2001 년 한국정보처리학회 추계 학술발표논문집 제 8 권 제 2 호

침입탐지시스템의 성능 향상을 위한 패킷 폐기 방안에 대한 연구

문종욱*, 임강빈*, 김종수*, 정기현*, 최경희**
* 아주대학교 전자공학부
** 아주대학교 정보 통신 대학원
e-mail : lache@madang.ajou.ac.kr

A Study on policy of packet dropping for enhancing IDS performance

Jong-Wook Moon*, Kang-Bin Yim*, Jong-Su Kim*,
Gihyun Jung*, Kyunghee Choi**
*Dept. of Electronics Engineering, Ajou University
**A Professional Graduate School
for Information and Communication Engineering, Ajou University

요 약

인터넷의 급속한 성장과 함께 보안이 사회적인 이슈가 됨에 따라 침입 탐지 시스템이 크게 각광을 받게 되었다. 하지만 이러한 침입 탐지 시스템의 대부분이 범용 하드웨어와 범용 운영 체제를 기반으로 하기때문에 본질적으로 패킷 수집의 성능이 떨어질 뿐 아니라 침입을 탐지하기 위한 비용도 너무나 커서 과도한 패킷이 유입될 때 제대로 처리를 못 하게 된다. 본 논문에서는 이러한 침입 탐지 시스템의 성능에 대한 문제점 개선을 위해서 네트웍 부하의 감소가 보장되고 불법 침입도 유해 정보도 아닌 내부 서버의 실제 웹 컨텐트를 담은 패킷의 폐기를 제안하고 실험을 통해서 제안 방안의 가능성을 보인다.

1. 서론

1980 년대 Anderson 에 의해 침입 탐지 시스템의 이론이 나온 이후로 침입 탐지 시스템의 성능 향상을 위한 연구가 많이 진행되었다. Headye 와 Luger 의 분류 시스템 기술을 이용한 침입탐지 기술과 Kumar 와 Spaffod 의 매턴 매칭 기법에 기반한 침입탐지 기술 등을 예로 들 수 있다[1][2]. 하지만 이러한 일련의 침입 탐지 시스템의 성능 향상을 위한 연구들은 모두 침입 탐지 알고리즘 자체에 편중되어 있다.

본 논문에서는 기존 침입 탐지 시스템 향상을 위한 연구와는 다른 관점에서 접근을 시도한다. 침입 탐지 시스템의 하드웨어적인 한계와 탐지 소프트웨어의 거대화에 따른 엄청난 시스템 부하로 인해서 자연히 발생하게 되는 비정상적인 패킷 폐기를 줄이기 위해서 탐지 시스템에 불필요한 패킷으로 분류될 수 있는 패킷을 미리 폐기하는 시스템을 제안한다.

침입 탐지 시스템의 성능 향상을 위해 본 논문에서는 폐기 가능한 패킷을 분류한다. 이 중에서 본 논문은 서버 팜 환경에서 성능이 극대화 될 수 있는 내부 서버로부터 외부 사용자에게로 나가는 HTML 패킷 폐기에 초점을 맞춘다.

본 논문 구성은 서론에 이어, 2 장에서는 연구의 동기가 된 침입 탐지 시스템의 성능 제한 고찰을 3 장에서는 패킷 폐기 분류를 4 장에서는 테스트 베드의 설계와 테스트 환경을 5 장에서는 모의 실험의 결과를 나타내었고, 마지막으로 6 장에서는 결론에 대해 서술하였다.

2. 제안 동기

본 장에서는 침입 탐지 시스템의 성능 문제를 침입 탐지 시스템의 구조적인 관점과 자체적인 관점에서 기술하고 패킷 폐기를 통해서 성능 향상을 얻을 수

있음을 살펴본다.

침입 탐지 시스템의 성능은 실제 탐지 소프트웨어의 알고리즘에 따라 많이 좌우되지만 또한 하드웨어로부터 상위 소프트웨어로 이어지는 구조에 의해서도 많은 차이가 난다. 이러한 구조적인 문제로부터 발생하는 침입 탐지 시스템의 성능 저하에 대한 두 가지 관점을 논한다. 하나는 범용 하드웨어 시스템의 한계이고 또 다른 하나는 범용 운영체제를 기반으로 한 소프트웨어의 불필요한 처리에 대해서이다.

첫째, AT&T Labs 에서 발표한 연구 조사서를 통해 알 수 있듯 네트웍 처리 성능 향상을 위해서는 전체적인 시스템 구조의 변경이 불가피하다[3]. 하지만 대부분의 침입 탐지 시스템은 범용 하드웨어 시스템을 기반으로 하기 때문에 이미 정형화되어 만들어져 있는 네트웍 인터페이스 카드를 사용하고 높은 지연이 유발되는 I/O 버스를 사용하게 될 수 밖에 없다. 이것은 침입 탐지 시스템에서 중요한 기능 중 하나인 패킷 수집의 성능이 떨어질 수 있음을 내포한다.

둘째, 대부분의 침입 탐지 시스템에서 운영 체제 또한 NT 또는 Linux 와 같은 범용 운영 체제를 사용한다. 커널이 침입 탐지를 위해 특화 되지 않았다는 것은 침입 탐지와 무관한 코드들이 많다는 뜻이다. 이는 엄청난 네트워크 로드가 걸릴 때 불필요한 코드의 처리 시간으로 인해 손해를 감소해야 됨을 의미한다. 그리고 대부분의 침입 탐지 시스템에서의 패킷 수집은 Berkeley 대학교에서 커널 내에 구현한 BPF(BSD Packet Filter)디바이스의 인터페이스를 위한 libpcap 라이브러리를 그대로 또는 약간의 변경을 통해서 이루어지고 있다. 비록 BPF 알고리즘이 다른 CSPF 나 NIT 의 것 보다 성능이 우수하다고는 하나 RISC 를 기반으로 창안된 것이어서 CISC 에서의 성능은 다시 고려해 봐야 할 것이고 사용자 측면에서는 중간의 libpcap 을 또 다시 거쳐야 하므로 오버 헤드가 있다[4].

이와 같은 침입 탐지 시스템은 네트워크에 흐르는 모든 패킷을 수집하고 이를 분석하여 불법적인 행동이나 유해 정보를 찾아내서 로그 정보를 남겨 놓아야 하므로 침입 탐지 시스템이 처리 해야 할 부하가 크다. 침입 탐지 시스템의 부하에 관련해서 충분히 예상이 되는 수행해야 할 일들에 간단히 살펴보고 이러한 부하로 인해 침입 탐지에 있어서 야기되는 문제점을 논하고 자 한다.

첫째, 침입 탐지 시스템은 침입을 탐지 하기 위해 내부에 가지고 있는 침입 패턴에 대한 데이터베이스를 검색하는 일이 필요하다. 데이터베이스 검색에 대한 소프트웨어적 알고리즘이 아무리 우수하더라도 데이터베이스의 양이 엄청나게 많을 때는 수행 시간이 길어지는 것이 당연하다.인터넷 보급 속도의 증가와 함께 인터넷에서의 침입을 위한 방법이 빠른 속도로 새롭게 생겨나고있고 더욱이 그들 많은 방법은 침입 탐지 시스템을 속이기 위해 위장을 한다. 이로 인해 단순한 규칙에 의존하는 데이터베이스에서 복잡하고 연산량이 많아지는 쪽으로 변경되어야만 한다.

둘째, 동시 다발적으로 일어나는 모든 불법 행동을 관리하기 위해서 침입 탐지 시스템은 이상적으로는 무한대의 세션을 관리를 해야 한다. 실질적으로는 메모리의 제한과 하드웨어 또는 운영체제의 한계로 인해서 세션의 양이 제한되어진다. 이러한 세션 관리는 침입 탐지 시스템의 큰 부하의 요소로 작용한다. 예를 들어서 침입 탐지 시스템으로 유입되는 패킷들에 의해 우연히 또는 악용적인 이유로 엄청나게 많은 양의 새로운 연결이 이루어 진다면 아무리 세련된 세션 관리 기법을 도용을 하더라도 시스템은 엄청난 부하에 직면한다.

셋째, 침입 탐지 시스템은 이른바 해킹이라 불리는 불법 침입에 대한 처리 뿐 아니라 패킷 자체 내용의 유해 정보의 패턴 검색이 필요하다.

기존의 침입탐지 시스템은 범용 하드웨어 시스템의 한계로 인해 망의 네트워크 부하가 클 때, 특히 작은 크기의 패킷이 burst 하게 들어 올 때 어쩔 수 없는 패킷 폐기와 커널 내에서 다른 처리의 우선 순위에 밀려서 일어나는 libpcap 의 자체적인 손실은 침입 탐지 시스템에 있어서는 가장 치명적인 침입의 패턴의 일부를 잃어버리는 결과를 낳는다. 하지만 본 논문에서 제안하는 방법인 침입 탐지 시스템에게 필요성이 떨어지는 패킷을 미연에 폐기하는 방식을 적용하면 버려진 양에 비례해서 네트워크 로드가 줄어들므로 보다 중요한 패킷의 수집 확률이 높아지게 되는 것이다.

침입 탐지 소프트웨어의 자체적인 무거움으로 인해서 발생하는 패킷 폐기는 탐지 소프트웨어가 인지는 하지만 처리 성능의 한계로 버려야 하는 경우이다. 이 경우에도 본 논문에서 제안하는 패킷 폐기 정책을 적용하면 현저한 성능 향상을 기대할 수 있다. 왜냐하면 침입과 무관한 패킷이 폐기되지 않고 탐지 소프트웨어까지 올라왔을 때 침입의 유형과 유해 정보에 관련된 데이터베이스를 뒤지는 등의 일련의 처리 과정을 모두 거친 후에야 스스로 그 패킷을 내부적으로 버리므로 미리 버려주게 되면 그러한 불필요한 처리를 보다 중요한 패킷에게 할애를 해 줄 수 있기 때문이다.

본 논문에서는 앞으로 침입 탐지 시스템의 성능 또는 기능에 문제가 없이 폐기가 가능한 항목에 대해 고찰을 한 다음 그 중 성능 측면에서 수확이 가장 기대되는 항목인 HTML 패킷 폐기의 방안을 모색하고 실험을 통해 그 실현 가능성 및 침입탐지 시스템의 성능 향상의 가능성을 살핀다.

3. 패킷 폐기 유형

본 장에서는 침입 탐지 시스템의 성능 향상을 위해 폐기 가능한 패킷의 유형을 나누어 보고 각 유형이 적용 가능한 모델을 설정하고 그것을 실현하기 위해 필요한 요소를 살펴 본다. 그 중에서도 본 논문에서 핵심적으로 다루는 외부로 나가는 HTML 패킷의 폐기에 대해서 자세히 살펴 본다.

웹의 폭발적인 성장에는 멀티미디어 서비스가 큰 역할을 하였다고 볼 수 있다. 즉 네트웍의 사용의 대부분이 웹을 차지하고 그 중에서도 멀티미디어 서비스가 차지하는 비중은 상당하다고 볼 수 있다.

내부 사용자가 외부 멀티미디어 서버로부터 동영상

이나 음악을 듣는 경우 연결 설정을 위한 일련의 패킷을 제외한 실제 영상 또는 음악을 담고 있는 패킷은 침입과 무관하므로 IDS 가 해킹 여부를 검토하지 않아도 상관없다. 하지만 멀티미디어 서비스를 위한 프로토콜 타입이 정의 되어 있지 않으므로 단일 패킷의 헤더만으로는 실제 데이터 패킷과 연결 설정을 위한 패킷을 구분할 방법이 없다. 하지만 대부분의 사용자가 웹을 통해서 멀티미디어 서비스를 받고 있다. 이 경우는 HTTP 의 MIME 정보를 모니터링 한 후 real-player 와 같은 잘 알려진 미디어 플레이어로 연결 될 때 해당 요청자의 포트 번호를 추적하면 구분이 가능하다. 그런데 이러한 기법은 멀티 미디어 서버로 접근하는 사용자별의 세션을 따로 관리하는 등 처리 비용이 많이 든다.

최근에 그 보급 속도가 급속히 확산되고 있는 VOIP(Voice Over IP)나 FOIP(Fax Over IP)등의 서비스에서 세션 정보를 제외한 대부분의 데이터 정보는 해킹의 가능성이 거의 없으므로 폐기 가능한 정보에 속한다.

인터넷의 급속한 보급은 사용에 편리하고 유려한 인터페이스를 가진 웹의 성장에서 기인한다고 볼 수 있다. 이러한 이유로 네트웍에 지나다니는 패킷의 상당수가 HTML 이라고 해도 과언이 아니다. 그런데 웹 서버로부터 나가는 HTML 패킷 중 실제 웹 컨텐트를 담고 있는 패킷은 침입과는 무관하므로 역시 폐기가 가능하다. 물론 웹의 실제 컨텐트는 침입 탐지 시스템에게 있어서 유해 정보 탐색의 대상이 되지만 내부 웹 서버의 내용이 검증 되었다고 한다면 불필요한 유해 패턴 검색의 처리를 하는 것에 지나지 않는다.

HTML 패킷 폐기가 제대로 이루어지기 위해서는 웹 서버로의 접속을 위해 사용된 TCP 연결 패킷은 침입 탐지 시스템이 불법 행동을 탐지하기 위한 세션을 맺는데 핵심이므로 절대 버려져서는 안 된다. 구체적인 폐기 알고리즘본은 다음 장에서 다룬다. 논문에서는 이러한 검증된 내부 웹 서버로부터 외부로 나가는 HTML 패킷에 대해서 패킷 폐기를 서버 팜 (server farm) 환경에서 효율적으로 사용될 수 있는 패킷 폐기 방안을 제안한다. 웹 서버 집합체인 서버 팜 환경에서는 네트웍에 HTML 패킷의 빈도가 크므로 침입 탐지 시스템의 부하를 획기적으로 줄일 수 가 있다..

4. 패킷 폐기 방안 구현

본 장에서는 서버 팜 환경에서 침입 탐지 시스템의 성능 향상을 위한 HTML 패킷 폐기의 알고리즘을 확인하고 성능 테스트를 위해 실제로 구현한 테스트 베드의 구성 요소와 그 디자인 이슈에 대해 알아 본다.

4.1. 소프트웨어

제안한 HTML 패킷 폐기 알고리즘의 중요한 요소는 침입 탐지 시스템에게 있어서 불법 행동 가능성의 유무를 정확하게 구분하여야 한다는 것이다. 불법 행동의 여지가 없는 HTML 실제 컨텐트를 담고 있는 패킷을 구분하는 알고리즘을 그림 1 의 순서도와 그림 2 의 의사 코드에서 표현한다.

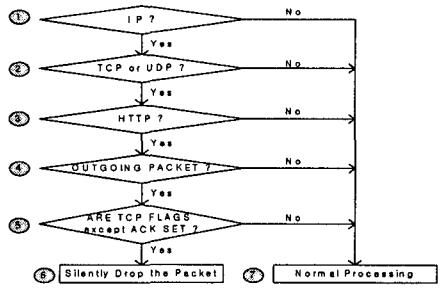

그림 1. HTML 패킷 폐기 알고리즘의 순서도

```
When packet is coming to PDE( packet dropper engine )
  drop_flag = FALSE
  if type field in MAC header is IP
    if protocol field in IP header is UDP or TCP
      if source port # in TCP or UDP header is HTTP
        if ((PDE's IP XOR dest. IP) AND NETMASK )
          if does any flag except ACK exist in TCP flags
            // This packet is real WEB Content
            drop_flag = TRUE
          else Nothing to do
        else Nothing to do
      else Nothing to do
    else Nothing to do
  else Nothing to do

  if drop_flag is FALSE
    Deliver this packet to Intrusion Detection System
  else
    Nothing to do( dropping packet )
```

그림 2. 패킷 폐기 알고리즘의 의사코드

제안되는 HTML 패킷 폐기 알고리즘은 그림 1 의 순서도에서 알 수 있듯 몇 가지 패킷 헤더의 특정 필드들의 패턴 매칭 만으로 가능하다. 이를 구체적으로 살펴보면 다음과 같다. 구체적인 고찰에 앞서 설명이 되어지는 일련의 패턴 매칭의 순서들은 패킷이 유입될 때 마다 매번 처리를 해야하는 행동임을 또 설명에 사용되어지는 패킷의 필드의 위치는 그림 3 에서 짙은 색으로 칠해 진 부분임을 밝힌다.

가장 먼저 비교하는 부분은 이더넷 헤더의 타입 필드가 0x800 인지를 확인해서 상위 프로토콜이 IP 인지를 비교한다. 다음은 IP 헤더의 프로토콜 필드를 비교해서 6 또는 17 인지를 확인해서 TCP 또는 UDP 임을 구분한다. 다음은 TCP 헤더의 목적지 포트 번호가 80 또는 8080 인지를 확인해서 HTTP 패킷 임을 구분한다. 다음은 이미 패킷 폐기기에 설정되어 있는 자신의 IP 정보와 Netmask 정보를 이용해서 패킷이 외부 사용자로 나가는 것인지를 구분한다. 지금까지의 패턴 매칭에 대한 조건이 만족되었다면 이 패킷은 외부로 나가는 HTTP 패킷이라고 할 수 있다. 하지만 이러한 패킷들 중에서 연결 설정에 관련된 패킷은 폐기를 하지

않아야 하므로 마지막으로 TCP 의 Flag 필드를 확인해서 연결 설정 유무를 판단한다.

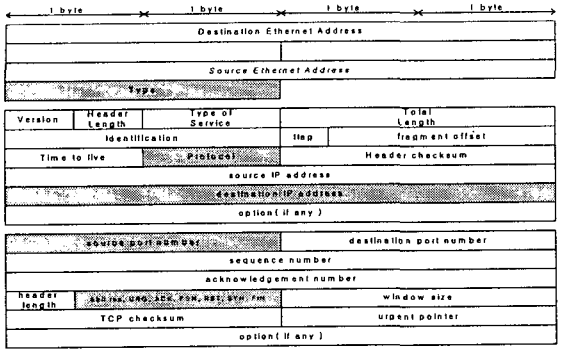

그림 3. 패킷 폐기 알고리즘에 사용되는 필드들

4.2. 하드웨어

패킷 폐기를 위한 테스트 베드는 그림 4 에서 볼 수 있듯이 크게 세 부분으로 나누어진다. 패킷의 수집과 패킷 폐기 여부 판단 및 처리를 담당하는 입력부와 프로세서간 통신을 위한 IPC 채널과 실제 침입 탐지 시스템에 패킷을 전달하는 출력부로 나눈다. 입력부는 과도한 네트웍의 유입에 대한 처리와 전처리 작업량이 많을 수 있으므로 보다 효율적인 네트웍 처리가 가능한 모토로라사의 32bit RISC 프로세서를 사용하였다.

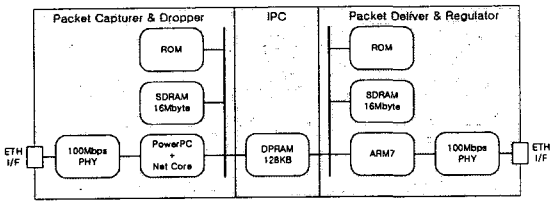

그림 4. 테스트 베드의 내부 구조도

본 논문에서 테스트 베드를 구성하는데 있어서 고려 되었던 이슈들은 다음과 같다. 첫째는 프로세서와 하위 네트웍 처리부의 연결 관계가 네트웍 처리에 있어서 최상의 성능을 발휘 할 수 있는 효율적인 구조를 가지는지에 대한 것이다. 둘째는 패킷 폐기를 위한 코드의 성격상 시간적 국지성이 뛰어나기 때문에 충분한 L1 캐시가 제공되는지에 대한 것이다. 셋째는 프로세서간 통신을 위한 것으로 가장 적합한 모델을 무엇인가에 대한 것이다.

5. 모의 실험

본 논문에서 HTML 패킷 폐기를 실험하기 위해 구성한 환경은 그림 5 와 같다. 망의 흐르는 패킷의 수집과 특정 패킷의 폐기가 제대로 되는지를 확인하고 네트웍의 부하에 미치는 영향을 확인하기 위해 NAI 사의 sniffer 를 패킷 폐기기의 입력부와 출력부에 설치하여서 실험을 하였다.

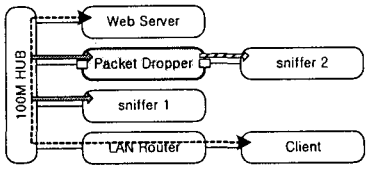

그림 5. 실험 환경

이러한 패킷 폐기를 통해서 네트웍 부하가 줄어드는 것을 확인하였다. 즉, 전체 네트웍 부하의 총량에서 HTML 웹 컨텐트의 비율 만큼의 부하가 줄어 들었다. 패킷 폐기를 통해 줄어든 네트웍 부하량 만큼 패킷 수집의 확률을 높일 수 있을 것이다. 또한 네트웍 부하량 감소에 따라서 침입 탐지 시스템이 처리해야 할 요소가 줄어 든 것이므로 보다 불법적인 행동을 감시하고 탐지하는데 성능을 할애할 수 있다. 그리고 이러한 패킷 폐기를 서버 팜 환경에서 사용을 한다면 유해 정보와 무관한 패킷 폐기를 통해 IDS 성능을 더욱 개선할 수 있게 될 것이다.

6. 결론

본 논문에서는 침입 탐지 시스템의 성능 향상을 위해 HTML 패킷 폐기 방안을 제안하고 실제 테스트 베드를 제작하여서 실험하였다. 본 연구를 통해서 침입 탐지 시스템의 기능적인 문제가 발생하지 않는 범위 내에서 의도적인 패킷 폐기가 침입 탐지 시스템의 성능 향상을 이룰 수 있음을 확인하였다.

현재는 단순히 네트웍 부하량만을 조사해서 침입 탐지 시스템의 성능 향상에 도움이 되리라는 것을 조사하고 있다. 네트웍의 부하량만을 테스트 하는 것은 burstness 등의 네트웍의 다양한 요소와 침입 탐지 소프트웨어의 실제 처리에 따른 영향이 반영되지 않는다. 향후에는 보다 정밀한 성능 측정 모델을 선정하는 것이 필요하리라 본다.

참고문헌

[1] R. Headye, G. Luger, A. Maccabe, and M.Servilla, "The architecture of a network level intrusion detection system", Technical Report, Department of Computer Science, University of New Mexico, August 1990

[2] S. Kumar. "Classification and Detection of Computer Intru-sions". PhD thesis, Purdue University, West Lafayette, IN 47907, 1995.

[3] Charles D. Cranor, R. Gopalakrishnan, Peter Z. Onufryk, "Architectural consideration for CPU and network interface integration", In IEEE Micro, January-February 2000

[4] S. McCanne and V. Jacobson. "The BSD Packet Filter: A New Architecture for User-level Packet Capture". In Proceedings of the 1993 Winter USENIX Conference,San Diego, CA, January 1993.

[5] T. H. Ptacek and T. N. Newsham. "Insertion, evasion, and de-nial of service: Eluding network intrusion detection", Technical report, Secure Networks, Inc., January 1998.

[6] W. R. Stevens. "TCP/IP Illustrated, volume Volume 1 -- The Protocols of Professional Computing Series". Addison-Wesley, 1994.