$ (ticket | data) | userID2], userID2)

close ticket을 받은 사용자2는 자신의 아이디와 close ticket을 검증자에게 전송한다.

IV. 요건 및 안전성 분석

본 논문에서 제시한 모바일 티켓팅 시스템이 갖춰야 할 요건 및 안전성에 대해 알아본다. 모바일 티켓팅 시스템에서 티켓의 복사와 수정이 안전성에 관한 가장 중요한 요소이다. 첫째, 본 논문에서 제시한 모바일 티켓팅 시스템은 S-expression을 사용하기 때문에 티켓의 표현이 간단하다. 둘째, 기존의 휴대폰, PDA, 스마트 카드 상에서 사용 가능하므로 별도의 인프라가 필요없다. 셋째, open ticket을 사용하기 때문에 티켓 구매자뿐만 아니라 다른 사람에게 티켓을 위임하여 사용 가능하다. 넷째, 데이터 전송량이 많지 않아서 모바일 환경에 적합하다. 다섯째, 티켓 구입자 이외의 사람이 같은 티켓을 사용하기 위해선 open ticket을 가져야 한다. open ticket을 갖기 위해서는 그림 3의 ④ OpenTicket(data)에서 open ticket을 추출해야 한다. 그러나 메시지가 티켓 구입자의 공개키로 암호화되어 있으므로 티켓 구입자만이 복호할 수 있다. 따라서 다른 사용자의 티켓 복사는 불가능하다. 또한 티켓 구입자가 티켓을 복사하는 경우 티켓 구입시 제공자가 검증자에게 티켓을 서명하여 보내기 때문에 중복사용이 불가능하다. 또한 티켓의 수정을 위해서는 open ticket에서 ticket 파라미터를 바꿔야 하는데 ticket 파라미터는 제공자가 서명했기 때문에 사용자가 수정할 수 없다. 그리고 만일 임의로 수정한 close ticket을 보내더라도 close ticket에 검증자가 서명된 ticket 파라미터를 포함하고 있기 때문에 티켓이 수정되었다는 것을 알 수 있다. 그러므로 사용자의 티켓 수정은 불가능하다.

V. 결론

이동 환경에 관한 각종 기술이 발달하면서 이동 환경에서 사용할 수 있는 서비스에 대한 관심이 증가하고 있다. 본 논문에서는 기존의 모바일 티켓팅 시스템의 안전성에 관한 문제점을 분석하고, 제시된 문제점을 바탕으로 모바일 티켓팅 시스템이 가져야할 요건과 안전성이 강화된 시스템을 제안하였다. 티켓 사용 과정에 제공자의 서명값을 포함시킴으로서 사용자가 임의로 티켓을 복사하거나 수정하기 어려운 시스템을 제시하고 그 안전성을 분석하였다. 제시된 시스템을 잘 활용한다면 보안적인 면에서 보다 안전한 시스템을 만들 수 있을 것이다.

참고문헌

[1] K. Fujimura and Y. Nakajima, "General-purpose Digital Ticket Framework," *Proceedings of the 3rd USENIX Workshop on Electronic Commerce,* pp. 177-186, Aug. 1998.

[2] K. Fujimura, H. Kuno, M. Terada, K. Matsuyama, Y. Mizuno, and J. Sekine, "Digital-Ticket-Controlled Digital Ticket Circulation," *Proceedings of the 8th USENIX Security Symposium.,* Aug. 1999.

[3] A. Maña, J. Martinez, S. Matamoros, and J. M. Troya, "GSM-Ticket: Generic Secure Mobile Ticketing Service," *Gemplus Developer Conference,* Jun. 2001.

[4] K. Fujimura, Y. Nakajima, and J. Sekine, "XML Ticket: Generalized Digital Ticket Definition Language," *http://www.w3.org/DSig/signed-XML99/pp/NTT_xml_ticket.html,* 1999.

[5] Y. Nakajima and K. Fujimura, "XML Ticket Model and Syntax Specification," *IETF Internet Draft,* to appear.

[6] D. Bider, "EDDF: Efficient Descriptive Data Format," *http://www.eddf.org,* 2000.

[7] "Resource Description Framework(RDF) Model and Syntax Specification," *The World Wide Web Consortium, Recommandation,* 1998.

[8] R. Rivest, "S-Expressions," *Internet Draft, Internet Engineering Task Force,* May 1997.