능동 노드를 위한 보안강화엔진 설계

김옥경° 임지영 김여진 나가진 나현정 채기준 나중찬 김영수
이화여자대학교 컴퓨터학과, ETRI
{kimok°, jylim, zzin97, nagajin, hjna, kjchae}@ewha.ac.kr, {njc, blitzkrieg}@etri.re.kr

Design of Security Enforcement Engine for Active Nodes

Okkyeung Kim° Jiyoung Lim, Hyunjung Na, Gajin Na, Yeojin Kim, Kijoon Chae
Jungchan Na Youngsoo Kim
Dept. of Computer Science and Engineering, Ewha Womans University
Information Security Technology Division, ETRI

요 약

본 논문은 액티브 네트워크 환경에서 액티브 노드를 위한 보안강화엔진의 구조와 기능을 설계하였다. 액티브 노드의 자원에 접근 시 발생되는 보안상의 문제점들을 해결하기 위한 보안강화엔진 구조를 제안하고 보안강화엔진 내에 Security, Authentication, Authorization 모듈을 두어 액티브 네트워크 환경에 노출되어있는 악의적인 위협 요소들로부터 액티브 노드들을 보호하고자 하였다. 본 논문에서는 보안강화엔진에서 Security, Authentication, Authorization 모듈의 설계 내용에 대해 기술한다.

1. 서 론

종단의 단말에만 집중되어있는 네트워크의 기능을 분산시키고 사용자의 망에 대한 요구를 적절하고 빠르게 반영하여 네트워크에 유연성을 제공하기 위해 액티브 네트워크라는 새로운 패러다임이 등장하였다. 액티브 네트워크란 라우터나 스위치가 프로그램 실행 능력을 가지고 있어서 프로그램을 포함하거나 또는 중간 노드의 프로그램을 실행하도록 하는 패킷을 처리하여 다양하고 유동적인 처리를 패킷에 행할 수 있는 환경을 가진 망을 말한다[1,2]. 액티브 네트워크에서 특별한 일을 수행하기 위하여 프로그램을 포함하고 있는 패킷을 액티브 패킷이라고 하고, 프로그래밍 수행 능력을 가지고 액티브 패킷을 처리할 수 있는 라우터를 액티브 라우터, 또는 액티브 노드라고 한다.

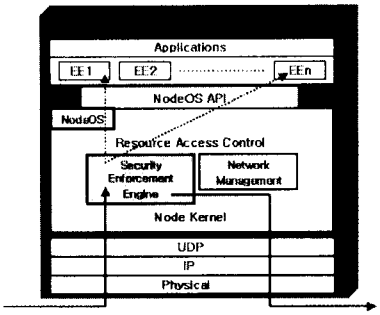
액티브 네트워크 기술은 중간 노드에서 여러 가지 처리를 가능하게 함으로써 기존의 네트워크가 제공하지 못하는 유연성과 다양한 장점을 제공할 수 있다. 하지만 동적이고 유연한 액티브 네트워크의 장점들이 보안의 측면에서는 매우 위험한 요소가 될 수 있다. 네트워크의 악의적인 사용자가 잘못된 코드 혹은 악의적인 코드를 네트워크내에서 실행함으로써 전체 네트워크에 결정적인 영향을 끼칠 수 있기 때문이다. 때문에 액티브 네트워크에서의 보안은 기존의 네트워크에서 보다 훨씬 더 중요한 이슈가 되고 있고, 이에 대한 활발한 연구가 진행중이다. 액티브 네트워크에서 보안을 제공하기 위해서는 기본적으로 인증과 권한부여가 필요하다. 인증과 권한부여에 대한 정보는 패킷 내에 첨가되어 전송된다. 또한 패킷 자체의 무결성을 보장하기 위해 전자 서명이 패킷 내에 첨가되어야 한다.

본 논문에서는 액티브 패킷이 중간 노드에서 실행되는 경우 발생하는 여러 가지 보안상의 문제점을 해결하기 위한 보안강화엔진 구조를 제안하고 그 설계 내용에 대해 기술하고자 한다.

2. 제안된 보안강화엔진 구조

액티브 네트워크는 전송중인 패킷의 프로그램 코드를 라우터에서 실행할 수 있으며 코드의 실행결과에 따라 라우터의 상태를 변경할 수 있음을 DARPA[3]에서 제안하였다. 이러한 액티브 네트워크는 패킷을 단순히 전달하는 기능만을 지닌 기존의 패시브 네트워크에 비해 유연성이 있는 반면, 심각한 보안 문제를 가지고 있다. 따라서 외부의 의도적인 공격에 대한 방어와 노드 자체 내의 안전성을 위해서 다양한 위협에 대응할 수 있는 안전한 액티브 노드 구조가 필요하다.

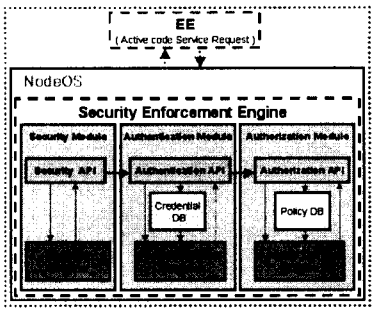
본 논문에서 제안한 보안성이 추가된 액티브 노드 구조는 < 그림 1 >과 같다.



< 그림 1 > 액티브 노드 구조

액티브 노드의 NodeOS 내에 위치하는 보안강화 엔진은 다음과 같은 기능을 수행한다. 노드로 들어온 패킷에 대한 무결성 검사와 암호화 / 복호화, 검증을 수행하고 수행한 결과가 안전하다고 판정되면 패킷에게 적절한 자원을 할당하고 패킷이 EE 에서 실행하도록 처리된다. 모두 처리된 후에는 다시 보안강화엔진을 거쳐 변경된 부분에 대한 무결성 검사와 암호화 / 복호화, Digital Signature 등을 처리한 후 다음 노드로 전달된다.

< 그림 2 >는 액티브 노드 내에 위치하는 보안강화 엔진의 상세 구조이다. 본 논문에서 제안한 보안강화 엔진은 Security, Authentication, Authorization 모듈로 구성되며 이 세 모듈간의 긴밀한 상호작용으로 액티브 네트워크 상에서의 안전한 통신을 가능하게 한다. Security 모듈과 Authentication 모듈을 통해 인증되고 Authorization 모듈을 통해 권한부여 서비스가 제공된다.

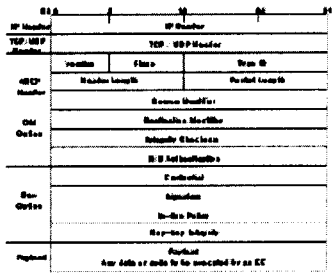


< 그림 2 > Security Enforcement Engine

액티브 패킷의 포맷으로는 ANEP(Active Network Encapsulation Protocol)[4] 패킷 구조를 이용하였고 패킷 포맷은 < 그림 3 >과 같다.

통신 프로토콜 스택은 기본적으로 IP 프로토콜을 사용하며 상위 계층으로 TCP 또는 UTP 프로토콜 모두 가능함을 제안한다.

본 논문에서는 IP 헤더와 UDP 헤더 다음에 ANEP 헤더를 붙이고 ANEP 헤더의 Old Option 과 New Option



< 그림 3 > ANEP 패킷 포맷

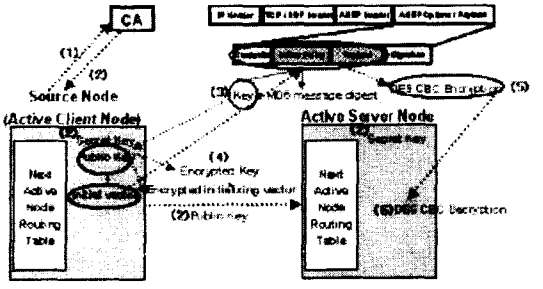
을 붙였다. 액티브 노드에 전달하고자 하는 데이터나 실행할 코드들은 Payload 필드에 담아서 보낸다. 액티브 패킷의 생성자와 소스노드를 검증하고 인증하기 위해 Credential 필드를 이용하며 X.509 포맷의 Credential 을 이용하여 구현하였다. 그리고 수신한 액티브 패킷의 무결성 검증하기 위해 Signature 필드를 생성하며 수신한 액티브 패킷의 자원이용에 대한 권한부여를 위해 In-line policy 를 이용한다. 또한 소스 액티브 노드 뿐만이 아니라 중간 홉에서의 인증, 메시지 무결성 등을 검증하기 위해 Hop-hop Integrity 를 이용할 수 있다. 중간 홉에서의 인증을 추가할 경우 해당 액티브 노드는 Credential 과 Signature 필드를 액티브 패킷에 추가로 첨가해야 한다.

2.1 Security 모듈

이 모듈에서는 메시지의 무결성을 검증하기 위해 처리되어 지는 부분이다. 사용한 알고리즘들은 다음과 같다. 메시지나 코드의 길이를 128bit 로 축약하는 Massage digest MD5 알고리즘, DES CBC 암호화와 복호화 알고리즘 그리고 수신한 메시지의 무결성을 검증하기위해 RSA Digital Signature 와 Verification 알고리즘을 적용하였다. 이 알고리즘들은 액티브 패킷내의 Credential, Signature, In-line policy, Payload 필드에 각각 적용하여 구현하였다.

< 그림 4 >는 Massage digest 와 암/복호화 동작 과정이다. 모든 과정은 사전에 CA (Certificate Authority) 로부터 Credential 과 비밀키, 공개키를 부여받고 난 다음 암 / 복호화, Signature 과정이 진행된다.

DES CBC 암호화를 하기 위해서는 자신의 공개키와 Initial Vector 값이 필요하고, 복호화를 위해 암호화된 공개키와 암호화된 Initial Vector 값이 필요하다. 그리고 이 알고리즘을 적용하기 전에 MD5 Massage Digest 알고리즘을 적용하고, (3)은 MD5 Message Digest 알고리즘을 적용하는 과정이다. (4)는 복호화를 위해 암호화된 공개키와 암호화된 Initial Vector 값을 다음 액티브 노드에 보내는 과정이다. (5)는 MD5 Message



< 그림 4 > Security 모듈에서의 암호화/복호화 동작

Digest 의 결과인 키 값과 Initial Vector 값으로 DES CBC 암호화를 수행하는 과정이고, (6)은 암호화된 공개키와 암호화된 Initial Vector 값으로 DES CBC 복호화 하는 과정이다.

다음 Signature 를 위해서는 소스 액티브 노드의 비밀키를 이용하여 메시지를 RSA Digital Signature 알고리즘으로 처리하고, 수신측은 수신 받은 액티브 패킷의 Signature 필드를 검증한다.

2.2 Authentication 모듈

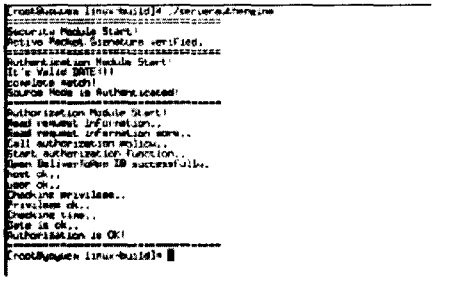
이 모듈에서는 소스 액티브 노드와 액티브 패킷을 보낸 송신자의 신원을 인증하기 위해 처리되어 지는 부분을 구현하였다. Security 모듈에서 처리과정이 성공적으로 수행되고 나면 Authentication 모듈에서는 액티브 패킷의 Credential 필드를 추출하여 액티브 패킷의 Credential 내용과 액티브 노드의 Credential DB 에 저장중인 CA 로부터 받은 소스 액티브 노드의 Credential 내용과 비교하여, 액티브 패킷이 유효한 액티브 노드와 사용자로부터 발신된 액티브 패킷임을 보증해 주게 된다.

2.3 Authorization 모듈

Authentication 모듈에서의 처리 후, 적합한 액티브 패킷이라 판단되면 Authorization 모듈로 넘어오고, 그렇지 않다면 액티브 패킷을 폐기시킨다. 이 모듈에서는 패킷의 In-line Policy 옵션필드(**C(Create),M(Modify),A(Append),R(Read),D(Delete),E(Execute)**)의 내용을 기반으로 권한 부여 기능을 수행한다. 인증된 액티브 패킷의 In-line Policy 옵션 필드에는 노드에서의 실행을 위한 자원 할당 요청이 들어 있다. 각 액티브 노드는 Policy DB 를 갖고 있으며, Policy DB 는 그 노드에서 실행 가능한 각 서비스 별 정책을 포함하고 있다. 해당 서비스의 정책에는 어떤 호스트와 사용자가 얼마만큼의 권한을 갖고 실행될 수 있는가가 명시되어 있다. 즉 액티브 패킷의 실행을 위한 권한을 줄 것인지 체크하는 기능을 담당한다.

3. 구현 환경

플랫폼으로는 Linux Red hat 7.3 version 으로 하고 NodeOS 의 기능을 대체하기 위해 ABONE 에서 제공하는 Anetd (Active Networks Daemon)를 설치하여 구현하였다. < 그림 5 >는 제안한 보안강화엔진을 통과한 구현결과 화면이다.



< 그림 5 > 보안강화엔진을 성공적으로 통과한 결과

4. 결 론

본 논문에서는 보안상의 문제점들을 해결하여 액티브 네트워크의 전체적인 보안성을 강화하기 위한 보안 엔진에 대해 제안하였다. 제안된 보안강화엔진은 Security, Authentication, 그리고 Authorization 모듈로 구성되었다. 본 논문에서 제안하는 보안강화엔진은 암호화함으로써 액티브 패킷이 네트워크를 통해 전송되는 동안 변질되지 않았고 메시지가 무결함을 보증하고, CA 로부터 받은 인증서를 비교함으로써 이 패킷을 보낸 사용자와 액티브 노드가 정당함을 인증한다. 또한 이렇게 정당하다고 인증된 패킷에게 부여된 권한 레벨을 체크하여 NodeOS 가 이 패킷에게 적합한 권한을 부여할 수 있도록 도와서 정당하지 않은 패킷이 NodeOS 의 자원에 불법적으로 침입하는 것을 방지한다.

참고문헌

[1] R. H. Campbell, et al., " Seraphim: Dynamic Interoperable Security Architecture for Active Networks," IEEE OPENARCH 2000, Tel-Aviv, Israel, Mar. 2000.

[2] Leon Dang, " CANSA (Certificate Active Network Security Architecture)," Basser Department of Computer Science, University of Sydney, 1998.

[3] Defense Advanced research Projects Agency, http://www.darpa.mil/ato/programs/activenetworks/actnet.htm.

[4] D. Scott Alexander, Bob Braden, Carl A. Gunter, Alden W. Jackson, Angelos D. Keromytis, Gary J. Minden, and David Wetherall, "Active Network Encapsulation Protocol (ANEP),"1997.