이동 코드를 이용한 공격자 대응 프레임워크

방 효 찬[†] · 김 진 오[†] · 나 중 찬[††] · 장 종 수[†††] · 이 영 석[††††]

요 약

사이버 공격의 형태가 나날이 다양해지고 복잡해지는데 반해 기존의 네트워크 보안 메커니즘은 지엽적인 영역의 방어적인 대응에 치중하고 있어 적시에 공격자를 탐지하고 신속하게 대응하는 것이 어려운 실정이다. 본 논문에서는 이러한 문제점을 해결하기 위한 방안으로 광역 네트워크 차원에서 다양한 사이버 공격에 쉽게 대응할 수 있고, 다수의 보안영역 간의 협력을 통해 공격자를 실시간으로 추적하고 고립화할 수 있는 새로운 네트워크 보안 구조를 제안하고자 한다. 제안하는 보안 구조는 액티브 네트워크를 기반으로 하는 이동코드를 포함한 액티브 패킷 기술을 이용하여 위조 IP 공격이나 DDoS(Distributed Denial of Service) 공격에 대하여 자율적이고 능동적으로 대응하는 것이 가능하다. 또한 다수의 보안영역 내의 보안 시스템 간의 협업을 통해 기존의 지엽적인 공격 대응 방식에 비해 보다 광역적이고 일관적인 보안 서비스를 제공한다. 또한, 본 논문에서는 제안한 공격자 대응 프레임워크의 실험 환경을 구축하고 실험한 결과를 분석함으로써 적용 가능성을 검증하였다.

Attacker Response Framework using Mobile Code

Hyo-Chan Bang[†] · Jin-Oh Kim[†] · Jung-Chan Na[††]
Joong-Su Jang[†††] · Young-Suk Lee[††††]

ABSTRACT

It has become more difficult to correspond an cyber attack quickly as patterns of attack become various and complex. However, current security mechanisms just have passive defense functionalities. In this paper, we propose new network security architecture to respond various cyber attacks rapidly and to chase and isolate the attackers through cooperation between security zones. The proposed architecture makes it possible to deal effectively with cyber attacks such as IP spoofing or DDoS(Distributed Denial of Service), by using active packet technology including a mobile code on active network. Also, it is designed to have more active correspondent than that of existing mechanisms. We implemented these mechanisms in Linux routers and experimented on a testbed to verify realization possibility of attacker response framework using mobile code. The experimentation results are analyzed.

키워드 : 액티브 네트워크(Active Network), 분산 서비스거부 공격(Distributed Denial-of-Service Attack), 공격자 고립(Attacker Isolation)

1. 서 론

네트워크는 컴퓨터 시스템간의 상호접속 및 정보 교환 등의 편리한 역할을 제공 하지만, 시스템에 대한 불특정 다수의 접근이 가능하기 때문에 시스템 침입자에 의한 보안 사고의 위협을 내포하고 있다. 특히, 네트워크의 물리적인 광범위함, 네트워크 경로 및 사용자의 다양성 등은 네트워크 상에서 특유의 보안 문제를 일으키며, 네트워크 구성요소 중 일부에 문제가 발생하더라도 전체 네트워크에 영향을 미칠 수 있다. 또한 최근 심각한 피해를 입히고 있는 웜 바이러스 형태의 해킹 기법은 수분 내지 수십 분 내에 해당되는 지역이나 공공기관 기간망을 마비시킬 수 있는 피해를 줄 수 있다.

따라서 사이버 공격을 시도하는 침입자에 대해 기존의 네트워크 보안에서 이루어지는 것 보다 좀 더 강력하고 능동적인 대응과 서비스의 품질을 보호하기 위한 보안 기술의 개발이 요구되고 있다.

네트워크 인프라 환경은 네트워크 자체의 목적을 최소화하기 위해 지금까지 많은 노력이 진행되어 왔다. 하지만 보안사고의 네트워크 위협에 대한 대응 프레임워크는 아직 현실화되지 않은 문제점들이 있다. 첫째, 기존의 네트워크 보안은 자신의 관리 도메인 내로 침입하는 공격을 어떻게 잘 탐지할 것인가와 탐지된 공격을 어떻게 효율적으로 차단하여 자신의 도메인을 잘 보호할 것인가에 초점이 맞추어져 있다. 이 경우 자신의 도메인에서 파악한 침입자 정보를 바탕으로 자신의 도메인 입구에서만 해당 트래픽을 차단함으로써 침입자는 자유로이 인터넷을 이용할 수 있을 뿐만 아니라, 다른 공격 기술이나 공격 루트를 이용한 제2, 제3의

† 정 회 원 : ETRI 능동보안기술연구팀 선임연구원
†† 정 회 원 : ETRI 능동보안기술연구팀 팀장
††† 정 회 원 : ETRI 네트워크보안그룹 그룹장
†††† 정 회 원 : 국립금산대학교 전자정보학부
논문접수 : 2004년 9월 24일, 심사완료 : 2004년 12월 7일

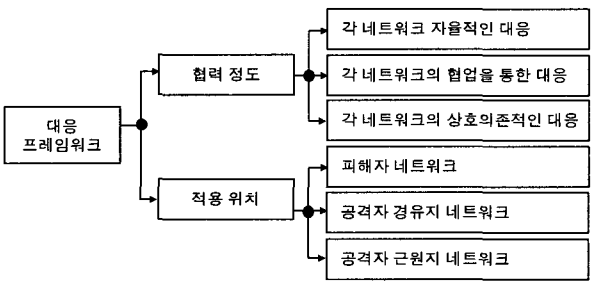
공격이 이루어 질 수 있다. 둘째, 최근 대부분의 네트워크 공격 형태는 하나의 에이전트에서 하나의 시스템을 공격하는 것이 아니라, 분산된 여러 에이전트에서 하나의 목표 시스템에 패킷을 범람시키는 DDoS 공격이 이루어지고 있다. 그러나 인터넷 관리는 분산화되어 지역적인 정책에 따라 각 네트워크 운영이 되고 있는 실정이다. 이러한 관리 환경은 네트워크 전체 차원에서의 특정한 한 보안 메커니즘 또는 보안 정책으로 보안을 강화하기는 더욱 어렵게 하고 있다. 셋째, 새로운 보안 기능 추가 시에 하드웨어 및 시스템의 교체가 수반되는 등 이종의 보안 장치 간, 이종의 네트워크 간, 이종의 사업자간의 상호 연동의 보안 서비스 환경을 제공하기 어렵고, 사이버 공격에 대응하기 위해 사용자 종단간의 안전하고 효율적인 보안 관리는 한계점이 있다.

이러한 문제점을 해결하기 위해서는 광역 네트워크 차원에서 공격을 탐지하고 대응할 수 있는 네트워크 보안구조와 보안환경 변화에 유연하게 적응할 수 있는 보안 응용 프로그램 구조 및 보안 도메인간의 협업을 통해 일원적인 대응 결정과 분산적인 대응 실행이 가능한 보안 서비스 환경의 구축이 필요하다.

본 논문에서는 이러한 새로운 네트워크 보안 요구사항을 만족하는 보안 프레임워크를 제시하고자 한다. 본 논문에서는 사이버 공격의 대응 기술과 관련된 외국의 연구동향과 국내의 연구개발 현황을 알아보고, 제안하는 이동 코드를 이용한 공격자 대응 프레임워크에 대하여 구체적으로 기술한다. 마지막으로 실험 환경 상에서 제안한 프레임워크의 실험 과정을 기술하고 도출된 실험 결과를 분석하여, 결론을 맺는다.

2. 관련 연구

네트워크 인프라의 공격에 대한 효과적인 대응을 위해서는 기존의 네트워크 보안에서 이루어지는 것 보다 좀 더 강력하고 능동적인 대응과 사용자 요구에 적합한 고객 지향 서비스를 지원하고 서비스의 품질을 보호하기 위한 대응 프레임워크를 제공해야 한다. 본 장에서는 (그림 2-1)과 같은 관점에서 네트워크 공격자 대응 프레임워크와 관련된 연구를 살펴본다.



(그림 2-1) 공격자 대응 프레임워크 뷰

2.1 DARPA's IDIP

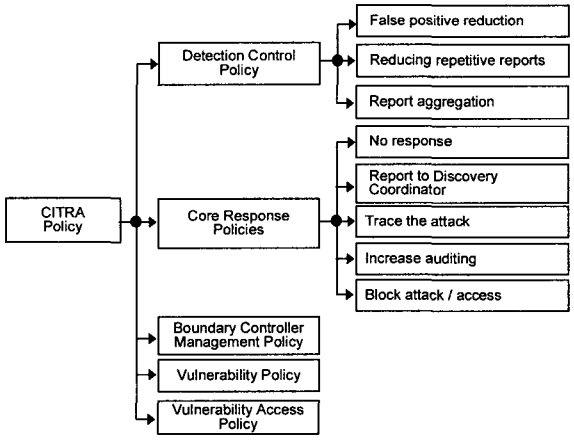
IDIP(Intrusion Detection Isolation Protocol)은 침입탐지 시스템, 방화벽, 호스트, 보안관리 관련 요소시스템들 간의 협력 작업을 통해 공격자의 실제 위치를 역추적하여 공격자를 네트워크로부터 고립화시키기 위한 프로토콜을 포함한 보안 기반구조로써, 미국 DARPA(Defense Advanced Research Projects Agency) SLSS(Survivability of Large Scale Systems) 프로그램의 일환으로 수행된 연구이다[1].

그러나 이 프로토콜은 다음의 결점을 갖는다. 첫 번째로 호스트에서 모든 연결에 대한 감시 기능을 수행해야 하며, 둘째 네트워크 도메인상의 모든 네트워크 노드들은 자신이 라우팅하는 모든 패킷에 대해 로그 정보를 유지할 수 있어야 하며, 마지막으로 IDIP가 실제 적용되기 위해서는 프로토콜 스택으로써 구현되어 시스템에 구축되어야 하는 정적인 문제점을 갖는다.

2.2 DARPA's CITRA

CITRA(Cooperative Intrusion Traceback and Response Architecture)는 공격자 역추적 및 고립화 기능을 프로토콜 형태로 구현하기 위한 목적을 가진 IDIP 과제로부터 시작되었다.

CITRA는 DARPA's IDIP의 방법을 그대로 사용하면서 (그림 2-2)와 같은 대응정책을 추가하여 운영하고자 하였다[2].



(그림 2-2) CITRA 정책

이전의 역추적 및 대응이 도메인에서의 공격 경로 상 마지막 노드인 경우에는 종료되었던 이전의 연구와는 달리, CITRA는 DARPA의 MCCD(Multi-Community Cyber Defense)과제를 통해 다수의 도메인간의 역추적 및 대응을 위해 확장되었다[3].

2.3 DARPA's AN-IDR

AN-IDR(Active Network-Intrusion Detection and Response)은 IDIP가 프로토콜로 구현될 경우에 발생하는 기능

변경의 정적인 특성으로 인한 유연성의 부족함과 특정 기능 수행 상에 있어서의 효율성 저하를 해결하기 위해 시작되었다. 이를 위해 IDIP 메커니즘과 액티브 네트워크 기술을 결합하여 상호 운용함으로써 기존의 정적인 IDIP에 이동성(mobility), 유연성(flexibility), 확장성(extensibility)을 부여함으로써 좀더 발전된 침입자 탐지 추적 기능을 수행하고자 하였다[4, 5].

AN-IDR의 경우 단순히 공격자의 추적 및 고립화뿐만 아니라, 액티브 패킷을 이용하여 공격용 툴로써 설치된 에이전트 프로그램을 스캐닝하고 해당 에이전트의 실행을 중지시키는 것과 같이 침해된 시스템을 복구하는 기능도 포함하였다.

2.4 IST's FAIN

FAIN(Future Active IP Network) 프로젝트는 IST(Information Society Technologies) 프로그램 산하에 유럽 8개국과 일본 및 미국 등 총 10개국 15개 기관이 컨소시엄 형태로 참여하는 유럽 중심의 연합 프로젝트로써 차세대 액티브 IP 네트워크 프레임워크에 관련된 연구를 활발히 수행하고 있다[6].

특히 FAIN(Future Active IP Network)의 프레임워크 상에 침입탐지와 대응 기능을 수행하는 이동 에이전트와 이들 간의 협력을 통해 DDoS 공격을 탐지하고, 라우터의 라우팅 테이블을 변경함으로써 특정 서브넷으로 향하는 모든 공격 트래픽을 차단하거나 해당 노드의 구성 및 정책을 변경하는 등의 대응 기법에 대한 연구를 수행하였다[7]. 또한 피해자 도메인의 경계에서 해당 공격자의 트래픽을 차단하는 것이 아니라, 에이전트와의 협력을 통해 공격자의 실제 위치를 역추적을 수행하고 공격자의 네트워크에 대한 접근성을 차단함으로써 고립화시키는 보다 강력한 대응을 수행할 수 있는 네트워크 보안 프레임워크를 구축하는 것을 목적으로 하고 있다.

2.5 기 타

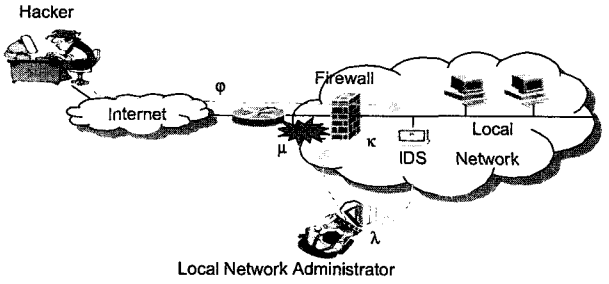
최근 중앙 집중화된 통합보안관리의 연산 과부하를 해결하기 위해 액티브 네트워크 기술과 네트워크 보안관리 기술과의 접목은 정보공유를 통해 상호 협력하게 함으로써 신속한 자동 대응이 가능한 예를 제공하고 있다[8, 9].

3. 공격자 대응 프레임워크

기존의 정보보호 방식은 시스템 설계 단계부터 반영된 것이 아니기 때문에 서비스 제공 이후에 발생 가능한 다양한 취약점 공격에 대한 효과적인 대응에 태생적 한계를 지니고 있다. 따라서 본 장에서는 침입에 대한 탐지 및 역추적, 대응 등의 기능을 효율적으로 수행할 수 있는 공격자 대응을 위한 프레임워크에 대하여 기술한다[13].

3.1 공격자 대응 프레임워크 고려사항

일반적으로 네트워크 보안 기술은 (그림 3-1)과 같이 특정 조직의 해당 도메인을 보호하기 위한 것으로 초점이 맞추어져 있다.



(그림 3-1) 현재의 네트워크 보안 시스템 구조

보안을 보장하기 위한 시스템은 크게 2가지로 구분될 수 있다. 침입 징후를 탐지하기 위한 침입 탐지 시스템과 탐지된 해당 침입자의 트래픽의 차단을 주목적으로 하는 방화벽이나 패킷 필터링 라우터와 같이 자신의 도메인을 보호하기 위한 대응 시스템이다. 초기에는 각 시스템이 별도로 운용되어 두 시스템간의 상호 연동을 위해서는 관리자의 개입이 필요하였으나 현재에는 두 시스템을 상호 결합하여 운용함으로써 탐지와 그에 따른 트래픽의 단절이 동시에 관리자의 개입이 없이 이루어지는 통합 보안 시스템이 주류를 이룬다. 최근에는 IDS와 F/W 연동 솔루션보다는 인라인에서 패킷을 분석하고, 빠르게 웜 바이러스를 감지해 바로 폐기하며, 또한 오탐이 많을 경우에는 트래픽을 제한하여 정상서비스 차단 확률을 최소화하는 IPS(Intrusion Prevention System)으로 출시되고 있다.

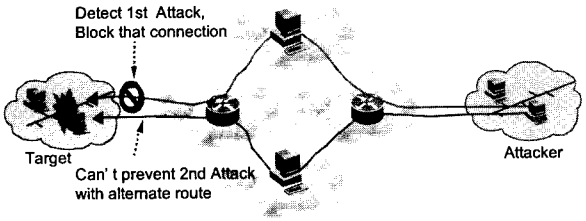
그러나 현재의 네트워크 보안 시스템이 가지는 한계는 해당 시스템의 구조가 급격하게 변화하는 보안 환경에 따라 쉽게 적응할 수 있는 유연성이 부족할 뿐만 아니라 해당 도메인만을 보호하기 때문에 침입자는 자유롭게 네트워크를 이용할 수 있다. 따라서 동일 시스템에 대한 추가적인 공격과 다른 도메인에 존재하는 다른 시스템에 대한 추가적인 공격이 가능하다는 단점이 있다. 따라서 공격자 대응 프레임워크를 설계함에 있어 다음과 같은 사항을 고려해야 한다.

3.1.1 네트워크 차원에서의 탐지 및 대응

현재 네트워크 보안 시스템은 해당 도메인에서 공격에 대한 지엽적인 판단과 지엽적인 대응만을 수행하는 한계를 내포하고 있다.

예를 들어, (그림 3-2)에서 보면 전체 네트워크 도메인 상에서 보면 동일한 침입자임에도 불구하고 1번째 공격과 2번째 공격 시 경유하는 중간 호스트를 달리 할 경우, 피해 도메인에서는 1번째 공격의 중간 경우 호스트를 기준으로 해당 트래픽을 차단함으로써 2번째 공격에 대해서는 똑같은

취약성을 가지게 된다.

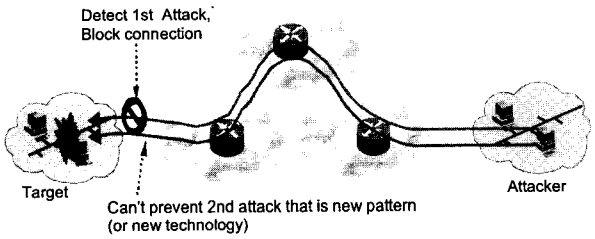


(그림 3-2) 다른 경유지를 이용하여 재침입하는 경우

3.1.2 보안환경변화에 대한 적응성 강화

동일한 공격에 대해서 전체 네트워크의 다른 부분에서 인식하게 되는 정보와 전체 네트워크 차원에서 해당 데이터를 상호 결합하는 기능이 부족하고, 침입자에 대한 대응에 있어서도 각 도메인 간의 협력이 없는 상태이다.

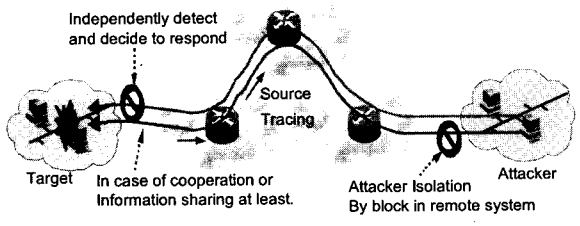
(그림 3-3)에서처럼 전체 네트워크를 구성하는 각 도메인 간의 데이터의 상호 결합, 대응에 있어서 상호 협력이 가능하다면 침입자의 실제 위치를 추적하여 해당 침입자를 네트워크로부터 단절시키는 것과 같은 좀더 강력한 대응이 가능할 것이다.



(그림 3-3) 새로운 기술을 적용하여 재공격하는 경우

3.1.3 협력을 통한 침입자 대응

동일한 공격에 대해서 전체 네트워크의 다른 부분에서 인식하게 되는 정보와 전체 네트워크 차원에서 해당 데이터를 상호 결합하는 기능이 부족하고, 침입자에 대한 대응에 있어서도 각 도메인 간의 협력이 없는 문제점이 있다.



(그림 3-4) 각 도메인 간의 협력을 통한 대응

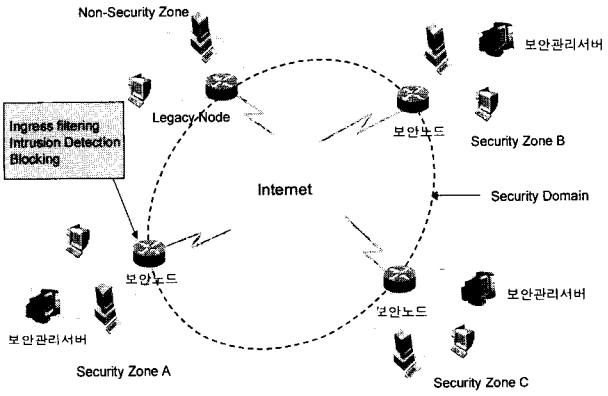
(그림 3-4)에서처럼 전체 네트워크를 구성하는 각 도메인 간의 데이터의 상호 결합, 대응에 있어서 상호 협력이 가능하다면 침입자의 실제 위치를 추적하여 해당 침입자를 네트워크로부터 단절시키는 것과 같은 좀더 강력한 대응이 가능할 것이다.

3.1.4 일원화된 분산 대응 시스템이 필요

대응 시스템은 인터넷상의 다양한 공격자와 피해자의 종류를 감당하기 위해서는 여러 지점에 일원화되게 적용하는 것은 매우 중요하다. 특히 DDoS 공격인 경우에는 보다 강력한 대응을 위해서는 분산되어 있는 각 공격 에이전트에 대한 대응 시스템도 분산되어 적용해야한다.

3.2. 공격자 대응 프레임워크 설계

본 논문에서 제안된 공격자 대응 프레임워크의 네트워크 구성도는 (그림 3-5)에 도시한 바와 같이 보안관리 영역(Security Zone)의 경계에서 이동코드 처리 및 보안 대응 기능을 제공하는 두 개의 시스템(보안노드와 보안관리서버)으로 구성되며, 두 시스템이 연동하여 하나의 보안관리 영역을 관리하고 제어한다.



(그림 3-5) 공격자 대응 프레임워크 네트워크 구성도

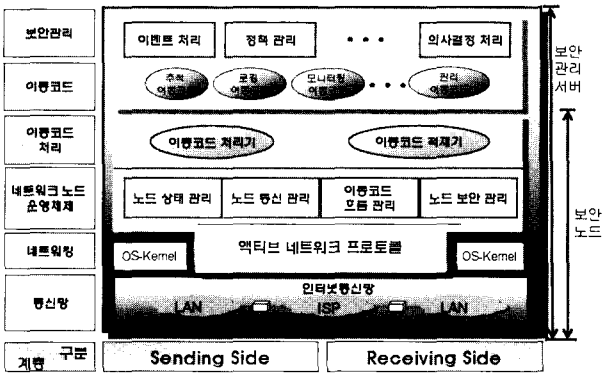
각 보안관리 영역은 전체 네트워크 상에 분산적으로 배치되어 상호간의 연동 및 협업을 수행하지만, 이를 위한 별도의 관리 계층은 갖지 않는다. 즉, 모든 보안 제어는 이동코드를 통해 이루어지며 보안관리 영역 간의 상호 연동과 협업 역시 이동코드에 의해 수행된다.

각각의 보안관리 영역은 (그림 3-5)에 도시한 바와 같이 이동코드를 통해 상호 연동함으로써 광역 네트워크 상에 논리적인 보안관리 도메인(Security Domain)을 형성한다. 이와 같이 기존의 네트워크(인터넷 백본)에 배치되어 있는 네트워크 시스템의 구성에 대한 변경 없이 새로운 보안 관리 영역을 형성할 수 있는 것이 공격자 대응 프레임워크의 큰 특징 중 하나이다.

3.2.1 공격자 대응 프레임워크 구성

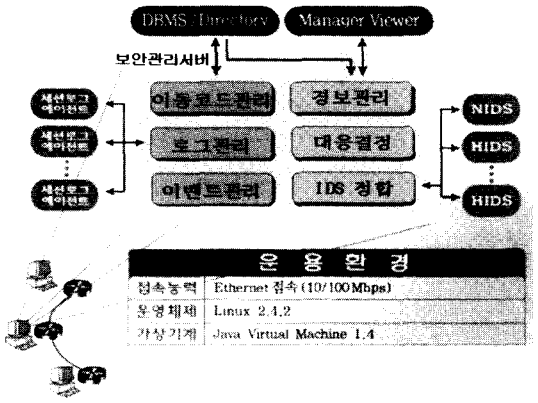
공격자 대응 프레임워크는 (그림 3-6)에 도시한 바와 같이 보안노드와 보안관리서버로 구성된다. 보안노드 및 보안

관리서버에는 이동코드를 수신하고 실행시킬 수 있는 이동코드 처리기가 공통적으로 탑재된다. 또한, 보안관리서버에는 이벤트관리, 의사결정처리 등의 보안관리를 위한 기능과 이동코드 및 정책을 관리하기 위한 저장소가 추가적으로 탑재되며, 보안노드는 이동코드에 의해 네트워크 차원의 대응 기능을 제공한다.



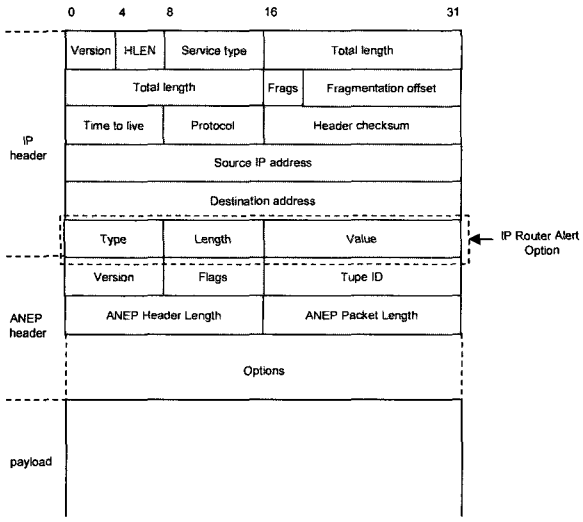
(그림 3-6) 공격자 대응 프레임워크 구성도

보안관리서버는 보안관리 영역 내에 배치된 침입탐지시스템 및 방화벽시스템으로부터 보고된 침입 행위에 대하여 이에 적합한 보안 대응을 결정하고, 이를 실행시킬 이동코드를 생성하여 네트워크에 송신함으로써 네트워크 차원의 보안상태를 동적으로 보안제어를 수행한다. 즉, 보안관리 영역 내의 보안노드를 제어함으로써 자신의 보안관리 영역을 관리하며, 다른 보안관리 영역을 관리 하는 보안관리서버와의 협업을 통해 전역적인 네트워크 보안관리 기능을 수행한다. 공격자 대응 프레임워크 상호간의 모든 제어 및 관리는 이동코드에 의해 수행된다. (그림 3-7)은 보안관리서버의 구조를 보여준다.



(그림 3-7) 보안관리서버 구조

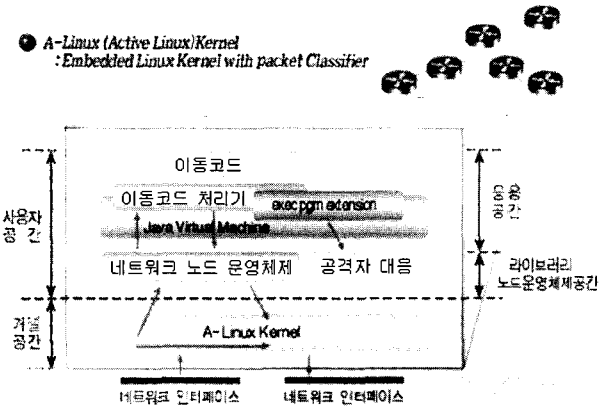
이동코드는 네트워크에 존재하는 다른 일반 노드에서도 전달될 수 있도록 기존 네트워크에서 사용하는 IP 패킷 형태로 구성한다. 이 IP 패킷을 액티브 패킷라고 하며, (그림 3-8)과 같은 구조를 갖는다.



(그림 3-8) 액티브 패킷 구조

액티브 패킷 헤더는 패킷의 IP 헤더와 ANEP(Active Network Encapsulation Protocol) 헤더로 구성되며, 페이로드에는 이동코드가 포함된다[10]. 'IP Router Alert Option'은 라우터가 패킷의 목적지 주소가 자신이 아닌 패킷을 가로챌 수 있도록 해주는 옵션으로써 일반 IP 패킷과 액티브 패킷을 구분하는 표시자 역할을 위해 활용한다[11].

보안노드는 (그림 3-9)과 같이 이동코드처리기, 액티브 패킷을 처리하는 A-Linux 커널, 공격자 대응, 그리고 보안노드 자원을 관리하는 네트워크노드 운영체제 등으로 구성된 구조를 갖는다.



(그림 3-9) 보안노드 구조

보안 노드는 보안관리 영역의 경계(가입자 네트워크의 에지라우터)에 이동코드처리 기능과 네트워크 차원의 보안대응을 수행하는 기능을 탑재한 시스템이다. 보안노드는 보호하고자 하는 네트워크의 가장 전단에 위치하여 유입되는 네트워크 패킷을 필터링하고 차단하는 기능을 수행한다. 또한, 위

조 IP(Internet Protocol) 역추적을 위한 MAC(Media Access Control) 주소 관리 기능과 DDoS 검출을 위한 트래픽 모니터링 기능 등을 제공한다. 이 외에도 전달된 이동코드를 수행하고 다른 네트워크로 송신하거나 보안관리서버로 전달하는 기능도 제공한다. 주요 기능은 다음과 같다.

• 이동코드처리기

이동코드를 이용하여 네트워크의 보안상태를 관리하기 위해서는 네트워크 계층에서 이동코드를 인지하고, 이를 상위에 전달하여 제한된 컴퓨팅 자원 내에서 실행시키는 기능을 수행할 수 있어야 한다. 이때 실행되는 이동코드가 수신된 패킷 내에 포함되어 있지 않은 경우에는 이동코드 저장소에서 다운로드 받아 실행한다. 이동코드 실행이 완료된 후, 다시 생성된 이동코드는 네트워크에 전송을 요구한다. 이동코드처리기는 이러한 기능을 수행하며, 자바가상머신(Java Virtual Machine) 위에서 수행된다.

• 액티브 패킷을 처리하는 A-Linux 커널

액티브 패킷 형태로 전송되는 이동코드를 네트워크 계층에서 인식과 함께 이를 수신하여 이동코드처리기로 전달하는 기능과 새로 생성된 이동코드를 액티브 패킷으로 캡슐화하여 네트워크에 전송하는 기능을 수행한다.

• 공격자 대응

보안 노드에서의 공격자 대응 기능은 이동코드가 보안노드의 네트워크 보안 기능을 이용하기 위한 상위 인터페이스를 제공한다. 즉, 보안노드 상에서 실질적으로 수행되는 이동코드가 패킷 필터링과 같은 보안 대응 기능을 제어하기 위해 필요한 인터페이스들을 제공한다. 각 인터페이스는 세션 관리, IP 관리, 위조 IP 관리, MAC 주소 관리, 그리고 DDoS 탐지 및 대응 기능을 포함하고 있다.

• 이동코드 저장소

이동코드의 저장은 관리자에 의해 생성되는 다양한 이동코드를 저장하고 관리하는 데이터베이스로써, 디렉토리 서버를 이용한다. 이동코드 저장소는 보안노드 및 보안관리서버와는 LDAPv3 프로토콜을 이용하여 이동코드를 전달한다.

이동코드는 액티브 네트워크 상에서 보안 기능을 수행하는 일종의 액티브 패킷으로써, 본 논문에서 설계한 이동코드의 종류는 <표 3-1>과 같다.

이러한 이동코드는 네트워크 침입에 능동적으로 대응하기 위한 소프트웨어로써, 액티브 패킷 내에서 실행 가능한 프로그램 코드 형식으로 전달된다. 코드는 이동성 유무에 따라 상주코드와 이동코드로 구분한다. 상주코드는 보안노드에 상주하며 필요에 따라 새로운 코드를 생성하고, 이동코드는 보안노드와 보안관리서버에서 수행되며 코드의 데이터를 변경할 수 있고 다른 보안노드나 보안관리서버로의 이동성을 갖는다.

〈표 3-1〉 이동코드 종류

유형	종 류	기 능
이동형	Spoofed_IP_Tracing	IP 패킷의 근원지 주소를 위조하는 위조 IP 공격 대응을 위한 역추적 기능
	DDoS_IP Tracing	트래픽을 세션 별로 조사하여 임계치를 넘는 트래픽을 보내는 노드에 대한 DDoS 추적
	Spoofed_IP_Tracing Complete	위조된 IP 역추적 완료 후 실제 공격자를 네트워크로부터 고립시키는 기능과 역추적 결과를 해당 보안관리서버에게 전달하는 기능
	DDoS_IP_Tracing_Complete	DDoS 공격 역추적 완료 후 실제 공격자를 네트워크로부터 고립시키는 기능과 역추적 결과를 해당 보안관리서버에게 전달하는 기능
	Packet_handling	추적 코드 수신 후, 보안노드에서 침입자로부터 공격이 불가능하도록 패킷을 차단하는 기능과 경유지로 사용되어 차단된 노드의 패킷을 차단을 해제하는 기능
상주형	Traffic_Monitoring	도메인 내로 유입되는 트래픽의 이상 변동을 감지하는 기능과 일정 수준을 넘는 트래픽이 발생했을 때 보안관리서버에게 보고하기 위한 기능
	DDoS_Traffic_Detector	트래픽 모니터링의 결과를 보안관리서버에게 전달하는 기능

3.3 공격 대응 메커니즘

현재 분류된 이동코드를 통해 제공 가능한 공격자 대응 서비스는 아래와 같으며, 각각의 동작 메커니즘을 시나리오에 기반을 두어 설명한다.

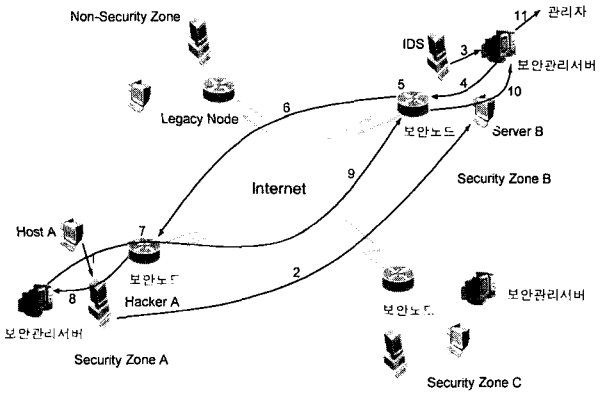
3.3.1 위조 IP 공격 대응 메커니즘

위조 IP 공격에 대한 대응 기능은 공격자가 IP 헤더 내의 근원지 IP 주소를 타인의 IP 주소로 위조하여 공격(IP Address Spoofing Attack)한 경우에 역추적을 통해 공격자를 보안영역에서 고립시키는 서비스를 제공한다[14]. 공격자 대응 프레임워크가 제공하는 위조 IP 역추적 메커니즘은 다음과 같은 기능을 제공한다.

- 침입탐지 및 차단을 위해 필요한 기존 보안장비(NIDS, Network Intrusion Detection System)와의 연동 기능
- 침입 근원지를 파악하기 위한 역추적 기능 및 침입자를 원천 봉쇄하기 위한 침입 근원지 고립화 기능
- 상기 기능의 유기적인 통합 관리를 통한 보안 관리 영역의 보안 상태 복구 기능

위조 IP 역추적 메커니즘은 기존의 네트워크 구성을 수정하지 않고도 이동코드를 통해 신속하게 실제 공격자를 검출할 수 있으며, 지금까지 수동적으로 이루어졌던 침입자 파악 수단보다 자동적이고 능동적인 대응을 가능하게 한다. 또한, 위조 IP 공격에 대한 대응 서비스를 제공하기 위하여 보안관리 영역의 망 접속점(Edge Point)에 설치된 보안노드에서 'Ingress Filtering' 기능을 수행한다[16]. 'Ingress Filtering'

을 통해 공격자에 의한 타 영역 내의 IP 주소 위조 및 조작을 사전에 방지함으로써 공격자에 의한 IP Spoofing 범위를 하나의 보안 관리 영역 내부로 한정한다.



(그림 3-10) 위조 IP 공격 대응 메커니즘

위조 IP 공격 대응 메커니즘은 (그림 3-10)과 같은 절차에 의해 수행되며, 각 단계별 수행 기능은 다음과 같다.

① 보안영역 A에 위치한 공격자 A는 동일한 보안영역 내에 위치하는 호스트 A의 IP 주소를 자신의 IP 주소로 위조한다.

② 보안영역 B에 위치하는 서버 B에게 DoS(Denial of Service) 공격을 시도한다.

③ 보안영역 B에 존재하는 IDS는 공격을 감지하여 침입탐지 경보를 보안관리서버(B)로 송신한다.

④ 보안관리서버(B)는 수신된 침입탐지 경보를 참조하여 유해패킷을 송신한 근원지 IP 주소를(보안영역 A 내의 호스트A 근원지 IP주소) 목적지 IP 주소로 하여 *Spoofed_IP_Tracing* 코드를 생성하여 전송한다.

⑤ *Spoofed_IP_Tracing* 코드를 수신한 보안노드(B)는 수행환경을 통해 수신된 코드를 실행하여 유해패킷의 유입을 *Packet_Handling* 코드를 통하여 차단한다. 이때, 보안노드(B)는 공격자의 위조 패킷과 IP 주소가 위조당한 호스트A의 정상적인 패킷까지 *Packet_Handling* 코드를 통하여 차단한다.

⑥ 보안노드(B)는 보안관리서버(B)로부터 수신한 *Spoofed_IP_Tracing* 코드를 목적지 주소로 전송한다.

⑦ 보안영역 A의 접속점에 위치하는 보안노드(A)에서 수신된 이동코드는 로그에 기록된 유출되는 이더넷 프레임 축약 정보를 검색하여 유해 패킷 정보와 일치하는 로그 정보를 추출한 후, 로그 정보에 기록된 MAC 근원지 주소와 ARP(Address Resolution Protocol) 테이블에 저장된 IP 주소를 비교하여 위조 여부 및 실제 근원지 IP 주소를 파악한다. 위조 여부가 판별되면 해당 MAC 주소로부터 유입되는 패킷을 *Packet_Handling* 코드를 통하여 차단한다.

⑧ 역추적 의뢰 정보, 성공 여부, 파악된 근원지 주소 등의 정보를 해당 보안영역 내에 위치하는 보안관리서버(A)로 송신한다.

⑨ 보안관리서버(A)는 역추적을 의뢰한 보안관리서버(B)에게 공격자의 고립결과를 *Spoofed_IP_Tracing Complete* 코드를 통해 보고한다.

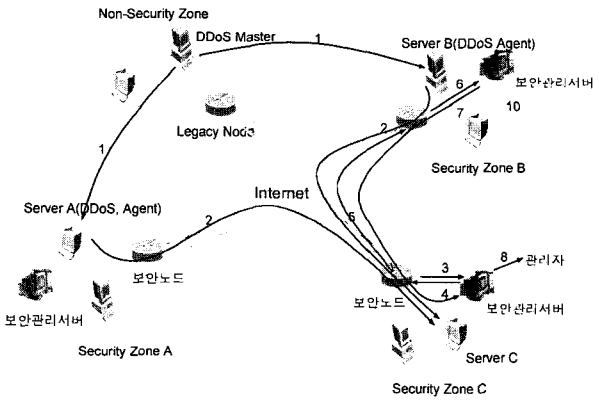
⑩ 전달 경로 상의 보안노드(B)는 *Spoofed_IP_Tracing Complete* 코드를 수신한 후, (단계 5)에서 IP 주소를 위조 당한 호스트A의 정상적인 패킷까지 차단한 세션을 *Packet_Handling* 코드를 통하여 복구하고, 보안관리서버(B)로 *Spoofed_IP_Tracing Complete* 코드를 전송한다.

⑪ 보안관리서버(B)는 수신된 *Spoofed_IP_Tracing Complete* 코드의 정보를 보안 관리자에게 통보한다.

3.3.2 DDoS 탐지 및 대응 메커니즘

DDoS 공격은 수십~수백 개의 시스템이 하나의 목표시스템을 집중 공격함으로써 피해 호스트가 서비스를 제공하지 못하도록 하는 공격 형태를 의미한다. 즉 다량의 호스트들에게 DoS 공격 에이전트를 분산시켜 설치하고, 이들을 중앙에서 제어함으로써 목표 시스템에 일제히 유해 패킷을 전송하도록 하여 시스템의 성능 저하 및 시스템 마비를 유발시키는 공격이다.

본 논문에서 제안된 공격자 대응 프레임워크는 이러한 DDoS 공격을 탐지하고 네트워크 상에 분산되어 있는 DDoS Agent를 찾아 내기 위한 역추적 메커니즘과 유해 패킷을 공격자 근원지에서 고립시키는 보안 서비스를 제공한다. DDoS 탐지 및 공격 대응 메커니즘은 (그림 3-11)과 같은 절차에 의해 수행되며, 각 단계별 수행 기능을 다음과 같다[16].



(그림 3-11) DDoS 탐지 및 공격 대응 메커니즘

① 비 보안영역(Non-Secure Zone)에 위치한 공격자는 보안영역 C 내의 서버 C를 공격하기 위해 DDoS 마스터를 기반으로 보안영역 A와 B에 위치한 서버 A와 B에 불법

적인 방법을 사용하여 DDoS 에이전트를 설치한다.

② 보안영역 A와 B에 있는 DDoS 에이전트는 보안영역 C에 위치한 서버 C로 DDoS 공격을 시도한다.

③ 보안노드(C) 내의 상주형 *Traffic_Monitoring* 코드에서 입력 패킷을 분석하여 특정 시간 내에 입력된 전체 패킷의 양이 임계치를 넘는다면 DDoS 공격으로 간주한다. DDoS 공격을 탐지하면 탐지된 결과를 포함하여 보안노드(C)는 *DDoS_Traffic_Detector* 코드를 생성하여 보안관리서버(C)로 송신한다.

④ 보안관리서버(C)는 *DDoS_Traffic_Detector* 코드를 수신한 후, *DDoS_Traffic_Detector* 코드 내의 공격 정보를 분석하여 *DDoS_IP_Tracing* 코드를 생성하여 보안노드(C)로 전송한다.

⑤ *DDoS_IP_Tracing* 코드를 수신한 보안노드(C)는 수행환경을 통해 수신된 코드를 실행하여 보안영역 C로 입력되는 패킷들을 분석한다. 만일 특정 발신자 주소들로부터 입력되는 패킷의 수가 일정한 값을 넘는다면, DDoS 공격으로 간주하고 공격 패킷을 전송하는 DDoS 공격 에이전트 수와 동일하게 *DDoS_IP_Tracing* 코드를 생성한다. *DDoS_IP_Tracing* 코드들은 DDoS 에이전트(서버 A와 B)들을 목적지 주소로 하여 해당 보안영역으로 전달된다.

⑥ *DDoS_IP_Tracing* 코드를 포함한 액티브 패킷은 전송경로에 따라 보안노드(B)에 의해 수신한다. *DDoS_IP_Tracing* 코드의 실행에 의해 보안영역 C의 서버 C로 나가는 패킷들을 검사한다. 검사한 패킷들이 서버 C의 특정 포트 번호를 공격하는 것으로 판명되면, 보안노드(B)는 서버 B에서 서버 C로 나가는 패킷 가운데 특정 포트 번호를 갖는 패킷들을 *Packet_Handling* 코드를 통하여 차단한다. 차단 이후에, 보안노드(B)는 *DDoS_IP_Tracing* 코드의 목적지 주소를 보안관리서버(B)로 변경하고 전송한다.

⑦ 보안관리서버(B)는 *DDoS_IP_Tracing* 코드를 수신하고, 처리 결과에 따라 *DDOS_Tracing_Complete* 코드를 생성한다. *DOS_Tracing_Complete* 코드의 목적지 주소는 *DDoS_IP_Tracing* 코드 내에 포함된 최초 전송자인 보안관리서버(C)의 주소로 결정된다. 전달 경로 상의 보안노드(B)에서는 *DOS_Tracing_Complete* 코드 수신 시, 별도의 실행 없이 재전송한다.

⑧ 보안관리서버(C)는 *DOS_Tracing_Complete* 코드를 수신한 후, 코드 내에 포함된 정보를 분석하여 관리자에게 전달한다.

단계 ⑤~단계 ⑦은 보안영역 A에도 동일하게 적용되며, DDoS 마스터에 대한 추적 및 대응 기능은 본 논문의 영역 외로 한다[12].

3.4 구 현

본 연구에서는 보안노드와 보안관리서버를 구현을 위해 사용한 환경은 다음과 같다.

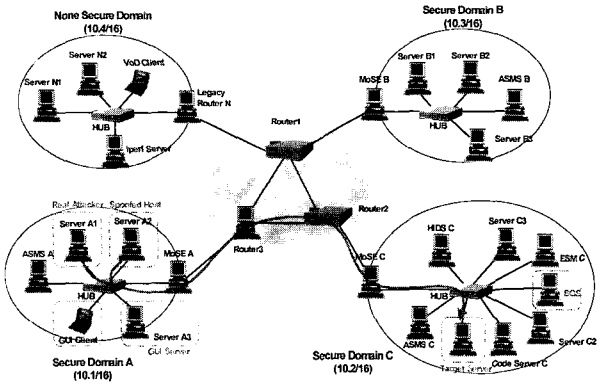
- 보안노드 및 보안관리서버 운영체제
  Linux 운영체제 커널(버전 2.4) 수정
- 이동코드를 실행하기 위한 실행 환경
  SUN Java Virtual Machine 버전 1.4.2
- 이동코드저장소
  - 이동코드 정보를 관리하기 위한 데이터베이스로서 PostgresSQL JDBC를 사용
  - 이동코드의 실행프로그램 저장소는 Netscape 사의 *iPlanet Directory Server version 5.1*을 사용
- 패킷필터링 라이브러리
  libpcap 라이브러리(*iptables* 명령)를 사용
- 네트워크 인터페이스
  10/100M Ethernet 인터페이스

4. 대응 프레임워크 실험

본 논문은 보안관리 영역 내에 침입 행위에 대하여 네트워크로부터 고립화하기 위해 액티브 네트워크 개념에 기반을 두어 이동코드를 포함한 액티브 패킷을 활용하였다. 앞서 기술한 바와 같이 해당 기술을 이용하여 네트워크 차원의 보안상태를 동적으로 보안제어를 수행하며, 다른 보안관리 영역을 관리 하는 보안관리서버와의 협업을 통해 전역적인 네트워크 보안관리 기능을 수행하도록 하였다. 이 경우 그 적용 범위가 광역 인터넷이라는 점에서 개발된 기능을 네트워크에 적용하고 검증하기가 매우 어렵다. 특히, 단일 로컬 네트워크 상에서는 개발된 기능을 검증할 방법이 없으므로, 공격자 대응 프레임워크의 적용 가능성 및 보안 기능 검증을 위한 실험환경을 구축하여 설계된 기능을 검증하도록 한다. 특히 제안된 공격자 대응 프레임워크에 대한 적용 가능성을 증명하기 위해 성능상의 중요한 요소 대응 수행시간을 분석하도록 한다.

4.1 실험 네트워크 환경

실험환경은 인터넷 백본 환경과 공격자 대응 프레임워크로 구성되는 복수의 보안관리 영역을 포함하고, 실험환경 상에서는 공격자 대응 프레임워크의 기능 검증, 네트워크 적용성 시험 등의 일련의 작업이 수행될 수 있는 제반 환경(NIDS, HIDS, ESM 등)을 제공한다. 또한, 공격자 대응 프레임워크의 각 기능들의 독립적인 시험 및 통합 연동 시험을 위한 제반 사항을 제공한다. (그림 4-1)은 본 논문에서 설계한 공격자 대응 프레임워크를 시험하기 위해 구성된 실험환경을 보여준다.



(그림 4-1) 실험 네트워크 환경 구성도

중앙에 위치한 1개의 네트워크 도메인은 공중망의 역할을 하는 가상 ISP(Internet Service Provider) 도메인이고, 나머지 4개의 도메인은 ISP에 연결되어 있는 로컬 네트워크 도메인으로 공격자가 존재하는 네트워크와 우회 공격 서버 혹은 직접적인 피해를 입는 서버가 존재하는 피해자 네트워크로 이용된다. 현 시점에서는 실제 네트워크를 축소한 실험 환경 수준의 망으로 구성하였다. 실험환경 시스템을 구성하는 각 주요 장비는 다음과 같다.

• 가상 ISP Edge Router

보안 관리 영역을 상호 연결하고, 가상 인터넷 환경을 구축하기 위한 백본 네트워크를 구성하기 위해 사용되는 라우터 시스템이다.

• 보안관리서버

보안관리서버는 보안관리 영역에 각각 한대씩 구축된다. 액티브 패킷을 처리할 수 있는 확장된 리눅스 기반 커널로 동작하는 서버이며, 이동코드 정보를 관리하기 위한 데이터베이스(PostgreSQL)가 탑재된다.

• 보안노드

보안 노드는 보안 관리 영역의 접속점에 각각 한대씩 구축된다. 액티브 패킷을 처리할 수 있는 확장된 리눅스 기반 커널로 동작하는 라우터이며, 패킷 차단, 프레임 모니터링(MAC, ARP 테이블 관리) 등 보안 대응 기능이 탑재된다.

• 가상 공격자 및 피해 시스템

가상 공격자를 가정하는 공격용 시스템과 공격 목표가 되는 시스템으로써 리눅스, SUN, Windows 등 다양한 플랫폼으로 구성된다.

• 공격 도구

DDoS 공격 도구인 Flitz를 사용하여 위조 IP 공격과 DDoS 공격을 이용한다.

4.2 실험 절차

제시된 공격자 대응 프레임워크를 실험은 3.3절에서 제시한 공격 대응 메커니즘을 시나리오에 기반을 두어 수행한다.

4.2.1 위조 IP 공격 대응 시험 절차

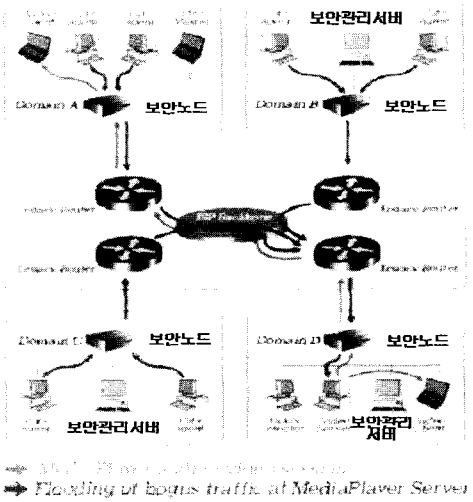
위조 IP 공격 대응 시험에서는 DDoS 공격용 툴 'Flitz'를 이용하여 동일 도메인 상에 존재하는 다른 시스템의 주소를 이용하였고, 도메인 C에 존재하는 서버를 대상으로 ICMP Flooding 공격을 수행하였다. 위조 IP 공격은 대부분 UDP 계열의 네트워크 공격이며, 본 실험에서는 이러한 공격을 탐지하기 위해서는 네트워크 기반 IDS 시스템을 설치하였다. (그림 4-1)의 화살표는 실험환경 상에서 위조 IP Flooding 공격 경로를 보여준다.

본 시험은 4.2.2절의 DDoS 공격과 성격상 유사한 점이 많지만, 단순히 위조 IP 기법만을 사용하여 공격을 시도하였고, 그에 따른 공격자 대응 프레임워크의 보안 체계의 구동 및 대응과 관련된 동작을 시험하였다. 시험 절차는 다음과 같다.

① Flitz를 이용한 위조 ICMP Flooding 공격 수행
② IDS의 침입 탐지 및 보안관리서버로 통지
③ 공격자 대응 메커니즘의 구동 및 공격자 호스트 역추적 실행
④ 실제 공격자 호스트 MAC 주소 필터링
⑤ 공격자 차단 확인

4.2.2 DDoS 공격 대응 시험 절차

DDoS 공격은 목표 시스템의 서비스 제공을 방해하기 위하여 해당 시스템이 처리할 수 없는 요청을 보내는 공격으로써, 무수히 많은 에이전트에서 특정 서비스 요청을 전송하여 부수적으로 해당 네트워크 트래픽을 범람시키는 효과를 가지게 된다. DDoS 공격 툴의 구성은 공격 요청을 보내는 무수히 많은 에이전트와 해당 에이전트에 공격 명령을 시달하기 위한 마스터로 구성된다.



(그림 4-2) DDoS 공격 시험 구성도

본 시험에서는 Flitz 툴을 이용하여 두개의 보안 도메인 상에 6개의 에이전트를 두고 비디오 스트리밍 서비스를 제공하는 서버를 공격하는 것으로 하였다. 또한 실제 네트워크 상황을 에뮬레이션하기 위해 iperf 툴을 이용하여 3.5M 정도의 백그라운드 트래픽을 생성하였다. 비디오 스트리밍 서비스로는 DVD(Digital Versatile Discs) 드라이브를 네트워크로 연결하여 DVD를 원격에서 구동하였다. (그림 4-2)의 화살표는 실험환경 상에서 DDoS Agent들의 공격 경로를 보여준다.

DDoS는 네트워크 기반의 IDS로 탐지하지만, 공격자 대응 프레임워크의 메커니즘을 이용하면 네트워크 기반 IDS가 필요하지는 않다. 즉, 공격자 대응 프레임워크를 구현한 각 도메인의 게이트웨이 라우터인 보안노드에서 네트워크 트래픽을 모니터링하고 이를 바탕으로 경보를 발령하면 실제 트래픽이 과다하게 발생시키는 에이전트 위치를 추적하여 단절하게 된다. 실험에서는 공격이 이루어짐에 따라 비디오 스트링밍 서비스의 품질이 저하되는 것을 보여주었고, 공격자 대응 프레임워크의 동작에 따라 해당 서비스의 품질이 회복되는 것을 보여 주었다. 시험 절차는 다음과 같다.

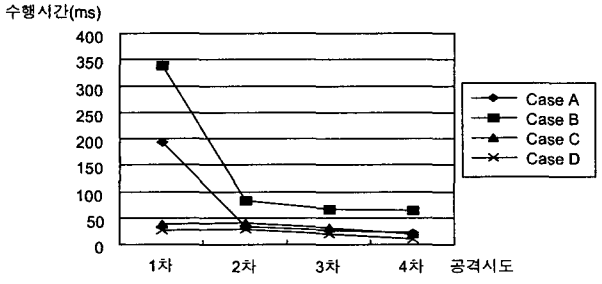
① 비디오 스트리밍 서비스 제공
② Flitz를 이용한 DVD 서버에 공격
③ 비디오 스트리밍 서비스의 품질 저하 확인
④ 공격자 대응 프레임워크의 구동 및 에이전트의 트래픽을 각 도메인의 게이트웨이에서 차단
⑤ 비디오 스트리밍 서비스 품질의 회복
⑥ 공격 트래픽 차단 확인

4.3 실험 결과

제안된 공격자 대응 프레임워크에 대한 적용 가능성을 증명하기 위해 성능상의 중요한 요소인 대응 수행시간을 분석하도록 한다.

4.3.1 위조 IP 공격 대응 실험 결과

위조 ICMP Flooding 공격 시에, 공격자 대응 프레임워크에서 이동코드를 이용한 대응 메커니즘의 코드 수행 시간은 (그림 4-3)과 같다.



(그림 4-3) *Spoofed_IP_Tracing* 및 *Spoofed_IP_Tracing_Complete* 수행 시간

Case A는 (그림 4-3)에서 피해 시스템이 속한 도메인 C의 경계에 위치한 보안노드(C)에서 수행된 *Spoofed_IP_Tracing* 코드의 수행 시간을 의미하며, Case B는 공격자가 속한 도메인 A의 경계에 위치한 보안노드(A)에서 수행된 *Spoofed_IP_Tracing* 코드의 수행 시간을 의미한다. Case C와 Case D는 보안노드(C)와 보안노드(A)에서 수행된 *Spoofed_IP_Tracing_Complete* 코드의 수행시간을 의미한다.

(그림 4-3)에서 보듯이, 보안노드(C, A)의 *Spoofed_IP_Tracing* 코드 수행 시간인 Case A와 Case B의 경우에서 공격자의 1차 공격 시도인 경우는 1차 이후의 공격에서보다 수행시간이 길다. 이것은 이동코드의 실행 프로그램이 저장된 코드저장소와 연결을 설정하고 실행 프로그램을 다운로드 하는 시간을 포함하기 때문이다. 특히, Case B의 경우는 보안노드(A)에서 *Spoofed_IP_Tracing* 코드를 수신한 후, ARP를 이용하여 공격자의 실제 MAC 주소를 차단하기 위해 수행되는 오버헤드가 포함되기 때문에 수행 시간이 가장 길다. 반면에, Case C와 Case D의 경우, 보안노드(C, A)에서는 *Spoofed_IP_Tracing_Complete* 코드를 수신하지만 이미 연결되어 있는 코드 서버로부터 *Spoofed_IP_Tracing_Complete* 코드를 다운로드 하며, 별도의 실행 없이 코드를 재전송한다. 따라서 이 경우에서의 이동코드 수행시간은 아주 작다.

그러나 위조 IP 공격 대응 실험은 1차 공격시도인 경우의 수행시간으로 인하여 제안된 공격 대응 프레임워크의 좋은 성능에 영향을 미칠 수 있다. 따라서 이동코드의 실행 프로그램을 다운로드하는 회수는 제안된 공격 대응 프레임워크 성능의 변화가 어떻게 되는 지에 대하여 신중을 기해야 할 것이다.

시험 결과 한 hop에서의 최대 대응 지연 시간이 평균 162ms임을 알 수 있다. 실제 인터넷의 hop 수는 평균 20hop 정도이며 최대 40hop 미만이다. 따라서 제안하는 기법은 탐지에서 대응까지 평균 3.2초에서 최대 6.5초 정도의 성능을 제공한다고 할 수 있다. 따라서 본 아키텍처는 인터넷에서도 충분히 이용할 수 있음을 시사하고 있다

4.3.2 DDoS 공격 대응 실험 결과

DDoS 공격에 대해 공격자 대응 프레임워크에서 이동코드를 이용한 대응 메커니즘의 코드 수행 시간은 <표 4-1>과 같다. <표 4-1>에서 보듯이, 보안노드(C, B, A)에서는 *DDoS_IP_Tracing* 코드를 수신하고 DDoS 에이전트로부터 입력 트래픽을 차단하는 기능을 수행하므로 코드 수행 시간이 길다. 그러나, *DDoS_IP_Tracing_Complete* 코드는 보안노드(C, B, A)에서 별도의 실행 과정 없이 보안관리서버(C)로 전달되기 때문에 보안노드에 실행 부담이 없다.

DDoS 공격 대응 실험에서 특이한 사항은 무수히 많은 분산된 공격 에이전트인 경우에서 매우 좋은 성능을 나타낼 것으로 예상된다. 이는 동시다발적인 분산된 공격 에이전트

에 대한 대응 메커니즘도 공격 에이전트 수와 동일하게 *DDoS_IP_Tracing* 코드를 생성하여 병렬적으로 수행되기 때문이다.

〈표 4-1〉 *DDoS_IP_Tracing* 및 *DDoS_IP_Tracing_Complete* 수행 시간

(단위 : millisecond)

수행지점	코드종류	1차	2차	3차	4차
보안노드(C)	*DDoS_IP_Tracing*	10478	10933	10509	10747
보안노드(B)	*DDoS_IP_Tracing*	10672	10652	10778	10561
보안노드(A)	*DDoS_IP_Tracing*	10708	10889	10801	10693
보안노드(C)	*DDoS_IP_Tracing_Complete*	30	17	22	29
보안노드(B)	*DDoS_IP_Tracing_Complete*	22	24	27	20
보안노드(A)	*DDoS_IP_Tracing_Complete*	25	28	24	21

5. 결 론

본 논문에서는 네트워크 보안 환경 변화에 따르는 요구사항을 반영할 수 있는 확장된 보안 구조로서 액티브 네트워크를 이용한 공격자 대응 프레임워크를 설계하였고, 위조 IP 공격이나 DDoS 공격에 대응하기 위한 메커니즘을 제안하였다. 제안한 공격자 대응 프레임워크는 새로운 공격 기술, 방어 기술의 등장이나 보안 환경 변화에 유연하게 적응할 수 있도록 전체 네트워크 수준에서 수행할 보안 기능을 이동코드 형태로 수행하도록 설계되었다.

제안된 공격자 대응 프레임워크의 적용 가능성을 증명하기 위해 실험환경을 구축하고, 해당 실험환경 상에서 실제 제공되는 서비스와 대표적인 공격 기법을 적용한 상태에서 대응 메커니즘을 실험하였다. 실험 결과에 의하면, 공격자 대응 프레임워크는 기존의 수동적인 침입 차단 및 침입탐지 시스템의 문제점을 해결하고, 동적인 보안 서비스를 제공함을 보였으며, 실제 필드에 적용할 수 있음을 확인하였다.

끝으로 현재의 네트워크 보안 기술이 제공하는 공격자 대응 수준이 보다 한층 더 강력하고 광범위한 대응에 대해 문제점으로 대두될 때, 역추적을 통한 공격자 고립화 기술은 비로소 자리매김을 할 것이다.

참 고 문 헌

[1] Dan Schnackenberg, Kelly Djahandari and Dan Sterne, "Infrastructure for Intrusion Detection and Response," DARPA Information Survivability Conference and Exposition(DISCEX 2000), Jan., 2000.

[2] Dan Schnackkenberg, Harley Holiday et al., "Cooperative Intrusion Traceback and Response Architecture(CITRA)," DISCEX 2001, June, 2001.

[3] Dan Schnackenber, Travis Rei, Kelly Djahandar, Brett Wilso, "Cooperative Intrusion Traceback and Response Architecture (CITRA)," NAI Labs Report #02-008 Feb., 2002.

[4] Dan Sterne, Kelly Djahandari, Ravindra Balupari, William La Cholter, Bill Babson, Brett Wilson, Priya Narasimhan, and Andrew Purtell, "Active Network Based DDoS Defense," Proceedings of the DARPA Active Networks Conference and Exposition (DANCE.02), p.193, May, 2002.

[5] Sterne, D., "Active Networks Intrusion Detection and Response (AN-IDR)," presentation at DARPA Fault Tolerant Networks Program Principal Investigators Meeting, Honolulu, HI, July, 2000.

[6] Spyros Denazis, "Overview FAIN Programmable Network and Management Architecture-Draft Ver. 2.0," WP3-HEL-056-D14-FAIN, FAIN Consortium, May 12th, 2003.

[7] Stamatis Karnouskos, "Dealing with Denial- of-Service Attacks in Agent-enabled Active and Programmable Infrastructures," IEEE 25th International Computer Software and Application Software (COMSAC 2001), Oct., 2001.

[8] B. Chang, D. Kimm Y. Kwon, T. Nam, T. Chung, "Security Management by Zone Cooperation in Active Network Environment," Proc. of the 2002 International Conference on Security Management (SAM '02), pp.187-192, 2002.

[9] Beom-Hwan Chang, Dong-Soo Kim, Hyun-Ku Kim, Jung- Chan Na, Tai-Myoung Chung, "Active security management based on secure zone cooperation," Future Generation Computer Systems, Vol.20, No.2, pp.283-293, February, 2004.

[10] 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)," Active Network Group Draft, July, 1997.

[11] D. Kat, "IP Router Alert Option," RFC 2113, IETF, Feb., 1997.

[12] Hyun Joo Kim, Jung C. Na and Sung W. Sohn, "Response To Distributed Denial-of Service Attack using Active Technology," IMSA2004, Apr., 2004.

[13] 이수형, 나중찬, 손승원, "액티브 네트워크 기반 보안 기술 동향", 한국전자통신연구원 주간기술동향, 제1076호, Dec., 2002.

[14] 이영석, 방효찬, 나중찬, "액티브 네트워크 기반의 위조 IP 공격 대응 메커니즘", 한국정보과학회 춘계학술발표논문집, Vol.4, No.4, 2003.

[15] 방효찬, 손선경, 나중찬, 손승원, "액티브 네트워크를 이용한 능동 보안 관리 프레임워크", COMSW2002, Jul., 2002.

[16] P. Ferguson, D.Senie, "Network Ingress Filtering : Defeating Denial of Service Attacks which employ IP Source Address Spoofing," IETF RFC2827, May, 2000.

방 효 찬

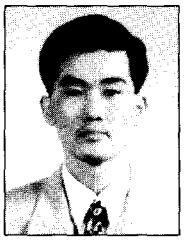
e-mail : bangs@etri.re.kr
1995년 북해도공업대학 경영공학과 공학사
1997년 북해도공업대학 기계시스템공학과 공학석사
1997년~1999년 한국통신 운용연구단 전임연구원
2000년~현재 ETRI 능동보안기술연구팀 선임연구원
관심분야 : 네트워크보안, 액티브네트워크

김 진 오

e-mail : zyno21@etri.re.kr
1994년 인하대학교 전자계산공학과 공학석사
1991년~1998년 ETRI 네트워크기술연구소
1998년~2001년 (주)팍스콤
2001년~현재 ETRI 능동보안기술연구팀 선임연구원
관심분야 : 네트워크 공격상황 분석, 정책기반 네트워크 보안관리

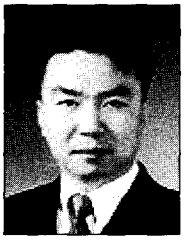
나 중 찬

e-mail : njc@etri.re.kr
1986년 충남대학교 계산통계학과 이학사
1989년 숭실대학교 전자계산학과 공학석사
2004년 충남대학교 컴퓨터과학과 이학박사
1989년~현재 ETRI 능동보안기술연구팀 팀장
관심분야 : 네트워크 트래픽 및 공격상황 분석



장 종 수

e-mail : jsjang@etri.re.kr
1984년 경북대학교 전자공학과 공학사
1986년 경북대학교 전자공학과 공학석사
2000년 충북대학교 컴퓨터공학과 공학박사
1989년~현재 ETRI 네트워크보안그룹 그룹장
관심분야 : 네트워크보안, 정책기반보안관리, 비정상트래픽탐지, 유해정보차단



이 영 석

e-mail : leeys@kunsan.ac.kr
1992년 충남대학교 컴퓨터공학과 공학사
1994년 충남대학교 컴퓨터공학과 공학석사
2002년 충남대학교 컴퓨터공학과 공학박사
1994년~1997년 LG전자연구원
2002년~2004년 ETRI 정보보호연구단 선임연구원
2004년~현재 국립군산대학교 전자정보학부
관심분야 : 분산시스템, 정보보호, 이동컴퓨팅