Kerberos상에서의 키 로밍 서비스에 관한 연구

이용호[0] 이임영
순천향대학교 정보기술공학부
abysskey@lycos.co.kr[0], imylee@sch.ac.kr

A Study on the Key Roaming Service in Kerberos

Yong-Ho Lee[0] Im-Yeong Lee
Division of Information Technology Eng. Soonchunhyang University

요 약

Kerberos는 MIT에서 Athena 프로젝트의 일환으로 개발된 인증 서비스로써 관용 암호 기법만을 사용하고 있다. 공개된 네트워크 환경에서의 허가되지 않은 사용자의 접속을 방지하기 위한 대안으로 모든 사용자의 패스워드를 중앙집중식 데이터베이스에 저장하는 인증 서버를 이용하고 있다. 이 인증 서버는 각 서버와 고유의 비밀키를 공유하게 된다. 이렇게 공유된 비밀키를 이용하여 사용자들에게 서비스를 제공하게 된다. 현재 Kerberos는 버전 5까지 개발되었고, IETF CAT Working Group에서 개발한 PKINIT 기반의 인증 서비스를 접목하려는 노력이 시도되고 있다.

키 로밍 서비스란 공개키 기반 구조하에서 사용되는 공개키, 개인키 쌍 중에서 개인키에 대한 안전한 관리를 목적으로 개발된 서비스로써 시간과 공간에 제약받지 않고, 패스워드만을 가지고 자신의 개인키에 접근할 수 있는 기법을 제공한다. 본 논문에서는 사용자의 개인키에 대한 키 로밍 서비스를 제공하는 Kerberos 시스템을 제안한다. 키 로밍 서비스를 제공하는데 있어서 기존에 Keberos가 가지고 있는 모든 서비스는 그대로 유지되고, 키 로밍을 위해 추가되는 통신상의 오버헤드를 최소화하였다.

1. 서 론

네트워크로 연결되어 있는 많은 사용자들과 서비스들 사이에 통신을 하기 위해서는 항상 인증, 무결성, 기밀성 등과 같은 보안 요구사항이 필요하다. 이러한 요구사항을 위해 MIT에서 Athena 프로젝트의 일환으로 개발한 네트워크 인증 프로토콜이 Kerberos이다. 최초로 사용된 버전은 버전 4이었으며, 1994년 이를 보완한 버전 5를 발표하였다. Kerberos는 암호화와 인증 작업을 위해서 비밀키 암호 기법을 사용했다. Kerberos는 문서 작성자의 확인 작업보다는 네트워크 자원의 사용과 관련된 요청을 인증하기 위해 고안되었다. 그러므로 Kerberos는 문서에 대한 제 3자 인증을 위한 장치가 존재하지 않는다. Kerberos 시스템에서는 각 네트워크상에서 키 관리를 포함한 제반 관리 기능을 중앙 집중식으로 처리하는 Kerberos 인증 서버를 가지고 있다. 인증 서버는 모든 사용자의 비밀키 정보들과 각 사용자들의 신분을 확인하기 위한 정보 그리고 각자를 확인하고 싶어하는 개인과 다른 영역의 인증 서버들을 위해 세션키를 공급하기 위한 정보 등을 관리하는 데이터베이스를 가지고 있다. Kerberos 시스템은 제 3자, 즉 Kerberos 인증 서버에 대한 신뢰를 요구한다. 만약 인증 서버의 보안 정보가 누설된다면 Kerberos 시스템 전체에 대한 신뢰성이 깨지게 된다. 최근들어 Kerberos가 기밀 유지를 제 3자에게 전적으로 의지해야하는 단점을 보완하기 위해 공개키 암호 시스템과의 접목이 시도되고 있다. Kerberos 시스템이 자신의 관리 영역 내에서만 사용된다면 커다란 문제가 없지만, 상이한 영역들 간에 사용되기 위해서는 좀 더 강력한 기능과 특성을 갖춘 공개키 시스템이 Kerberos 시스템보다 더 적당할 것이다.[1][3][4] 이러한 배경에 의해서 IETF(Internet Engineering Task Force) CAT Working Group에서는 Kerberos 시스템 두 영역간의 인증 서버를 공개키로 상호 서비스해 주는 PKINIT(Public Key Cryptography for Initial Authentication)/PKCROSS(Public Key Cryptography for Cross-Realm Authentication)을 개발하였다.[5][6]

공개키 암호 기술에서 가장 중요한 것은 사용되는 공개키와 개인키의 관리라 할 수 있다. 공개키는 키 쌍 생성시 인증기관에 의해 안전하게 관리되는 반면 개인키는 사용자에 의해서 관리된다. 최근 이슈가 되는 있는 개인키 관리 기술로 키 로밍 기술이 있다. 키 로밍 기술에서는 사용자와 제 3의 기관간에 신뢰 관계를 가지고 있어야 한다. 사용자는 암기할 수 있는 패스워드를 이용하여 신뢰된 제 3의 기관과 안전한 채널을 형성하여 개인키를 전송받는 구조를 가지고 있다.[2][8]

본 논문에서는 기존 Kerberos 시스템에 대해 알아보고, 키 로밍 서비스를 제공하는 Kerberos 시스템을 제안한다. 키 로밍 서비스를 제공함에 있어서 기존의 서비스들과는 독립적으로 운용되며, 추가되는 통신 오버헤드는 최소화하였다.

본 논문의 구성은 다음과 같다. 2장에서 기존 방식으로써 Kerberos 버전 5에 대해 알아보고, 3장에서 제안 방식을 소개한다. 마지막으로 4장에서 결론을 맺도록 한다.

2. Kerberos 버전 5

Kerberos 버전 5 시스템은 4개의 개체를 가지며, 3개의 단계로 구성된다. 4개의 개체는 클라이언트 C, 인증 서버 AS, 티켓-승인 서버 TGS 그리고 서버(Server) 이고, 3개의 단계는 인증 서비스 교환 단계, 티켓-승인 서비스 교환 단계, 클라이언트-서버 인증 교환 단계이다. 각각은 다음과 같은 역할을 수행한다.[7]

- 4개의 개체
  - C(Client) : 특정 S에서 서비스를 얻고자하는 개체로써 AS, TGS와 차례로 통신을 수행해서 S의 서비스를 이용할 수 있는 권한을 획득하게 된다.
  - AS(Authentication Server) : Kerberos 시스템을 구성하는 개체로써 C에게 접근 권한을 제공하는 기능을 수행한다.
  - TGS(Ticket-Granting Server) : Kerberos 시스템을 구성하는 개체로써 S의 서비스를 이용할 수 있는 권한을 제공하는 기능을 수행한다.
  - S(Server) : 인증되고, 권한을 획득한 C에게 해당하는 서비스를 제공하는 기능을 수행한다.
- 3개의 단계
  - 인증 서비스 교환 단계 : C와 AS간에 수행되는 단계로써 인증 과정과 티켓-승인 티켓 요청 그리고 발급 과정으로 이루어진다.

- 티켓-승인 서비스 교환 단계 : C와 TGS간에 수행되는 단계로써 서비스-승인 티켓 요청과 발급 과정으로 이루어진다.
- 클라이언트-서버 인증 교환 단계 : C와 S간에 수행되는 단계로써 서비스 요청과 서비스 제공 과정으로 이루어진다.

2.1 시스템 계수

다음은 Kerberos 버전 5에서 사용하는 시스템 계수에 대해 설명한다.

- *Options* : 티켓-승인 티켓이 포함해야 하는 플래그들로 구성. 플래그는 서비스들을 나타내는 것으로써 사용자는 자신이 필요한 서비스를 선택 가능
- ID_c : C의 식별자
- $Realm_c$: C의 영역을 나타내는 식별자
- *Times* : AS가 C에게 발급하는 티켓에 설정한 시간
- *Nonce* : 전송되는 메시지가 새로운 것이고, 제 3자에 의해 재전송된 것이 아님을 보증하기 위해 사용되는 난수 값
- K_c : 사용자의 패스워드를 기반으로 한 암호화키
- $Realm_{tgs}$: TGS의 영역을 나타내는 식별자
- ID_{tgs} : TGS의 식별자
- $Ticket_{tgs}$: 티켓-승인 티켓
- $Ticket_{tgs}=E_{Ktgs}(Flags||K_{c,tgs}||Realm_c||ID_c||AD_c||Times)$
- K_{tgs} : TGS의 패스워드를 기반으로 한 암호화키
- *Flags* : *Option*에 의해 설정된 플래그 값
- $K_{c,tgs}$: C와 TGS간의 공유키, AS에게 생성
- AD_c : C의 네트워크 주소
- ID_v : S의 식별자
- $Authenticator_{c1}$: 인증자, 티켓 사용자의 객체 인증을 수행. 티켓 제출자가 티켓이 발행된 C와 같음을 보증
- $Authenticator_{c1} = E_{Kc,tgs}(ID_c||Realm_c||TS_1)$
- $Ticket_v$: 서비스-승인 티켓
- $Ticket_v=E_{Kv}(Flags||K_{c,v}||Realm_c||ID_c||AD_c||Times)$
- K_v : S의 패스워드를 기반으로 한 암호화키
- $K_{c,v}$: C와 S간의 공유키, TGS에서 생성
- $Authenticatior_{c2}$: 인증자, 티켓 사용자의 객체 인증을 수행. 티켓 제출자가 티켓이 발행된 C와 같음을 보증
- $Authenticator_{c2}=E_{Kc,v}(ID_c||Realm_c||TS_2||Subkey||Seq\#)$
- *Subkey* : C가 S와 공유하는 서브키, 사용자가 생성
- *Seq#* : C가 S를 인증할 경우 S가 사용할 시작 순서번호를 지정하는 선택적인 필드, 메시지 재전송을 방지

2.2 프로토콜

다음은 Kerberos 버전 5에서 이루어지는 프로토콜에 대해 설명한다.

1) 인증 서비스 교환

C가 AS에게 티켓-승인 티켓을 얻기 위해 수행되는 과정으로 다음과 같이 이루어진다.

① C는 AS에게 아래의 정보를 전송한다.

$Options || ID_c || Realm_c || Times || Nonce_1$

② AS는 전송 정보를 이용하여 C에게 아래의 정보를 전송한다.

$E_{Kc}(K_{c,tgs} || Times || Nonce_1 || Realm_{tgs} || ID_{tgs}) || Realm_c || ID_c || Ticket_{tgs}$

2) 티켓-승인 서비스 교환

C가 TGS에게 서비스-승인 티켓을 얻기 위해 수행되는 과정으로 다음과 같이 이루어진다.

③ C는 TGS에게 다음 정보를 전송한다.

$Options || ID_s || Times || Nonce_2 || Ticket_{tgs} || Authenticator_{c1}$

④ TGS는 전송된 정보를 이용하여 다음 정보를 전송한다.

$Realm_c || ID_c || Ticket_v || E_{Kc,tgs}(K_{c,v} || Times || Nonce_2 || Realm_v || ID_v)$

3) 클라이언트-서버 인증 교환

C가 S에게 서비스를 얻기 위해 수행되는 과정으로 다음과 같이 이루어진다.

⑤ C는 S에게 아래의 정보를 전송한다.

$Options || Ticket_V || Authenticator_{c2}$

⑥ S는 전송된 정보를 이용하여 C에게 다음 정보를 전송한다.

$E_{Kc,v}(TS_5 || Subkey || Seq\#)$

Kerberos 시스템은 상기와 같이 총 6번의 통신을 수행하여 C와 S간에 비밀키를 공유하게 된다. C는 이 비밀키를 이용하여 S에게 서비스를 제공받게 된다.

3. 제안 방식

이번 장에서는 Kerberos 버전 5 프로토콜을 기초하여 키 로밍 서비스를 제공할 수 있는 새로운 프로토콜을 소개한다.

3.1 시스템 계수

여기서는 Kerberos 버전 5의 시스템에서 사용하는 것을 그대로 사용하고, 새로이 추가되는 시스템 계수만을 소개한다.

- p : 큰 소수
- g : Z_p상의 원시원소
- pw_c : C의 패스워드
- x_c : C의 개인키
- x_{AS} : AS의 개인키
- y_{AS} : AS의 공개키(단, $y_{AS} = g^{x_{AS}}$)
- *EPK* : C의 패스워드로 암호화된 C의 개인키
- hp : 패스워드 해쉬값($=h(g^{pw_c})$)
- hpv : 패스워드 검증자($=g^{hp}$)
- r, n : 랜덤 수
- $h(\ )$: 안전한 일방향 해쉬 함수

3.2 프로토콜

다음은 키 로밍 서비스를 제공하는 Kerberos 시스템에서 이루어지는 프로토콜에 대해 설명한다.

1) 키 로밍 서비스 제공을 위한 플래그 추가

Kerberos 시스템을 이용하는 C는 Options으로써 다양한 플래그들을 설정할 수 있다. Kerberos 시스템이 제공하는 플래그에 다음과 같은 플래그를 추가한다.

- KEY-ROAMING : C가 KEY-ROAMING 플래그를 선택하면 AS와 TGS는 각각 발행하는 티켓에 C가 키 로밍 서비스를 제공받을 수 있는 권한이 있음을 가리키는 기능을 설정한다. 이 티켓을 받은 S는 C에게 전자서명을 요구할 수 있게 된다.

2) 키 로밍 서비스 제공을 위한 수행 단계

이 단계는 인증 서비스 교환 단계와 같이 이루어진다. 개인키 정보 위탁은 초기에 한번만 수행되고, 그 이후에는 KEY-ROAMING 플래그만을 설정한다. 키 로밍 서비스를 제공하기

위해서 Kerberos 버전 5의 인증 서비스 교환 단계는 다음과 같이 변경된다. 그리고 티켓-승인 서비스 교환 단계와 클라이언트-서버 인증 교환 단계는 Kerberos 버전 5와 동일하게 수행된다.

① C에 의해서 수행되는 과정

$$EPK = E_{pwc}(x_c)$$

$$hp = h(g^{pwc} \bmod p),\ hpv = g^{hp} \bmod p$$

$$KR\_req = E_{Kc}(ID_c \| AD_c \| EPK \| hpv)$$

② C는 아래의 정보를 AS에게 전송한다.

$$Options \| ID_c \| Realm_c \| Times \| Nonce_1 \| KR\_req$$

③ AS에 의해서 수행되는 과정

KR_req을 확인하고 ID_c, AD_c, EPK, hpv을 안전하게 보관한다.

④ AS는 다음 정보를 C에게 전송한다.

$$E_{Kc}(K_{c,tgs} \| Times \| Nonce_1 \| Realm_{tgs} \| ID_{tgs}) \| Realm_c \| ID_c \| Ticket_{tgs}$$

3) 키 로밍을 위한 다운로드 프로토콜

이 단계는 C가 AS에게 키 로밍 서비스 제공을 위한 수행 단계를 거친 후 실제로 키 로밍 서비스를 제공받는 경우 수행되는 과정을 소개한다.

(1) C에서 수행되는 과정

① 사용자는 AS에서 키 로밍을 위한 S/W를 다운로드하여 설치 및 실행한다. 그리고 자신의 ID_c와 pw_c를 입력한다.

② S/W는 입력값을 이용하여 패스워드 검증자를 계산한다.

$$hp = h(g^{pwc} \bmod p),\ hpv = g^{hp} \bmod p \quad (식1)$$

③ AS의 공개키를 이용하여 PDH를 계산하고, 다음 정보를 AS에게 전송한다.

$$PDH = y_{AS}{}^{hp} \bmod p$$

$$ID_c \| E_{PDH}(r) \| h(r \| hpv) \quad (식2)$$

(2) AS에서 수행되는 과정

① AS는 전송된 ID_c에 해당하는 사용자의 hpv를 획득하고, 자신의 개인키를 이용해 PDH를 계산한다.

$$PDH = hpv^{xAS} \bmod p \quad (식3)$$

② PDH를 이용하여 해쉬값을 다음과 같이 비교 검증한다.

$$r' = D_{PDH}(E_{PDH}(r))$$

$$h(r \| hpv) \stackrel{?}{=} h(r' \| hpv) \quad (식4)$$

③ 랜덤 수 n을 선택해서 Pr과 세션키 SK를 계산한다.

$$Pr = PDH \oplus hpv \oplus r$$

$$SK = h(PDH \| n \| r) \quad (식5)$$

④ 다음 정보 C에게 전송한다.

$$E_{Pr}(n) \| h(n \| r) \| E_{SK}(EPK) \quad (식6)$$

(3) C에서 수행되는 과정

① PDH와 hpv 그리고 r을 이용하여 Pr을 계산한다.

$$Pr = PDH \oplus hpv \oplus r \quad (식7)$$

② Pr을 이용하여 n을 계산하고, 해쉬값을 구성하여 전송된 해쉬값과 비교 검증한다.

$$n' = D_{Pr}(E_{Pr}(n))$$

$$h(n \| r) \stackrel{?}{=} h(n' \| r) \quad (식8)$$

③ PDH와 n 그리고 r을 이용하여 세션키 SK를 계산하고, 이것을 이용하여 EPK를 계산한다.

$$SK = h(PDH \| n \| r)$$

$$EPK = D_{SK}(E_{SK}(EPK)) \quad (식9)$$

④ pw_c를 이용하여 사용자의 개인키 x_u를 계산한다.

$$x_u = D_{PWc}(EPK) \quad (식10)$$

3.3 제안 방식의 특징

제안된 방식은 Kerberos 버전 5 프로토콜에 기반하고 있다. 그러나 키 로밍 서비스를 지원하기 위해 추가되는 과정들은 기존에 제공하고 있었던 서비스와는 별도로 다음과 같다.

- 키 로밍을 위해 C에서 수행하는 과정
- 위 수행한 결과를 AS에게 위탁하는 과정
- 위탁된 개인키 정보를 받아오는 다운로드 프로토콜

C는 전자서명 등 개인키가 필요할 경우 키 로밍을 수행하기 위해 AS에서 제공하는 S/W를 설치해야 한다. 이 S/W에서는 키 로밍을 위한 다운로드 프로토콜이 수행되고, 수행이 완료되면 최종적으로 개인키를 영구 삭제해야 한다. 그리고 본 논문에서 S/W의 안전성은 논의하지 않는다.

4. 결 론

분산 네트워크 환경에서 인증, 무결성, 기밀성 등과 같은 보안 서비스를 제공하기 위해서 많은 연구들이 진행되고 있다. 이 중에서 상기 보안 요구사항을 만족하면서 사용자들에게 서비스를 제공하는 네트워크 인증 시스템인 Kerberos에 대해 알아보았고, 현재 공개키 암호 시스템에서 사용되는 개인키에 대한 관리 기술로써 주목받고 있는 키 로밍 서비스에 대해 알아보았다. 초기 Kerberos는 대칭키 암호 시스템에 기반하여 운용되었으나 현재는 공개키 암호 시스템을 접목하려는 연구가 시도되고 있다. 이렇게 공개키 암호 시스템이 사용되면서 Kerberos에서도 키 로밍 서비스가 가능하게 되었다. 이러한 배경을 바탕으로 우리는 키 로밍 서비스를 제공하는 Kerberos 시스템을 제안했다.

향후 영역에 구분없이 Kerberos 시스템을 이용하여 키 로밍 서비스를 제공받을 수 있도록 하는 방법에 대한 연구가 진행되어야 하리라 생각된다.

참 고 문 헌

[1] 최용락, 소우영, 이재광, 이임영, "*컴퓨터 통신 보안*", 도서출판 그린, 2001.

[2] 이용호, 이임영, "개선된 패스워드 기반 키 로밍 프로토콜", 한국통신학회 하계종합학술발표회, pp.592, 2002.

[3] 신광철, 정일용, 정진욱, "PKINIT기반의 Kerberos 인증과 키 교환에 관한 연구", 정보처리학회논문지, 제9-C권, 제3호, 2002.

[4] 신광철, 정진욱, "네트웍 환경에서 안전한 Kerberos 인증 메커니즘에 관한 연구", 정보보호학회논문지, 제12권, 제2호, 2002.

[5] B.Tung, C.Neuman, M.Hur, A.Medvinsky, S. Medvinsky, J.Wray and J.Trostle, "Public Key Cryptography for Initial Authentication in Kerberos", draft-ietf-cat- kerberos -pk-init-14.txt.

[6] B.Tung, C.Neuman, M.Hur, A.Medvinsky, S.Medvinsky, "Public Key Cryptography for Cross-Realm Authentication in Kerberos", draft-ietf-cat-kerberos-pk -cross-07.txt.

[7] J.Kohl and C.Neuman, "The Kerberos Network Authentication Service (V5)", RFC 1510, 1993.

[8] Seungjoo Kim, Byungchun Kim and Sungjun Park, "Comments on password-based private key download protocol of NDSS'99", Electronics Letters 35(22), IEE Press, pp.1937-1938, 1999.