2001년 한국정보처리학회 추계 학술발표논문집 제8권 제2호

효율적인 연결 관리를 위한 VPN 설계

김정범 이윤정 김태윤
고 려 대 학 교 컴 퓨 터 학 과
{qston, nspark, tykim}@netlab.korea.ac.kr

Design of VPN for Efficient Session management

Jeong-Beom kim, Yun-Jeong Lee, Tai-Yun Kim
Dept. of Computer Science and Engineering, Korea University

요 약

최근, 네트워크의 사용 증가에 따른 보안의 필요성이 대두되어 암호 사용이 급속히 확산되고 있다. 그러나, 암호는 본래 가지고 있는 키 관리의 어려움 때문에 여러 가지 문제가 발생할 수 있다. 이러한 암호의 사용이 야기하는 역기능을 해소하고 순기능을 지향하기 위해 키 복구에 대한 연구가 활발히 진행되고 있으며, 지금까지 많은 키 복구 기술들이 제시되어왔다. 본 논문에서는 IPSec(IP Security)로 구현된 Host-to-Gateway VPN(Virtual Private Network) 환경 하에서 SG(Security Gateway)와 호스트 사이에 연결이 중단되었을 경우 이에 따른 연결 복구에서의 시간적 소모를 줄이기 위한 방안으로 키 복구 기술을 이용한 메커니즘을 제안한다. 키 복구 방식을 기반으로 한 메커니즘은 VPN에서 SG와 호스트 사이의 터널 형성을 위한 세션 정보를 분실할 경우에 대해 세션 정보를 미리 저장해두고, 필요시 이전 연결 상태를 복구 할 수 있다. 제안한 키 복구 메커니즘은 기존 SG를 확장하여, IPSec 기반의 Host-to-Gateway VPN에서 세션 복구에 따른 시간적 지연을 해결한다.

1. 서론

재택 근무가 활발해지고 기업 외부와도 네트워크 구성이 필요하게 되는 등의 기업 네트워크가 점차 확대되어감에 따라 막대한 시설 투자가 필요하게 되었다. 네트워크의 확대와 함께 네트워크에 연결된 서로간에 안전한 통신을 하기 위해 사용해오던 전용망에 투자해야 하는 비용과 그에 따른 운영과 관리가 커다란 문제가 되고 있다.

VPN(Virtual Private Network)이란 이런 문제들의 해결을 위한 방안으로, 기업의 네트워크를 구성할 때 전용 임대회선을 사용하는 것이 아니라 공용망인 인터넷망을 이용하는 연결망이다. VPN은 터널링이라는 기법을 사용하여 일대일 연결과 같은 터널을 형성하며 데이터 패킷들은 터널을 통해 안전하게 전달된다. 이러한 터널링을 구현하는 기술로는 PPTP(Point to Point Tunneling Protocol), VTP(Virtual Tunneling Protocol), L2F(Layer 2 Forwarding Protocol), L2TP(Layer 2 Tunneling Protocol), IPSec(IP Security Protocol) 등이 있다. 본 논문에서는 이러한 터널링 기법 중 IPSec으로 구현된 VPN의 환경을 기반으로 연구한다.

IETF(Internet Engineering Task Force)에 의해서 IP 계층 보안을 위한 개방 구조로 설계된 IPSec은 네트워크 계층의 보안에 대해서 안정적이고 영구적인 기초를 제공한다. IPSec은 오늘날의 암호화 알고리즘을 수용할 수 있을 뿐만 아니라 새로운 알고리즘을 수용할 수 있다.

이러한 기존의 IPSec에서는 터널이 중단되었을 경우 다시 세션 키를 위해 재협상을 하여 터널을 복구해야 한다. 하지만 많은 호스트와 연결된 SG(Security Gateway)에서는 세션 복구를 해주는데 많은 시간이 소모된다. 이러한 문제를 해결하기 위한 방안으로는 SG에 키 복구 방식을 이용한 메커니즘을 기반으로, 세션 재연결시 저장된 키 복구를 이용하여 세션 키를 복구 해줌으로써 호스트와 SG 간에 재협상 과정을 생략하였다. 이러한 재협상에 따른 시간적 소모를 줄이므로 세션 복구에 따르는 전체 시간을 줄일 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 암호 키 기반 관리 구조와 IPSec 기반의 VPN 구조에 대해 분석한다. 3장에서는 키 복구 메커니즘을 제안하며 이를 기반으로 기존 IPSec 기반 VPN 구조에 대한 SG를 확장한다. 4장에서는 제안한 메커니즘의 성능을 분석한다. 5장에서는 결론 향후 연구 과제를 제시한다.

2. 관련연구

2.1 키 복구 기반 관리

암호 키 관리 기반구조는 일반적으로 암호문의 소유자가 아닐지라도 사전에 약속된 어떤 특정한 조건하에서 허가된 사용자에게 복호를 가능하게 하는 시스템으로 정의 할 수 있다. 암호 키 기반 관리 구조가 키의 분실이나 손실로 접근할 수 없는 경우와 국가가 범죄 수사 등의 적법한 이유로 키에 접근할 필요가 있을 경우, 암호가 오용됨으로써 발생할 수 있는 잠재적인 이유들을 방지하기 위하여 필요하다.

현재까지 제안된 암호 키 관리 방식은 크게 위탁 방식과 TTP(Trusted Third Party) 방식, 캡슐화 방식으로 나눌 수 있다.

위탁방식은 암호문 복호화를 위한 키 또는 키의 조각들을 신뢰하는 기관에 위탁하고 필요시에 그 정보들을 얻어내어 키를 복구해내는 방식으로 유사시에 확실한 키 복구가 가능하다는 장점이 있다. 반면에 이 방식에서 위탁되는 키나 키 조각들은 사용자들의 비밀키와 관련된 것들이므로 사용자의 프라이버시 보호를 위해서는 위탁기관의 신뢰성이 절대적으로 보장되어야 하는 문제가 있다. 이를 위한 방안으로 비밀 분산 방식(Secret sharing scheme)이 주로 사용되고 있다. 또한 사용자의 키가 복구되었을 경우 키의 사용기간을 제한하는 것과 위탁되는 정보가 유효한 것인가를 확인하는 것도 해결해야 할 문제이다.

TTP 방식은 신뢰할 수 있는 제 3자인 TTP가 복구를 요청하는 사용자의 비밀키를 생성하고 사용자에게 분배하는 방식으로 실제적인 키 위탁은 일어나지 않으나 사용자의 오래 보존해야 하는 키를 TTP가 직접 보관하고 있으므로 위탁된다고도 할 수 있다. 이 방식에서 TTP는 사용자의 비밀키를 생성·분배하므로 신뢰성 보장이 절대적으로 중요하다. 이 방식의 장점은 TTP가 사용자들의 비밀키를 모두 가지고 있으므로 필요시에 TTP에 의한 키 복구가 확실히 보장되며 TTP 사이의 키 생성 방식을 통일하면 국가간 호환이 용이하다는 것이다. 반면에 많은 TTP가 필요하며 TTP와 사용자 그리고 TTP와 TTP 사이의 병목현상이 심하다는 단점이 있다.

캡슐화 방식은 생성되는 각각의 암호문에 대해 키 복구 정보를 생성하여 암호문과 함께 전송 또는 저장하는 방식으로 복구되는 키는 키 위탁 방식과는 달리 세션 키이다. 이 방식에서는 복구되는 키가 세션 키이므로 감청 기관의 복구능력을 제한할 수 있어 키 위탁 방식보다는 사용자의 프라이버시 보호에 유리하며 기존 프로토콜의 확장 필드를 이용하여 간단하게 사용할 수 있다는 이점이 있다. 그러나 키 복구에 필요한 정보를 사용자들이 생성하므로 조작이나 변조를 통해 키 복구 능력을 악용 할 수 있다는 문제가 있다.

그림 1은 캡슐화 방식의 키 복구를 사용하여 암호 통신을 하는 두 사용자 단말 장치 사이의 상호 작용을 나타내고 있다.

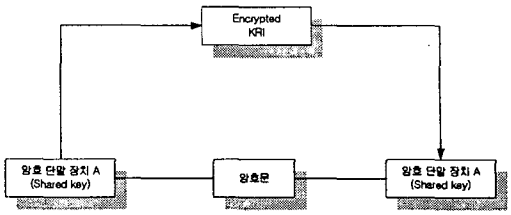

그림 1 캡슐화 방식

이 방식에서는 먼저 목적키를 복구 가능하게 하기 위해서 사용자 단말 장치 내의 키 복구 정보 생성 기능은 목적키에 대응하는 키 복구 정보(KRI: Key Recovery Information)를 생성하여 캡슐화 한 후, 상대편 사용자 단말 장치로 암호문과 함께 전송한다.

본 논문에서 제안한 메커니즘은 이러한 암호 키 관리 방식 중 캡슐화 방식을 기반으로 제안한다.

2.2 기존의 VPN 연결 관리 문제점

기존의 Host-to-Gateway VPN은 세션 복구에 따른 시간이 너무 오래 걸린다는 문제가 있다. 그림 3과 그림4는 기존의 VPN의 세션 복구 문제점을 보여주고 있다. 그림 3은 호스트가 안정된 네트워크 환경 속에서 IPSec 터널을 이용하여 보안 처리된 패킷을 순조롭게 보내는 것을 나타낸 것이다.

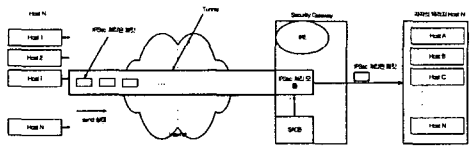

그림 2 기존의 IPSec으로 구현한 Host-to-Gateway VPN

이때 네트워크 상태의 불안정 등의 이유로 호스트와 SG 사이의 터널 연결은 모두 해체된 경우에 터널연결이 모두 중단된다. 그림 4는 네트워크 상태가 원상태로 복구된 경우 재동작한 SG의 세션 처리를 나타낸다.

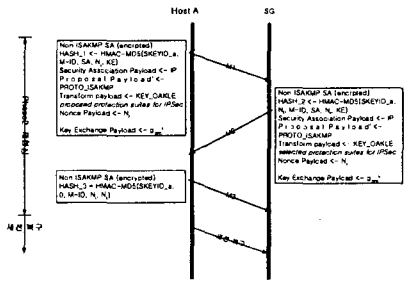

그림 3 세션 재복구 시

그림 4의 SG는 각 호스트에 대한 세션 정보가 순간 소실되었기 때문에 각 터널에 대한 정보를 알 수 없다. SG는 세션 정보를 모르는 호스트들에 대해서 다시 재접속을 요청하고 요청하기 이전에 키 협상 프로토콜인 ISAKMP의 Phase2 과정의 재시도를 통한 새로운 세션 정보를 서로 공유하게 된다

그림 4에서 보여주는 일련의 작업들은 여러 개의 호스트가 연결되어 있는 현 네트워크 상황에 적합하지 못하다. 이러한 문제점을 해결하기 위한 방안으로 본 논문에서는 키 복구 개념을 도입한 IPSec을 위한 새로운 SG를 제안한다.

3. 본론

3.1 제안된 IPSec의 키 복구 메커니즘

그리고, Host A와 SG간의 연결에서 SG가 KRFSH 내의 KRI를 갖고 어떻게 세션키를 복구하고 백업하는지를 그림 8에 나타냈다.

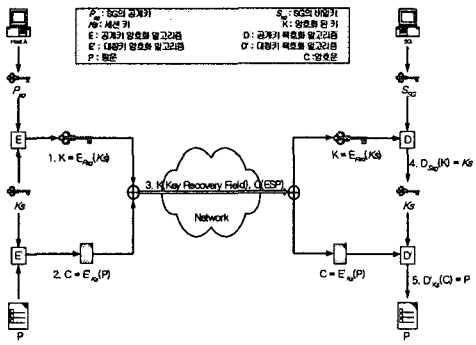

그림 4 세션 키 복구와 백업 절차

그림 8을 살펴보면 다음과 같다.

1. 먼저 호스트 A에서 SG의 공개키 *Psg*와 세션 키 *Ks*를 공개키 암호 알고리즘을 써서 암호화 된 키 **K**를 산출한다.
2. 이와 동시에 세션 키 *Ks*로 대칭키 암호 알고리즘을 써서 평문 P를 암호화한다.
3. 호스트 A는 만들어진 암호문 **C**와 **K**를, **C**는 ESP 형식에 **K**는 KRFSH 형식에 맞춰서 SG에게 전송한다.
4. SG 측에서는 *Ssg*와 전송받은 **K**를 공개키 복호화 알고리즘을 이용하여 *Ks*를 획득한다.
5. 이 *Ks*가 유효한지를 검증하기 위해 획득한 *Ks*로 대칭키 복호화 알고리즘을 써서 암호문 **C**를 평문 **P**로 복호화한다.

5의 결과가 정당하다면 SG는 획득한 *Ks*를 백업해 둘 수 있고 세션이 중단 될 경우, 백업해 둔 세션 키를 가지고 세션을 복구 할 수 있다.

3.2 제안된 SG의 동작 메커니즘

KRI의 주요 목적은 이 헤더 안에다가 키 복구 정보를 들여놓음으로써, 이 헤더를 삽입한 패킷이 호스트에서 SG로 전송되어질 때, SG에서는 이 정보를 저장해둠으로써 네트워크 상태의 불안정으로 인한 상황 속에서 세션이 끊어졌을 경우에 저장된 각 호스트에 대한 세션 정보를 해당 호스트에게 보냄으로써 일방적인 세션 복구가 이루어진다.

이러한 키 복구 기술을 탑재한 SG를 이용한 IPSec 터널링 VPN의 세션에 이상이 없을 시에 동작원리는 그림 9와 같다.

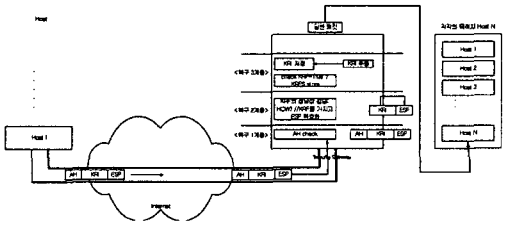

그림 5 제안한 SG 확장 메커니즘

그림 9에서 호스트가 KRI를 보내는 IP 패킷에 첨가해서 보내게 되면 TTL이 만료될 때까지 KRI를 소스 주소와 함께 저장 해 놓는다.

SG와 연결이 해제된 경우가 발생했을 때 제안한 SG를 가진 IPSec 터널에서의 세션 재연결 절차는 그림 10과 같다.

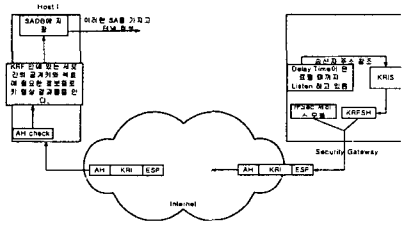

그림 6 제안한 세션 복구 처리 메커니즘

그림 10은 보낸 패킷에 포함된 KRI안에 있는 세션 정보를 바탕으로 각 호스트에 대한 세션을 복구 해 주는 것을 보여준다. 호스트는 TTL이 만료될 때까지 KRI 속의 세션 정보를 포함하는 KRF를 소스주소와 함께 임시 저장 해 놓는다. 사고가 일어났을 경우, 저장되어 있는 세션 정보를 가지고 각각의 호스트의 세션 정보에 맞게 AH+KRI+ESP 형식으로 전송하게 된다. 그러면 호스트에서는 KRI속에 있는 세션 정보에 맞게 연결을 다시 복구한다.

이렇게 함으로써 각 호스트와 SG 사이에서 세션 재복구를 위한 메시지 3개로 구성된 재협상 과정을 생략 할 수 있어서 전체적인 시간 지연의 문제도 해결할 수 있다.

4. 성능 평가 및 분석

4.1 성능 예측

성능에 대한 예측은 다음과 같다.

먼저 호스트 A가 n개의 호스트와 연결되어 있다고 가정한다. Phase1 과정에서 소비되는 시간을 t_{p1}, Phase2 과정에서 소비되는 시간을 t_{p2}, 각각의 호스트의 시스템 내부의 프로세싱 시간을 t_{proc}라고 하고, 프레임 하나를 전송하는데 걸리는 시간을 t_{frame}, 각각의 호스트에서 i개의 데이터 프레임을 보내다가 세션이 중단되었을 경우 세션 복구 처리에 걸리는 총 시간 지연 Ψ 은 다음과 같다.

$$\Psi = n \times \left(t_{p1} + 2t_{p2} + t_{proc} + \sum_{i=1}^{i} i t_{frame} \right)$$

$$= n \times \left(t_{p1} + 2t_{p2} + t_{proc} + \frac{i(i+1)}{2} t_{frame} \right)$$

여기서, $n>0$, $t_{p1}>0$, $t_{p2}>0$, $t_{proc} \ll 1$이므로 위 식은 아래의 (1)식과 같다.

$$(1)\ \Psi = n \times \left(t_{p1} + 2t_{p2} + \frac{i(i+1)}{2} t_{frame} \right)$$

반면에 그림 9와 같은 VPN에서 걸리는 지연시간은 세션에 대한 협상이 없으므로 총 지연시간 T'은 $\Psi' = n \times (t_{p1} + t_{p2} + i t_{frame})$ (2)이다.

따라서 키 복구 기반이 있는 경우 없는 경우보다 세션 재 연결에 따른 예측할 수 있는 총 시간 절약 Δ 은 다음과 같다.

$$\therefore \Delta = \Psi - \Psi' = n \times \left(t_{p2} + \frac{i^2+i-1}{2} t_{frame} \right)$$

위의 (3)식에서와 같이, 정수배 만큼의 시간을 절약할 수 있다.

4. 결론 및 향후 연구 과제

암호의 급속한 사용은 일상 생활에 있어서 많은 편리함을 제공하게 되었지만 범죄 집단에 의한 암호의 악용과 키의 분실 및 손상에 따른 암호문의 복호 불가와 같은 부작용 또한 크게 대두되고 있다. 이러한 암호의 부작용에 대한 여러 가지 대처 방안들 중에서 현재 세계 각 국에서는 키 복구에 대한 연구가 활발히 진행되고 있다. 본 논문에서는 암호화 된 연결이 끊어졌을 때의 문제에 대한 해결책으로서 SG를 위한 키 복구 메커니즘을 제안한다.

본 논문은 IPSec으로 설계된 Host-to-Gateway VPN에서 SG가 작동 불능 상태에서 재동작할 때 끊어진 세션에 대해 재협상을 하여 세션키를 얻는데 따르는 시간적 소모와 암호의 악용을 해결하기 위해서 여러 가지 키 복구 시스템 중에서 대표적인 캡슐화 방식을 이용하여 세션 정보를 저장함으로써 SG의 재동작에 의한 세션 복구를 재협상이 없이 복구 가능하도록 하였으며, 이에 따른 시간적 소모도 해결하여 좀 더 효율적임을 보였다.

향후 연구 과제로는 무선에서 사용자의 세션 키를 안전하게 복구할 수 있는 키 복구 시스템과 이러한 키 복구 시스템을 이용한 VPN 설계이다.

5. 참고 문헌

[1] Dave Kosiur, "Building and Managing Virtual Private Networks?" John Wiley & Sons, 1998.

[2] Atkinson, R., "Security Architecture for the Internet Protocol", RFC 2401, NRL, November 1998.

[3] Brigit Pfizmann and Micheal Waidner, "How to Break Fraud Detectable Key Recovery", ACM Operating Systems Review 32, 1998.

[4] D. Maughan, M. Schertler, M. Schneider, J. Tunner, "Internet Security Association and Key Management Protocol(ISAKMP)", RFC 2408, NRL, November 1998.

[5] Tom Markham, Charles Williams. Key Recovery Header for IPSec, DRAFT Key Recovery Alliance Recommendation 2, April 1998.

[6] Atkinson, R., "IP Authentication Header", RFC 2402, NRL, November 1998.

[7] Atkinson, R., "IP Encapsulation Security Payload", RFC 2406, NRL, November 1998.

[8] Sabari Gupta, A Common Key Recovery Block Format: promoting Interoperability between dissimilar key recovery schemes, KRA white-paper, 1998.