유무선 장치간의 인증된 채널 설정 프로토콜

양종필, 서 철, 이경현*

부경대학교 전자계산학과
*부경대학교 전자컴퓨터정보통신공학부

Improved Authentication Protocol between Wired and Wireless Device

Jong-Phil Yang, Chul Sur, Kyung Hyune Rhee*

Department of Computer Science
*Division of Electronic, Computer and Telecommunication Engineering

요 약

무선 네트워킹 기술의 발전으로 인하여 무선환경에서의 안전한 상거래 및 뱅킹업무에 대한 요구가 점차적으로 증가하고 있다. 통신 당사자간의 인증 문제는 면대면 방식이 아닌 무선 환경에서는 안전한 m-business를 위한 필수적인 선결 조건이라 할 수 있다. 본 논문에서는 무선환경에서 가용한 인증된 채널을 설정할 수 있는 프로토콜을 제안하고자 한다. 제안 프로토콜은 통신 당사자들 사이의 상호 신분확인을 위한 인증서 교환시에 인증서 취소 상태의 검증에 대한 대역폭 소모를 최소화하며, 통신 당사자들 사이의 종단간의 인증된 채널 설정이 가능하다.

1. 서론

무선 네트워킹 기술의 발전으로 인하여 유·무선환경에서의 안전한 상거래 및 뱅킹 업무에 대한 새로운 시장이 점차적으로 증대되고 있다. 따라서, 유·무선 장치간의 안전한 거래를 위한 인증된 채널을 형성하는 것은 위와 같은 요구사항을 만족시키기 위한 중요한 연구과제가 되고 있다. 무선환경에서는 기존의 유선환경에서 일반적으로 사용되는 인증서를 통한 상호 인증과 키의 분배를 수행하는 기법은 무선 장치의 처리 능력과 대역폭 등의 성능적인 한계로 인하여 직접적으로 적용하기 어렵다.

본 논문에서는 [5]에서 소개되었던 유·무선간의 인증된 채널을 설정하는 프로토콜을 개선하고자 한다. 본 논문의 프로토콜은 기존의 무선환경을 위하여 정의된 WPKI에 기반하는 것이 아니라, [4]에서 소개된 개선된 공개키 프레임워크에 기반하고 있다. 즉, [4]에서 소개된 공개키 기반 구조를 바탕으로 유·무선 장치간의 인증된 채널을 설정하기 위하여, 통신 비용과 계산 비용이 효율적으로 감소된 프로토콜을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 사용되는 기 제안된 기법들을 간략히 소개한다. 3장에서는 기 제안된 기법들을 적용한 새로운 인증 프로토콜을 제안하고, 4장에서 제안된 기법에 대한 보안성에 대한 간략한 평가를 수행한다. 그리고, 5장에서 결론을 맺는다.

2. 관련 연구

2.1 개선된 공개키 프레임워크

인증서 취소에 대한 처리는 공개키 기반 구조(PKI) 환경에서 지속적으로 연구되어 해결해야 할 중요한 연구 과제이다[1]. 좀 더 효율적이고 즉시성이 있는 인증서 취소 기술을 개발하고자 indirect-CRL[1], Delta-CRL[2], OCSP[6], CRL[7], CRS[10], Windowed Certificate Revocation[8]과 같은 다양한 연구 결과들이 발표되었다.

한편, J.Zhou는 [3]에서 기존의 연구와는 차별화된 기법을 소개하였다. 즉, 인증기관이 아니라 사용자 스

스로가 자신의 인증서 취소 유무를 해쉬체인에 기반하여 인증서 수신자에게 증명할 수 있는 새로운 개념을 제안하였다. 하지만, [3] 논문에서의 해결방안은 기존의 CRL[9]과 같이 인증서 취소 정보의 즉시성에 대한 해결책이 부족하며, 또한 현실의 공개키 기반 구조(PKI)에서 적용하기에 다소 무리가 따르는 구조를 가지고 있다.

[4]에서의 개선된 공개키 프레임워크는 제어 윈도우(Control Window)라는 개념을 통하여 인증서 취소 정보에 따르는 통신 대역폭에 대한 소모를 최소화하고자 하였다. 이는 [3]에서 제시된 단순히 해쉬체인에 의존하여 인증서 취소 유무를 판단하는 것이 아니라, 제어 윈도우라는 허용된 타임구간(time period)내에서만 해쉬체인에 의존한 인증서 취소 유무에 대한 검증을 수행하는 기법이다. 또한, [4]에서의 개선된 공개키 프레임워크는 [3]과는 달리 기존의 PKI에서 쉽게 적용 가능하다.

2.2 E2ESP

[5]에서는 기존의 무선 인터넷 환경에서 가장 주요한 문제점인 통신 개체사이의 종단간의 인증을 수행하기 위하여 End-to-End Security Protocol(E2ESP)를 제안하였다. [5]에서는 CRL-Agent라는 신뢰되는 개체가 무선 단말기의 사용자들이 수행해야할 인증서 취소에 대한 처리를 사용자들을 대신하여 수행함으로써, 사용자들은 통신을 하고자 하는 상대방의 인증서 취소 유무를 판단하기 위한 대역폭의 낭비를 최소화하는 기법이다. 하지만, [5]에서는 CRL-Agent라는 부가적인 신뢰 개체에 의존하는 단점을 가지고 있다. 따라서, 본 논문에서는 기 제안된 E2ESP를 [4]에서 제안된 개선된 공개키 프레임워크를 사용하여 CRL-Agent의 필요성을 효율적으로 제거하고자 한다.

3. 개선된 인증 프로토콜

3.1 용어 및 가정사항

본 장에서는 [5]에서 소개되었던 프로토콜을 [4]를 기반으로 하여 좀 더 효율적으로 변경한다. 본 논문의 제안 방안을 통하여 효율적인 유선 장치와 무선 장치 사이의 인증된 채널 설정이 가능하다. 또한, [5]에서 사용되었던 신뢰기관인 CRL-Agent를 제거함으로써, 무선 장치에서의 통신 비용과 계산 비용을 줄일 수 있다. 본 절을 위해서 아래의 용어들이 사용된다.

- PK_X : 통신개체 X의 공개키 암호 알고리즘에서의 공개키(public key).
- $SIGN_X(M)$: 메시지 M은 통신개체 X에 의해서 전자서명.
- $E_K(M)$, $D_K(M)$: 메시지 M을 키 K로서 암호화 및 복호화.
- $seqN$: 전송되는 메시지의 순차 번호.
- ref : 얼마나 자주 세션키들이 갱신되는 가를 의미한다. 새로운 세션키는 $n = 2^{ref}$ 메시지 마다 계산된다. 즉, 새로운 세션키의 $seqN$는 $0, n, 2n, 3n, \cdots$.
- $Krls$: 유선 장치인 서버가 제공 가능한 모든 갱신 주기들의 리스트.

제안하는 안전한 채널 설정 프로토콜은 서버와 무선장치 사용자간의 비대칭적 상호 인증을 제공한다. 여기서, 비대칭적이라는 의미는 서버는 사용자의 패스워드를 통한 인증을 수행하고, 사용자는 서버를 인증서를 통해서 인증한다. 현실적으로, 사용자의 무선 장치에서의 인증서의 사용은 현재 WPKI가 표준화되어 있다고는 하지만 장치적인 한계로 인하여, 위와 같은 상호 인증기법이 현실적인 대안이 될 것이라 판단된다. 본 절을 위한 가정사항은 아래와 같다.

- 시스템을 위한 공개 매개 변수 : 통신 개체들은 큰 소수 p와 생성원 g를 "g가 mod p의 원시근(primitive)"으로 협정한다. 여기서, p와 g는 공개값이다.
- 사용자의 무선 장치와 유선 장치인 서버는 인증기관의 공개키를 신뢰한다.
- 유선 장치인 서버 S는 공개키 인증서($CERT_S$)를 소유하고 있다.
- 사용자는 접속하고자 하는 서버에 이미 등록되어 있음을 가정한다. 여기서, 등록의 의미는 사용자의 패스워드 관련 정보를 아래와 같은 간략화된 방식으로 서버가 안전하게 저장하고 있다.
  ① 사용자(U)는 자신을 위한 패스워드를 $Upass$로 설정하고, 임의의 랜덤비트열 λ를 생성한다.
  ② U는 $E_{Upass}(\lambda)$를 계산하여, 무선 장치에 저장한다. 또한, $v = g^{H(\lambda, U, S)} \bmod p$를 계산하여 서버에게 등록한다.
  ③ 서버는 사용자 인증을 위해서, 사용자 패스워드에 관련된 정보인 v를 시스템의 패스워드 파일에 안전하게 저장한다.

본 장에서 제안되는 프로토콜은 새로운 연결 프로토콜(New Connection Protocol : NCP)와 축약된 연결 프로토콜(Reduced Connection Protocol : RCP)의 두 가지 형태의 프로토콜로 구성되어 있다: 사용자가 최초로 서버와 인증된 채널을 설정하고자 할 때는 NCP를 통하여 채널을 설정하게 된다. 그 후에 발생하는 인증된 채널 설정 시에는 서버 인증서내의 제어 윈도우 동안은 RCP를 통하여 채널을 설정한다.

3.2 새로운 연결 프로토콜(NCP)

단순화를 위해서 mod p 연산은 생략하며, 사용되는 보안 파라메터는 [4],[5]과 동일하다. NCP 프로토콜의 절차는 아래와 같다.

(1) 사용자 U는 서버 S에게 단순한 로그인 요청 메시지를 전송한다.

[LoginReq] $U \rightarrow S$: "User Login Request"

(2) S는 랜덤한 큰 수 정수 x를 선택하고, g^x 를 계산한다. 물론, x는 서버의 비밀정보가 된다. 그리고, S는 랜덤 챌린지로서 r_S를 생성하고, ServerRep1을 생성하여 U에게 전송한다.

[ServerRep1]

$$S \rightarrow U : \quad CERT_S, SIGN_S(g^x, Krls, r_S)$$

(3) U는 $CERT_S$내의 인증기관(CA)의 전자서명을 검증하고, CA에게 인증서 취소 정보를 질의한다. 만약, 두 개의 검증이 성공하면, U는 S의 공개키가 PK_S이고, 마지막 해쉬값은 $H^j(r)$임을 확인한다. U는 $CERT_S$의 제어 윈도우에 의해서 "유효성 시작 지점"과 "유효성 만료 지점"을 설정한다[4]. 그리고, U는 ServerRep1내의 서명된 부분을 PK_S로 검증하여, 유효한 g^x, $Krls$, r_S를 얻는다. 그리고, U는 자신의 패스워드를 입력하여 $E_{pass}(\lambda)$를 복호화한다. 그리고, U는 $Krls$로부터 적절한 ref를 선택하고, 랜덤 챌린지로서 r_C를 생성한다. U는 마스터 비밀 키(MS)와 세션키 (SK)를 아래와 같이 계산한다. 단, 여기서 초기 $seqN$는 0로 설정한다.

$$MS = H((g^x)^{H(\lambda, U, S)})$$
$$SK = H(MS, r_S, r_C, seqN)$$

그리고, U는 UserRep를 S에게 전송한다.

[UserRep] $U \rightarrow S$: $U, E_{SK}(ref, r_S), r_C$

(4) S는 $MS = H(v^x)$와 $SK = H(MS, r_S, r_C, seqN)$를 계산하고, UserRep에서 SK로 암호화된 부분을 복호화한다. S는 복호된 r_S와 ServerRep1에서 자신이 전송한 r_S를 비교한다. 만약 두 값이 동일하면, S는 사용자 U를 인증한다. 최종적으로, S는 Finish를 U에게 전송한다.

[Finish] $S \rightarrow U$: $E_{SK}(r_C)$

(5) U는 Finish를 복호화하여, r_C와 비교를 한다. 두 값이 동일하면, U는 생성된 세션키가 안전한 채널을 위해서 사용될 수 있음을 알게 된다. 최종적으로, U는 S에 대한 $H(PK_S)$, $H^j(r), j, MS$를 "유효성 만료 지점"까지 캐쉬한다. 여기서, MS의 경우는 U의 무선 장치내에 안전하게 저장되어야 한다.

$seqN$는 U와 S사이에 교환되는 메시지마다 매번 갱신된다. ref에 의하여, SK는 단지 $seqN$만이 갱신된 상태로 재계산될 것이다.

3.3 축약된 연결 프로토콜(RCP)

NCP가 성공적으로 수행된 후, 서버 인증서의 제어 윈도우 기간 동안 사용자는 서버와의 인증된 채널을 설정하기 위하여 RCP를 수행할 수 있다. RCP 프로토콜의 수행 절차는 아래와 같다.

(1) U는 LoginReq를 S에게 전송한다.

(2) S는 현재의 $H^i(r), i$를 계산하고, ServerReq2를 U에게 전송한다.

[ServerRep2] $S \rightarrow U$: $PK_S, H^i(r), i, Krls, r_S$

(3) U는 ServerRep2를 수신한 현재 시간이 "유효성 만료 지점"을 지났는가를 검사한다. 만약, 지나지 않았다면, U는 $H^{j-i}(H^i(r)) = H^j(r)$를 검사한다. 유효할 경우, U는 PK_S를 해쉬처리하여, 캐쉬된 값과 비교한다. 만약 참이면, U는 S의 인증서는 아직 유효하며, S의 공개키는 PK_S라고 결정하게 된다. 그리고, U는 랜덤 챌린지로서 r_C를 생성하고, $seqN$를 0으로 설정한다. U는 안전하게 저장되었던 MS를 로드하여, MS와 SK를 NCP와 동일한 방법으로 생성한다. 그 이후의 단계는 NCP와 동일하게 수행된다.

RCP 프로토콜에 의해서, 사용자는 더 이상 인증서

내의 CA 서명값의 유효성 및 취소 유무 판단을 위한 계산 비용 및 통신 비용을 낭비할 필요가 없어진다. 또한, 사용자는 MS를 계산하기 위한 지수 연산을 수행할 필요도 없다. 따라서, 사용자는 신속하고 안전하게 서버와의 인증된 채널을 설정 가능하다.

4. 보안성 평가

본 논문에서 제안된 프로토콜은 기존의 PKI를 고려하여 설계된 E2ESP에 비하여, 무선 장치에서의 통신 비용 및 계산 비용을 감소시켰으며, CRL-Agent라는 부가적인 신뢰개체를 효율적으로 제거하였다. 또한, 제안된 프로토콜에서 서버내의 패스워드 파일이 공격자에게 노출되더라도, 그 패스워드 파일에는 사용자의 패스워드와 직접적인 관련이 없는 값이 포함되어 있기 때문에 오프라인 사전 공격이 어렵다. 또한, 만약 전송되는 메시지에 대한 메시지 인증이 요구되는 환경에서는, 세션키를 유도하는 함수의 간단한 변경을 통하여 메시지 인증 코드(MAC)를 위한 대칭키의 생성이 가능하다.

제안 프로토콜에서 NCP 수행 이후, 사용자의 무선 장치는 제어 윈도우 기간동안 MS를 안전하게 보관하고 있어야 하는 오버헤드를 가지고 있다. 따라서, 만약 제시된 ServerRep2를 아래와 같이 변경하면, 사용자의 무선 장치는 MS를 안전하게 보관할 필요가 없어진다.

[ServerRep2] $S \rightarrow U$: $PK_S, H^i(r), i, g^x, Krls, r_S$

하지만, 이 경우에는 사용자의 무선 장치는 RCP 프로토콜 수행시에 새로운 MS 계산을 위한 지수 연산을 부가적으로 수행해야하는 단점을 가진다.

5. 결론

본 논문에서는 유·무선 장치사이에 인증된 채널을 성공적으로 설정할 수 있는 개선된 암호학적 프로토콜을 제안하였다. 본 논문에서 제안된 프로토콜은 기제안된 프로토콜에서 불필요한 신뢰 개체를 제거하였으며, 그에 따르는 통신 및 계산 비용 또한 효율적으로 감소시켰다.

[참고문헌]

[1] C. Adams, S. Lloyd, "Understanding public-key infrastructure", Indianapolis: Macmillan Technical Publishing, 1999

[2] D. Cooper, "A more efficient use of delta-CRLs", Proceeding of 2000 IEEE Symposium on Security and Privacy, pp.190-202. May 2000

[3] J. Zhou, F. Bao, R. Deng, "An Efficient Public-Key Framework", 5th International Conference on Information and Communication Security, LNCS 2836. Oct. 2003

[4] Jong-Phil YANG, Chul Sur, Hwa-Sik Jang, Kyung-Hyune Rhee, "Practical Modification of An Efficient Public-Key Framework", The 2004 IEEE-EEE'04, pp.554-557, 2004, 4.

[5] Jong-Phil Yang, Weon Shin, Kyung Hyune Rhee, "An end-to-end authentication protocol in Wireless Application Protocol", ACISP 2001, LNCS 2119, 2001. 7.11

[6] M. Myers, R.Ankney, A. Malpani, S. Galperin, C. Adams, "X.509 Internet public key infrastructure on-line certificate status protocol(OCSP)", RFC 2560. June 1999

[7] M. Naor, K. Nissim, "Certificate revocation and certificate update", Proceedings 7th USENIX Security Symposium. January 1998

[8] P. McDaniel, S. Jamin, "Windowed certificate revocation", Proceedings of IEEE INFOCOM'2000. March 2000

[9] R. Housley, W. Ford, W. Polk, D. Solo, "Internet X.509 Public key infrastructure certificate and CRL profile", RFC 2459. January 1999

[10] S. Micali, "Efficient Certificate revocation", Technical Memo MIT/LCS/TM-542b. 1996