제 18 회 한국정보처리학회 추계학술발표대회 논문집 제 9 권 제 2 호 (2002.11)

Kerberos 인증 메커니즘의 인증 관리 인터페이스 연구

정해진*, 문정훈*, 이명선*, 이희규**
*한국과학기술정보연구원 슈퍼컴퓨팅센터, **한남대학교
e-mail : hjjung@kreonet2.net

Research on implementation of Authentication administration interface in Kerberos

Jung Hae Jun*,Moon Jung Hoon*, Lee Myung Sun*, Lee Hee gu**

September, 2002

*Supercomputing Center Korea Institute of Science & Technology Information (KISTI)
** Hannam Univ.

Abstract

최근 개방형 분산 네트워크 상의 강력한 인증 시스템으로 MIT 에서 개발된 Kerberos 인증 메커니즘이 많이 쓰이고 있다. Kerberos 는 비밀키 기반의 DES 알고리즘에 기반을 둔 인증 시스템으로 보편화된 인증 방법들을 우회하는 공격과 보다 넓은 네트워크 환경에 적합한 인증 시스템을 염두에 두고 개발되었다. 많은 OS 의 최신 버전에서는 서비스 패키지를 통해 Kerberos 를 제공하고 있으나 복잡한 인증 정보 및 인증 단계를 손쉽게 사용할 수 있도록 하는 관리 툴은 많이 개발되지 않은 상태이다. 또한 분산 네트워크 서비스에서 인증 시스템을 사용할 때 여러 대의 인증 시스템으로 인증 서비스를 지원해야 하는데 그 인증 시스템을 관리할 때마다 로컬 인증 시스템에 직접 작업을 해야 하는 불편이 따르며, 또 관리 시 원격에서 접속할 때 OS 패키지 마다 지원해주는 인증 관리 인터페이스는 사용할 수가 없다. 따라서 본 논문에서는 Kerberos 의 인증 메커니즘을 소개하며 또 Kerberos 의 인증 메커니즘을 이용하여 원격에서도 인증 관리 인터페이스를 통해 인증 관리를 할 수 있는 인증 관리 인터페이스를 JAVA 로 설계하여 손쉽게 Kerberos 를 관리할 수 있는 방법을 제시한다.

1. 서론

개방형 시스템인 네트워크를 통한 클라이언트와 서버간의 서비스 교환에서 오용과 남용을 막기 위해서는 클라이언트와 서버 사이에 클라이언트가 서버에게 자신의 신원을 확인 시켜주고 동시에 서버의 신원을 클라이언트에게 확인시켜주는 과정과 메시지의 출처를 확인시켜주는 과정이 필요하다. 현재는 많은 시스템들이 개방형 네트워크상에서 다중 사용자에게 서비스를 지원해주며 또 분산 시스템 환경에서 원격으로 사용자들에게 서비스를 제공해주고 있다. 그러나 이러한 과정에서의 사용자와 서비스를 제공해주는 서버와의 신원확인은 로그인시 패스워드로 인증과정을 끝낸다. 이런 ID/Password 방식의 인증은 네트워크 상으로 패스워드와 아이디가 노출되기 쉽기 때문에 보안상 매우 취약한 구조이다. 이런 점에 고려해 볼 때 MIT 에서 개발한 Kerberos 는 개방형 분산 통신망에서 클라이언트와 서버간에 상호인증을 지원하는 대표적인 3 part 인증 시스템이다.[1] Kerberos 는 기본적으로 비밀키 암호 알고리즘인 DES 를 기반으로 하는 상호 인증 시스템으로 최근 개발된 Kerberos V5 는 보다 넓은 환경에서의 지원이 가능 하다는 특징을 가지고 있어 넓은 분야의 응용들이 많이 개발되고 있다.[3]

본 논문에서도 Kerberos V5 의 인증 메커니즘을 응용하여 인증 관리 인터페이스대해서 조사하고 새로운 인증 관리 인터페이스를 제안하고자 한다. Kerberos V5 는 강력한 인증 서비스를 제공하는 시스템으로 많이 알려져 상용 OS 에 패키지로 포함되어 있는 경우가 많은 것에 반해 클라이언트나 서버간의 인증을 관리해주는 인터페이스는 극히 드물다. 또한 개방형 분

산 통신망의 인증을 관리하기 위해서는 많은 클라이언트들의 인증에 대한 관리가 필요하기 때문에 로컬로 분산되어있는 인증 서버들을 통합하여 관리할 수 있는 인터페이스를 제안한다.

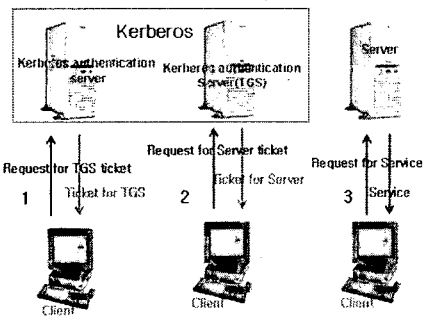
본 논문의 2 장에서는 현재 Kerberos V5 의 특징을 알아보고 분산 통신망의 로컬 인증 시스템 구축 시 인증 관리자의 관리 상태에 대한 문제점들을 알아봤다. 3 장의 1 절에서는 현재 리눅스 기반으로 만들어져 있는 인증 관리 인터페이스인 gkadmin 에 대한 조사해보고 2 절에서는 새로운 인증 관리 인터페이스인 jkadmin 을 소개한다.

2. Kerberos V5

Kerberos 는 1980 년 중반에 MIT 의 Athena 프로젝트로써 개발되었다. Kerberos 는 개방형 네트워크상에서 기존의 ID/Password 기반의 인증으로 인한 보안상 취약점을 해결하며 비밀키 기반의 DES 암호 알고리즘을 사용하여 서버와 클라이언트간의 통신 내용을 암호화 하는 분산된 인증 시스템이다. 따라서 보다 넓은 네트워크상에서의 인증을 효율적으로 인증해 줄 수 있다.

Kerberos 인증 메커니즘은 인증 서버인 KDC(Key distribution Center)와 티켓 허가 서버(TGS : Ticket Granting Server)로 나누어져 있어 사용자는 Kerberos 로부터 인증을 받고 TGS 에서 사용할 서비스에 대한 티켓을 얻은 후 그 티켓의 유효 시간까지 서비스를 제공 받을 수 있다. 또 다른 서비스를 받고자 할 때 티켓을 포워딩 함으로써 여러 번 패스워드를 입력해야 하는 불편을 없앴을 뿐 아니라 네트워크상 패스워드가 전송되면서 공격자에게 도청 될 기회도 줄여 보안상 안전을 보장한다. Kerberos 는 클라이언트에 대한 인증과 인증 서버와 티켓 허가 서버에 대한 인증도 동시에 수행하는 상호 인증(Mutual Authentication)도 제공한다.[2]

Kerberos V5 의 기본 인증 수행 과정은 다음과 같다.



[그림 2-1] Kerberos 인증 과정

Kerberos 에서는 사용자에 대한 인증을 워크스테이션이 맡아서 하게 된다. 즉, 인증에 관해 사용자에게 투명성을 제공해주는데 기본 알고리즘에서의 클라이언트는 사용자를 대신하는 워크스테이션이라고 할 수 있다. 첫 번째에서 사용자는 KDC 즉 인증 서버로부터 TGS 를 사용할 수 있는 티켓을 받게 된다. 두 번째로 사용자는 인증서버로부터 받은 TGS 티켓을 이용해서 TGS 에게 요청할 서비스를 사용할 수 있는 티켓을 얻게 된다. 마지막으로 사용자는 TGS 로부터 허가 받은 티켓을 이용해서 요구한 서비스를 제공 받을 수 있다.[2] 1 과정은 사용자가 워크스테이션에 로그인 하는 과정에서 수행되고 2 과정은 어떤 서비스에 대한 티켓을 얻는 과정이며 마지막 단계에서는 부여 받은 티켓으로 서비스를 사용하는데 이 단계부터는 더 이상의 패스워드 입력은 없어도 된다.

Kerberos 는 서비스를 제공하는 서버와 독립적이여서 보다 넓은 네트워크 환경이나 고성능 시스템을 다중 사용자에게 제공해야 하는 환경에 적합한 인증 시스템이다. 많은 사용자들에 대한 KDC 를 독립적으로 구축하여 여러 대를 설치하여 관리할 수 있다. 그러나 각각의 KDC 에 따로 접속을 하여서 사용자들에 대한 티켓을 관리해야 함으로 불편이 따른다.

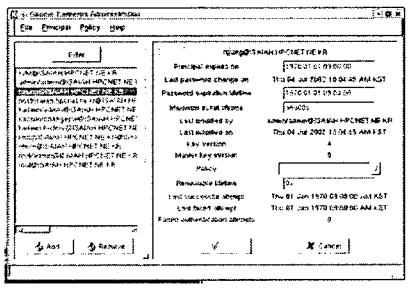
3. 인증 관리 인터페이스

Kerberos 는 강력한 인증 시스템이면서 보안적으로 많은 정보를 제공해준다. 모든 서버와 클라이언트의 접속을 log 파일로 저장하며 다른 보안 시스템에 비해 로그 파일이 풍부한 편이다.[2] 그러나 이런 로그 파일을 분석하는 일이나 많은 사용자들의 티켓을 관리하는 일에 있어서는 관리자에게 있어 많은 불편이 따른다. 직접적으로 잘 정리되어 있지 않은 로그 파일을 꺼내서 봐야 하며 텍스트 기반의 인증 관리 응용 인터페이스로 많은 불편이 따른다. 그래서 본 논문에서는 보다 넓은 네트워크 환경에서 Kerberos 인증을 사용하고자 할 때 관리자의 불 편을 덜고자 새로운 인증 관리 인터페이스를 제시한다.

3-1 기존 인증 관리 인터페이스 gkadmin

인증 관리 인터페이스로 현재 쓰이고 있는 것은 리눅스 기반의 gkadmin 이라는 것이 있다.

Gkadmin 은 리눅스 기반의 그래픽 인터페이스로 인증 관리를 지원해 주고 있다. 기본 기능으로는 현 리눅스에 구축된 KDC 에 접속하여 principal 을 추가해 주고 삭제하며 또 각 principal 에 대한 티켓의 정보를 수정 및 추가 해 줄 수 있다. 다음 그림은 gkamin 의 인증 관리 인터페이스 이다.



[그림 3-1]gkadmin

gkadmin 한 KDC 만을 관리하며 단순히 principal 에

대한 관리와 티켓에 대한 관리 이외의 기능은 없기 때문에 다중 환경을 지원해야 하는 네트워크 상에서는 불편 사항이 따르게 된다. 여러 대의 KDC 를 구축했을 경우 각각의 KDC 서버에서 직접 작업을 해야 한다. 그렇지 않을 경우 원격에서 해야 한다면 telnet 상으로 접속하여 텍스트 기반으로 작업을 해야 한다. 또한 gkadmin 은 로그 파일을 볼 수 있는 기능이 없을 뿐 아니라 여러 명의 principal 이 정리되지 않은 상태에서 혼잡한 인터페이스를 지원하고 있다.

3-2 제안하는 인증 관리 인터페이스 jkadmin

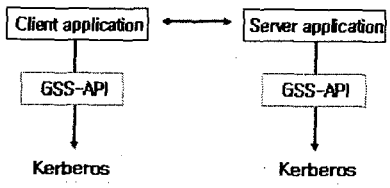
본 논문에서는 자바로 구현하는 Kerberos 전용 인증 관리 인터페이스를 설계하고자 한다. 임의로 새로운 인터페이스는 jkadmin 이라고 명명했다. jkadmin 는 자바가 지원해주는 장점들을 모두 보유할 수 있다. 또한 jkadmin 은 로컬 KDC 여러 개를 원격 접속으로 통합 관리 할 수 있다.

Kerberos 는 GSS-API 를 통해 제어가 가능 한데 자바를 GSS-API 를 포함하는 언어로써 자바로 인증 관리 인터페이스를 설계할 때 자연스럽게 jkadmin 은 kerberos 의 각 KDC 를 제어 할 수 있다.

Java GSS-API

Java GSS-API 는 일반적 보안 서비스를 위한 API 이며 IETF 와 RFC 2853 로 문서화 되어 있으면 정의되고 있다.

Java 를 만든 Sun 에서의 Java GSS-API 의 구현은 최초로 Kerberos V5 인증 메커니즘만을 지원하기 위해 구현되었다. Java GSS-API framework 는 자체적으로 매우 간다하며 보안에 관련된 모든 기능은 메커니즘 아래로부터 포함된 요소들에 대해 위임된다.[4]

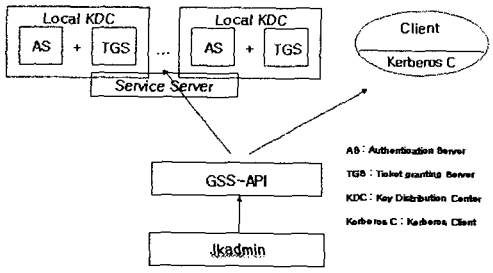


[그림 3-2] Kerberos 와 GSS-API 구조

jkadmin

본 논문에서 제안하는 jkadmin 에는 gkadmin 에 확장 기능로써 Local KDC 별로 통합 관리 하는 기능과 Kerberos 의 로그 파일을 분석할 수 있는 기능을 추가한다. 또한 윈도우기반 어플리케이션으로 만들어 원격에서 로컬 KDC 로 접속이 가능하게 하는 기능을 추가한다.

다음 [그림 3-3]은 jkadmin 과 Kerberos 의 전체적인 구조를 나타낸 그림이다.



[그림 3-3] jkadmin 수행 구조

설계 중 중요한 것은 jkadmin 이 GSS-API 에 접근하여 Kerberos 를 제어하는것이다. Jkadmin 은 앞에서 말했듯이 Java 로 구현 하기 때문에 여러 플랫폼에 구애를 받지 않으며 또한 슈퍼 클래스를 사용하여 상속하는 장점을 이용해서 GSS-API 에 접근할 수 있다. 또 Java 에서는 GSS-API 를 전격 지원하는 프로그래밍 언어로써 단순히 GSSManager 만 호출해 내면 jkadmin 으로 kerberos 를 제어하는 GSS-API 에 쉽게 접근 할 수 있다.[4]

다음은 GSSManager 의 기본 구현 코드이다.

```
GSSManager manager = GSSManager.getInctance();
```

GSSManager 는 또한 세가지 중요한 인터페이스를 위한 factory class 를 지원하는데 그것은 GSSName, GSSCredential 그리고 GSSContext 가 있다. 이 세가지들은 Kerberos 만을 위한 Java 의 GSS-API 의 중요한 인터페이스들이다. GSSName 은 Kerberos 의 클라이언트와 서버간의 이름 즉 Principal 에 해당되고 GSSCredential 은 인증서에 해당하며 GSSContext 는 보안 서비스를 수행하는 것의 인터페이스 이다. 이 세 가지의 인터페이스를 위한 코딩 또한 다음과 같이 간단하다.[4]

GSSName 의 서버와 클라이언트의 예

```
GSSName clientName = manager.createName
                        ("exodus", GSSName.NT_USER_NAME);

GSSName serverName = manager.createName(
    "root@isaiah.hpcnet.ne.kr", GSSName.NT_HOSTBASED_SERVICE);
```

GSSCredential 의 예

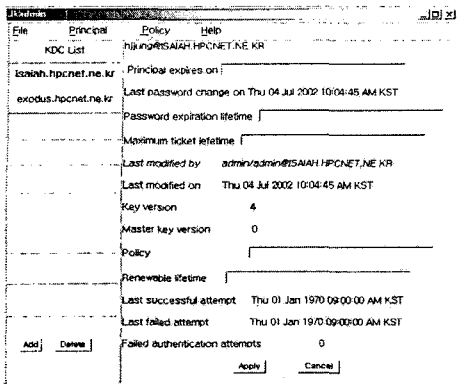
```
GSSCredential clientCreds = manager.createCredential(clientName,
                8*3600, desiredMechs, GSSCredential.INITIATE_ONLY);

GSSCredential serverCreds = manager.createCredential(serverName,
                GSSCredential.INDEFINITE_LIFETIME, desiredMechs,
                GSSCredential.ACCEPT_ONLY);
```

GSSContext 의 예

```
GSSContext GSSManager.createContext(GSSName peer, Oid mech,
                GSSCredential clientCreds, int lifetime)
throws GSSException
```

GSSCredential 에 보면은 클라이언트와 서버로 두가지로 나뉘어 있다. 이것은 서버와 클라이언트의 인증서 부분에 해당되며 상호 인증을 지원하기 위함이다. GSSContext 에는 누구에게 이 보안 서비스를 제공해주고 누구로부터 제공되는 서비스 인가를 확일할 수 있다.



[그림 3-4] jkadmin 의 인터페이스

[그림 3-4]는 jkadmin 의 그래픽 인터페이스를 대략 나타내보았다. Principal 의 풀 다운 메뉴에는 Principal 에 대한 수정 삭제 및 추가 extract key tap 설정 등의 기능이 있고 Policy 의 풀 다운 메뉴에는 Principal 에 대한 보안 정책과 전체 Kerberos 의 각 local KDC 의 로그파일을 볼 수 있는 기능이 있다. 맨 왼쪽의 리스트는 각 영역마다의 KDC 에 접속할 수 있는 리스트를 나타낸 것이다.
다음 [표 1]은 기존에 리눅스 기반의 인증 관리 인터페이스인 gkadmin 과 새로 제안된 jkadmin 을 비교한 표이다.

	gkadmin	Jkadmin
원격접속	Impossible	Possible
로그파일분석	Impossible	Possible
Principal 관리	Possible	Possible
Ticket 관리	Possible	Possible
다중 KDC 통합관리	Impossible	Possible

[표 1] gkadmin 과 jkamin 의 비교

*jkadmin : 제안된 인터페이스

4. 결론

Kerberos 는 강력한 인증 체계를 제공하는 인증 시스템으로써 넓은 개방형 네트워크 환경에서 인증 시스템으로 쓰기에 적합하다[7]. 그러나 넓은 개방형 분산 통신 망에서 많은 Principal 들을 관리하려면 여러 대의 Local KDC 를 구축해야 하는데 이 KDC 를 통합으로 관리하기란 어려운 일이다. 그래서 본 논문에서는 KDC 를 통합관리해주면서 원격으로 접속이 가능한 인증 관리 인터페이스에 대한 제안을 했다. 기존의 인증 관리 인터페이스에 비해 제안된 인증 관리 인터페이스는 인증 시스템을 관리하는 관리자에게 있어 편리한 구조 가지며 원격 접속이라든지 로그파일 분석과 다중 KDC 관리에 있어 가능성을 보여주었다.

4-1 향후 과제

현재는 아직 Version 1.0 도 나온 상태는 아니지만 계속하여 꾸준히 Java 로 개발할 때 로그 분석을 통한 침입 탐지도 할 수 있다. 또 여러 연구를 통해 Kerberos 가 DES 비밀키 암호 알고리즘을 사용하므로써 갖게 되는 보안적 Holl 을 RSA 공개키 기반 암호 알고리즘을 사용하여 보완하고자 하는 연구도 한창 진행 중이다.[6] 이런 점에서 미루어 볼 때 Kerberos 의 사용 수가 증가할 것이고, 그에 적합한 인증 관리 인터페이스가 개발 되어야 한다.

5. Reference

[1] *Kerberos : An Authentication Service for Computer* Network
B. Clifford Neuman and Theodore Ts'o
IEEE Communications Magazine
September 1994

[2] An Authentication Service for Open Network Systems
J. G. Steiner, B. Clifford Neuman, and J.I. Schiller.
In *Proceedings of the Winter 1988 Usenix Conference.*
February, 1988.

[3] The Evolution of the Kerberos Authentication System. In
Distributed Open Systems
John T. Kohl, B. Clifford Neuman, and Theodore Y. T'so
IEEE Computer Society Press, 1994

[4] Single Sign-on Using Kerberos in Java
Mayank Upadhyay, Ram Marti
Sun Microsystems, Inc. 2001

[5] Limitation of the Kerberos Authentication System
Steven M. Bellovin , Michael Merritt
AT&T Bell Laboratories 1991

[6] 안전한 Kerberos 인증 메커니즘 설계
조성아, 한태창, 함호상, 이동훈
고려대학교 1997, 정보처리 제 4 권 특집 논문

[7] 네트웍 환경에서 안전한 Kerberos 인증 메커니즘에 관한 연구
신광철*, 정진욱**
*벽성대학교, **성균관대학교
정보처리학회 논문지 vol 12 .no 2. 2002, 4