제18회 한국정보처리학회 추계학술발표대회 논문집 제9권 제2호 (2002.11)

키 복구를 제공하는 Kerberos에 관한 연구

이용호, 이임영
순천향대학교 정보기술공학부
e-mail:abysskey@lycos.co.kr

A Study on the Kerberos support Key Recovery

Yong-Ho Lee, Im-Yeong Lee
Division of Information Technology Eng. Soonchunhyang University

요 약

Kerberos 시스템은 MIT에서 Athena 프로젝트의 일환으로 개발된 인증 시스템으로 인증되지 않은 사용자들의 불법 접속을 방지하고, 인증된 사용자에게 서비스 이용 권한을 제공하려는 목적을 가지고 있다. 이 시스템은 인증 서버(AS; Authentication Server)와 티켓 승인 서버(TGS; Ticket Grantig Server) 그리고 서비스 제공 서버(SPS; Service Provide Server)로 구성된다. 사용자는 서비스 제공 서버와 관용 암호 알고리즘에 사용할 비밀키를 공유하고, 이를 이용하여 서비스를 제공받는다.

본 논문에서는 키 복구 기능과 향상된 인증 서비스를 제공하는 Kerberos 시스템을 제안한다. 제안하는 시스템은 사용자와 서비스 제공 서버간에 키 분배를 수행하고, 서비스 제공 서버는 분배된 키를 이용하여 서비스를 제공한다. 이때, 분배된 키는 사용자의 요청에 따라 인증 서버와 티켓-승인 서버에 의해 복구될 수 있다.

1. 서론

Kerberos 시스템은 개방형 시스템인 네트워크 환경에서 사용자를 인증하고 서비스를 제공하는 서버에 대한 접근 제어를 수행하기 위해서 개발되었다. Kerberos 시스템은 인증 서버와 서비스를 제공하는 서버에 대한 접근을 제어하는 티켓 발급 서버를 가지고 있다. 인증 서버는 모든 사용자의 식별자와 패스워드를 저장하고 있으며, 패스워드를 기반으로 생성한 비밀키를 각 사용자와 공유하고 있다. 이를 기반하에서 인증, 무결성, 기밀성을 제공하고 있다. 그러나 부인봉쇄는 제공하지 못하고 있다. 본 논문에서는 공개키 암호 기법과 키 복구 기술을 적절히 적용하여 기존 Kerberos 버전 5가 가지고 있었던 문제점을 개선하는데 중점을 두었다.[1][3][4]

본 고에서는 2장에서 기존 방식의 문제점과 요구사항을 알아보고, 3장에서 제안 방식을 소개한다. 마지막으로 4장에서 결론을 맺는다.

2. 기존 방식 소개 및 분석

이 장에서는 기존 방식으로써 Kerberos 버전 5에 대해 소개하고 분석한다.[1][2]

2.1 Kerberos 버전 5의 참여 개체 및 구성

Kerberos 시스템은 4개의 개체를 가지며, 3개의 단계로 구성된다. 4개의 개체는 사용자, 인증 서버, 티켓 승인 서버 그리고 서비스 제공 서버이며, 3개의 단계는 인증 서비스 교환 단계, 티켓-승인 서비스 교환 단계, 클라이언트-서버 인증 교환 단계이다. Kerberos 버전 5 프로토콜은 다음과 같다.

1) 인증 서비스 교환 단계

① A는 AS에게 아래의 정보를 전송한다.

$$ID_a \parallel Time \parallel Nonce_1$$

② AS는 전송 정보를 이용하여 $Ticket_{tgs}$을 구성하고, A에게 아래의 정보를 전송한다.

$$Ticket_{tgs}=E_{Ktgs}(K_{a,tgs} \parallel ID_a \parallel AD_a \parallel Time)$$

$$E_{Kc}(K_{a,tgs} \parallel Time \parallel Nonce_1 \parallel ID_{tgs}) \parallel ID_a \parallel Ticket_{tgs}$$

2) 티켓-승인 서비스 교환 단계

③ A는 $Auth_1$을 계산하고, TGS에게 다음 정보를 전송한다.

$$Auth_1=E_{Ka,tgs}(ID_a \parallel TS_1)$$

$$ID_s \parallel Time \parallel Nonce_2 \parallel Ticket_{tgs} \parallel Auth_1$$

④ TGS는 전송된 정보를 이용하여 $Ticket_v$을 계산

하고, 다음 정보를 A에게 전송한다.

$$Ticket_v=E_{Kv}(K_{a,v} \| ID_a \| AD_c \| Time)$$

$$ID_c \| Ticket_v \| E_{Ka,tgs}(K_{a,v} \| Time \| Nonce_2 \| ID_s)$$

3) 클라이언트-서버 인증 교환

⑤ A는 $Auth_2$을 계산하고, 아래의 정보를 S에게 전송한다.

$$Auth_2=E_{Ka,v}(ID_a \| TS_2 \| Subkey \| Seq)$$

$$Ticket_v \| Auth_2$$

⑥ S는 전송된 정보를 이용하여 A에게 다음 정보를 전송한다.

$$E_{Ka,v}(TS_2 \| Subkey \| Seq)$$

Kerberos 시스템은 상기와 같이 총 6번의 통신을 수행하여 A와 S간에 비밀키를 공유하고, 이것을 이용하여 기밀성을 가지는 암호화 통신을 수행하게 된다.

2.2 Kerberos 버전 5의 분석 및 요구사항 도출

여기서는 Kerberos 버전 5 프로토콜을 문제점을 분석하고 이를 통해 얻어진 사항을 기반으로 Kerberos가 가져야 하는 요구사항을 도출한다.

1) 문제점 분석

- Kerberos 버전 5에서는 모든 비밀 통신에 관용 암호 알고리즘만을 이용한다. 따라서 인증 서버에서는 많은 비밀키를 보관하고 관리해야 한다. 이것은 이용하려는 사용자가 증가함에 그 비중이 더해지게 된다.[5]
- 인증 서버는 사용자의 식별자와 패스워드를 중앙 집중식으로 관리하고, 패스워드를 기반으로 비밀키를 생성한다. Kerberos에서 사용되는 패스워드는 사전공격이 취약하다는 문제점이 제기되었다. 따라서 이를 기반으로 생성한 비밀키 또한 취약하게 된다.[6]
- 인증 서버와 티켓 발급 서버는 각각 독립적으로 키를 생성하고 분배하는 기능을 가지고 있다. 따라서 사용자와 서비스 제공 서버 사이의 모든 통신은 인증 서버와 티켓 발급 서버에 의해 각각 노출된다.

2) 요구사항 및 추가사항 도출

- 암호 알고리즘 - 관용 암호의 사용에 따른 비밀키의 부담은 공개키 암호로 해결할 수 있다. 그러나 공개키 암호의 사용은 효율성이 떨어진다는 문제점을 가지고 있다. 따라서 프로토콜 상에서 필요한 부분에 대해서만 적절히 적용해야 한다.
- 패스워드 기반 비밀키 생성의 문제점 - 패스워드 기반 비밀키 생성 부분을 모두 공개키 암호를 이용한 방법으로 변경하면 해결할 수 있다. 를 이용하면 이 문제점 또한 쉽게 해결할 수 있다.
- 사용자와 SPS간의 비밀 통신에 대한 불법적인 키 복구 및 메시지 복구 - 사용자와 SPS간의 통신에 사용되는 비밀키는 인증 서버와 티켓 발급 서버가 생성할 수 없는 방법으로 생성해야 한다. 이와는 다르게 키 유실과 같은 상황이 발생하면 중요한 데이터에 대한 접근이 불가능해진다는 문제점을 가지고 있다. 이렇게 상반되는 두 가지 문제는 키 복구 기술을 이용하여 해결할 수 있다. 키 복구 기술을 이용하면 사용자와 SPS간의 비밀 통신은 보장하면서, 특수한 상황이 발생할 경우 인증 서버와 티켓 발행 서버가 모두 참여하여 세션키를 복구할 수 있도록 구성할 수 있다.

3. 제안 방식

다음은 본 고에서 제안하는 키 복구 기능을 가지는 개선된 Kerberos 시스템에 대해 살펴본다.

3.1 구성 개체

다음은 제안 방식을 구성하는 개체들에 대한 설명이다.

- A(Alice) : 임의의 사용자로서 SPS에서 제공하는 서비스를 제공받기 원하는 개체이다. 서비스를 제공받기 위해서 AS와 TGS를 통하여 인증을 수행하고, SPS와 세션키를 교환하게 된다. 또한 세션키를 분식할 경우 AS에게 키 복구 요청을 할 수 있다.
- AS(Authentication Server) : 사용자의 인증을 수행하고, TGS의 접근 권한을 제공하는 개체로써 모든 서버들의 공개정보를 저장하고 안전하게 관리하는 역할을 수행하게 된다. 인증된 사용자에게 티켓-승인-티켓을 발급한다. 또한 사용자의 키 복구 요청에 따라 해당하는 세션키를 복구하는 기능을 가지고 있다.
- TGS(Ticket Granting Server) : AS로부터 인증된 사용자에게 SPS의 접근 권한을 제공하는 개체로써 AS의 보조 역할을 수행하고 있다. 사용자에게

서비스-승인 티켓을 발급한다. 또한 AS의 키 복구 지원 요청에 따라 키 복구를 지원하는 역할을 수행하고 있다.

- SPS(Service Provide Server) : TGS에 의해 승인된 사용자에게 서비스를 제공하는 개체로써 사용자와 키 교환을 수행하게 된다.

3.2 시스템 계수

다음은 제안 방식에서 표기되는 시스템 계수에 대해 알아본다.

- # : 참여 개체(A, AS, TGS, SPS)를 가리키는 지시자로써 개체 이름의 소문자로 표시
- $ID_{\#}$: #에 해당하는 개체의 식별자
- T : 티켓-승인 티켓의 유효시간
- N_1, N_2 : 제 3자에 의해 재전송된 것이 아님을 알리기 위해 사용되는 랜덤수
- $(x\_\#, y\_\#)$: # 해당 개체의 비밀키와 공개키
- k_{a_tgs} : AS에게 생성하는 일회용 비밀키로써 A와 TGS간에 암호화 통신에 사용
- AD_a : A의 네트워크 주소
- E() : 공개키 암호 알고리즘으로 생성된 암호문
- SE() : 대칭키 암호 알고리즘으로 생성된 암호문
- Sig() : 공개키 암호 알고리즘으로 생성된 서명문
- h() : 안전한 일방향 해쉬함수로 생성된 해쉬값
- TS_1, TS_2 : Time-Stamp
- Seq : Alice와 SPS간에 키 교환을 할 경우 서버가 사용할 시작 번호를 지정하는 필드로써 메시지 재전송을 방지하는 기능을 수행
- ssk : Alice와 SPS간에 생성되는 세션키로써 키 복구의 대상이 된다.

3.3 프로토콜

3.3.1 식별 및 티켓-승인 티켓 발급 단계

다음은 Alice와 AS간에 이루어지는 인증과 티켓-승인 티켓 발급 과정에 대해 설명한다.

1) Alice는 AS에게 다음 메시지를 전송한다.

$$ID_a \| T \| N_1 \quad (식1)$$

2) AS는 Alice를 식별하고, Alice에게 티켓-승인 티켓 및 관련 정보를 발급한다.

$$d = g^{x\_tgs * x\_as}$$

$$m_1 = k_{a\_tgs} \| T \| N_1 \| ID_{tgs} \| d$$

$$Ticket_{tgs} = E_{y\_tgs}(k_{a\_tgs} \| ID_a \| ID_{tgs} \| AD_a \| T)$$

$$ID_a \| Ticket_{tgs} \| SE_{k1}(m_1) \| E_{y\_a}(Sig_{x\_as}(k_1)) \quad (식2)$$

3) Alice는 전송된 정보를 이용하여 m_1과 $Ticket_{tgs}$를 획득한다.

3.3.2 인증 및 서비스-승인 티켓 발급 단계

다음은 Alice와 TGS간에 이루어지는 인증과 서비스-승인 티켓 발급 과정에 대해 설명한다.

1) 인증을 위해 Alice는 TGS에게 다음 정보를 전송한다.

$$info_1 = ID_a \| TS_1$$

$$Auth_1 = E_{ka\_tgs}(info_1) \| Sig_{xa}(h(info_1))$$

$$ID_{sps} \| T \| N_2 \| Ticket_{tgs} \| Auth_1 \quad (식3)$$

2) TGS는 Alice에게 서비스-승인 티켓을 발급한다.

$$m_2 = T \| N_2 \| ID_{sps}$$

$$Ticket_{sps} = E_{y\_sps}(ID_a \| AD_a \| T)$$

$$ID_a \| Ticket_{sps} \| SE_{k2}(m_2) \| E_{y\_a}(Sig_{x\_tgs}(k_2)) \quad (식4)$$

3) Alice는 전송된 정보를 이용하여 m_2와 $Ticket_{sps}$를 획득한다.

3.3.3 인증 및 키 설립 단계

다음은 Alice와 SPS간에 이루어지는 인증과 키 설립 과정에 대해 설명한다.

1) 인증을 위해 Alice는 SPS에게 다음 정보를 전송한다.

$$D = d^{x\_a}$$

$$info_2 = ID_a \| TS_2 \| D \| Seq$$

$$Auth_2 = E_{y\_sps}(info_2) \| Sig_{x\_a}(h(info_2))$$

$$Ticket_{sps} \| Auth_2 \quad (식5)$$

2) SPS는 전송된 D를 이용하여 키 생성 정보 d'와 세션키 ssk를 계산한다.

$$d' = (D * y\_a^{-1})^{x\_sps}$$

ssk의 계산 과정은 다음과 같다.

$$\begin{aligned} ssk &= D * d' \\ &= g^{x\_a * x\_as * x\_tgs} * (g^{x\_a * x\_as * x\_tgs} * g^{-x\_a})^{x\_sps} \\ &= g^{x\_a * x\_as * x\_tgs} * g^{x\_as * x\_tgs * x\_sps} \\ &= g^{x\_as * x\_tgs(x\_a + x\_sps)} \end{aligned}$$

3) 키 교환을 위해 SPS는 Alice에게 다음 정보를 전송한다.

$$info_3 = TS_2 \| Seq \| d'$$
$$E_{y\_a}(info_3) \| Sig_{x\_sps}(h(info_3)) \quad (식6)$$

4) Alice는 d'를 획득하고, 이를 이용해 세션키 ssk를 계산한다.

$$ssk = D*d' = g^{x\_as*x\_tgs(x\_a + x\_sps)}$$

5) Alice와 SPS는 세션키 ssk를 공유하게 되고, Alice는 기밀성을 가지는 암호 통신을 수행해 서비스를 제공받을 수 있게 된다.

3.3.4 키 복구 단계

다음은 Alice와 SPS간에 설립된 세션키 ssk를 복구하는 과정에 대해 설명한다. 키 복구는 Alice의 요청이나 키를 복구해야만 하는 특수한 상황이 주어질 경우 수행될 수 있다. 다음은 Alice가 AS에게 복구를 요청하는 경우에 대해서 설명한다.

1) Alice의 키 복구 요청을 받은 AS는 TGS에게 다음을 전송한다.

$$value_1 = y\_a * y\_sps$$

2) TGS는 $value_1$을 다음과 같이 계산하여 AS에게 전송한다.

$$value_2 = (value_1)^{x\_tgs}$$

3) AS는 전송된 정보를 이용하여 Alice와 SPS간에 공유된 세션키 ssk를 계산한다.

$$ssk = (value2)^{x\_as}$$

ssk는 다음과 같이 계산된다.

$$(value2)^{x\_as} = (value_1)^{x\_tgs*x\_as} = (g^{x\_a + x\_sps})^{x\_tgs*x\_as} = ssk$$

4) AS는 Alice에게 ssk를 안전하게 전송한다. 그러면 Alice는 ssk를 이용하여 저장되어 있는 서비스를 복구할 수 있게된다.

3.4 비교 분석

표1은 기존 Kerberos와 제안 방식을 비교 분석한 것이다.

표1. 비교 분석표

	Kerberos V5	제안방식
인증	O	O
기밀성	O	O
무결성	O	O
부인봉쇄	X	O
서비스 정보 불법노출방지	X	O
키 복구 기능	X	O
적용암호기술	관용키 암호	관용키 암호, 공개키 암호, 해쉬 알고리즘

4. 결론

공개된 네트워크 환경에서 사용자가 특정 서비스를 제공하는 서버에 접근하여 정당하게 서비스를 제공받기 위해서는 인증, 무결성, 기밀성, 부인봉쇄 등과 같은 보안 서비스를 제공해야 한다. Kerberos 시스템을 주축으로 이에 대한 많은 연구가 진행되고 있다. 본 고에서는 Kerberos 시스템을 분석하여 문제점들을 지적하였고, 키 복구의 필요성에 대해 알아보았다. 그리고 키 복구를 지원하고 향상된 보안 서비스를 제공하는 Kerberos 시스템을 제안하였다.

본 고에서는 다중 Kerberos 환경은 고려하고 있지 않다. 향후 다중 Kerberos 환경에서 이용할 수 있는 시스템에 대한 연구가 진행되어야 하리라 생각된다.

참고문헌

[1] J.Kohl and C.Neuman, The Kerberos Network Authentication Service, RFC1510, 1993.

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

[3] Kaufman, Perlman and Speciner, *Network Security*, Prentice Hall, 1995.

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

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

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