HomeWork #5 연습문제

정보보안 이론과 실제

9장

6. 다음 상호인증 프로토콜을 생각해 보자. 여기서 K_{AB}는 공유 대칭키이다.

트루디가 밥이 트루디가 앨리스인 것처럼 믿게 하기 위해 사용할 수 있는 두 가지 다른 공격 방법을 제시하라.

답변

한 가지 공격 방법은 트루디가 통신 내용을 스니핑하여 NONCE R과 E(R+1, K_{AB})를 저장해두고 자신이 앨리스인 척 같은 NONCE R을 밥에게 전송하고 추후에 저장해두었던 E(R+1, K_{AB})을 전송함으로써 인증 받는다.
다른 공격 방법은 트루디가 밥에게 NONCE R과 함께 인증을 요청하고 응답을 하기 전에 밥에게 NONCE R+1을 보내 E(R+1, K_AB)를 수신한다. 이 수신한 암호문을 처음 인증을 요청한 것에 대한 응답문으로 사용함으로써 밥을 속일 수 있다.

9. 다음 프로토콜을 생각해 보자. 여기서 K=h(S, R_A, R_B) 및 CLNT와 SRVR는 상수이다.

a. 앨리스는 밥을 인증하는가? 그리고 그렇게 생각하는 이유는 무엇인가?
b. 밥은 앨리스를 인증하는가? 그리고 그렇게 생각하는 이유는 무엇인가?

답변

a. 앨리스는 밥을 인증한다. 그 이유는 K를 얻기 위해서는 {S}_밥을 해독할 수 있어야 한다. {S}_밥을 해독할 수 있는 사람은 밥밖에 없다.
b. 밥은 앨리스를 인증할 수 없다. 그 이유는 앨리스가 보내는 메시지들은 굳이 앨리스가 아닌 누구라도 만들 수 있는 내용들이기 때문이다.

10. 다음 프로토콜을 생각해 보자. 여기서 K=h(S, R_A, R_B) 및 CLNT와 SRVR는 상수이다.

a. 앨리스는 밥을 인증하는가? 그리고 그렇게 생각하는 이유는 무엇인가?
b. 밥은 앨리스를 인증하는가? 그리고 그렇게 생각하는 이유는 무엇인가?

답변

a. 앨리스는 밥을 인증한다. 역시 K를 얻기 위해서는 S가 필요한데 이 S는 E(S, K_{AB})를 복호화할 수 있어야 한다. 앨리스와 밥 사이에 K_{AB}를 안전하게 공유하고 있다는 가정 하에 S를 얻을 수 있는 것은 밥밖에 없기 때문이다.

b. 밥 또한 앨리스를 인증한다. 그 이유는 S를 암호화하기 위해서는 앨리스와 밥 사이에 미리 공유된 K_{AB}가 필요하다. 만약 트루디가 임의의 키를 사용할 경우 밥은 자신이 해독한 S를 사용해 만든 해시값과 K가 다르다는 것을 알 수 있게 된다.

14. 트루디가 어떠한 관찰자(앨리스 또는 밥을 포함해)에게든 앨리스와 밥 사이의 유효한 메시지처럼 보이는 메시지를 구성할 수 있다면 이 프로토콜은 그럴듯한 부인권(plausible deniability)을 제공한다고 말한다. 다음 프로토콜을 생각해 보자.

여기서 K=h(R_A, R_B, S)이다. 이 프로토콜은 "그럴듯한 부인권"을 제공할까? 만약 그렇다면 이유를 설명하라. 그리고 그렇지 않다면 상호인증과 안전한 세션 키 제공을 유지한 상태에서 "그럴듯한 부인권"을 제공할 수 있도록 프로토콜을 약간 변경하라.

답변

우선 그럴듯한 부인권이란 누군가가 어떤 일을 했건 하지 않았건 그 누군가는 그 일을 하지 않았다고 주장하고 사람들을 믿게 할 수 있는 능력을 말한다. 위의 프로토콜에서는 그럴듯한 부인권을 제공하지 않는다. 그 이유는 서

명은 오로지 자신만이 할 수 있기 때문이다. 위의 프로토콜에서 그럴듯한 부인권을 제공하기 위해서는 아래의 그림과 같이 조금만 수정하면 된다.

기존의 프로토콜과 비교해 NONCE를 주고받는 과정에서의 서명 과정을 제거했다. 위의 프로토콜은 상호 인증과 안전한 세션키를 제공한다. 그리고 인증은 실패할지라도 트루디 또한 위의 메시지를 생성할 수 있다. 따라서 그럴듯한 부인권을 제공하게 된다.

18. 다음은 공유 대칭키 K_{AB}에 기초한 상호인증 프로토콜이다.

밥이 트루디가 앨리스인 것처럼 믿도록 트루디가 이 프로토콜을 공격할 수 있다는 것을 설명하라. 여기서 암호화는 안전하다고 가정한다. 트루디의 공격을 막을 수 있도록 프로토콜을 수정하라.

답변

트루디는 우선 밥에게 인증을 요청한다. 그러면 밥은 프로토콜에 따라 R_B에

대한 암호문을 요청하게 된다. 트루디는 이 암호문을 생성할 수 없으므로 이 통신은 잠깐 보류해두고 다시 밥에게 R_B를 전송하며 인증을 요청한다. 그러면 밥은 R_B에 대한 암호문을 트루디에게 건네준다. 트루디는 이 연결을 끊고 대기 중이던 통신에 대한 응답으로 좀 전에 받았던 암호문을 그대로 전송한다. 따라서 트루디는 앨리스로서 인증을 받을 수 있게 된다.

23. 이 장 도입 부분에 피아식별(IFF) 프로토콜을 설명하였다. 더 이상 중간 미그기 공격에 당하지 않도록 IFF 프로토콜을 수정하라.

답변

이 공격법은 프로토콜 과정에서 기지국을 인증하지 않기 때문에 가능하다. 즉, 상호인증이 필요하다.

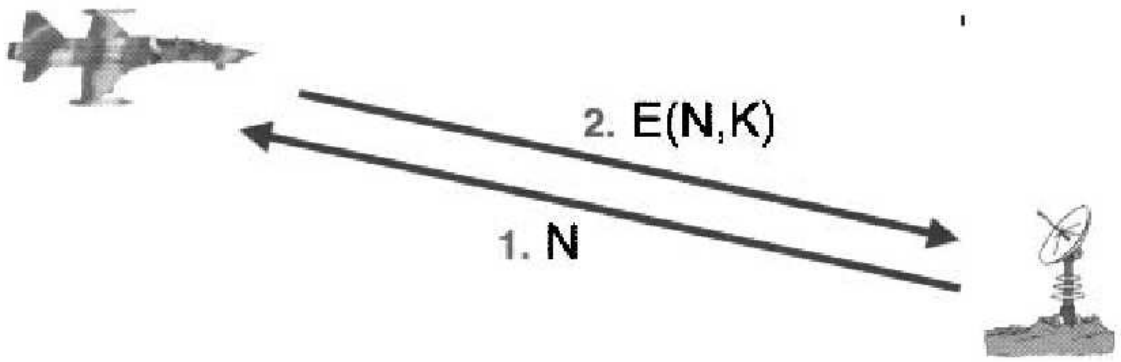

Figure 9.1: Identify Friend or Foe (IFF)

Figure 9.1은 기존의 IFF 프로토콜이다. 아래의 그림은 대칭키를 사용한 안전한 상호인증 프로토콜을 적용시킨 IFF 프로토콜이다.

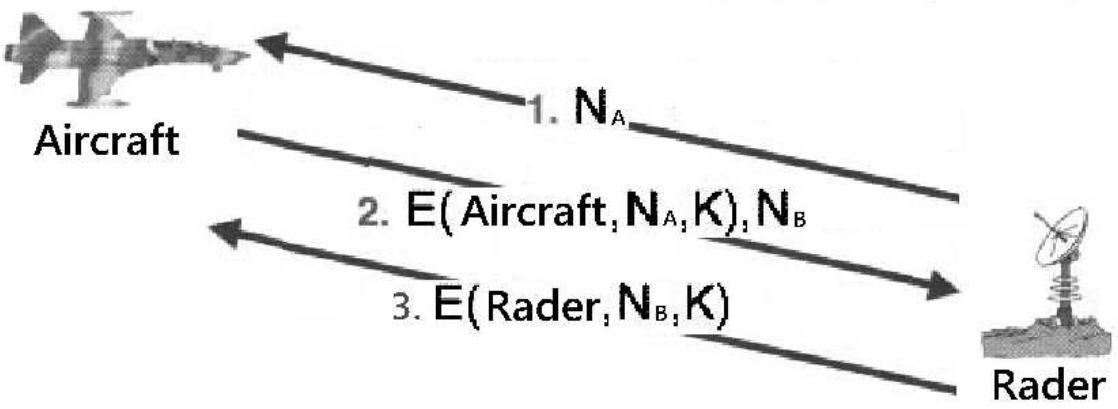

Figure 9.1: Identify Friend or Foe (IFF)

서로 제공한 NONCE를 공유된 키 K로 암호화하여 주고받으면서 인증을 시도한다. 조금 더 추가된 점은 평문에 자신의 식별 ID도 함께 추가하여 암호화한다는 점이다. 이렇게 하면 안전하게 상호인증을 할 수 있으며 중간 미그기 공격 또한 방지할 수 있다.

10장

1. 10.4.2절에서 설명한 커베로스의 절차를 보면 "밥으로의 티켓"이 밥에게 직접 전달되지 않고 앨리스에게 보내졌는데 앨리스는 이를 다시 밥에게 전달해야만 한다. 커베로스는 왜 이러한 방식으로 구현하였을까?

답변

개인적인 생각으로는 "밥으로의 티켓"이 밥에게 직접 전달될 경우 앨리스가 밥에게 최초로 접촉할 때까지 밥은 해당 티켓을 기억하고 있어야 하는 오버헤드가 발생한다. 하지만 앨리스에게 해당 티켓을 준다면 밥은 앨리스가 최초로 접촉할 때 티켓을 바로 복호화하여 비밀키를 얻을 수 있기 때문에 저장해 둘 필요가 없다. 이 답변은 지극히 개인적인 답변이므로 실제 의도와는 크게 다를 수 있다.

3. SSL이 IPSec에 비해 중요한 잇점은 무엇일까? 또 IPSec이 SSL보다 중요한 잇점은 무엇일까?

답변

우선 SSL이 IPSec에 비해 가지는 이점은 다음과 같다. IPSec은 네트워크 계층에 위치하기 때문에 구현하기 위해서는 운영체제를 고쳐야한다. 또한 과도하게 복잡하다. 이에 반해 SSL은 응용 계층과 전송 계층 사이에 위치하기 때문에 운영체제를 고치는 대신 응용 프로그램을 수정함으로써 구현할 수 있고 비교적 쉽게 구현할 수 있다는 장점이 있다.
다음으로 IPSec가 SSL에 비해 가지는 이점을 설명한다. IPSec가 구현되어 있다고 가정한 뒤에 SSL을 사용하면 각 응용 프로그램을 모두 수정해야하는 반면에 IPSec는 응용프로그램에 독립적으로 동작할 수 있다. 또한 응용 프로그램을 개발할 때 통신에 관한 보안적인 요소를 고려하지 않아도 된다는 장점이 있다.

13. (문제 생략)

a. 만약 와일드카드 기능이 없다면 테드는 교환권을 5개 숫자를 복원하기 위해 추정을 몇 번 해야 할까?

전체 경우의 수가 16^5이므로 1,048,576번이다. 따라서 평균적으로 1,048,576/2인 524,288번 추정해야 한다.

b. 와일드카드 기능을 사용하면 테드는 교환권의 5개 숫자를 복원하기 위해 추정을 몇 번 해야 할까?

테드가 받은 교환권은 마지막 5개 숫자를 전부 알 수 없으므로 한 자리씩 추정을 해 내는 방법을 사용해야 한다. 16진수 코드이기 때문에 한 자리에 올 수 있는 숫자는 16가지이다. 즉 평균적으로 한 자리당 8번의 추정을 해야 한다. 즉, 8^5번의 추정을 해야 한다.

c. 정직하지 않은 직원인 데이브가 어떻게 와일드카드 기능을 활용해 기계를 속일 수 있는지 설명하라.(힌트 : 데이브는 거의 1년 가까이 지나서 분실된 것으로

간주할 수 있는 교환권에 집중하고자 할 것이다. 데이브는 그러한 교환권이 없으면서도 와일드카드 기능을 이용해 현금교환을 하려 할 것이다.)

교환권의 코드 혹은 그 코드를 저장해 둔 데이터베이스에는 코드의 유효 기간을 확인하기 위한 부분이 분명 존재할 것이다. 데이브는 그 정보를 통해 1년 가까이 교환해 가지 않은 교환권을 찾아낼 수 있을 것이다. 데이브는 그런 교환권을 찾아내기 위해 우선 처음에는 마지막 5자리를 모두 *로 채워 검색한다. 그리고 yes가 나오면 *을 한자리씩 임의로 체크하며 yes가 나오면 앞의 과정을 반복하는 방법을 사용할 수 있다. 이 방법은 앞의 10자리와 일치하는 고액 교환권이 한 장이라고 가정했을 때 평균적으로 8^5번 즉, 32,768번의 시도로 고액 교환권 코드를 찾아낼 수 있다.

d. 데이브의 위험부담은 무엇일까? 즉, 현 시스템에서 데이브를 어떻게 발각할 수 있을까?

현 시스템에서는 그저 비정상적으로 와일드카드 기능을 많이 사용한 모든 사람을 의심하는 방법밖에 없다고 생각한다. 그 이유는 앞서 나온 불쌍한 테드의 경우와 데이브의 악의적인 행위는 전산 처리 과정을 봐서는 크게 다를 것이 없기 때문이다.

e. 기계가 자동으로 스캔할 수 없는 교환권을 직원들이 안전하고 효율적으로 처리할 수 있으면서도 데이브가 기계를 속이는 것이 불가능하도록(적어도 매우 어렵도록) 시스템을 개선하라.

지금 데이브는 고액 교환권이 없으면서도 분실되었을 확률이 있는 코드만을 훔치려고 하는 것이다. 따라서 스캔이 안 되더라도 교환권이 있어야만 현금이 일치되도록 시스템을 개선한다. 따라서 데이브가 목적을 달성하기 위해서는 데이터베이스에 남은 날짜가 거의 1년 전이면서 발급 날짜가 그 날짜와 같고 뒤의 5자리가 판별이 안 되는 교환권을 가지고 있어야 한다.