TPM에 기반한 공유 네트워크 스토리지 암호화 및 키 관리

최종욱[O] 박우람 박찬익
포항공과대학교

TPM Based Encryption of Shared Network Storage and Key Management

Jongwook Choi[O] Wooram Park Chanik Park
Pohang University of Science and Technology

요 약

iSCSI기반의 네트워크 스토리지는 IPSec을 사용함으로써, 전송하는 데이터의 기밀성과 무결성을 보장하고, 인증을 지원하고 있다. 결국 네트워크 스토리지에 전송되는 데이터의 기밀성을 지원하기 위해서는, IPSec을 사용하여야 하고, 이 경우에 IPSec은 데이터를 전송하는 과정에서 암호화를 위해 많은 CPU 오버헤드를 발생시킴으로써 전체적인 네트워크 스토리지의 성능 저하를 가져 올 수 있다. 또한, IPSec은 실제 타겟 서버에 저장되는 데이터들을 암호화하지 않으며, 이러한 데이터의 암호화를 지원하기 위해서는 별도의 방법을 사용하여야 한다. 본 논문에서는 네트워크 스토리지에 저장되는 데이터를 안전하게 암호화 하고, 해당 데이터 암호화에 사용되는 키들을 관리하고 사용하는 방법을 TPM에 기반하여 디자인하고 구현 하고자 한다. 또한, 네트워크 스토리지를 여러 명의 사용자가 공유한다고 가정 할 때, TPM을 사용하여 효율적인 그룹과 사용자 관리에 대해서 고려해 보고자 한다. 본 논문에서 제안하는 네트워크 스토리지의 데이터 암호화 방법과 IPSec의 AH 프로토콜을 사용한다면, 전송되는 데이터의 기밀성, 무결성, 인증을 제공할 수 있을 것이다. 또한 TPM에 기반하여 키와 사용자를 관리 함으로써, 특정 클라이언트만 해당 키를 사용할 수 있도록 하고, 암호화 되어 저장된 데이터가 외부로 유출 될 경우에도 데이터를 복호화 하는 것을 막을 수 있도록 한다.

1. 서 론

많은 기업과 기관들은 사용자가 네트워크를 통해서 사용할 서버의 저장 장치(Storage Device)를 사용할 수 있도록, SAN(Storage Area Network)과 NAS(Network Attached Storage) 시스템을 구축하고 있다. 이러한 환경에서, 컴퓨팅 시스템과 애플리케이션이 네트워크를 통해 서버의 저장 장치에 있는 데이터에 접근할 수 있는 네트워크 스토리지를 사용한다. 특히, iSCSI는 IETF에서 개발한 IP에 기초한 네트워크 스토리지 프로토콜로써, 이더넷이 구축되어 있으면 어디에서나 네트워크 스토리지 환경을 만들 수 있다. 그렇기 때문에, 고비용의 복잡한 파이버 채널(Fibre Channel) 없이도, 네트워크 스토리지의 유연한 데이터 관리 기능을 충분히 활용할 수 있다는 장점이 있다. 하지만, iSCSI는 IP를 기반으로 동작하기 때문에, 네트워크 환경에서 존재하는 제3자 공격(Man-In-the-Middle Attack), 위장 공격(Masquerade Attack), 변조 공격(Spoofing Attack) 등에 취약하다. 그래서, 네트워크 보안은 네트워크 스토리지 구축에 중요한 요소이다.

현재 iSCSI[1]에서는 네트워크 스토리지 보안을 위해서 CHAP[4] (Challenge-Handshake Authentication Protocol) 사용자 인증 프로토콜과 IPSec[2][3] (IP Security) 프로토콜을 사용하고 있다. IPSec 프로토콜은 IP 계층의 프로토콜로써, 전송되는 패킷을 암호화 하여서 정보의 기밀성(Confidentiality)을 유지하기 위한 ESP 프로토콜과 MD5/SHA1과 같은 해시 알고리즘을 사용해서 전송하고자 하는 패킷의 해시 코드를 생성하고, 이 정보를 바탕으로 송신 데이터에 대한 무결성(Integrity)과 인증 (Authentication)을 제공하는 AH 프로토콜로 구성된다.

IPSec의 ESP 프로토콜은 데이터의 기밀성을 보장하기 위해서 근원지 호스트는 데이터를 암호화 하고, 목적지 호스트는 데이터의 복호화를 수행한다. 이러한 과정에서 IPSec은 데이터 암호화와 복호화를 위해 CPU 오버헤드를 발생시킬 수 있다. 또한, 클라이언트가 증가하면 증가할수록 서버의 과부하는 커질 것이다. 본 논문에서는, IPSec의 ESP 프로토콜을 사용하지 않고 네트워크 스토리지 전송 데이터의 무결성을 보장하면서 저장되는 데이터를 암호화 하는데 초점을 두고 있으며, 위와 같은 상황에서 클라이언트의 수가 증가하더라도

[1] 본 연구는 지식경제부 및 정보통신 연구진흥원의 IT R&D 프로그램의 연구 결과로 수행되었음 [2008-S034-01, Development of Collaborative Virtual Machine Technology for SoD (System on-Demand) Service]

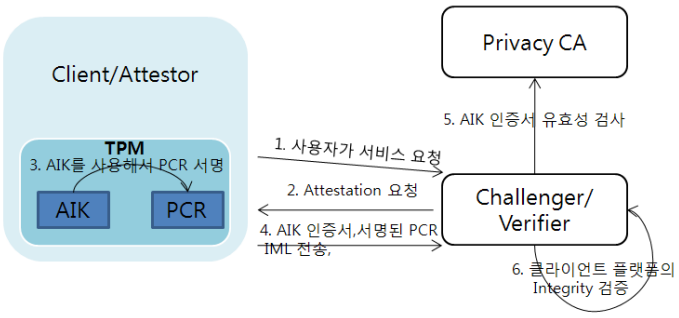

그림 1 원격 검증 절차(Remote Attestation)

서버에게는 최대한 적은 과부하를 주기 위해 시스템을 디자인 하였다. 또한, 데이터 암호화 키의 관리와 사용을 TPM[5]에 기반하여 구현 함으로써 더욱 안전한 키 관리를 목표로 하고, 같은 LUN을 여러 명의 사용자가 사용하는 공유 네트워크 스토리지를 가정하였을 때 발생할 수 있는 사용자와 그룹 관리 문제점을 TPM으로 해결해 보고자 한다.

본 논문의 구성은 다음과 같다. 2장은 TPM과 함께 사용되는 기술들에 대한 설명, 3장은 전체 시스템에 대한 디자인과 각각의 구현 방법, 4장에서는 관련 연구, 5장에서는 결론을 맺도록 한다.

2. 배경 지식

네트워크 스토리지 서버는 여러 클라이언트에게 자신의 물리 또는 가상 디스크를 제공해 준다. 클라이언트는 이러한 네트워크 스토리지를 자신에게 직접 연결된 물리디스크처럼 사용할 수 있다. 본 논문에서는, 네트워크 스토리지를 iSCSI를 사용하고 있으며, 클라이언트 측의 iSCSI Initiator 계층에서 서버에 저장되는 데이터를 직접 암호화 함으로써, IP 레이어에서 암호화 하는 ESP 프로토콜의 오버헤드를 줄이고자 한다. 또한, 암호화에 사용되는 키는 서버가 관리하며 원격 검증[6](Remote Attestation) 이라고 하는 기법을 사용하여 클라이언트를 인증하고, 인증된 클라이언트에게만 네트워크 스토리지를 암호화, 복호화 할 수 있는 키를 제공한다. 이러한 과정에서 키 관리 및 전달, 원격 검증을 위한 프레임워크를 구축하기 위해 TPM(Trusted Platform Module)이라고 하는 특수한 하드웨어 보안 칩을 사용한다.

아래에서, 본 논문에서 제안하는 기법을 구현하기 위해 필요한 기술들에 대해서 차례대로 소개하도록 하겠다.

2.1 IMA(Integrity Measurement Architecture)

IMA[7]는 SHA-1 알고리즘을 사용하여서, 시스템 상에서 실행되는 모든 프로세스와 라이브러리의 코드에 대한 해시 값을 계산하고, 그 결과를 TPM 내부에 존재하는 특정 PCR(Platform Configuration Register)에 업데이트 하는 코드이다. 그리고 수행된 프로세스와 라이브러리 코드의 해시 값은 실행 순서대로, 로그 파일형태로 저장이 된다. 로그 파일에 저장되는 각각의 항목을 IML(Integrity Measurement Log)라고 하며, 프로세스(혹은 라이브러리) 경로, 해시 값, 해시 값이 업데이트 된 PCR 번호로 구성되어 있다. 추후에 각 IML의 해시 값을 계산(Extend) 하여서 특정 PCR 값과 동일한지를 비교하여 무결성 (Integrity) 검사를 수행할 수 있다.

2.2 원격 검증(Remote Attestation)

원격 검증이란 상대방의 플랫폼이 신뢰할 수 있는 상태인지 검증하는 것을 말한다. 그림 1은 원격 검증 절차를 보여 주고 있다. 클라이언트가 서버(Challenger)에게 어떠한 서비스를 요청하면, 서버는 서비스를 제공 하기 전에 클라이언트 플랫폼의 신뢰성을 검증하기 위한 원격 검증을 수행한다. 서버는 새롭게 생성한 난수와 함께 원격 검증 요청 메시지를 전송한다. 클라이언트는 수신한 난수와 현재 PCR 값, IML을 자신의 AIK로 서명하고, 이것을 서버에 전송한다. 서버는 TTP(Trusted Third Party)로부터 AIK 인증서를 확인하고, AIK로 서명한 값들의 무결성을 검증한다. 그리고 IML의 해시 값들을 차례로 Extend 하여서 수신한 PCR값과 일치하는지 확인함으로써 클라이언트가 신뢰할 수 있는 플랫폼인지 증명한다.

2.3 Key Migration

TPM은 여러 종류의 비대칭 키들을 생성할 수 있으며, 내부에서 관리 할 수 있고 비대칭 키의 개인키는 밖으로 유출 시키지 않는 특징을 가지고 있다. 해당 키들은 다른 TPM으로 이동할 수 있는 옵션이 있는데, 특정 TPM 내부에 저장된 키를 다른 TPM으로 옮겨서 사용할 수 있는 Key Migration 기법을 제공하고 있다. 이러한 방식을 통해서 지정된 TPM만 마이그레이션 되는 키를 사용할 수 있도록 허용할 수 있다. 키를 이동(Migration)하는 과정에서 비대칭 키의 공개 키에 대해서는 무결성을 제공하고, 비공개 키에 대해서는 무결성과 기밀성 모두를 제공하여 한다[7].

3. 디자인 및 구현

본 논문에서는 IPSec의 ESP 프로토콜에 의해 발생하는 CPU의 과부하를 줄이고 네트워크 스토리지의 성능을 높이며, 데이터를 암호화 하여 저장함으로써 데이터의 기밀성을 보장하고자 한다. 또한, 데이터의 기밀성을 위해서 사용되는 암호화 키는 TPM 칩에서 관리 함으로써 더욱 안전한 키 관리 및 사용을 위한 프레임 워크를 구축하고자 한다. 본 논문에서 제안하는 시스템의 기능을 정리해 보면 아래와 같다.

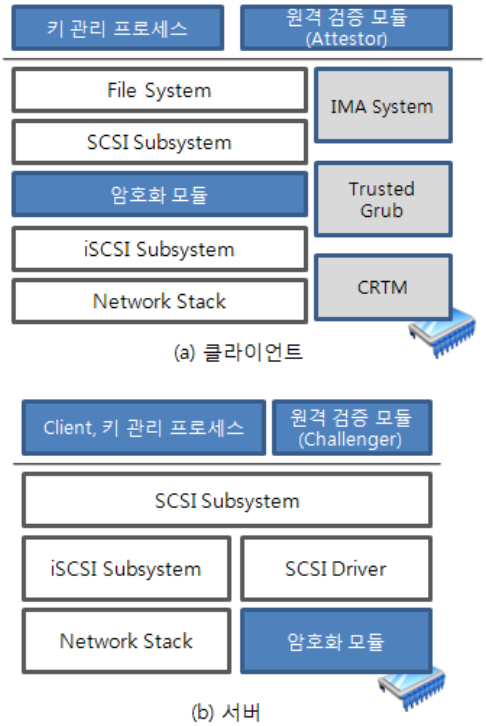

그림 2 클라이언트와 서버의 구조

- 클라이언트에서의 전송 데이터의 암호화
- 클라이언트에서 동작하는 소프트웨어 스택의 검증(Remote Attestation)
- TPM의 인증서를 사용한 사용자 인증
- TPM 기반의 키 관리 및 전달

네트워크 스토리지 환경에서 IPSec을 사용할 경우 클라이언트는 해당 디스크에 쓰기 연산을 수행할 때에는 서버로 전송하는 데이터의 암호화를 수행하고, 읽기 연산을 수행할 때에는 전송 받은 암호화된 데이터에 대해 복호화를 수행한다. 이러한 데이터의 암호화와 복호화에 사용되는 키는 서버가 TPM을 사용하여 관리하며, 클라이언트가 원격 검증을 통해 안전한 플랫폼이라고 증명 되고, 해당 클라이언트가 등록되어 있을 경우에만 제공하도록 한다. 전체 사용 시나리오는 다음과 같이 요약할 수 있다. 첫 번째로, 클라이언트는 iSCSI 서버에 접속하기 전에 원격 검증을 통해서 자신의 키 관리 프로세스와 iSCSI 암호화 모듈뿐만 아니라, 수행되고 있는 모든 프로세스와 커널 모듈 등에 대한 상태를 검증하게 된다. 이와 함께, 클라이언트 TPM은 신뢰할 수 있는 제 3자에 의해 인증된 고유의 인증서를 서버에게 제공하고, 서버는 클라이언트의 인증서를 통해서 클라이언트 플랫폼을 인증 한다. 두 번째로, 클라이언트의 상태를 검사가 성공적으로 수행 된 다음에는 서버가 클라이언트에게 해당 iSCSI 디스크를 암호화, 복호화 할 수 있는 키를 마이그레이션 하도록 한다. 그리고 세 번째로, 실제 iSCSI 연결을 설정하고 데이터를 암호화, 복호화 하여서

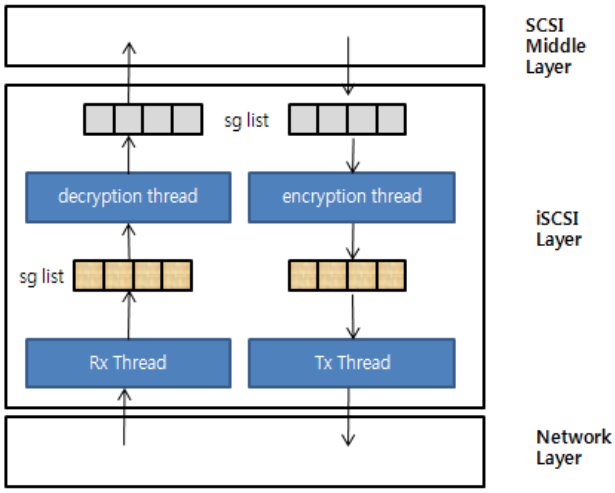

그림 3 클라이언트의 암호화/복호화 모듈

읽기/쓰기 할 수 있도록 한다. 이러한 과정을 거쳐서 클라이언트에게 네트워크 스토리지를 안전하게 제공하고 사용하도록 한다.

위와 같이 TPM을 사용하여 키를 관리하고 전달하고 TPM의 인증서를 사용함으로써, 등록된 특정 플랫폼만 해당 네트워크 스토리지에 접근할 수 있고, 서버의 데이터가 유출되더라도 TPM에서 관리하는 키의 유출이 힘들기 때문에, 데이터를 복호화 해서 사용하기가 불가능 하다.

그림 2 에서는 전체적인 시스템 아키텍처를 보여주고 있다. 클라이언트에서는 원격 검증을 위해서 플랫폼의 Integrity를 측정하는 Trusted Boot, IMA를 사용하고 있으며, 이렇게 측정된 Integrity의 결과를 원격 검증 모듈이 원격 검증을 위해서 사용한다. 또한, 서버로부터 전송 받는 키를 암호화 모듈에 전달하는 키 관리 프로세스가 있고, iSCSI 데이터를 암호화하는 암호화 모듈이 존재한다.

서버 측에서는 각 클라이언트의 키를 관리하는 모듈과 클라이언트의 원격 검증 모듈과 상호작용하는 검증 모듈, 암호화 모듈이 있다. 실제 암호화는 클라이언트의 암호화 모듈이 수행을 하고, 추후에 서버 관리자가 특정 디스크의 암호화된 데이터를 보고자 할 때에는, 자신의 암호화 모듈을 사용할 수 있다. 아래에서는 각 모듈이 수행하는 일과 각 모듈의 구현 방법에 대해서 설명하도록 하겠다.

3.1 네트워크 스토리지 암호화

먼저, 클라이언트에서 어떻게 네트워크 스토리지를 암호화 하고 복호화 하는지에 대해서 설명하도록 하겠다. 키 관리 프로세스가 서버로부터 마이그레이션 받은 키를 암호화 모듈에게 전달 해 줌으로써, 암호화 모듈은 해당 키를 사용해서 데이터를 암호화 하거나 복호화 한다. 암호화 모듈은 iSCSI와 SCSI Subsystem

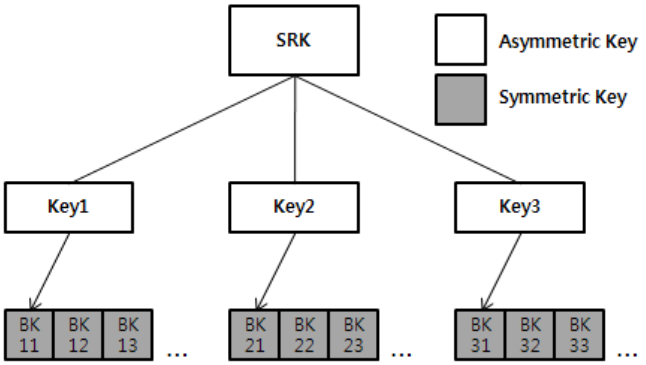

그림 4 암호화에 사용되는 대칭키의 관리

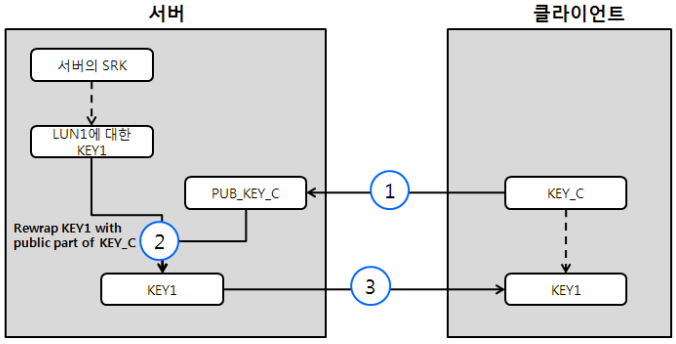

그림 5 키 마이그레이션(Migration)

사이에서 수행하도록 구현하였다. 그림 3은 데이터의 암호화와 복호화에 대한 구현을 보여 주고 있다.

쓰기 연산은 SCSI Subsystem에서부터, 쓰고자 하는 데이터들의 scatter/gather list(sg list)를 iSCSI Subsystem에게 알려 준다. 그러면 Tx 스레드는 해당 scatter list에 있는 데이터를 여러 번 나누어서 서버로 전송하게 된다. 이때, Tx 스레드가 해당 데이터를 전송하기 전에, 새로운 워크 스레드를 생성하고 해당 커맨드 구조체에 락을 건 후, 해당 sg list의 모든 데이터를 암호화 하는 작업을 수행하도록 한다. 암호화 작업이 끝나게 되면 구조체의 락을 풀게 되고, 이때 Tx 스레드는 대기하고 있다가 해당 sg list의 모든 데이터를 전송하도록 한다.

읽기 연산에서는 쓰기 연산과 마찬 가지로 Rx 스레드가 하나의 Read 커맨드에 대한 모든 데이터를 서버로부터 전송 받은 후, 새로운 sg list를 생성하고 전송받은 데이터들을 저장하게 된다. 이때 전송 받은 모든 데이터는 암호화 되어 있기 때문에 복호화를 해야 하는데, 복호화를 수행하는 새로운 워크 스레드를 생성해서 work 큐에 삽입 후, iSCSI Layer는 자신의 일을 마치게 된다. 그러면 해당 work 스레드는 sg list의 모든 데이터에 대해서 복호화를 수행하고, SCSI Middle Layer가 등록한 call back 함수를 호출함으로써 읽기 연산을 마치게 된다.

IP Layer에서 수행되는 IPSec의 ESP 프로토콜은 Transport Layer에서 최대 전송 크기에 의해 나누어진 모든 패킷를 각각 암호화해야 하는 오버헤드를 가져 올 수 있지만, 위와 같은 방법을 사용할 경우에는 이미 iSCSI Layer에서 모든 데이터를 암호화 함으로써 IP Layer에서 수행하는 것 보다 더 효율 적으로 데이터를 암호화 할 수 있다. 또한, 서버 측에서는 별도의 암호화 연산을 수행하지 않기 때문에 오버헤드를 줄일 수 있다.

3.2 키 전달 및 관리

그림 3 은 TPM을 사용하였을 경우 암호화 키를 관리하는 구조를 보여 주고 있다. SRK는 TPM 내부의 키로서, 절대 외부로 유출 되지 않는다. 또한, SRK의 아래 계층에 있는 키들은 각 부모 키에 의해서 암호화가 되어서 디스크나 메모리에 영구 저장할 수 있다. SRK의 개인키가 절대 외부로 유출되지 않기 때문에 위와 같은 구조가 안전하다고 말할 수 있다. Key1, Key2, Key3는 iSCSI 서버가 제공하는 각 LUN에 할당된 비대칭 키들이다. 이 키들은 BK11, BK12와 같은 실제 블락을 암호화하는 대칭 키를 암호화 하고 있다. 하나의 비대칭(BK) 키는, 정해진 크기(4MB)의 블락을 암호화 하는데 사용하고 있다.

실제 클라이언트에 대한 원격 검증과 인증서를 사용한 인증이 끝나고 나면, 해당 클라이언트가 사용하는 LUN에 해당하는 비대칭 키와 대칭 키의 리스트를 클라이언트에게 전송해 주어야 한다. 이러한 키를 전송하는 과정을 키 마이그레이션이라 한다. 그림 4는 TPM Key 마이그레이션에 대해서 보여 주고 있다. 먼저, 클라이언트의 TPM이 생성한 KEY_C에 대한 공개키를 서버에게 전송한다. 이때, 서버는 전송하고자 하는 KEY1에 대한 인증서를 TPM에게 요청한다. TPM은 KEY1을 KEY_C의 공개키로 암호화 하고, 이 결과물과 KEY1의 공개 키에 대한 무결성을 보증하는 데이터를 인증서와 함께 제공한다[7]. 이 전체 데이터를 클라이언트에게 전송하게 되면, 클라이언트는 TPM에게 해당 키를 사용할 수 있도록 복호화를 요청하게 된다. 이때, TPM 은 내부적으로 서버가 전달 해 준 키를 복호화 하고 사용할 수 있도록 TPM 내부에 로딩하게 된다.

위와 같은 방식으로 서버의 TPM에서 클라이언트의 TPM으로 키를 마이그레이션 할 수 있다. KEY1과 같은 비대칭 키와 함께, 각 블락 별로 암호화에 사용되는 대칭 키(BK)도 함께 클라이언트에게 전송하게 된다. 이때, 클라이언트는 해당 대칭 키를 TPM 내부에 마이그레이션 된 비대칭 키를 사용하여 복호화 할 수 있다. 이러한 방법으로 마이그레이션 되는 키는 오직 해당 클라이언트의 TPM에서만 복호화 할 수 있을 것이다.

3.3. 사용자 관리와 인증

본 논문에서는 네트워크 스토리지가 여러 사용자가 함께 정보를 공유 할 수 있는 상태를 가정하고 디자인 하였다. 비록, iSCSI는 여러 사람이 공유하기에 동기화 문제가 발생할 수 있지만, 그러한 문제를 해결할 수 있는 솔루션이 이미 나와 있는 상태이기 때문에 간단하게 해결 할 수 있을 것이다. 이러한 문제점을 배제한 상태에서 어떻게 사용자를 그룹에 추가하고 제거할 것인지에 대해서 생각해 보기로 한다.

여기서 사용자에 대한 인증은 TPM의 고유 인증서를 사용하여 인증하고 있다. TPM은 자신을 구분할 수 있는 유일한 키를 가지게 되고, 이것을 사용해서 자신을 인증할 수 있는 인증서를 만들고, 이 인증서를 신뢰할 수 있는 제 3자에 의해 배포 할 수 있다. 이 인증서를 사용해서 특정 TPM을 가진 플랫폼만 네트워크 스토리지에 접근하게 함으로써, 플랫폼에 종속적으로 사용자의 접근을 허가할 수 있다[10]. 이것은 패스워드를 사용하는 CHAP 프로토콜 보다 더 안전할 수 있을 것이다.

4. 관련 연구

네트워크 스토리지의 성능 향상을 위해서 많은 연구가 수행되었었다.

[9]에서는 기존 IPSec의 ESP 프로토콜은 IP Layer에서 데이터를 암호화 하기 때문에 발생할 수 있는 문제점을 지적하고, IPSec의 AH 프로토콜을 그대로 사용하여 데이터의 무결성을 보장하고, iSCSI Level에서 Data를 암호화 함으로써 기밀성을 유지한다. 이 같은 경우 서버와 클라이언트 모두가 iSCSI Level에서 데이터를 암호화 복호화 함으로써 IPSec의 ESP 프로토콜을 iSCSI Layer로 올린 것에 해당한다.

[10]에서는 IPSec을 사용하지 않고 User layer에 Middle layer를 두고서 데이터를 암호화 한다. 또한 [9]과 같이 서버에 저장되는 데이터는 모두 복호화 되어서 저장이 된다.

네트워크 스토리지의 보안을 위해서 IPSec을 사용할 때 발생하는 오버헤드를 줄이기 위해 많은 연구들이 진해되었다. 본 논문은, 네트워크 스토리지의 성능 향상 뿐만 아니라 하드웨어 보안 칩인 TPM을 사용하여 더욱 안전한 키 관리 및 키 사용에 대해서 초점을 맞추고 있다.

5. 결론

본 논문에서는, IPSec을 사용하는 경우 발생할 수 있는 오버헤드를 줄이기 위해서 클라이언트에서 암호화하는 방식을 사용하고 있다. 또한, 클라이언트 플랫폼을 계속해서 원격 검증하는 방식과 암호화에 사용되는 키를 TPM 기반으로 관리하고 전달함으로써 더욱 안전한 키 관리 및 네트워크 스토리지 접근 방법을 제안하고 있다.

참고 자료

[1] Julian Satran, Kalman Meth, Costa Sapuntzakis, Mallikarjun Chadalapaka, Efri Zeidner, “iSCSI Draft”

[2] S. Kent, R. Atkinson, “IP Authentication Header”, RFC 2402, Nov. 1998.

[3] S. Kent, R. Atkinson, “IP Encapsulating Security Payload”, RFC 2406, Nov. 1998.

[4] RFC 1994 - PPP Challenge Handshake Authentication Protocol (CHAP)

[5] Trusted Platform Module Work Group Web Site: http://www.trustedcomputinggroup.org/developers/trusted_platform_module

[6] Dries Schellekens, Brecht Wyseur, Bart Preneel, "Remote Attestation on Legacy Operating Systems With Trusted Platform Modules", Science of Computer Programming Vol74, Dec. 2008.

[7] Trusted Computing Group Web Site : http://www.trustedcomputinggroup.org/

[8] Reiner Sailer, Xiaolan Zhang, Trent Jaeger, and Leendert van Doorn, "Design and Implementation of a TCG-based Integrity Measurement Architecture", USENIX Security Symposium, Vol13, 2004.

[9] Bo Mao, Dan Feng, Suzhen Wu, Jianxi Chen, Lingfang Zeng: Performance-Directed iSCSI Security with Parallel Encryption. AINA 2008: 855-860

[10] Kikuko Kamisaka, Masato Oguchi, Saneyasu Yamaguchi : Performance Evaluation of iSCSI System Optimized for Encryption Processing in the Upper Layer, 21st ICDEW, 2005.