제19회 한국정보처리학회 춘계학술발표대회 논문집 제10권 제1호 (2003.5)

하드웨어 방화벽의 구현에 적합한 패킷 필터링 알고리즘

홍재인*, 채현석**, 조태경***, 최명렬*
*한양대학교 전기전자제어계측학과
**동원대학 인터넷 정보과
***한양사이버대학교 컴퓨터 학과
e-mail : monadic1979@hanyang.ac.kr

A Packet Filtering Algorithm for H/W Firewall Implementation

Jae-In Hong*, Hyun-Seok Chae**, Tae-Kyung Cho***, Myung-Ryul Choi*
*Dept of EECI, Hanyang University
**Dept of internet & information retrieval, Tongwon college
***Dept of Computer Science, Hanyang Cyber University

요 약

본 논문에서는 내부 네트워크의 보안을 담당하는 방화벽 시스템 중에서 기존의 패킷 필터링 형태를 설명하고, 하드웨어 기반의 패킷 필터링 방화벽 시스템구성을 위한 알고리즘과 구조를 제안한다. 소프트웨어 기반의 필터링은 처리 속도가 느리기 때문에 성능저하를 우려하여, 실제 보안에서는 사용하지 않는 경우가 많았다. 그러나 제안한 하드웨어 기반의 필터링을 수행하면, 만족스러운 처리 속도를 보장할 수 있고 성능 저하 없이 보안을 위해 유용하게 동작하는 장점이 있다.

1. 서론

최근 인터넷에 연결하여 사용하는 내부 네트워크의 보안 시스템에 대한 연구가 국내에서 활발히 진행되고 있으며, 많은 상용 제품들이 개발되고 있다. 내부 네트워크를 보호하기 위한 방법으로 네트워크를 연결해 주는 장치에서 입출력되는 패킷을 분석하여 패킷 트래픽을 제어 및 차단하는 방화벽 시스템을 사용할 수 있다. 방화벽 시스템의 주요 기능으로는 사용자 인증, 접근 제어, 트래픽 암호화, 감시 추적 기능 등이 있으며, 그 형태는 1)응용 기반(Application-based)의 형태, 2)패킷 필터링(Packet filtering)형태, 3)상태 감시(Stateful-inspection)형태, 3)프락시 서버(Proxy server)형태, 4)NAT(Network Address Translation)형태등으로 나뉘고, 네트워크 계층, 전송 계층, 응용계층에서의 보안을 담당할 수 있다[1]. 또한 구현의 측면에서 방화벽 시스템은 크게 하드웨어 기반의 시스템과 소프트웨어 기반의 시스템으로 나눌 수 있다[2].

현재 방화벽 시스템의 필터링 기능은 대부분 소프트웨어 기반으로 만들어졌다. 소프트웨어 기반 필터링의 경우 보안 정책의 변화에 따라 즉각적으로 규칙을 바꿀 수 있지만, 속도가 느리고 코딩이 복잡해지는 단점을 가지고 있어, 방화벽의 성능 저하를 가져오며, 그로 인해 실제로는 필터링 기능을 거의 사용하지 않는다. 따라서 하드웨어를 기반으로 패킷 필터링을 구현하면 속도의 문제점이 해소되어, 필터링 기능을 효율적으로 사용할 수 있는 장점이 있다.

이에 따라 본 논문에서는 하드웨어 기반 패킷 필터링 방화벽 시스템의 구현을 위한 알고리즘을 제안하였으며, 2장에서 기존의 패킷 필터링의 규칙에 대하여 알아보고, 3장에서 제안한 알고리즘을 설명하고 4장에서 결론을 맺는다.

2. 기존의 패킷 필터링 규칙

패킷 필터링 방화벽 시스템에서는 네트워크에서 사용하는 통신 프로토콜의 형태, 송신지 주소와 수신지 주소, 통신 프로토콜의 제어 필드 그리고 통신 시 사용하는 포트 번호를 분석하여 내부에서 외부 네트워크로 전송되는 패킷 트래픽과, 외부에서 진입하는 패킷 트래픽의 허가 혹은 거절을 결정한다

[3,4,5]. 아래의 <그림 1>에서 네트워크 상에서 패킷 필터링이 이루어지는 전송 계층과 네트워크 계층에서의 위치를 보여준다.

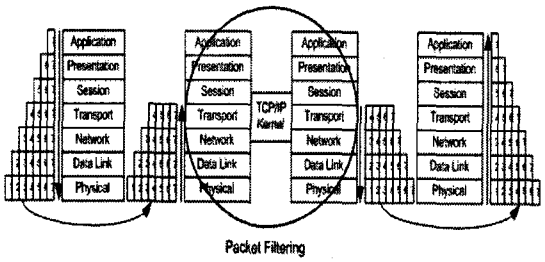

그림 1. 패킷 필터 동작

패킷 필터링에 사용되는 정보는 다음과 같다.

1. IP 송신지 주소 (IP source address)
2. IP 수신지 주소 (IP destination address)
3. 프로토콜 (TCP/UDP/ICMP)
4. TCP 혹은 UDP 송신지 포트 번호
5. TCP 혹은 UDP 수신지 포트 번호

이러한 정보들을 이용하여, 패킷 필터 규칙을 정하게 되고, 규칙이 정해지면 인터넷 주소에 적용하는 허가 혹은 거절의 조건의 순차적인 액세스 집합인 액세스 규칙을 정의하게 된다. 이러한 액세스 규칙을 순차적으로 적용하여 패킷의 수락 여부를 결정하게 된다. 다음 표1에서, 패킷 필터링 초기의 액세스 규칙을 나타내었다.

표1. 패킷 필터 초기 규칙

규칙	방향	송신지	수신지	타입	포트	결정
A	In	External	Internal	TCP	25	수락
B	Out	Internal	External	TCP	>1023	수락
C	Out	Internal	External	TCP	25	수락
D	In	External	Internal	TCP	>1023	수락
E	Either	Any	Any	Any	Any	수락

위의 초기 규칙은 관리자의 보안 정책에 따라 변경이 가능하며, 시스템의 상태나 네트워크의 상태에 따라서도 변경이 가능하다. 따라서 패킷 필터링에서 생길 수 있는 규칙은 매우 많아질 수 있다.

이러한 패킷 필터링 규칙을 적용하여 직접 패킷을 수락하거나 혹은 거절하는 예를 표2에서 설명하였다. 표2의 패킷 1에서 4까지는 내부의 호스트 IP 주소가 172.16.1.1이고, 외부의 호스트 IP 주소가 192.168.3.4 일때 외부 포트 1234에서 내부의 포트 25번으로의 접근을 허락하는 예제를 보여준다. 또한 패킷 5와 6의 경우에는 외부(10.1.2.3)에서 포트 5150을 사용하여 내부(172.16.3.4)의 8080 포트로의 공격이 성공한 경우를 보여준다. 이는, 초기의 필터링 규칙에서 1023이상 포트의 데이터를 받아들이도록 하였기 때문에 일어난 현상이다.

표2. 패킷 필터링 예제

패킷	방향	송신지	수신지	타입	포트	결정
1	In	192.168.3.4	172.16.1.1	TCP	25	수락(A)
2	Out	172.16.1.1	192.168.3.4	TCP	1234	수락(B)
3	Out	172.16.1.1	192.168.3.4	TCP	25	수락(C)
4	In	192.168.3.4	172.16.1.1	TCP	1357	수락(D)
5	In	10.1.2.3	172.16.3.4	TCP	8080	수락(D)
6	Out	172.16.3.4	10.1.2.3	TCP	5150	수락(B)

3. 제안한 하드웨어 패킷 필터링 방화벽 시스템

2장에서 기존의 패킷 필터의 규칙과 그 예제등을 살펴보았다. 패킷 필터링 방화벽 시스템에서 중요하게 고려해야 할 요소는 1)방화벽의 규칙을 어떻게 정의할 것인가 2) 보안 정책에 따라 요구되는 규칙을 어떻게 변화시킬 것인가 3) 규칙을 어떻게 적용할 것인가 4) 패킷의 동작 속도를 어떻게 높일 것인가 등이다.

기존의 소프트웨어 기반의 패킷 필터링 방화벽 시스템은 규칙의 정의와 적용 및 동작 속도 등이 CPU의 속도에 의해 결정되어 만족할 만한 성능을 보여주지 못한다. 이러한 단점은 패킷 필터링을 하드웨어로 구현함으로써 해결할 수 있다. 하드웨어 기반의 방화벽 시스템은 보안 정책의 변화에 따른 규칙의 변화에 민감하게 대응할 수 없다는 문제점이 있었지만, 이는 전원이 차단되면, 내용이 지워지는 FPGA의 특성을 이용하여 해결 할 수 있다.

3.1 제안한 하드웨어 패킷 필터링 알고리즘

패킷 필터링 방화벽 시스템에서 처리하는 패킷은 크게 외부 네트워크에서 내부 네트워크로 들어오는 패킷과 내부 네트워크에서 외부로 나가는 패킷으로 나눌 수 있다. 제안한 방식에서는 내부에서 내보내는 패킷에는 간단한 규칙을 적용하고, 내부로 진입하는 패킷에는 패킷의 IP 송신지 주소, IP 수신지 주소, 포트 번호등의 정보를 가지고 일정한 연산을 취하여, 패킷의 진입을 허락 혹은 거절하는 방식을 사용한다.

먼저 내부에서 외부로 나가는 경우에는 특별히 패킷을 처리하지 않고 아래의 규칙을 적용하며, 연결이 종료되거나 일정 시간 동안 어떠한 패킷도 연결되지 않은 경우를 제외하고는 변경하지 않는다.

- 규칙 1 : 패킷에 대한 프로토콜을 명시한다.
- 규칙 2 : 주소가 바뀐 경우를 제외하고, 패킷의 송신지 주소와 수신지 주소를 명시한다.
- 규칙 3 : 번호가 바뀐 경우를 제외하고, 패킷의 송신지 포트와 수신지 포트를 명시한다.

다음으로 내부로 집입하는 패킷은 네트워크의 보안 및 안정을 위해서 패킷의 헤더 정보를 확인하고, 패킷의 진입을 결정한다. 이 때, 패킷 헤더로부터 받는 정보들을 표3에 나열하였다.

표3. 패킷 필터링을 위한 정보

IP 헤더	TCP 헤더
· 버전(4-비트 필드) · 프로토콜(4-비트 필드) · 송신지 및 수신지 주소(32-비트 필드)	· 송신지 및 수신지 포트 번호(16-비트 필드) · SYN/ACK 비트 · RST/FIN 비트

이러한 정보들 중에서 직접적으로 패킷의 비교에 사용되는 정보는 다음과 같이 세 가지이다.

- IP 패킷의 송신지 주소
- IP 패킷의 수신지 주소
- 수신지 포트 번호

이들 정보는 각각의 모듈로 구현되며, 미리 정해진 규칙과 비교하여 0또는 1의 출력을 내보낸다.

패킷 필터링 프로세스를 설명하면, IP 헤더와 TCP헤더의 정보를 받아 필터링을 위한 IP 송신지 주소 32 비트 필드, 수신지 주소 32 비트 필드, TCP 포트 번호 16 비트 필드를 각각 추출한다. 추출한 데이터들은 비교 모듈로 입력되어, 저장된 초기 규칙과 비교하게 된다. 이때 규칙과 부합되면 1, 부합되지 않으면 0을 출력하고 4비트의 포맷으로 만들어진다. 이 비트 포맷의 최상위 비트는 첫 번째 패킷인지 아닌지를 판별하는 상태 비트가 된다.

패킷의 정보인 4비트 포맷은 네트워크 상태와 관리자의 보안 정책에 따라 만들어진 규칙 4 비트 포맷과 연산이 되며, 그 결과를 이용하여 패킷을 수락할지 혹은 거부할 지를 결정한다.

이때, 네트워크 상태가 혼잡한 경우에는 0000 bit를 출력하여 어떠한 패킷의 진입도 허락하지 않는다.

제안한 패킷 필터링 프로세스를 아래 그림2에 수도(Pseudo) 코드 형식으로 표현하였다.

```
[1] Get the TCP/IP packet
[2] Extract packet information( IP source address and
    IP Destination address and TCP port number)
[3] Compare the packet information and initial rule
    if(start_packet)
       d_start = 1; /* packet start information */
    else
       d_start = 0;
    if (IPS_addr == init_rule_IPS)
       d_saddr = 1; /* IP source address info. */
    else
       d_saddr = 0;
    if (IPD_addr == init_rule_IPD)
       d_daddr = 1; /*IP destination address info. */
    else
       dec_daddr = 0;
    if (TCP_port == init_rule_TCPN)
       d_port = 1; /* TCP port number info. */
    else
       d_port = 0;
    pac_info = d_start • d_saddr • d_daddr • d_port;
[4] Compare pac_info and rule set
    if(congested)
      rule_set = 0000; /* deny all packet */
    else
      rule_set = rule_set;
    decision = pac_info | rule set;
[5] Decide the inbound packet
    if(decision)
      permit inbound packet
    else
      deny inbound packet
```

그림 2. 패킷 필터링 프로세스

필터링의 예를 들면, 내부 네트워크에서 송신지 주소와 포트 번호 정보를 통해 패킷을 수락하는 초기 규칙을 가지고 있을 때, 초기 규칙은 0101 비트로 표현된다. 이때, 진입하고자 하는 패킷이 송신지 주소만 맞다면 연산을 통해, 0100 비트가 된다. 네트워크가 혼잡하지 않다고 가정하고 하나 이상의 정보가 부합될 때 패킷을 수락한다면, 다시 내부 규칙과 OR 연산을 통해 0101 비트를 출력하며, 패킷을 수락한다.

3.2 제안한 하드웨어 패킷 필터링 시스템 설계

제안한 하드웨어 패킷 필터에서는 필터링 연산을 위해서 IP 송신지 주소 비교 모듈, IP 수신지 주소 비교 모듈, TCP 수신지 포트 번호 비교 모듈을 정의였다. 이 모듈들에서는 내부 네트워크에서 정의한 초기 규칙들과 비교하여 각각 0 또는 1의 비트를 출

력하도록 하여, 네트워크 상태나 관리자의 보안 정책에 따른 규칙과의 연산을 통해 패킷 수락을 결정한다. 패킷에 적용하는 규칙은 송신지 주소, 수신지 주소, 포트 번호 중 하나만 고려하는 경우, 두개를 만족하는 경우 혹은 모든 조건을 만족하는 경우 등으로 다양하게 만들어질 수 있다.

제안한 패킷 필터링 시스템 모듈은 크게 4가지 부분으로 나눌 수 있는데, 먼저 입-출력 인터페이스 부분, 그리고 IP 헤더와 TCP 헤더 정보를 저장하는 부분, 정보에서 필요한 주소 및 포트 번호등을 추출해내는 부분, 패킷 필터링 연산을 행하는 부분등이다.

입-출력 인터페이스 부분은 패킷 필터 방화벽 칩과 네트워크 프로세서 간의 연결을 담당하고, 메모리 부분은 IP 헤더 정보와 TCP 헤더 정보 부분이 저장되는 부분이다. 메모리 1은 버퍼와 연결되어, 헤더의 정보들을 각가 분리 할 수 있도록 추출기로 전달된다. 추출기에서 분리된 정보는 필터 모듈로 전송되어 필터 연산을 수행하게 된다. 필터 모듈에서는 추출기에서 전송된 정보들과, 메모리 2에 저장된 규칙 정보를 비교하여 패킷을 수용할지 거부할 지에 대한 결정을 내린다. 다음 그림 3에 전체적인 패킷 필터링 방화벽 시스템의 모듈 간의 연결등을 도시화하였다.

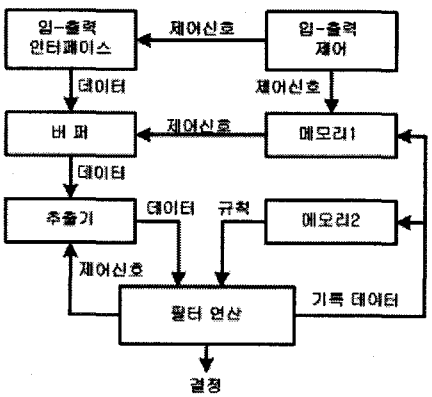

그림 3. 패킷 필터링 시스템 구조

4. 결론

본 논문에서는 인터넷과 같은 외부 네트워크와 연결된 내부 네트워크의 보안을 위한 패킷 필터링 방화벽 시스템을 하드웨어로 구현하기 위한 알고리즘 및 구조를 제안하였다. 실제로 패킷 필터링은 보안을 위해 중요한 과정임에도 불구하고, KT등 네트워크 서비스 업체에서는 성능 저하를 우려하여 필터링 과정을 행하지 않는다. 그 이유는 기존의 필터링 과정이 모두 소프트웨어 기반으로 되어 원하는 속도를 만족 시킬 수 없기 때문이다. 그러나 필터링 과정을 하드웨어로 구현하면 처리 속도가 증가되므로 보안 시스템의 성능 저하 우려 없이 패킷 필터링을 할 수 있는 장점이 있다. 따라서 하드웨어 기반 필터링을 통해 실제적으로 관리자의 보안 정책에 맞는 시스템 보안을 효율적으로 할 수 있다.

향후 제안한 하드웨어 패킷 필터링 방화벽 시스템을 VHDL등의 언어로 시뮬레이션 및 검증할 예정이며, 직접 FPGA를 이용하여 시스템을 구성할 예정이다.

참고문헌

[1] Zalenski, R. "Firewall technologies", IEEE Potential, Volume: 21, Feb/Mar 2002, P.24 - 29

[2] Ayman Kayssi, Louis Harik, Rony Ferzli, and Mohamed Fawaz "FPGA-based Internet Protocol Firewall chip" Electronics, Circuits and Systems, 2000. ICECS 2000. The 7th IEEE International Conference on, Volume:1, 2000 P.316 - 319

[3] Gupta, P., McKeown, N., "Algorithms for packet classification", IEEE Network, Volume : 15, Mar/Apr 2001, P.24 -32

[4] Scott Hazelhurst, Addi Attar "Algorithm for Improving the Dependability of Firewall and Filter Rule Lists", Dependable Systems and Networks, 2000. Proceedings International Conference on, 2000. P.576 - 585

[5] Lili Qiu, George Varghese, Subhash Suri, "Fast Firewall Implementation for Software and Hardware-based Routers", Network Protocols ninth International Conference on ICNP, 2001, P. 241- 250