고성능 패킷 분류를 위한 TCAM 분할

준회원 김 규 호, 강 석 민, 송 일 섭, 정회원 권 택 근*

TCAM Partitioning for High-Performance Packet Classification

Kyu-Ho Kim, Seok-Min Kang, Il-Seop Song *Associate Members*,
Teack-Geun Kwon* *Regular Member*

요 약

네트워크 대역폭 증가에 따라 다양한 서비스의 등장과 함께 네트워크 위협이 꾸준히 증가하고 있다. 고성능 네트워크 보안의 실현을 위해, TCAM 등의 하드웨어를 통한 고속 네트워크에서의 빠른 패킷 분류 방법이 일반적으로 사용된다. 이러한 디바이스는 상대적으로 가격이 비싸고 용량이 충분치 않기 때문에 효율적으로 사용하기 위한 방법이 필요하다. 본 논문에서는 대표적인 침입탐지시스템인 Snort의 규칙집합을 이용하여 고속의 패킷 분류에 적합한 디바이스인 TCAM을 통한 효율적인 패킷 분류방법을 제안하였다. 제안한 방법에서는 값비싼 TCAM의 효율적인 사용을 위하여, TCAM을 분할함으로써 규칙상의 IP 주소와 포트의 중복 필드를 없애고 부정(negation), 범위(range) 규칙을 최소의 엔트리로 표현하도록 한다. 또한 포트번호 조합으로 TCAM 분할을 줄여 용량상의 이점은 유지하고, TCAM 검색횟수를 줄인다. 시뮬레이션을 통해 TCAM 용량을 최대 98%까지 줄이면서 대용량의 규칙을 사용하는 고속 패킷 분류에도 성능저하를 줄일 수 있음을 보인다.

Key Words : Network Security, Intrusion Detection, Packet Classification, Internet

ABSTRACT

As increasing the network bandwidth, the threat of a network also increases with emerging various new services. For a high-performance network security, It is generally used that high-speed packet classification methods which employ hardware like TCAM. There needs an method using these devices efficiently because they are expensive and their capacity is not sufficient. In this paper, we propose an efficient packet classification using a Ternary-CAM(TCAM) which is widely used device for high-speed packet classification in which we have applied Snort rule set for the well-known intrusion detection system. In order to save the size of an expensive TCAM, we have eliminated duplicated IP addresses and port numbers in the rule according to the partitioning of a table in the TCAM, and we have represented negation and range rules with reduced TCAM size. We also keep advantages of low TCAM capacity consumption and reduce the number of TCAM lookups by decreasing the TCAM partitioning using combining port numbers. According to simulation results on our TCAM partitioning, the size of a TCAM can be reduced by upto 98% and the performance does not degrade significantly for high-speed packet classification with a large amount of rules.

Ⅰ. 서 론

네트워크의 발달과 대역폭의 증가에 따라 사용자들에 의해 다양한 서비스에 대한 요구가 증가되고 그로 인해 다양한 서비스가 생성됨에 따라 고속의 패킷 처리와 네트워크 위협들로 사용자를 보호하는

※ 본 연구는 정보통신부의 대학 ITRC 지원사업의 연구비 지원으로 수행되었음.
* 교신저자: 충남대학교 컴퓨터공학과 (tgkwon@cnu.ac.kr)
논문번호 : KICS2005-11-446, 접수일자 : 2005년 11월 2일

것들이 중요한 문제로 대두되고 있다. 인터넷 망이 보급된 이래 대역폭은 6개월에 두 배씩 증가[1]하여 네트워크 백본망에는 OC-192(10Gbps), OC-768 (40Gbps) 등 고속의 링크가 적용되고 있으며, 네트워크 주변부에는 기가비트 이더넷 속도가 보급되고 있다. 그에 따라 네트워크 위협들에 의한 피해도 대역폭의 증가에 맞춰 점점 더 거대한 규모의 피해액을 발생시킨다. 정량적인 피해 외에도 네트워크 상에서 일어나는 기업들의 업무 효율의 저하, 정보 유출, 사용 가능한 대역폭의 감소 같은 정성적 피해도 많이 발생하고 있다[2]. 네트워크 위협이 증가하면서 사용자 및 시스템 등을 보호하기 위해 네트워크를 위협하거나 비정상적인 트래픽을 감시하는 침입탐지 시스템(IDS: Intrusion Detection System)의 중요성이 대두되고 있고, 대역폭의 증가에 적용할 수 있도록 최근의 침입탐지시스템 기술은 고속화에 초점을 맞추어 발전하고 있다. 또한 침입방지시스템(IPS: Intrusion Prevention System), 방화벽, 가상 사설망(VPN: Virtual Private Network), 정책기반 라우터 등 여러 응용이 네트워크 보호를 위해 개발되고 사용되고 있다. 이러한 네트워크에서 교환되는 패킷들의 서로 다른 처리를 요구하는 응용을 구현하기 위해서 패킷을 분류하는 방법이 필요하다.

패킷 분류는 패킷 헤더의 정보 중 프로토콜, IP 주소, 포트번호의 5개 필드를 가지고 5-튜플을 구성하고 패킷을 분류하는 방법이 많이 사용되고 있다. 그리고 대역폭이 증가함에 따라 패킷 분류를 빠른 속도로 하기 위한 알고리즘의 효율이 중요해지고 있다. 현재까지 제안된 패킷 분류방법에는 재귀흐름 분류(Recursive Flow Classification)[3]를 사용하여 패킷이 규칙에 매칭되는 패턴을 간략화 시키는 방법, 규칙 집합을 비트 벡터(Bit Vector)로 표현한 뒤 이를 분할하여 매칭된 규칙을 찾는 방법(Aggregated Bit Vector)[4], TCAM(Ternary Content Addressable Memory)을 사용하여 하드웨어적으로 처리하는 방법[5, 6] 등이 있다. 이러한 방법 중 소프트웨어 기반의 알고리즘으로 10Gbps를 처리하기는 쉽지가 않다. 하드웨어 기반의 TCAM을 사용하여 패킷 분류를 하는 방법은 한번의 검색으로 패킷을 분류할 수 있기 때문에 고속의 패킷 분류에 가장 적합하다. 그러나 상대적으로 고가의 디바이스이고, 밀집도가 약해 많은 공간을 필요로 하기 때문에[1] 큰 용량의 TCAM을 사용하는 데에는 많은 제약이 따라 현존하는 알고리즘으로는 수천 개의 규칙을 처리하기에 부담이 크다. 본 논문에서는 일반 메모리보다 검색 능력이 뛰어난 TCAM의 용량을 절약하면서 고속의 패킷 분류를 할 수 있는 방법을 제시한다.

본 논문의 구성은 다음과 같다. 2장에서는 TCAM의 특징에 대한 소개와 TCAM을 이용한 고성능 패킷 분류 방법에 대해서 기술한다. 3장에서는 TCAM에서 Snort 규칙을 이용한 패킷 분류 방법과 TCAM의 분할을 통해서 TCAM의 용량을 적게 사용하고, 그에 따른 교환 관계인 TCAM 접근 횟수로 인한 성능저하를 줄이는 방법을 제시하고, 4장에서는 본 논문에서 제시한 패킷 분류를 위한 TCAM 분할의 성능을 분석한다. 마지막으로 결론은 5장에서 기술한다.

Ⅱ. TCAM을 이용한 고성능 패킷 분류

TCAM은 0, 1뿐 아니라 'don't care' 상태를 포함하는 3 종류의 논리 상태를 갖고, 이에 매치되는 데이터를 찾는 메모리를 포함하는 프로세서로써 네트워크 프로세서의 패킷 분류 기능을 처리하는데 널리 사용된다[7]. 그림 1은 기본적인 TCAM의 동작을 설명하고 있다. 입력 패턴 1000에 대하여 일치되는 TCAM 엔트리는 1000과 10--, 그리고 마지막 디폴트 엔트리이지만 처음으로 일치되는 엔트리의 관련 데이터(associated data)의 내용이 반환된다. 멀티-매치(multi-match)를 제공하는 TCAM의 경우엔 일치되는 모든 TCAM 엔트리가 차례대로 반환된다.

TCAM은 LPM(Longest Prefix Matching) 등 패킷 포워딩을 위한 IP 주소 검색에 적합하고, 처리 속도 또한 빨라 고속 패킷분류에 사용되지만 상대적으로 고가의 디바이스로, 이를 효율적으로 사용하기 위한 여러 가지의 방법들이 제안되었다[8][9]. 하지만 기존의 연구는 TCAM을 사용한 고속 라우터에서 라우팅 테이블의 크기를 줄이는 방법을 제안하였고, 정책 기반 라우팅 혹은 방화벽, IDS 등을 위

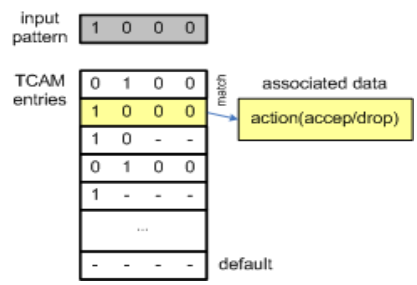


그림 1. TCAM 동작 예

한 L4 패킷 분류에 대한 연구[10]에 치중되었다. 본 논문에서는 이전에 수행되었던 연구와 다른 방향의 TCAM을 이용한 고성능 패킷 분류에서 TCAM 용량 최적화에 대해 논하였다.

Ⅲ. 패킷 분류 정책을 위한 TCAM 분할

침입탐지시스템, 방화벽, 가상 사설망 등과 같이 통신하는 패킷들 사이에서 서로 다른 처리를 하기 위해서는 패킷 분류가 필요하다. 또한, 패킷 분류의 결과를 입력으로 하고, 패킷의 페이로드 부분을 패턴 매칭[11]하여 IPS를 구성하는 등 패킷 분류는 여러 응용으로 사용될 수 있다.

3.1 패킷 분류를 위한 TCAM 구성

표 1은 공개 침입탐지시스템으로 대표적으로 사용되는 Snort의 규칙을 나타낸 것이다. Snort는 헤더와 옵션의 두 부분으로 나뉘어져 있다. Snort 규칙 헤더에는 패킷 분류를 위한 5-튜플 정보를 나타내고 있으며, Snort 규칙 옵션은 규칙에 하나 또는 여러 개의 옵션이 올 수 있어 보다 상세한 패킷 분류와 패턴 매칭을 할 수 있게 한다[12]. 본 논문에서는 5-튜플 정보만을 이용하여 패킷 분류를 하기 때문에 Snort 규칙의 헤더만을 이용한다.

표 1에서 sid 2523인 규칙을 R1이라 할 때, R1에 해당하는 플로우는 근원 IP가 $EXTERNAL_NET이고 목적 IP가 $HOME_NET인 트래픽과 방향이 반대인 외부 망으로의 트래픽을 모두 포함하고 있다. 그리고 sid 3089인 규칙을 R2라 할 때, R2에 해당하는 플로우는 외부 망에서 내부 망으로의 트래픽 만을 포함하고 있다.

표 1. Snort 규칙 예

```
alert tcp $EXTERNAL_NET any <> $HOME_NET
179 (msg:"DOS BGP spoofed connection reset at-
tempt"; flow:established; flags:RSF*; threshold:type
both,track by_dst,count 10,seconds 10; reference:bug-
traq,10183; refer-
ence:url,www.uniras.gov.uk/vuls/2004/236929/in-
dex.htm; classtype:attempted-dos; sid:2523; rev:7)

alert udp $EXTERNAL_NET any -> $HOME_NET
2048(msg:"DOS squid WCCP I_SEE_YOU message
overflow attempt"; content:"|00 00 00 08|"; depth:4;
byte_test:4,>,32,16; reference:cve,CAN-2005-0095; ref-
erence:bugtraq,12275; classtype:attempted-user;
sid:3089; rev:1;)
```

표 1의 Snort 규칙은 그림 2에서와 같이 3개의 엔트리를 가진 TCAM과 관련 데이터(associated data)로 표현할 수 있다. 그림 2에서는 5-튜플의 정보가 모두 하나의 엔트리에 포함되어 있어 TCAM 테이블 크기가 104 비트인 DB-104를 사용하게 된다. 이는 엔트리 하나가 추가될 때 TCAM 용량이 104 비트 만큼 늘어나 한정된 TCAM의 용량을 사용하는데 부담이 된다. 또한 5-튜플이 하나라도 같지 않으면 IP, 포트, 프로토콜의 정보가 중복되는 내용이 있어도 중복되는 내용을 무시하고 엔트리를 추가하여 저장 공간을 낭비하게 되는 단점이 있다.

규칙 저장 공간의 용량이 중요한 TCAM은 그림 2와 같은 방법으로 사용하기에는 많은 부담이 따른다. 이것을 IP, 포트, 프로토콜을 TCAM 필드 용량에 따라 DB-32, DB-16, DB-8로 분할하면 5-튜플의 중복을 줄일 수가 있다. TCAM을 분할하는 경우에는 TCAM 용량은 크게 줄어드는 반면, 다수의 TCAM 데이터베이스를 여러 번 접근해야 하는 부담이 있다.

그림 2의 5개 필드를 가진 TCAM 데이터베이스에서 근원 IP 주소와 목적 IP 주소는 32 비트로 구성되고, 근원 및 목적 포트번호는 16 비트로 구성되며, 마지막 프로토콜 필드는 8 비트로 구성된다. 이를 각각 32 비트, 16 비트, 그리고 8 비트인 3개의 TCAM 데이터베이스, 즉 DB-32, DB-16, 그리고 DB-8로 분할하여 표현할 경우에 중복되는 필드의 값을 제거하여 표현할 수 있다. 분할된 필드의 관련 데이터는 전체 규칙 중 일부를 표현하는 서브규칙(subrule)이므로 이 서브규칙으로부터 원래의 규칙을 찾을 수 있도록 추가 정보, 즉 RMT(Rule Matching Table)가 필요하다. 이 테이블 역시 TCAM으로 구성하여 한 번의 검색으로 관련 데이터의 내용을 반환받을 수 있도록 한다. 그림 3은 TCAM 분할 방법을 사용한 플로우 표현을 보여준다. (a)는 필드 용량에 따라 TCAM을 분할하여 엔트리를 구성한 모습을 보이고 매치되는 엔트리가 있으면 관련 데이터로 서브규칙의 값이 반환되어 그림 (b)의 RMT와 다시 검색을 하여 매칭되는 규칙이 있는지 확인

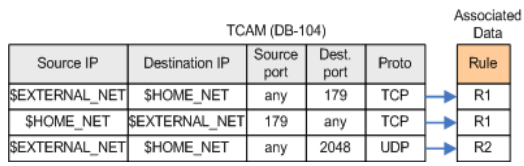
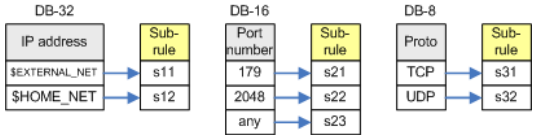
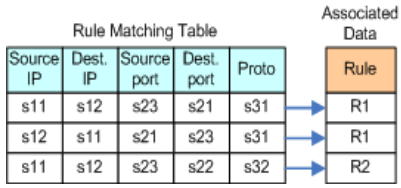


그림 2. Snort 규칙에 따른 TCAM 및 관련 데이터 표현



(a) 3개의 TCAM 분할



(b) RMT 구성

그림 3. Snort 규칙에 따른 TCAM 분할

한다. 예를 들어 표 1의 Snort 규칙을 검색하기 위한 패킷의 5-튜플이 [10.0.0.1, 192.168.1.100, 1200, 2048, UDP]이라고 하면, 근원 IP와 목적 IP 검색을 위하여 10.0.0.1, 192.168.1.100을 키 값으로 DB-32를 검색하여 s11, s12를 구하고, 근원 포트번호와 목적 포트번호를 위하여 1200, 2048을 키 값으로 DB-16을 검색하여 s23, s22를 구한다. 마지막으로 s32를 구하여 이들의 연접(concatenation), s11 | s12 | s23 | s22 | s32의 값으로 RMT를 통해 최종 규칙 R2를 검색한다.

분할된 TCAM 테이블에서 IP 주소에 해당하는 DB-32는 $HOME_NET의 설정에 따라 달라진다. 그리고 $EXTERNAL_NET이 $HOME_NET을 제외한 모든 IP를 나타낸다고 하면, 이러한 부정(negation)을 처리하는 방법에 따라 달라진다. [13]에 의한 'bit-by-bit flip'의 방법을 사용할 경우에 $HOME_NET을 16과 32 비트 네트워크 주소를 사용할 경우에 DB-32의 엔트리 수는 각각 42개와 58개가 된다. 단, Snort 규칙의 다른 서브 주소는 각각 한 개의 IP 주소를 갖는 것으로 가정한다.

TCAM 분할은 중복되는 필드를 없애 TCAM 용량을 줄일 수 있다. 그러나 IP, 포트, 프로토콜을 따로 검색을 해야 하는 부담이 있고, 5-튜플 검색의 결과로 나오는 서브규칙의 연접을 다시 RMT와 매칭 시켜야 하는 단점이 있다. 이점을 개선하기 위해 TCAM 검색 횟수를 줄이는 방법이 필요하게 된다.

3.2 포트번호 조합을 통한 TCAM 분할

TCAM을 분할할 경우에 필요한 TCAM의 크기는 줄어들지만 TCAM의 분할된 테이블을 액세스하는 횟수는 늘어난다. 고성능 패킷 분류를 위하여 TCAM 테이블의 액세스 횟수를 줄이는 것 또한 중요하다.

IP 주소 검색을 위하여 32 비트의 TCAM 테이블을 사용하고, 포트번호 검색을 위하여 16 비트의 TCAM 테이블을 사용한다. 하지만 TCAM에서는 임의 크기의 테이블을 제공하지 않고 일반적으로 미리 정해진 특정 크기의 테이블 만을 제공한다. 본 논문에서 사용한 TCAM인 IDT 75K62134는 최소 36 비트 크기의 테이블을 제공하므로 16 비트의 포트번호를 위한 테이블을 위하여 36 비트 크기의 테이블을 사용하여야 하므로 TCAM 용량의 낭비 요인이 존재한다. 또한 Snort에서는 현재 4개의 프로토콜 TCP, UDP, ICMP, IP만 지원하기 때문에 패킷 분류를 위한 프로토콜 필드를 별도의 TCAM 테이블을 유지하는 것보다 프로토콜 필드만 따로 처리하는 것이 유리하다.

본 논문에서는 이를 위하여 근원-목적 포트번호를 조합한 하나의 테이블을 유지할 경우에 IP 주소와 같은 32 비트 크기의 TCAM 테이블을 이용할 수 있다. Snort 규칙에서 별도의 포트번호를 사용할 경우에 241개의 TCAM 엔트리가 사용되는 것에 비하여 근원-목적 포트를 조합할 경우에 358개의 TCAM 엔트리가 필요하다. 따라서 약 50%의 TCAM 용량이 더 필요하다. 하지만 이와 같이 포트번호를 조합할 경우에 근원-목적 포트번호를 두 번 액세스 하는 것에 비하여 TCAM 액세스는 한 번으로 줄어든다.

이와 같이 포트번호의 조합이 가능한 이유는 Snort 규칙에서 이상 트래픽이 대부분 웹 응용 등의 특정 응용에 치우쳐 분포되어 있기 때문인데, 현재의 Snort 규칙을 분석하면 표 2와 같이 상위 10

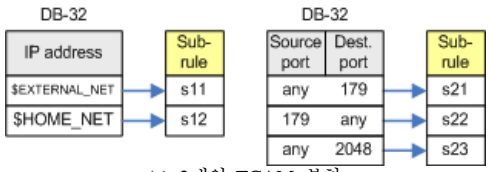
표 2. 상위 15개 포트 번호 조합의 분포(전체 2758개 규칙 중)

순위	SrcPort	DstPort	빈도	누적률(%)
1	Any	80	1016	36.8
2	Any	Any	263	46.4
3	110	Any	148	51.7
4	Any	21	92	55.1
5	Any	111	69	57.6
6	Any	25	68	60.0
7	60000	2140	53	62.0
8	Any	139	49	63.7
9	80	Any	48	65.5
10	Any	1521	26	66.4
11	Any	143	24	67.3
12	Any	110	22	68.1
13	2000	Any	22	68.9
14	Any	23	21	69.7
15	Any	445	19	70.3

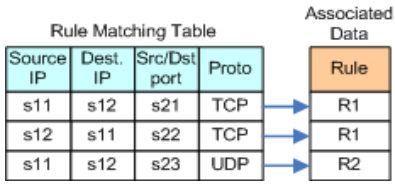
개와 15개의 포트번호 조합이 전체 2758개의 규칙의 각각 66%와 70% 이상을 차지하고 있음을 알 수 있는데, 이로 인하여 포트번호를 조합할 때 엔트리 수가 크게 증가하지 않는다. 예를 들어 근원 포트번호가 Any이고 목적 포트번호가 80인 하나의 포트번호 조합은 2758개의 전체 규칙 중에 1016개의 빈도로 36.8%의 많은 비중을 차지하고 있다.

포트번호의 조합과 프로토콜 필드의 별도 처리를 하는 경우, TCAM 분할을 통해 3번의 TCAM 액세스로 5-튜플의 패킷 분류 과정을 처리할 수 있다. 그림 4는 그림 3의 TCAM 분할을 포트번호 조합과 별도의 프로토콜 필드 처리를 통하여 TCAM 액세스 횟수를 줄인 확장된 TCAM 분할을 보이고 있다.

그림 5는 3번의 TCAM 액세스로 패킷을 분류하고 RMT를 검색하여 Snort 규칙에 해당되는지 비교하는 과정을 보이고 있다. 그림 5의 (a)에서 Snort 규칙을 검색하기 위한 패킷의 5-튜플이 [$HOME_NET, $EXTERNAL_NET, 179, 80, TCP]이라고 할 때, 서브규칙들의 연접 [s12, s11, s22, TCP]가 R1에 매치되는 모습을 보이고 있고, Snort 규칙을 검색하기 위한 패킷의 5-튜플이 [$EXTERNAL_NET, $HOME_NET, 2048, 22, TCP]이라고 할 때에는 포트번호에서 서브규칙을 찾을 수 없으므로 그림 5의 (b)와 같이 매치 실패가 나타나게 된다.

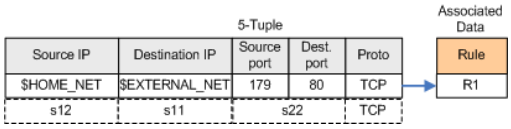


(a) 2개의 TCAM 분할

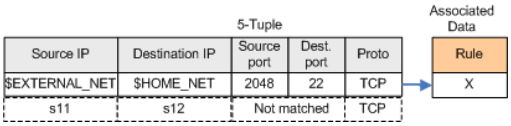


(b) RMT 구성 및 관련 데이터 표현

그림 4. 그림 3의 TCAM 분할 방법의 확장



(a) 매치 성공



(b) 매치 실패

그림 5. 포트번호 조합에서의 검색

Ⅳ. 성능 평가

TCAM 액세스 횟수와 TCAM 용량은 성능을 평가하는 데 중요하다. 패킷 분류를 할 때 적은 TCAM 액세스를 보일수록 속도는 빨라지고, TCAM의 용량을 적게 사용할수록 많은 규칙집합을 적용시킬 수 있고 다른 응용에서도 TCAM을 공유해서 사용할 수 있기 때문이다.

RMT의 각 필드 길이는 필드 엔트리의 수 N에 의해 결정되는데 그 값은 ⌈log(N)⌉으로 실제 Snort 규칙 2394개에 따른 일반적인 TCAM 사용방법, TCAM 분할 방법, 포트번호 조합방법의 3가지 방법에 대한 필드 엔트리 수와 필드 길이는 표 3과 같다. TCAM 용량 향상률은 분할 전 TCAM 용량을 1로 할 때 절약되는 TCAM의 용량을 %로 나타낸 것이다. 예를 들어 분할 후 TCAM 용량 향상률은 분할 전 126.58KB의 용량이 분할 후 2.54KB로 줄어 필요한 용량이 분할 전 용량의 2.01%에 불과하여 TCAM 용량 향상률이 97.99% 이다.

본 논문에서 제시한 방법의 성능 평가를 위해 학교망의 트래픽을 대상으로 하여 실험하였다. 표 3과 그림 6은 2004년 7월 9일에 수집한 트래픽 정보 중 10만개 패킷을 가지고 리눅스 환경에서 pcap 라이브러리를 사용하여 시뮬레이션 하였다. 시뮬레이션에서는 TCP, UDP 패킷만 유효한 데이터로 사용

표 3. Snort 규칙에 대한 분할 전후의 TCAM 용량(프리픽스 길이 16일 경우)

구분	분할 전	분할 후				포트번호 조합 후	
		DB-32	DB-16	DB-8	RMT	DB-32	RMT
엔트리 수	7032	33	241	4	287	358	289
TCAM 폭(bits)	144	36	36	36	36	36	36
TCAM 용량(KB)	126.58	0.15	1.08	0.02	1.29	1.61	1.3
		2.54				3.06	
평균 TCAM 액세스 횟수	1	7.38				5.53	
TCAM 용량 향상률(%)	-	97.99				97.58	

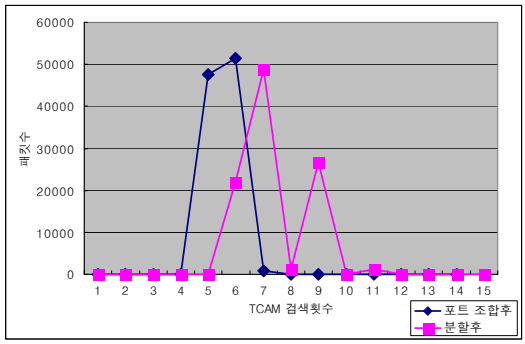


그림 6. 10만 개 패킷의 TCAM 검색횟수 빈도

하였으며, ICMP, ARP, IPX 등의 패킷은 시뮬레이션에서 유효하지 않은 데이터로 간주하였다. 10만개 패킷 중 TCP 패킷은 97822개, UDP 패킷은 2082개 이었으며 유효하지 않은 패킷 수는 96개 이었다. 시뮬레이션의 결과로 서브규칙들의 연접으로 인한 TCAM의 검색횟수는 다음 그림 6과 같다.

포트번호 조합 후의 TCAM 검색횟수는 대부분이 5~6번으로 분할 후의 TCAM 검색횟수에 비해 규칙적인 검색횟수를 나타내고 있다. 포트번호 조합 후의 평균 TCAM 검색횟수는 약 5.53번이고, 분할 후의 평균 TCAM 검색횟수는 약 7.38번으로 포트번호 조합 후의 TCAM 평균 검색횟수가 분할 후의 방법보다 검색횟수가 약 1.85번 감소되는 것으로 나타났다.

V. 결 론

본 논문에서는 네트워크 환경 발달과 대역폭의 증가에 따라 요구되는 다양한 서비스를 제공하고, 네트워크 위협에 대응하기 위해 고가의 디바이스인 TCAM을 사용하여 고속의 패킷 분류를 하는 방법에 대해 논하였다. 상대적으로 용량에 제한적인 TCAM을 활용하여 많은 규칙 집합의 패킷 분류를 하기 위해 우리는 TCAM의 분할에 의해 TCAM 용량을 절약함으로써 해결하였다. 실험을 통해 본 논문에서 제시한 TCAM 분할 방법은 RMT 등의 추가적인 데이터 검색이 필요하여 TCAM을 분할하기 전보다 검색은 약 7배 정도 증가하지만 추가된 RMT 데이터 공간을 포함하여도 전체 TCAM 용량은 약 97~98% 정도 줄임으로써 많은 규칙 집합의 분류를 가능하게 하였고, 검색 횟수의 증가는 포트 조합을 통해 약 2번 정도의 검색횟수 감소를 이끌어 내면서 TCAM 용량은 크게 늘어나지 않는 방법을 논하였다. 본 논문을 통해 앞으로 IPv4에 비해 월등히 많은 주소를 가지는 IPv6 주소 체계에서도 TCAM을 이용하여 패킷 분류를 할 수 있는 가능성을 제시하였다. 또한 본 논문에서 제시한 방법은 분류된 패킷 정보와 절약된 TCAM 용량을 침입탐지 시스템, 패킷 필터링 등 다른 응용프로그램에서 이용할 수 있다는 장점을 가지고 있다.

향후 연구과제로는 본 논문에서 제안된 고성능 패킷 분류 방법을 기반으로 네트워크 프로세서와 같은 장비에 직접 구현하여 침입탐지시스템에 적용하는 작업이 필요하다.

참 고 문 헌

〔1〕 F. Baboescu, S. Singh, G. Varghese, "Packet Classification for Core Routers: Is there an alternative to CAMs?," *IEEE Infocomm* 2003.

〔2〕 P. Jungck, S. S. Y.Shim, "Issue in High-Speed Internet Security," *IEEE Computer*, May. 2004.

〔3〕 P. Gupta, N. McKeown, "Packet Classification on Multiple Fields," *ACM Sigcomm*, Sept. 1999.

〔4〕 F. Baboescu, G. Varghese, "Scalable Packet Classification," *ACM Sigcomm*, 2001.

〔5〕 E. Spitznagel, D. Taylor, and J. Turner, "Packet Classification Using Extended TCAMs," *ICNP*, Nov. 2003.

〔6〕 Seok-Min Kang, Yoshiaki Kasahara, Taeck-Geun Kwon, "Packet Classification using Dual TCAM Tables," *Proceedings of ITC-CSCC*, 4, pp.1431-1432, Jun. 2005.

〔7〕 Z. J. Wang, H. Che, M. Kumar, and S. Das, "CoPTUA: Consistent Policy Table Update Algorithm for TCAM without Table Lock," *IEEE Transactions on Computers*, 53(12), pp. 1602-1628, Dec. 2004.

〔8〕 H. Liu, "Routing Table Compaction in Ternary CAM," *IEEE Micro*, 22(1), pp. 58-64, Jan-Feb. 2002.

〔9〕 V.C. Ravikumar, R. N. Mahapatra, "TCAM Architecture for IP Lookup Using Prefix Properties," *IEEE Micro*, 24(2), pp. 60-69, Mar-Apr. 2004.

〔10〕 T. V. Lakshman and D. Stiliadis, "High-

Speed Policy-Based Packet Forwarding Using Efficient Multi-Dimensional Range Matching," *ACM Sigcomm*, pp. 203-214, 1998.

〔11〕 Jung-Sik Sung, Seok-Min Kang, Youngseok Lee, Taeck-Geun Kwon, and Bong-Tae Kim, "A Multi-gigabit Rate Deep Packet Inspection Algorithm using TCAM," *Globecom*, Nov. 2005.

〔12〕 SNORT network intrusion detection system, www.snort.org.

〔13〕 Fang Yu, Randy H. Katz and T.V. Lakshman, "Efficient Multi-Match Packet Classification with TCAM," *IEEE Micro*, Feb. 2005.

〔14〕 D. Shah and P. Gupta, "Fast Incremental Updates on Ternary-CAMs for Routing Lookups and Packet Classification," *Proceedings of Hot Interconnects*, 2000. http://citeseer.csail.mit.edu/shah00fast.html

〔15〕 IDT, Network Search Engine(NSE) with QDRTM Interface, http://www1.idt.com/pcms/tempDocs/75K6213452134_DS_80635.pdf

김 규 호 (Kyu-Ho Kim) 준회원

2005년 2월 충남대학교 정보통신공학부 컴퓨터 학사

2005년 3월~현재 충남대학교 컴퓨터공학과 석사과정

<관심분야> 통신시스템, 인터넷 보안 등

강 석 민 (Seok-Min Kang) 준회원

1999년 2월 충남대학교 컴퓨터공학과 학사

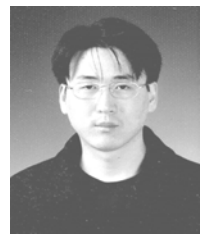
2002년 2월 충남대학교 컴퓨터공학과 석사

2002년~2003년 한국전자통신연구원

2004년 3월~현재 충남대학교 컴퓨터공학과 박사과정

<관심분야> 초고속 인터넷, 통신 시스템 등

송 일 섭 (Il-Seop Song) 준회원



2003년 2월 충남대학교 컴퓨터공학과 학사

2005년 2월 충남대학교 컴퓨터공학과 석사

2005년 3월~현재 충남대학교 컴퓨터공학과 박사과정

<관심분야> 컴퓨터 통신 및 보안, 네트워크 프로세서 등

권 택 근 (Taeck-Geun Kwon) 정회원

1988년 2월 서울대학교 컴퓨터공학과 졸업

1990년 2월 서울대학교 컴퓨터공학과 석사

1996년 2월 서울대학교 대학원 컴퓨터공학과 박사

1992년~1998년 LG전자 정보통신연구소 연구원

1998년~현재 충남대학교 전기정보통신공학부 컴퓨터전공 부교수

<관심분야> 네트워크 프로세서, 초고속 인터넷, 통신 시스템, 인터넷 보안 등