멀티패스 SCTP의 경로검증을 위한 인증메커니즘

김강혁, 송주석
연세대학교 컴퓨터과학과
e-mail:{drface,jssong}@emerald.yonsei.ac.kr

Authentication mechanism for path verification in SCTP

GangHeok Kim, JooSeok Song
Dept of Computer Science, Yonsei University

요 약

SCTP(Stream Control Transmission Protocol)는 새로운 전송계층 프로토콜로 TCP와 UDP의 장점을 결합한 대체 프로토콜로 기대되고 있다. 다양한 환경에서의 연구와 함께 SCTP에서 보안서비스를 제공하는 보안메커니즘에 관한 연구도 활발히 진행되고 있다. SCTP를 위한 E2E 보안메커니즘은 대부분 4계층 이상에서의 보안서비스를 결합한 방식으로 제안되고 있으며, IP계층에서의 보안을 위한 IPSEC은 멀티호밍의 특성을 적절히 지원하지 못하고 있다. 그래서 IP계층에서의 IP주소인증 및 IP패킷의 보안서비스 제공을 위한 보안메커니즘이 요구된다. 본 논문에서는 멀티호밍의 SCTP에서 각 IP 주소를 인증하고 비밀키를 공유하는 인증메커니즘을 제안한다.

1. 서론

SCTP(Stream control transmission protocol)[1]는 원래 IETF의 SIGTRAN에서 PSTN망에서의 Signalling 목적을 위해 개발된 프로토콜이었다. 그러나 SCTP가 제공하는 멀티호밍과 멀티스트리밍 등 여러 특징들로 인해 다양한 응용프로그램의 요구조건을 충족할 것으로 기대되고 있다. 다양한 연구와 함께 SCTP의 E2E 보안에 관련된 연구도 진행되고 있다. TLS over SCTP, SCTP over IPSEC, S-SCTP, SS-SCTP, SCTP aware DTLS등 각종 보안메커니즘이 제안되고 있으며, 비교 및 실험이 진행되었다.[2][3] 또한 SCTP 청크의 인증을 위한 SCTP-AUTH도 RFC4895로 제정되어 표준화 되었다.

다양한 보안 메커니즘이 제안되고 있으나, 대부분 4계층 위로 보안서비스를 제공하며 여러 레이어의 보안서비스를 결합해 보안을 제공하는 메커니즘들이 제안되고 있으며, 멀티호밍의 특성으로 인해 IP계층에서의 보안서비스는 IPSEC으로도 적절히 지원하지 못하고 있다. 그래서 멑티호밍의 SCTP를 사용하는 호스트의 IP 주소인증 및 IP패킷의 보안서비스는 제공되지 못하고 있다. 이러한 서비스를 위해서 IP주소 인증을 위한 보안메커니즘이 필요하다.

본 논문에서는 경로인증은 물로 기밀성 제공을 위한 비밀키를 생성하기 위한 인증메커니즘을 제안한다. 2장에서는 제안된 보안 메커니즘에 대한 내용을 설명하고 과정을 기술하였고, 3장에서 결론 및 향후과제에 대해 설명한다.

*이 논문은 2008년도 정부(과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. R01-2006-000-10614-0)

2. 본론

SCTP에서는 멀티호밍을 지원하기 위해 하나의 어소시에이션으로 여러개의 IP를 관리할 수 있다. 그래서 두 개의 호스트가 서로간의 어소시에이션을 맺을 때 두 호스트는 가용한 IP 리스트를 인증없이 교환하게 된다. 그림1은 SCTP의 어소시에이션 과정을 나타내고 있다. 그림과 같이 SCTP는 DoS 공격을 방어하기 위해 4-way 핸드쉐이크 어소시에이션 과정을 거친다.

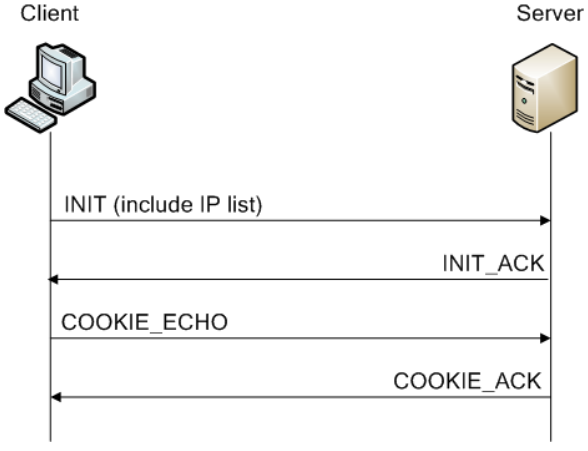

그림 1. SCTP 어소시에이션 과정

이렇게 교환된 IP리스트는 미확인 상태로 유지되다가, RTO시간 이후에 송신 호스트가 수신 호스트에게 HEARTBEAT 청크를 보내 응답이 오는 지를 통해 유효한 경로인지 또는 실제 소유하고 있는 IP인지를 확인하게 된다. 이러한 과정은 공격자에 의해 IP stealing공격을 당할 수 있는 취약점이 있으며, 경로인증을 위한 보안메커니즘이 요구된다.

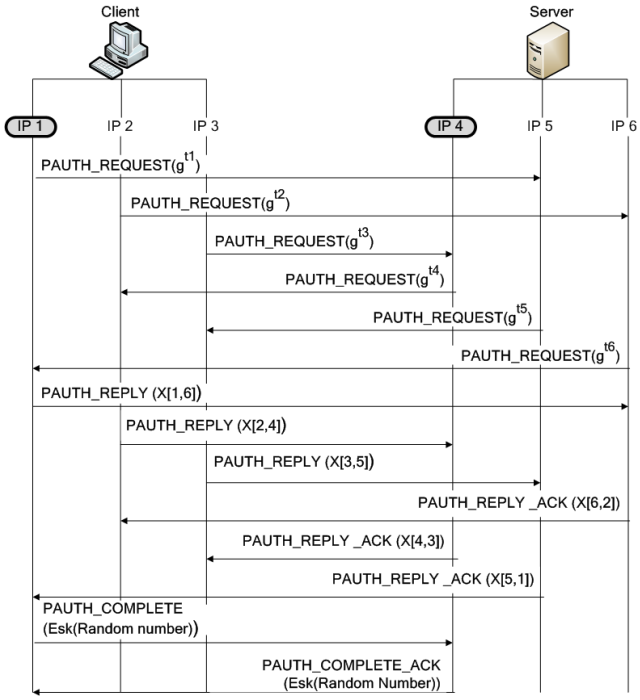

그림 2. 제안된 경로인증 보안메커니즘

본 논문에서는 경로인증을 위한 보안메커니즘을 위해 센서네트워크에서 사용하기 위해 제안된 BD-GKA protocol 를 기반으로 한다.[4] 그림2는 제안된 경로인증 보안메커니즘을 나타낸 것이다. 경로인증을 위한 보안인증 과정은 다음과 같다.

Round 1

1. 각 호스트에서 각 IP리스트 수만큼 t_i를 선택해서 $Z_i=g^{ti}$를 생성한다. 그래서각각의 송신IP로부터 각각의 Z_i를 목적지 IP로 보낸다. 목적지 IP는 IP리스트의 순으로 하되, 주경로를 제외한 IP부터 보낸다.

Round 2

2. 각각의 IP로부터 받은 Z_i로 $X_{[k,i,j]}=(Z_j/Z_k)_{ti}$를 구하여 비밀키 sk를 생성한다.

$$sk_{a_i} = (z_{a_{i-1}})^{mt_{ai}} \bullet X^{m-1_{a_i}} \bullet X^{m-2_{a_{i+1}}} \cdots\cdots X_{a_{i-2}}$$

Round 3

3. 비밀키를 사용하여 계산된 Xi를 암호화하여 보내고, 수신 호스트는 비밀키로 복호화하여 Xi의 값을 확인하여 송신IP를 인증한다.

비밀키 생성방식을 토대로, SCTP 어소시에이션 과정은 다음과 같다. 먼저 어소시에이션을 맺기를 시도하는 호스트, 클라이언트는 주경로(IP1)를 통해 서버(IP4)에게 INIT 청크를 보내 연결을 시도한다. 이때 두 호스트가 공유할 부분키 생성을 위한 g값을 선정해 서버에게 보낸다.

다음으로, g값을 포함하여 INIT청크를 받은 서버는 INIT_ACK 청크를 보내고 g값을 토대로 Z_4, Z_5, Z_6을 계산한다. 그리고 나서 두 호스트는 경로인증을 위해 PAUTH_REQUEST 청크에 부분키를 각 IP로 전송한다. 이 과정을 통해 하나의 IP를 도용하는 공격자의 경우에는 비밀키 생성을 위한 모든 부분키를 알지 못하기 때문에 경로인증에 성공할 수 없게 된다. 각 PAUTH-REQUEST 청크를 받은 두 호스트는 X_i를 각각 계산하여, 부분키 X_i를 상대 호스트 주경로(IP1↔ IP4)로 보내고 비밀키를 생성한다. 그러면 두 호스트는 생성된 비밀키를 통해 난수를 주고받아 서로 동일한 비밀키를 공유했는 지를 확인한다

3. 결론 및 향후 과제

SCTP는 멀티호밍, 멀티스트리밍 등 많은 특성들로 인해 TCP와 UDP를 대체할 프로토콜로 간주되고 있다. SCTP에 관한 많은 연구가 실제로 진행중에 있으며, 보안과 관련된 많은 프로토콜도 표준화 되었다. 그러나 멀티호밍 환경에서 IP에 관한 인증은 이루어지지 않아, IP stealing 공격 등 보안공격에 취약할 수가 있다. 본 논문에서 제안한 그룹키 보안메커니즘을 통해 교환되는 IP리스트는 서로 인증되어, 여러 보안 공격에 안전하며, 인증에 사용된 부분키를 사용하여 연결후 기밀성을 위한 비밀키로 사용되어, 성능면에서도 오버헤드가 크지 않다. 또한 키교환에 사용된 청크로부터 멀티패스의 상태를 측정하는 데에도 활용할 수 있는 이점이 있다. 향후 제안한 보안메커니즘과 관련하여 성능측정 및 시뮬레이션을 통하여 안전한 인증메커니즘이 되도록 발전시켜야 할 것이다.

참고문헌

[1] R. Stewart, “Stream control transmission protocol,” Sep, 2007.

[2] S. Lindskog and A. Brunstrom, “An end-to-end security solution for sctp,” in Proc. Third International Conference on Availability, Reliability and Security ARES 08, 2008, pp. 526 - 531.

[3] C. Hohendorf, P. Rathgeb, E. Unurkhaan, and M. Tuxen, “Secure end-to-end transport over sctp protocol,” in Telecommunication Systems, vol. 27, no. 2-4, June, 2007, pp. 31 - 40.

[4] Jarecki, S., Kim, J., and Tsudik, G. “Robust group key agreement using short broadcasts” In Proceedings of the 14th ACM Conference on Computer and Communications Security (Alexandria, Virginia, USA, October 28 - 31, 2007). CCS '07. ACM, New York, NY, 411-420.