역추적 기술 동향: TCP Connection Traceback 중심

최양서* 서동일** 손승원***

최근 인터넷 사용자가 급증하면서 인터넷을 이용한 각종 해킹 및 사이버 범죄가 크게 증가되고 있다. 이러한 상황에서 각종 침해사고로부터 시스템 및 네트워크를 보호하기 위해 각종 보안 강화 시스템이 개발되어 운용되고는 있으나, 현재 사용중인 각종 보안 강화 도구들은 수동적인 해킹 방어 시스템으로 해커의 해킹 시도 자체를 제한하는 것이 아니라 해킹이 시도된 후 이를 막기 위한 제품으로 해킹 시도 자체를 방지하는 데는 한계를 가지고 있다. 이와 같은 한계를 극복하기 위해서는 해커의 해킹을 보다 효과적으로 방지할 수 있는 능동적인 해킹 방지 기술이 필요하게 되었다. 능동적인 해킹 방어를 위한 가장 기본적인 기술은 해커의 실제 위치를 추적하는 역추적 시스템이라 할 수 있는데, 현재는 이에 대한 연구가 활발히 진행 중에 있다. 이에 본 문서에서는 현재 연구 개발되고 있는 역추적 시스템들의 동향에 대해 알아보도록 한다.

I. 서 론

인터넷은 이미 일상 생활에 깊게 자리 잡았다. 이러한 인터넷을 이용해서 실생활에서 수행하여야만 했던 많은 일들을 인터넷을 통해 수행할 수 있게 되었고, 인터넷의 편리함 때문에 인터넷 사용자 역시 크게 증가하였다. 이러한 인터넷 사용자의 증가와 더불어 (그림 1)에서 볼 수 있듯이 인터넷을 통한 각종 침해사고 역시 크게 증가하였다[1].

이에, 보안 업체들은 각종 침해로부터 시스템 및 네트워크를 보호하기 위해 각종 보안 강화 시스템을 개발하였고 각 시스템 및 네트워크 관리자들은 이 시스템들을 적용하였다. 그러나, 현재까지 개발되어 사용되고 있는 대부분의 보안 강화 시스템들은 해커의 해킹 시도 자체를 제한하는 것이 아니라, 해커가 해킹을 시도하는 경우, 이를 좀더 어렵게 만드는 수준에 지나지 않았다. 즉, 해커의 해킹 시도를 능동적으로 대처하지 못하고, 수동적으로 방어하는 수준인 것이다. 또한 인터넷 상에 동작하는 여러 보안 강화 시스템들은 매우 다양하기 때문에 해커의 해킹에 대한 상호 협력을 통한 대응이 거의 불가능한 상황이다. 이와 같은 현재의 보안 시스템 환경 때문에 해킹 시도는 날로 증가하고 있고, 이를 효과적으로 방어하지 못하는 것이 현실이다.

이와 같은 문제점을 해결하기 위해 해커의 해킹 시도 자체를 제한할 수 있는 능동적인 해킹 방지 시스템을 개발하고자 하는 노력이 시도되었고, 능동적인 해킹 방지를 위해 가장 시급한 것이 바로 역추적 시스템임을 주지하게 되었다. 따라서, 최근 역추적 기술에 대한 관심이 날로 커지고 있고, 비록 아직은 초보적인 수준이나 역추적 기술에 대한 연구가 진행되기 시작하였다[2]. 이에 본 문서에서는 현재 진행되고 있는 역추적 기술 연구의 동향에 대해

알아보도록 하고, 향후 역추적 기술의 발전 방향에 대해 논의해 보도록 한다.

본 문서는 다음과 같이 구성되어 있다. II장에서는 역추적이란 무엇인지에 대해 알아보도록 하고, III장에서는 현재 발표된 역추적 기술에는 어떤 것들이 있는지 알아보도록 하고, IV장에서는 결론과 향후 연구 방향에 대해 논의하도록 한다.

II. 역추적 기술이란 무엇인가

역추적(traceback)이란 해킹을 시도하는 해커의 실제 위치를 실시간으로 추적하는 기술을 말하는 것으로 크게 2가지로 분류할 수 있다. 본 장에서는 역추적 기술의 정의와 분류 그리고 현재 사용되고 있는 역추적 기법의 문제점에 대해 살펴 보도록 한다.

1. 역추적의 정의 및 현재의 문제점

가. 역추적의 정의

[정의 1] 역추적: 사이버 범죄를 시도하는 공격자의 네트워크 상의 실제 위치를 탐색하는 기술

이와 같이 해킹을 시도하는 해커의 실제 위치를 추적하는 기술을 역추적 기술이라 한다. 물론 이러한 역추적은 매우 어려운 일이다. 이는 인터넷의 익명성과 다양성이라는 가장 큰 특징에 의해 더욱 어려운 문제가 되고 있다. 이에, 아직까지 역추적을 수행하기 위해 사용되고 있는 방법은 인간에 의해 수동적으로 해커의 흔적을 따라가는 방식인데, 이는 많은 문제점을 가지고 있다.

나. 현재의 역추적 기술의 문제점

현재 사용되고 있는 역추적 기법은 사람에 의해 진행되는 것이 전부이다. 이와 같이 사람에 의해 직접 해커의 흔적을 찾아내 역추적을 수행하는 경우, 해커의 실제 위치를 찾기 위해서는 공격을 당하고 있는 시스템((그림 2)의 Host B)의 로그 기록뿐만 아니라, 공격을 수행하고 있는 시스템((그림 2)의 Host A)의 로그 기록을 분석해야 한다. 왜냐하면 Host B의 로그기록에서 얻을 수 있는 정보는 Host A에서 해킹이 시도되었다는 것 뿐이기 때문이다. 그런데, 실제 해커의 위치는 Host A가 아니므로, 이를 파악하기 위해서는 Host A의 로그 기록을 확인해야 한다. 즉, 역추적 경로상에 존재하는 모든 중간 경유지에 대해 해커의 흔적을 조사하여야 한다. (그림 2)에서는 경유 시스템이 1개 뿐이지만 실제 상황에서는 다수의

경유 시스템이 존재할 수 있다. 따라서 다음과 같은 문제점들이 존재한다.

① Host A와 Host B의 거리가 지리적으로 먼 거리에 위치하여 직접 그 시스템을 확인할 수 없는 경우라면, 실제 해커의 위치를 파악하는 데 많은 시간과 노력이 필요하게 된다.

② 수작업이 주인 역추적은 기술 인력의 부족으로 늘어나는 해킹 사고에 빠르게 대응하지 못하고 있다.

③ 역추적을 수행하는 과정에서 중간 경유 시스템으로부터 공격 경로상의 이전 시스템에 대한 정보를 얻을 수 없는 경우, 역추적이 불가능하게 된다. 이는 역추적을 위하여 오직 로그 파일에만 의존하기 때문이다.

이와 같은 문제점들 때문에, 신속한 역추적을 수행하기 위해 자동화된 시스템이 절실히 필요한 것이다.

2. 역추적 기술의 분류

역추적 기술은 일반적으로 크게 2가지 분야로 분류되는데, 이는 해커가 우회공격을 시도하는 경우, 해커의 실제 위치를 추적하는 기술과, IP주소가 변경된 패킷의 실제 송신지를 추적하는 기술이다. 이때, 우회 공격을 시도하는 해커의 실제 위치를 추적하는 기술을 TCP 연결 역추적(TCP connection traceback) 혹은 연결 역추적(connection traceback)이라 하고, IP 주소가 변경된 패킷의 실제 송신지를 추적하는 기술을 IP 패킷 역추적(IP packet traceback) 혹은 패킷 역추적(packet traceback)이라 한다. 본 문서에서는 TCP 연결 역추적 기술을 중심으로 다루도록 한다.

가. TCP 연결 역추적

TCP 연결 역추적은 다음과 같이 정의될 수 있다.

[정의 2] TCP 연결 역추적: TCP연결을 기반으로 우회 공격을 시도하는 해커의 실제 위치를 실시간으로 추적하는 기법

여기서 우회 공격이란, (그림 2)에서 볼 수 있듯이, 해커가 최종 침입 대상(Host B)을 공격하는 데 있어서, 직접 Host B로 침입을 시도하지 않고, 다수의 중간 시스템(Host A) 을 경유하여 침입을 수행하는 공격을 의미한다. 이와 같이 해커가 우회 공격을 시도하게 되면, 직접 공격을 당하는 피해 시스템(Host B)에서는 해커의 실제 위치를 파악할 수 없게 된다. Host B에서 확인할 수 있는 것은 오직 현재 해당 시스템을 공격하고 있는 바로 이전의 시

스템(Host A)에 대한 정보만을 얻을 수 있다.

이와 같은 TCP 연결 역추적 기술은 흔히 연결 체인 역추적 기술이라고 불리기도 한다. 여기서, 연결 체인이란 (그림 3)에서 H0에서 Hn까지의 연결들의 집합을 연결 체인이라고 한다. 즉, 해커가 실제로 위치한 시스템으로부터 여러 시스템을 경유하여 실제 공격을 당하고 있는 시스템까지의 연결들의 집합을 말하는 것으로 다음과 같이 정의된다.

[정의 3] 연결 체인(Connection Chain)[3]: 컴퓨터 H0의 한 사용자가 네트워크를 통해 다른 시스템 H1으로 로그인하면, 두 시스템 H0와 H1간에는 TCP 연결 C1이 생성된다. 이때, 같은 사용자가 시스템 H1에서 H2로, 또 H3, …, Hn으로 로그인하게 되면, 각각의 해당 시스템들 간에는 TCP 연결 C2, C3, …, Cn이 같은 방식으로 생성되게 된다. 이때 이 일련의 연결들의 집합 C = (C1, C2, . . . , Cn)를 연결체인이라 한다.

TCP 연결 역추적 기술은 다시, 크게 2가지로 분류할 수 있다. 이는 호스트 기반 연결 역추적(host-based connection traceback) 기술과 네트워크 기반 연결 역추적(network-based connection traceback) 기술로 분류된다.

1) 호스트 기반 연결 역추적 기술

호스트 기반 연결 역추적 기술은 역추적을 위한 모듈이 인터넷 상의 호스트들에 설치되는 역추적 기법으로 호스트에서 발생하는 로그 기록 등의 다양한 정보를 바탕으로 역추적을 진행하는 기술이다. 그러나 이러한 방법을 이용하여 역추적을 수행하기 위해서는 인터넷 상의 모든 호스트에 역추적 모듈이 설치되어야 하고, 역추적 경로 상의 단 1개의 시스템에서라도 어떤 문제에 의해서 역추적 정보를 얻을 수 없게 되는 경우가 발생하면 역추적이 불가능하게 되는 단점을 가지고 있다. 이와 같은 문제점들로 인해 현재의 인터넷 환경에 적용하는 것은 거의 불가능하다고 할 수 있다[4-6].

2) 네트워크 기반 연결 역추적 기술

네트워크 기반 연결 역추적 기술은 네트워크 상에 송수신되는 패킷들로부터 역추적을 수행할 수 있는 정보를 추출하여 역추적을 수행하는 것으로 역추적 모듈이 네트워크 상에 송수신되는 패킷을 확인할 수 있는 위치에 설치된다. 현재 제안되고 있는 방법은 대부분 송수신 패킷을 확인할 수 있는 위치에서 공격 연결과 같은 연결 체인에 속하는 연결을 추출하여 역추적을 수행하는 방법을 취하고 있다.

그러나 아직까지 네트워크 기반 연결 역추적 기술을 현재의 인터넷에 적용하여 사용할 수 있는 전체 시스템은 제안되지 못했다. 다만 네트워크 상에서 얻을 수 있는 패킷으로부터 어떤 정보를 활용해야 공격 연결과 같은 연결에 속하는가를 판단할 수 있을지에 대한 알고리듬만이 제기되고 있는 상황이다[3,7,8]. 이는 네트워크 상의 패킷들로부터 얻게 되는 각종

연결 정보들을 네트워크 상에 존재하는 역추적 시스템들과 공유하는 데 있어서, 생성되는 정보의 순서관계 및 동기화가 매우 어렵고, 네트워크 상에서 발생하는 모든 연결에 대한 정보를 지속적으로 보유하고 있어야 하는 문제가 발생할 수 있기 때문이다.

또 다른 네트워크 기반 연결 역추적 기술로는 액세스 네트워크상에서 동작하는 기술들이 있다. 그러나 액세스 네트워크를 기본으로 하기 때문에 현재의 인터넷 환경에 적용하는 데 많은 어려움이 있는 것이 사실이다[9,10].

나. IP 패킷 역추적[11,12]

IP 패킷 역추적 기술은 앞서 잠시 언급한 바와 같이 IP주소가 변경된 패킷의 실제 송신지를 추적하기 위한 기술을 말한다.

[정의 4] IP 패킷 역추적: IP 주소가 변경된 패킷의 실제 송신지를 추적하는 기술

일반적으로 IP 주소가 변경된 패킷은 악의적으로 사용되는 경우가 대부분이다. 특히 서비스 거부(Denial of Service: DoS), 혹은 분산 서비스 거부(Distributed Denial of Service: DDoS) 공격에 주로 사용된다. IP 주소가 변경되는 경우에는 TCP 연결을 유지할 수 없기 때문에, 일방적인 패킷 송신으로 공격이 가능한 DoS 혹은 DDoS에서 주로 사용되는 것이다. 물론 과거 IP spoofing이라 알려져 있는 해킹 기법을 이용하는 경우, IP 주소가 변경된 패킷을 이용하여 공격하고자 하는 대상 시스템에 백도어를 설치하도록 하는 기법[13]이 사용되기도 하였으나, 이를 위해서는 TCP sequence number guessing 과정이 필요하기 때문에 최근에는 거의 사용되지 않고 있다. 또한 IP 패킷 역추적은 현재 특정 시스템으로 IP 주소가 변경된 패킷을 송신하는 시스템을 찾는 기술로서, 여러 중간 경유지를 추적하여 실제 해커의 위치를 찾는 TCP 연결 역추적 기술과는 해결하고자 하는 문제의 대상에 약간의 차이가 있다.

IP 패킷 역추적 기법으로는, (그림 4)에서 볼 수 있듯이 해커가 전송하는 패킷에 해당 패킷을 전달한 라우터를 표시함으로써 추적할 수 있게 하는 패킷 표시 기법[11]을 이용한 연구와 다른 여러 기법을 통한 IP 패킷 역추적을 위한 연구[12]가 진행중이다.

III. 제안된 TCP 연결 역추적 기술

이미 앞서 언급한 바와 같이 본 문서에서 언급하는 역추적 기술은 TCP 연결 역추적 기술이다.

가. 호스트 기반 연결 역추적 시스템

1) CIS[4]

CIS(Caller Identification System)는 H.T. Jung에 의해 1993년 제안된 시스템이다. CIS는 사용자가 특정 시스템에 접속하고자 할 때, 해당 시스템은 접속을 시도하는 사용자가 그 이전에 거쳐 왔던 모든 시스템에 대한 시스템 목록과 로그인 ID 등의 정보를 요구한다. 그리고 요구에 따라 이전의 경유 시스템 목록을 입력 받게 되면, 모든 경유 시스템과의 통신을 통해 각 시스템에 대해 입력된 시스템 및 로그인 ID 목록이 정당한 것인지를 확인하게 되고, 이러한 목록이 유효할 때만 접속을 허락한다. 즉 한 시스템에 접속하기 위해서는 자신이 경유한 모든 시스템에 대한 목록을 제공해야 하는 것이다.

이런 형태의 역추적 시스템은 실제 역추적이라기 보다는 미리 사용자가 거쳐온 시스템의 목록을 관리하는 것으로, 정상적인 사용자들이 접속하는 데도 많은 지연을 초래하게 된다. 또한 침입이 발생하기 이전에 수행하는 작업이 많기 때문에, 자원 활용 면에서 비효율적이라고 할 수 있다. 그리고, CIS는 접속을 원하는 사용자가 거쳐온 시스템 각각에 대한 인증을 거쳐가는 시스템마다 요구하므로 이로 인한 네트워크 부하가 크고, CIS에 오고 가는 인증을 위한 메시지의 무결성을 보장하지 못하는 단점이 있다.

2) AIAA[5]

AIAA(Autonomous Intrusion Analysis Agent) 시스템은 침해를 당한 서버의 해킹 피해 분석과 추적을 위한 로그 분석을 에이전트를 이용해 자동화한 역추적 시스템이다. AIAA 시스템은 침입자가 거쳐온 경유 시스템의 관리자의 도움을 받아 AIAA를 설치하고, 이 시스템에서 바로 이전의 침입경로와 해킹 흔적을 분석하고 다시 이전의 침입시스템으로 분석을 옮겨가서 최종 경유지 서버까지 거슬러 간다.

본 시스템은 역추적 경로상에 존재하는 시스템들의 관리자의 도움을 받아 설치하기 때문에 역추적을 완료하기까지 많은 시간이 필요하게 된다. 또한 역추적 경로상에 존재하는 모든 시스템에 직접 접속해야 하기 때문에 만약 관리자와의 협조가 불가능하여 시스템으로의 접근이 불가능한 경우 역추적 자체가 불가능할 수도 있다.

나. 네트워크 기반 연결 역추적 시스템

1) Thumbprints based algorithm

Thumbprint란 말 그대로 지문을 의미한다. Thumbprint를 이용하는 방법은 역추적 시스템 전체를 의미하는 것이 아니라 역추적을 위해 공격자의 시스템으로부터 공격 대상 시스템까지의 연결 체인을 구성하는 알고리듬이다. 본 알고리듬은 연결 체인에 속하는 호스트들이 속한 네트워크 상에 송수신되는 데이터를 수집하여 비교한다.

Thumbprints를 이용한 방법의 아이디어는 공격자의 시스템으로부터 해커가 위치하고 있는 시스템까지의 연결 체인에 송수신되는 데이터는 동일할 것이라는 점을 이용한다. 즉, 공격에 사용되는 연결에서 송수신 되는 데이터로부터 추출한 내용 정보를 특정 함수를 적용하여 얻어낸 thumbprints가 일정수준 이상 동일한 경우 두 연결은 하나의 연결 체인상에 존재하는 것으로 판단한다.

그러나 패킷이 암호화되거나 터널링되어 패킷의 내용이 변경되는 경우, 해당 연결 체인을 구성할 수 없는 경우가 발생할 수 있다. 그리고, 같은 연결 체인에 속하지 않는 다른 연결의 송수신 데이터가 우연히 동일하게 되는 경우, 같은 연결 체인으로 분류할 수 있는 경우가 발생할 수 있으며, 서로 같은 연결 체인에 속하지만 같지 않은 연결 체인에 속하는 것으로 판단할 수 있는 경우가 발생할 수 있다.

2) Timing based algorithm

Timing을 이용한 방법 역시 연결 체인을 구성하기 위한 알고리듬이다. 본 알고리듬은 해커가 입력하는 키보드 입력에 의해 발생하는 데이터 송신 간격은 프로그램이 송신하는 데이터에 비해 매우 크기 때문에, 이를 쉽게 파악할 수 있고, 만약 같은 연결 체인에 속한다면 그 간격이 매우 유사할 것이라는 점을 이용한다.

이 시스템은 ON period와 OFF period를 이용하여 각각의 상태가 변화하는 시점과 한 상태를 유지하는 시간 간격을 분석하여 같은 연결 체인에 속하는지 여부를 판단하게 된다.

3) TCP sequence number의 증가 정도를 이용한 알고리듬

TCP sequence number를 이용하는 알고리듬은 비록 송수신되는 데이터가 암호화 되더라도 데이터의 양은 크게 변하지 않는다는 점에 착안하여 sequence number의 증가 정도를 변동 폭의 조정을 통해 비교하고 연결 체인을 구성하는 알고리듬이다.

4) Sleepy Watermark Tracing(SWT)

Sleepy watermark 역추적 시스템은 침입에 대한 응답 패킷에 워터마크를 삽입하여 역추적을 수행한다. SWT 기법은 다음과 같은 형태로 이루어진다.

한 네트워크에는 guardian gateway가 존재하고, 이와 연동되어 동작하는 guarded host가

존재한다. 최초 침입이 발생할 때까지는 아무런 추가적인 동작이 진행되지 않은 일반적인 상태로 존재한다. 침입이 발생되면—이는 guarded host내의 IDS에 의해 탐지된다—guarded host의 SWT subsystem의 sleepy intrusion response 모듈의 작동이 시작되고 이때부터 일반 host에 도착되는 패킷에 의한 응답은 watermark enabled application에 의해 작성되기 시작한다. 이는 일반적인 응답패킷에 워터마크를 삽입하여 송신을 시작한다. 이렇게 역추적이 시작되면 이는 guardian gateway의 active tracing 모듈과 연동되어 워터마크가 삽입된 패킷을 찾기 시작한다.

본 SWT 역추적 기법은 공격에 대한 응답 패킷을 이용하여 해커의 위치를 추적하기 때문에 빠르고 정확한 역추적이 가능하다. 그러나, watermark enabled application이 필요하다는 문제로 인해 실제 인터넷 환경에 적용하기에는 큰 문제를 가지고 있다. 또한 해커에 의해 사용되는 연결이 암호화 되는 경우에는 역추적이 전혀 불가능할 수 있다는 단점이 존재한다.

IV. 결 론

본 문서에서는 최근 활발히 연구되고 있는 TCP 연결 역추적 기법에 대해 알아보았다.

해킹시도의 급증과 함께 능동적인 해킹 방지 기법에 대한 필요성이 인식되어, 역추적의 중요성이 크게 대두되고 역추적을 위한 연구가 활발히 진행되고는 있으나, 현재의 인터넷 환경에 적용할 수 있는 TCP 연결 역추적 기술은 아직까지 제안되고 있지 못한 것이 현실이다.

호스트 기반의 역추적 시스템의 경우에는 인터넷의 다양성이라는 큰 특징으로 인해 역추적에 성공하기 위해서는 역추적을 위해 필요한 모듈이 인터넷 상의 모든 호스트에 설치되어야 한다. 또한 비록 인터넷 상의 모든 호스트에 역추적 모듈을 설치한다 하더라도, 임의의 호스트에 대한 무결성을 유지하기가 매우 어렵기 때문에 호스트 기반의 역추적은 현재의 인터넷 환경에 적용할 수 있는 가능성이 거의 없다고 할 수 있겠다.

네트워크 기반의 역추적 시스템의 경우에는 임의의 네트워크 노드에서 얻을 수 있는 패킷 정보를 이용하여 특정 연결이 다음단계의 어떤 연결과 같은 연결 체인에 속하는지를 찾는 기법이 주로 제공되고 있다. 즉, 이를 활용한 역추적 시스템 전체 프레임 워크를 구성하는데 문제가 있기 때문이다. 이는 실제로 동일한 연결체인에 속하는 연결들을 찾을 수 있다고 하더라도 네트워크 상에 송수신되는 패킷으로부터 얻은 정보를 서로 비교하려 할 때 발생하는 순서관계의 일치 즉, 시퀀싱(sequencing) 문제를 해결하기가 매우 어렵고, false positive 혹은 false negative가 발생하기 쉬우며, 네트워크 상에 존재하는 각각의 모든 연결에 대한 정보를 수집 기록해야만 하기 때문이다.

이와 같은 문제를 가지고 있지 않았던 방법이 SWT이다. SWT는 송수신되는 패킷을 지속적

으로 관찰하고 이를 이용하여 연결 정보를 추출하는 기존의 방식이 아니라, 해커의 공격에 대한 응답 패킷에 워터마크를 삽입하고 이를 탐지함으로써, 실시간으로 해커의 위치를 추적하는 방식을 이용하기 때문이다. 그러나 SWT를 활용하기 위해서는 watermark enabled application을 이용해야 하는데, 이는 현실적으로 무리가 있는 방법이다. 또한 SWT는 송수신되는 패킷의 데이터 영역을 이용하기 때문에, 해커가 암호화된 연결을 사용하게 되면, 역추적이 불가능하게 된다.

지금까지 제안된 각종 역추적 기법은 아직까지 실제 인터넷에 적용하여 사용할 수 있는 기법이 존재하지 않는다. 그러나, 이는 앞서 언급한 현재 인터넷의 특징으로 인해 발생하는 것으로 이를 해결하기가 사실상 매우 어렵다. 이에, 역추적 자체를 액티브 네트워크 기반에서 수행하고자 하는 시도가 계속되고 있다. 그러나, 현재의 인터넷 망에 적용할 수 있는 역추적 기법의 개발을 위한 노력은 지속되어야 하고, 그렇게 될 것이다.

앞으로의 역추적은 SWT 기법을 확장하고, 암호화된 연결을 추적할 수 있는 기법이 추가되어 역추적을 수행할 수 있게 되지 않을까 예상한다.

<참 고 문 헌>

[1] http://www.cert.org

[2] Buchholz, Thomas E. Daniels, Benjamin Kuperman, Clay Shields, “Packet Tracker Final Report,” CERIAS Technical Report 2000-23, Purdue University, 2000.

[3] K. Yoda and H. Etoh, “Finding a Connection Chain for Tracing Intruders,” In F. Guppens, Y. Deswarte, D. Gollamann, and M. Waidner, editors, 6th European Symposisum on Research in Computer Security - ESORICS 2000 LNCS -1985, Toulouse, France, Oct. 2000.

[4] H.T. Jung et al. “Caller Identification System in the Internet Environment.,” Proceedings of the 4th Usenix Security Symposium, 1993.

[5] Chaeho Lim, “Semi-Auto Intruder Retracing Using Autonomous Intrusion Analysis Agent,” FIRST Conference on Computer Security Incident Handling & Response 1999.

[6] Steven R. Snapp, James Brentano, Gihan V. Dias, “DIDS(Distributed Intrusion Detection System) - Motivation, Architecture, and An Early Prototype,” Proceedings of the 14th National Computer Security Conference, 1991.

[7] S. Staniford-Chen and L.T. Heberlein. “Holding Intruders Accountable on the Internet,” In Proceedings of the 1995 IEEE Symposium on Security and Privacy, 1995.

[8] Y. Zhang and V. Paxson, “Detecting Stepping Stones,” Proceedings of 9th USENIX Security Symposium, Aug. 2000.

[9] D. Schnackenberg, K. Djahandari, and D. Sterene, “Infrastructure for Intrusion Detection and Response,” Proceedings of DISCEX, Jan. 2000.

[10] D. Schnackenberg, K. Djahandary, and D Strene, “Cooperative Intrusion Traceback and Response Architecture(CITRA),” Proceedings of the 2nd DARPA Information Survivability Conference and Exposition(DISCEXII), June 2001.

[11] Dawn X. Song and Adrian Perrig, “Advanced and Authenticated Marking Schemes for IP Traceback,” Proceedings of InfoCom 2001.

[12] Stefan Savage, David Wetherall, Anna Karlin “Practical Network Support for IP Traceback,” Proceedings of the 2000 ACM SIGCOMM Conference, Stockholm, Sweden, Aug. 2000, pp295-306.

[13] 포항공대 유닉스 보안 연구회 저, “Security Plus for Unix” 영진.com, 2000.

[14] 정현철, “Unix 로그 분석을 통한 침입자 추적 및 로그 관리 Part II,” Korea Computer Emergency Response Team Coordination Center 기술문서, 2001. 7.

[15] Wright Stevens, “Unix Network Programming,” Prentice Hall, 1998.

[16] W.R. Stevens. TCP/IP Illustrated, Vol.1, Addison Wesley, 1994.