(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년11월03일
(11) 등록번호 10-1455288
(24) 등록일자 2014년10월21일

(51) 국제특허분류(Int. Cl.)
*H04L 12/26* (2006.01) *H04L 12/24* (2006.01)
(21) 출원번호 10-2013-0052100
(22) 출원일자 2013년05월08일
심사청구일자 2013년05월08일
(56) 선행기술조사문헌
KR1020050100143 A
KR1020120010535 A

(73) 특허권자
**고려대학교 산학협력단**
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
**김명섭**
충청남도 연기군 조치원읍 신흥리 푸르지오아파트 111-902
**박태영**
부산 금정구 서곡로 46-47, 2동 101호 (서동, 대영하이츠빌라)
**윤성호**
경상북도 포항시 북구 우현동 금호어울림 103동 702호
(74) 대리인
**특허법인엠에이피에스**

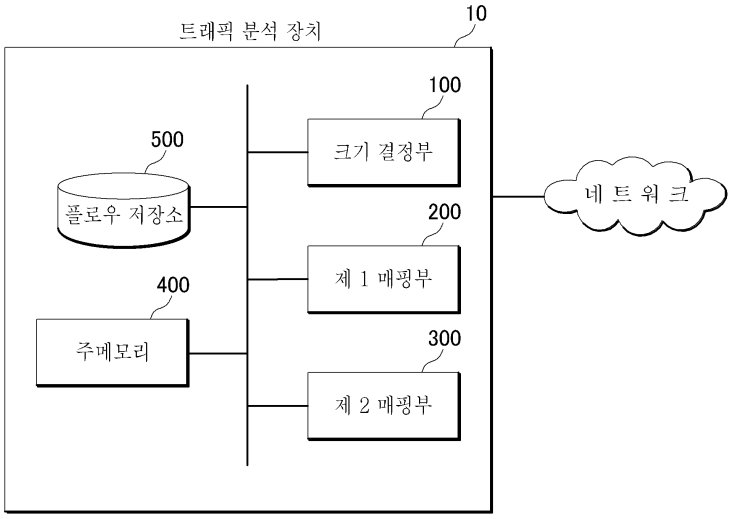
전체 청구항 수 : 총 9 항 심사관 : 석상문

(54) 발명의 명칭 **트래픽 분석 장치 및 방법**

## (57) 요 약

본 발명은 네트워크 트래픽 분석 장치에 있어서, 크기가 가변적인 제 1 해시 테이블, 상기 제 1 해시 테이블의 테이블 인덱스를 통해 참조되는 다수 개의 크기가 고정적인 제 2 해시 테이블, 및 상기 제 2 해시 테이블의 버킷 인덱스를 통해 참조되는 연결 리스트 저장 공간을 유지하는 주메모리; 수집된 네트워크 트래픽 플로우의 개수에 기초하여, 상기 제 1 해시 테이블의 크기를 결정하는 크기 결정부; 각 네트워크 트래픽 플로우에 대해, 상기 테이블 인덱스를 결정하는 제 1 매핑부; 상기 각 네트워크 트래픽 플로우에 대해, 상기 버킷 인덱스를 결정하는 제 2 매핑부;를 포함하되, 상기 각 네트워크 트래픽 플로우는 상기 산출된 테이블 인덱스 및 버킷 인덱스로 결정되는 연결 리스트 저장 공간에 연결 리스트로 연결되어 저장되며, 상기 제 1 해시 테이블의 크기는 상기 제 2 해시 테이블의 개수와 일치하고, 상기 제 2 해시 테이블의 크기는 서로 동일하게 설정되는 장치를 제공한다.

*대 표 도* - 도1



## *특허청구의 범위*

**청구항 1**

네트워크 트래픽 분석 장치에 있어서,

크기가 가변적인 제 1 해시 테이블, 상기 제 1 해시 테이블의 테이블 인덱스를 통해 참조되는 다수 개의 제 2 해시 테이블, 및 상기 제 2 해시 테이블의 버킷 인덱스를 통해 참조되는 연결 리스트 저장 공간을 유지하는 주 메모리;

수집된 네트워크 트래픽 플로우의 개수에 기초하여, 상기 제 1 해시 테이블의 크기를 결정하는 크기 결정부;

각 네트워크 트래픽 플로우에 대해, 상기 테이블 인덱스를 결정하는 제 1 매핑부;

상기 각 네트워크 트래픽 플로우에 대해, 상기 버킷 인덱스를 결정하는 제 2 매핑부;를 포함하되,

상기 각 네트워크 트래픽 플로우는 상기 결정된 테이블 인덱스 및 버킷 인덱스로 결정되는 연결 리스트 저장 공간에 연결 리스트로 연결되어 저장되며,

상기 제 1 해시 테이블의 크기는 상기 제 2 해시 테이블의 개수를 나타내고,

상기 제 2 해시 테이블의 크기는 서로 동일하게 설정되며 고정적인 트래픽 분석 장치.

**청구항 2**

제 1 항에 있어서,

상기 크기 결정부는 네트워크 트래픽 플로우의 개수가 변하면 상기 제 1 해시 테이블의 크기를 다시 결정하는 트래픽 분석 장치.

**청구항 3**

제 1 항에 있어서,

상기 크기 결정부는 상기 제 1 해시 테이블의 사용률이 80% 이하가 되는 최대값을 상기 제 1 해시 테이블의 크기로 결정하는 트래픽 분석 장치.

**청구항 4**

제 1 항에 있어서,

상기 제 1 매핑부는 상기 각 네트워크 트래픽 플로우가 포함하는 소스 주소, 목적지 주소, 소스 포트, 목적지 포트, 및 프로토콜의 마지막 1 바이트를 모두 합산하고, 상기 합산한 값을 상기 제 1 해시 테이블의 크기와 AND 연산하여 상기 테이블 인덱스를 산출하는 트래픽 분석 장치.

**청구항 5**

제 1 항에 있어서,

상기 제 2 매핑부는 상기 각 네트워크 트래픽 플로우가 포함하는 소스 주소, 목적지 주소, 소스 포트, 및 목적지 포트의 처음 2 바이트 및 마지막 2 바이트를 모두 합산하고, 상기 합산한 값을 상기 제 2 해시 테이블의 크기와 AND 연산하여 상기 버킷 인덱스를 산출하는 트래픽 분석 장치.

**청구항 6**

네트워크 트래픽 분석 장치를 사용한 네트워크 트래픽 분석 방법에 있어서,

수집된 네트워크 트래픽 플로우의 개수에 기초하여 제 1 해시 테이블의 크기를 결정하고, 상기 제 1 해시 테이블의 크기와 일치하도록 제 2 해시 테이블의 개수를 조절하는 단계;

각 네트워크 트래픽 플로우에 대해, 상기 제 1 해시 테이블의 테이블 인덱스를 결정하는 단계;

상기 각 네트워크 트래픽 플로우에 대해, 상기 제 1 해시 테이블의 테이블 인덱스를 통해 참조되는 제 2 해시 테이블의 버킷 인덱스를 결정하는 단계; 및

상기 각 네트워크 트래픽 플로우를 상기 결정된 테이블 인덱스 및 버킷 인덱스로 결정되는 연결 리스트 저장 공간에 연결 리스트로 연결하여 저장하는 단계;를 포함하되,

상기 제 2 해시 테이블의 크기는 서로 동일하게 설정되며 고정적인 트래픽 분석 방법.

**청구항 7**

제 6 항에 있어서,

상기 제 1 해시 테이블의 크기는 상기 네트워크 트래픽 플로우의 개수의 변화에 따라 상기 제 1 해시 테이블의 사용률이 80% 이하가 되도록 조절되는 트래픽 분석 방법.

**청구항 8**

제 6 항에 있어서,

상기 테이블 인덱스를 결정하는 단계는

상기 각 네트워크 트래픽 플로우가 포함하는 소스 주소, 목적지 주소, 소스 포트, 목적지 포트, 및 프로토콜의 마지막 1 바이트를 모두 합산하고, 상기 합산한 값을 상기 제 1 해시 테이블의 크기와 AND 연산하여 상기 테이블 인덱스를 산출하는 트래픽 분석 방법.

**청구항 9**

제 6 항에 있어서,

상기 버킷 인덱스를 결정하는 단계는

상기 각 네트워크 트래픽 플로우가 포함하는 소스 주소, 목적지 주소, 소스 포트, 및 목적지 포트의 처음 2 바이트 및 마지막 2 바이트를 모두 합산하고, 상기 합산한 값을 상기 제 2 해시 테이블의 크기와 AND 연산하여 상기 버킷 인덱스를 산출하는 트래픽 분석 방법.

# *명 세 서*

## *기 술 분 야*

[0001] 본 발명은 네트워크 트래픽 분석 장치 및 방법에 관한 것으로, 수집한 네트워크 트래픽을 듀얼 해시 테이블 구조를 사용하여 주메모리에 적재하는 장치 및 방법에 관한 것이다.

## *배 경 기 술*

[0002] 최근 인터넷의 급격한 발전으로 인해 효율적인 네트워크 관리를 위한 네트워크 트래픽 데이터 분석의 중요성이 강조되고 있다. 인터넷 사용자의 증가와 고속 네트워크의 보급으로 인해 네트워크 트래픽이 급증하고 있으며, 트래픽의 종류 또한 다양해지고 있다. 따라서 급증하는 인터넷 트래픽을 효과적으로 관리하기 위해 다양한 트래픽 분석 방법이 개발되고 있다.

[0003] 트래픽을 분석하기 위해서는 트래픽 데이터를 주메모리에 적재하고 조건에 맞는 데이터를 검색하여 트래픽의 원천(예: 응용 프로그램, 프로토콜 등)에 따라 분류하는 과정을 반복하는 것이 필요하다. 따라서, 트래픽 양이 증가하면 데이터 적재 및 검색 시간이 증가하므로, 대용량 트래픽의 실시간 분석을 위해서는 효과적인 데이터 적재 및 검색 방법이 필요하다.

[0004] 일반적으로 이를 위해 해시 테이블(hash table)이 사용된다. 그러나 도 2을 통해 후술하는 바와 같이, 이러한 종래의 고정 크기 싱글 해시 테이블 구조는 증가하는 트래픽 데이터 양에 따른 해시 성능 저하 문제를 안고 있다.

[0005] 예를 들어, 고정된 크기의 해시 테이블을 사용하기 때문에 대용량 트래픽 발생시 해시 테이블의 동일한 버킷(bucket)으로 매핑(mapping)되는 충돌(collision) 현상이 급격히 증가함에 따른 오버헤드가 발생할 수 있다. 연결 리스트(linked list)가 길어짐에 따라 메모리 접근 횟수가 늘어나기 때문이다. 이러한 데이터 적재 및 검색

처리 시간 증가는 실시간으로 대용량 트래픽을 분석하는 데 장애물이 될 수 있다

[0006] 또한 트래픽 증가에 따라 해시 테이블을 확장하려고 할 때 해시 함수를 재정의 해야 하는 문제도 가지고 있다.

[0007] 그러므로 대용량 트래픽을 효과적으로 분석하기 위해서는 트래픽 분석에 사용하는 해시 테이블의 구조를 개선할 필요가 있다.

[0008] 이와 관련하여 미국특허 제7779143호("Scalable methods for detecting significant traffic patterns in a data network")에는 트래픽 양이 많거나 트래픽 양이 급격히 증가하는 트래픽 패턴을 탐지하는 구성이 개시되어 있다.

[0009] 또한, 한국특허 제10-2004-0054110호("튜플 공간 검색 기반의 IP 패킷 분류장치 및 분류방법")에는 2차원 해시 테이블을 사용하여 IP 패킷 트래픽을 분류하는 구성이 개시되어 있다.

## 발명의 내용

### 해결하려는 과제

[0010] 본 발명은 전술한 문제를 해결하기 위한 것으로서, 그 목적은 메모리 공간 및 메모리 접근 면에서 효율적인 트래픽 분석 장치 및 방법을 제공하는 것이다.

### 과제의 해결 수단

[0011] 상기와 같은 목적을 달성하기 위한 본 발명의 제 1 측면에 따른 장치는 네트워크 트래픽 분석 장치에 있어서, 크기가 가변적인 제 1 해시 테이블, 상기 제 1 해시 테이블의 테이블 인덱스를 통해 참조되는 다수 개의 크기가 고정적인 제 2 해시 테이블, 및 상기 제 2 해시 테이블의 버킷 인덱스를 통해 참조되는 연결 리스트 저장 공간을 유지하는 주메모리; 수집된 네트워크 트래픽 플로우의 개수에 기초하여, 상기 제 1 해시 테이블의 크기를 결정하는 크기 결정부; 각 네트워크 트래픽 플로우에 대해, 상기 테이블 인덱스를 결정하는 제 1 매핑부; 상기 각 네트워크 트래픽 플로우에 대해, 상기 버킷 인덱스를 결정하는 제 2 매핑부;를 포함하되, 상기 각 네트워크 트래픽 플로우는 상기 산출된 테이블 인덱스 및 버킷 인덱스로 결정되는 연결 리스트 저장 공간에 연결 리스트로 연결되어 저장되며, 상기 제 1 해시 테이블의 크기는 상기 제 2 해시 테이블의 개수와 일치하고, 상기 제 2 해시 테이블의 크기는 서로 동일하게 설정되는 것을 특징으로 한다.

[0012] 상기와 같은 목적을 달성하기 위한 본 발명의 제 2 측면에 따른 방법은 네트워크 트래픽 분석 장치를 사용한 네트워크 트래픽 분석 방법에 있어서, 수집된 네트워크 트래픽 플로우의 개수에 기초하여 제 1 해시 테이블의 크기를 결정하고, 상기 제 1 해시 테이블의 크기와 일치하도록 제 2 해시 테이블의 개수를 조절하는 단계; 각 네트워크 트래픽 플로우에 대해, 상기 제 1 해시 테이블의 테이블 인덱스를 결정하는 단계; 상기 각 네트워크 트래픽 플로우에 대해, 상기 제 1 해시 테이블의 테이블 인덱스를 통해 참조되는 제 2 해시 테이블의 버킷 인덱스를 결정하는 단계; 및 상기 각 네트워크 트래픽 플로우를 상기 산출된 테이블 인덱스 및 버킷 인덱스로 결정되는 연결 리스트 저장 공간에 연결 리스트로 연결하여 저장하는 단계;를 포함하되, 상기 제 2 해시 테이블의 크기는 서로 동일하게 설정되어 변하지 않는 것을 특징으로 한다.

### 발명의 효과

[0013] 본 발명은 트래픽 분석 장치 및 방법에 있어, 메모리 공간 및 메모리 접근이 효율적이라는 효과를 얻는다.

[0014] 고정된 싱글 해시 구조는 트래픽 양이 증감함에 따라, 충돌로 인해 연결 리스트가 길어져 해시 성능 저하가 발생하나, 본 발명은 데이터를 메모리에 로드하기 전 트래픽 양에 따라 제 1 해시 테이블의 크기를 조절함으로써, 적절한 크기의 해시 테이블을 구축하고 두 번의 해시 함수를 거친다. 따라서 충돌을 줄일 수 있으며, 따라서 연결 리스트 길이가 줄어든다.

[0015] 또한 본 발명은 제 1 해시 테이블의 크기와 제 2 해시 테이블의 개수가 일치하므로, 제 2 해시 함수를 중복 사용하기 때문에, 해시 함수를 재정의 할 필요가 없다.

## 도면의 간단한 설명

[0016] 도 1은 본 발명의 일실시예에 따른 트래픽 분석 장치의 구조를 도시함.

도 2은 종래의 싱글 해시 테이블 구조를 도시함.

도 3는 본 발명의 일실시예에 따른 듀얼 해시 테이블 구조를 도시함.

도 4는 본 발명의 일실시예에 따른 트래픽 분석 방법의 흐름을 도시함.

도 5는 본 발명의 일실시예에 따른 제 1 해시 테이블의 크기를 산출하는 방법을 도시함.

도 6 및 도 7은 본 발명의 일실시예에 따른 제 1 해시 테이블의 인덱스를 산출하는 방법을 도시함.

도 8 및 도 9는 본 발명의 일실시예에 따른 제 2 해시 테이블의 인덱스를 산출하는 방법을 도시함.

도 10 내지 도 12는 본 발명의 효과를 검증하기 위한 실험을 도시함.

## *발명을 실시하기 위한 구체적인 내용*

[0017] 아래에서는 첨부한 도면을 참조하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본 발명의 실시예를 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.

[0018] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다.

[0019] 도 1은 본 발명의 일실시예에 따른 트래픽 분석 장치의 구조를 도시하고 있다.

[0020] 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 크기 결정부(100), 제 1 매핑부(200), 제 2 매핑부(300), 주메모리(400), 및 플로우 저장소(500)을 포함한다.

[0021] 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 네트워크로부터 트래픽 데이터를 수집하며, 수집된 트래픽 데이터는 플로우 저장소(500)에 저장될 수 있다. 일실시예에서 트래픽 데이터는 5-튜플(5-tuple)로 구조의 플로우(flow) 형태로 수집되어 저장된다. 5-튜플은 트래픽 플로우가 포함하고 있는 소스 주소(예: source IP address), 목적지 주소(예: destination IP address), 소스 포트(예: source port), 목적지 포트(예: destination port), 및 프로토콜(예: layer 4 protocol)의 5가지 정보를 나타낸다.

[0022] 본 명세서는 단방향 플로우(One-way-flow)는 5-튜플이 나타내는 정보가 동일한 단일 방향의 연속된 패킷(packet)의 집합으로 정의한다. 또한 양방향 플로우(Two-way-flow)는 방향이 반대인 두개의 단방향 플로우의 집합으로 정의한다. 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 양방향 플로우를 사용한다.

[0023] 전술한 바와 같이, 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 수집된 네트워크 트래픽 플로우 데이터를 분류하기 위해 주메모리(400)로 적재한다. 이때 각 플로우는 각 플로우의 5-튜플을 키(key)로 하여 매핑(mapping)되는 해시 테이블 구조를 사용한다. 해시 테이블에 대해 도 2을 통해 설명한다.

[0024] 도 2은 종래의 싱글 해시 테이블 구조를 도시하고 있다.

[0025] 해시 테이블(hash table)은 해시 함수(hash function)에 의해 데이터의 키로부터 산출된 인덱스(index)를 통하여 참조되는 저장 공간인 버킷(bucket)에 데이터를 저장하는 데이터 구조(data structure)이다. 즉, 해시 함수는 데이터의 키를 사용하여 데이터를 해시 테이블의 각 버킷에 매핑한다.

[0026] 이때 도시된 것과 같이, 여러 개의 데이터가 같은 버킷으로 매핑될 수 있는데, 이를 충돌(collision)이라고 한다. 도시되어 있는 해시 테이블은 이를 해결하기 위해 같은 버킷으로 매핑되는 데이터를 연결 리스트(linked list)로 연결하여 저장하고, 해당 버킷은 해당 버킷으로 매핑된 첫번째 데이터를 가리키는 방식을 사용한다. 본 명세서에서는 데이터가 저장되는 공간을 편의상 연결 리스트 저장 공간이라고 지칭하기로 한다. 또는 해당 기술 분야의 관례대로 버킷과 실제의 데이터 저장 공간을 구분하여 지칭하지 않고 데이터가 해당 버킷에 저장된다고 표현할 수도 있다.

[0027] 즉, 싱글 해시 구조는 고정된 크기의 해시 테이블을 생성하고 데이터에서 추출한 키 값으로 해시 함수를 통해 인덱스를 계산하여 해당 인덱스에 의해 참조되는 버킷에 데이터를 적재한다. 만약 동일한 인덱스에 둘 이상의 데이터를 적재해야 하는 경우, 즉, 충돌이 일어난 경우, 연결 리스트 데이터 구조를 사용하여 해당 데이터들을 연결한다.

[0028] 도시된 해시 테이블의 크기는 a이다. 즉, 도시된 해시 테이블은 0, 1, ... a-1이라는 인덱스를 통해 참조되는 a 개의 버킷으로 구성된다. 도시되어 있는 바와 같이 플로우 1과 플로우 2는 동일한 버킷으로 매핑되었다.

[0029] 트래픽 분석을 위해 종래에는 일반적으로 이러한 싱글 해시 테이블 구조를 사용해 왔지만, 싱글 해시 구조는 응용의 종류가 다양해지고 사용자가 증가함에 따라 데이터 적재 및 검색 시 처리 시간이 증가하여 트래픽 분석에 있어 큰 오버헤드로 작용한다. 이는 해시 테이블의 크기는 고정되어 있지만, 트래픽 데이터의 양은 계속 증가하기 때문으로, 해시 테이블 사용률 및 동일 버킷에 매핑되어 동일 인덱스를 통해 연결 리스트로 적재되는 데이터의 개수가 증가하여, 적재 및 검색 시 데이터 비교 횟수가 증가하기 때문이다.

[0030] 이러한 문제를 해결하기 위한 가장 간단한 방법은 해시 테이블의 크기를 증가시키는 것이다. 하지만, 분석 시스템에서 사용할 수 있는 메모리의 크기는 제한이 있기 때문에 무한정 해시 테이블의 크기를 증가시키는 것은 불가능할 뿐만 아니라, 적절한 크기의 해시 테이블을 찾는 것 또한 쉽지 않다.

[0031] 또한, 한시적으로 트래픽 양에 적절한 해시 테이블 크기를 찾더라도, 그에 적합한 해시 함수를 재정의 해야 할 뿐만 아니라 트래픽 데이터의 양은 지속적으로 변하기 때문에 반복된 작업을 계속해야 하는 어려움이 있다.

[0032] 따라서 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 듀얼 해시 테이블 구조를 사용한다. 이를 도 3를 통해 살펴보도록 하자.

[0033] 도 3는 본 발명의 일실시예에 따른 듀얼 해시 테이블 구조를 도시하고 있다.

[0034] 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 기존의 싱글 해시 구조의 한계점을 보완한 듀얼 해시 구조를 사용한다. 제 1 해시 테이블의 길이는 저장되는 트래픽 데이터의 양에 따라 변동 가능한 a로 설정하고, 제 2 해시 테이블의 길이는 b로 고정하여 설정한다.

[0035] 트래픽 데이터의 양은 데이터를 메모리에 로드하기 전에 트래픽 데이터의 파일 크기를 데이터 크기로 나누어 구한다. 트래픽 데이터의 양(N)에 기초하여 산출한 크기인 a로 제 1 해시 테이블을 생성하고, 제 1 해시 테이블의 각 인덱스에 참조되는 a 개의 제 2 해시 테이블을 생성한다. 즉, 제 1 해시 테이블의 크기는 제 2 해시 테이블의 개수와 같다. 이때, 제 1 해시 테이블의 길이인 a는 트래픽 플로우 양에 따라 가변적으로 조정되며, 제 2 해시 테이블의 크기는 모두 b로 고정되어 있다.

[0036] 편의상 본명세서에서는 제 1 해시 테이블의 인덱스를 테이블 인덱스, 제 2 해시 테이블의 인덱스를 버킷 인덱스라고 지칭하기로 한다. 제 1 해시 테이블의 인덱스는 제 2 해시 테이블을 참조하는 데 사용되고, 제 2 해시 테이블의 인덱스는 실제로 데이터가 연결되어 저장될 버킷을 참조하는 데 사용되기 때문이다.

[0037] 예를 들어, 인덱스 0가 첫번째 항목을 가리키는 일실시예에서, 어떤 트래픽 플로우 데이터에 대해 제 1 해시 테이블의 인덱스가 0, 제 2 해시 테이블의 인덱스가 1로 산출되었다면, a개의 제 2 해시 테이블 중 첫번째 제 2 해시 테이블이 선택되고, 첫번째 제 2 해시 테이블의 b 개의 버킷 중 두번째 버킷이 선택된다. 즉, 해당 트래픽 플로우는 첫번째 제 2 해시 테이블의 두번째 버킷으로 매핑된다.

[0038] 따라서 해당 트래픽 플로우 데이터는 테이블 인덱스 0 및 버킷 인덱스 1을 사용하여 참조될 수 있다. 즉, 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 해당 트래픽 플로우 데이터를 주메모리(400)에 적재할 때, 테이블 인덱스 0 및 버킷 인덱스 1을 사용하여 해당 트래픽 플로우 데이터를 첫번째 제 2 해시 테이블의 두번째 버킷과 연결된 연결 리스트 저장 공간에 저장한다. 또한, 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 해당 트래픽 플로우 데이터를 주메모리(400)에서 검색할 때, 테이블 인덱스 0 및 버킷 인덱스 1을 사용하여 해당 트래픽 플로우 데이터를 첫번째 제 2 해시 테이블의 두번째 버킷과 연결된 연결 리스트 저장 공간에서 검색한다.

[0039] 즉, 트래픽 플로우 데이터는 산출된 제 1 해시 테이블의 인덱스 및 제 2 해시 테이블의 인덱스에 의해 참조되는 버킷에 적재되며, 서로 다른 데이터가 동일한 버킷으로 매핑되는 경우, 즉, 충돌이 일어난 경우에는 싱글 해시 테이블에서와 마찬가지로 연결 리스트 데이터 구조로 연결하여 저장된다.

[0040] 다시 도 1로 돌아가서, 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 이러한 듀얼 해시 테이블을 사용하여 주메모리(400)에 네트워크 트래픽 플로우 데이터를 적재하기 위해 크기 결정부(100), 제 1 매핑부(200), 및 제 2 매핑부(300)를 포함한다.

[0041] 크기 결정부(100)는 제 1 해시 테이블의 크기를 결정하며, 제 1 매핑부(200)는 테이블 인덱스, 즉, 제 1 해시 테이블의 인덱스를 산출하고, 제 2 매핑부(300)는 버킷 인덱스, 즉, 제 2 해시 테이블의 인덱스를 산출한다.

[0042] 자세한 내용은 이하 본 발명의 일실시예에 따른 트래픽 분석 방법을 설명하고 있는 도면들인 도 4 내지 도 9를 사용하여 설명한다.

[0043] 도 4는 본 발명의 일실시예에 따른 트래픽 분석 방법의 흐름을 도시하고 있다.

[0044] 먼저, 트래픽 플로우 양에 기초하여 제 1 해시 테이블 크기를 결정한다(S100). 본 발명의 일실시예에 따른 크기 결정부(100)는 도 5에 설명한 알고리즘을 사용하여 제 1 해시 테이블의 크기를 결정할 수 있다.

[0045] 제 1 해시 테이블의 테이블 인덱스를 결정한다(S200). 본 발명의 일실시예에 따른 제 1 매핑부(200)는 도 6 및 도 7에서 설명한 방법, 즉, 제 1 해시 함수를 사용하여 테이블 인덱스를 결정할 수 있다.

[0046] 제 2 해시 테이블의 버킷 인덱스를 결정한다(S300). 본 발명의 일실시예에 따른 제 1 매핑부(200)는 도 6 및 도 7에서 설명한 방법, 즉, 제 2 해시 함수를 사용하여 버킷 인덱스를 결정할 수 있다.

[0047] 산출된 테이블 인덱스 및 버킷 인덱스로 결정되는 버킷에 해당 트래픽 플로우 데이터를 연결 저장한다(S400).

[0048] 도 5는 본 발명의 일실시예에 따른 제 1 해시 테이블의 크기를 산출하는 방법을 도시하고 있다.

[0049] 전술한 바와 같이, 제 1 해시 테이블의 길이는 저장되는 트래픽 데이터의 양에 따라 변동 가능한 a로 설정하고, 제 2 해시 테이블의 길이는 b 로 고정하여 설정한다.

[0050] 일실시예에서 b는 65,536이다. 이는 후술하는 바와 같이, 제 2 해시 테이블의 버킷 인덱스를 결정할 때, IP 주소의 앞, 뒤 2 바이트씩 폴딩하여 제 2 해시 함수의 키로 사용하기 때문에 2 바이트가 나타낼 수 있는 수, 즉 65,536(=2의 16승)로 정의한 것이다.

[0051] 통계적으로 해시 테이블 사용률이 80% 이상이 되면 해시 테이블의 성능이 저하되기 시작되기 때문에 해시 테이블 사용률이 80% 이하가 되는 최대값 a를 선정하여 제 1 해시 테이블로 설정한다.

[0052] 도시된 알고리즘은 제 1 해시 테이블의 크기 a를 계산하는 과정을 나타내고 있다. 해시 사용률이 80% 이상 되면 성능 저하가 시작 되기 때문에, 해시 사용률이 80% 이하가 되는 최대값 a를 선정한다. 해당 알고리즘에 사용된 수식은 실험을 통하여 도출하였으나, 여기에 한정되지 않는다.

[0053] 도 6 및 도 7은 본 발명의 일실시예에 따른 제 1 해시 테이블의 인덱스를 산출하는 방법을 도시하고 있으며, 도 8 및 도 9는 본 발명의 일실시예에 따른 제 2 해시 테이블의 인덱스를 산출하는 방법을 도시하고 있다.

[0054] 전술한 바와 같이, 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 각 네트워크 트래픽 플로우의 5-튜플을 해당 플로우를 해시 테이블로 매핑하기 위한 키로 사용한다. 전술한 바와 같이 데이터의 키를 사용하여 해시 테이블의 각 버킷으로 매핑하는 것은 해시 함수가 담당한다. 트래픽 분석의 효율성을 높이기 위해서는 충돌이 적은 해시 함수를 사용하는 것이 바람직하다는 것은 당업자에게 쉽게 이해될 것이다. 그러나 충돌이 적은 해시 함수를 설계하는 것은 쉬운 일이 아니기 때문에, 전술한 바와 같이, 종래의 싱글 해시 테이블 구조에서는 트래픽 양이 늘어남에 따라 해시 함수를 재정의해야 하는 필요가 발생할 경우 오버헤드로 작용한다.

[0055] 그러나 본 발명의 일실시예에 따른 트래픽 분석 장치(10)는 각각의 해시 테이블에 대해 해시 함수를 재정의하지 않고 중복하여 사용하므로, 트래픽 양의 변동에 따라 해시 함수를 재정의해야 하는 추가적인 오버헤드가 없다. 즉, a개의 제 2 해시 테이블은 모두 동일한 해시 함수(제 2 해시 함수)를 사용하며, 이는 제 1 해시 테이블이 사용하는 해시 함수(제 1 해시 함수)와 거의 동일하다. 또한, 제 1 해시 테이블의 길이인 a가 변하더라도 해시 함수는 재정의할 필요가 없다.

[0056] 도 6은 도 4의 S200 단계, 즉, 제 1 해시 함수의 흐름을 도시하고 있다.

[0057] 플로우의 각 튜플에서 1 바이트씩 추출하여 합산한다(S210).

[0058] 제 1 해시 테이블 크기와 AND 연산하여 테이블 인덱스를 산출한다(S220).

[0059] 도 7을 통해 자세히 살펴보면, 플로우 5-튜플의 소스 주소와 목적지 주소, 소스 포트와 목적지 포트, 및 프로토콜의 마지막 1 바이트는 각각 0에서 255 사이 수를 가지고 있다. 이들을 모두 산술적으로 합산한 후, 제 1 해시 테이블 크기 a에서 1을 뺀 값, 즉, a-1과 AND 연산하면 제 1 해시 테이블의 인덱스인 0 내지 a-1을 얻을 수 있다.

[0060] 도 8 및 도 9는 본 발명의 일실시예에 따른 제 2 해시 테이블의 인덱스를 산출하는 방법을 도시하고 있다.

[0061] 도 8은 도 4의 S300 단계, 즉, 제 2 해시 함수의 흐름을 도시하고 있다.

[0062] 플로우의 각 튜플에서 앞뒤 2 바이트씩 추출하여 폴딩 합산한다(S310).

[0063] 제 2 해시 테이블 크기와 AND 연산하여 버킷 인덱스를 산출한다(S320).

[0064] 도 9을 통해 자세히 살펴보면, 테이블 인덱스를 산출할 때와 같이 플로우 5-튜플의 소스 주소와 목적지 주소, 소스 포트와 목적지 포트, 및 프로토콜의 일부 바이트들을 모두 산술적으로 합산하되, 1 바이트가 아니라 2 바이트씩 합산하며, 소스 주소와 목적지 주소의 시작 2 바이트 및 마지막 2 바이트는 폴딩(folding)하여 합산한다. 소스 주소와 목적지 주소 각각의 시작 2 바이트 및 마지막 2 바이트, 소스 포트와 목적지 포트의 2 바이트들을 모두 산술적으로 합산한 후, 제 2 해시 테이블 크기 b에서 1을 뺀 값, 즉, b-1과 AND 연산하면 제 2 해시 테이블의 인덱스인 0 내지 b-1(=65,535)을 얻을 수 있다.

[0065] 도 10 내지 도 12는 본 발명의 효과를 검증하기 위한 실험을 도시하고 있다. 듀얼 해시 구조의 성능을 증명 하기 위해 증가하는 트래픽 데이터 양에 따른 싱글 해시 구조와 듀얼 해시 구조의 성능 비교 결과를 기술한다. 도시하지 않았지만 이들 각 도면에서 각 데이터의 단위는 10의 6승이다.

[0066] 듀얼 해시 구조는 기존의 싱글 해시 구조와 달리,트래픽 데이터를 메모리에 로드하기 전에 트래픽 데이터 양을 확인하고, 그에 적절한 크기의 해시 테이블로 확장가능하다. 테이블이 확장될 때, 제 2 해시 함수가 중복 사용되는 구조이므로, 해시 함수를 재정의 할 필요없이 성능유지가 가능하기 때문이다. 네트워크 트래픽 양은 유동적이기 때문에, 그에 적합한 가변적인 해시 테이블을 사용함으로써 효과적인 데이터 메모리 적재 및 검색이 가능하다.

[0067] 도 10은 트래픽 데이터 양에 따른 검색 값과 해시 테이블 크기에 대한 비교 결과이다. 여기에서 검색 값은 해시 테이블에 저장된 모든 트래픽 데이터를 1회씩 비교 검색할 때 소요되는 총 비교 횟수로 정의하였으며, 해시 테이블의 성능을 평가하기 위한 것이다. 비교를 위해 싱글 해시 테이블 구조는 해시 테이블 길이를 각각 다르게 설정하였다.

[0068] 도 11과 같이 싱글 해시 구조는 해시 테이블의 길이가 고정되어 있기 때문에, 트래픽 양이 증가할수록 충돌 횟수 또한 증가하고, 따라서 연결 리스트가 길어지게 되어 검색 값이 급격하게 증가한다. 반면에, 듀얼 해시 구조는 트래픽 데이터 양의 증가에 적절한 크기의 해시 테이블이 생성되기 때문에 검색 값이 점진적으로 증가하는 것을 확인할 수 있다.

[0069] 도12와 같이 듀얼 해시는 해시 테이블이 확장되어 메모리를 많이 사용하였음에도 해시 성능에서 트래픽 양의 변화와 관련없이 좋은 성능을 유지 할 수 있다. 저장되는 데이터의 양에 비해서 적은 메모리를 사용하는 것이고, 성능 유지를 위한 최소한의 메모리를 사용하기 때문에 효율적이다. 그리고, 트래픽 양의 변화에 대해, 해시 함수의 재정의없이 자동적으로 해시 테이블을 확장시키므로, 오버헤드가 없어 효율적이다.

[0070] 본 발명의 일 실시예는 컴퓨터에 의해 실행되는 프로그램 모듈과 같은 컴퓨터에 의해 실행가능한 명령어를 포함하는 기록 매체의 형태로도 구현될 수 있다. 컴퓨터 판독 가능 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 가용 매체일 수 있고, 휘발성 및 비휘발성 매체, 분리형 및 비분리형 매체를 모두 포함한다. 또한, 컴퓨터 판독 가능 매체는 컴퓨터 저장 매체 및 통신 매체를 모두 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함한다. 통신 매체는 전형적으로 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈, 또는 반송파와 같은 변조된 데이터 신호의 기타 데이터, 또는 기타 전송 메커니즘을 포함하며, 임의의 정보 전달 매체를 포함한다.

[0071] 전술한 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.

[0072] 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

## 부호의 설명

[0073] 10: 트래픽 분석 장치

100: 크기 결정부

200: 제 1 매핑부

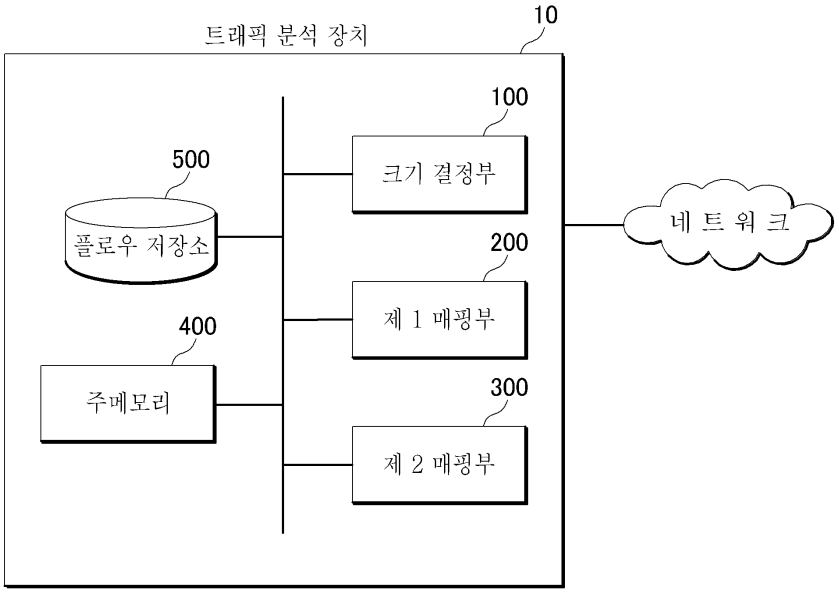
300: 제 2 매핑부

400: 주메모리

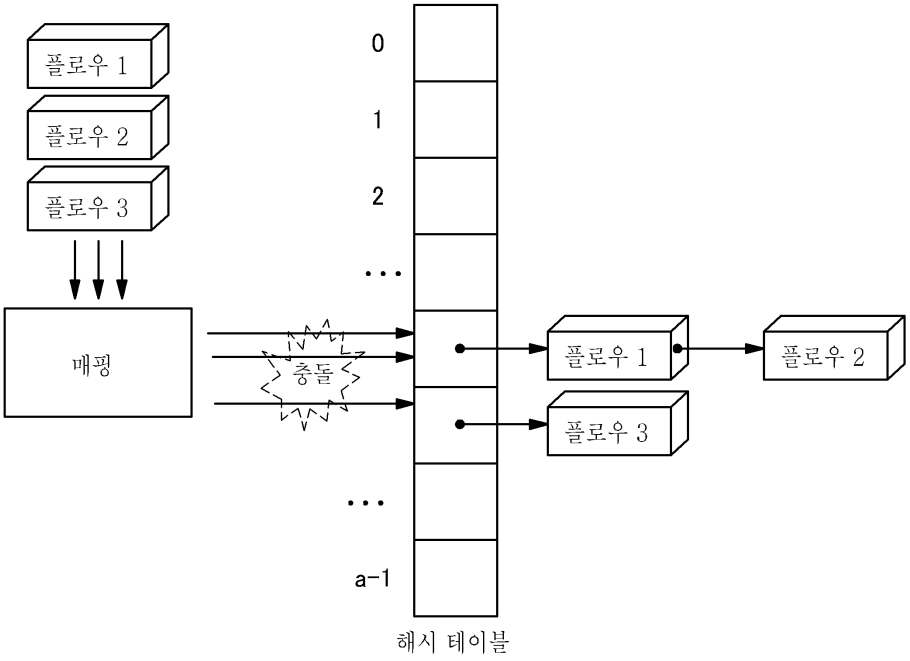
500: 플로우 저장소

## 도면

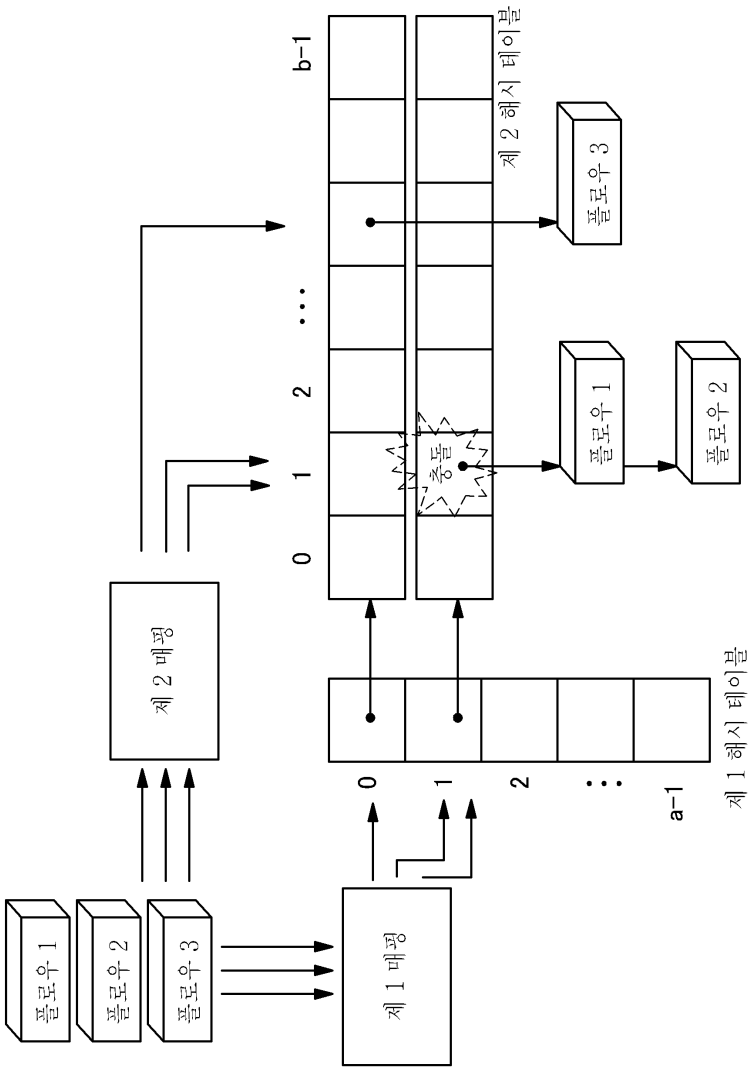
### 도면1



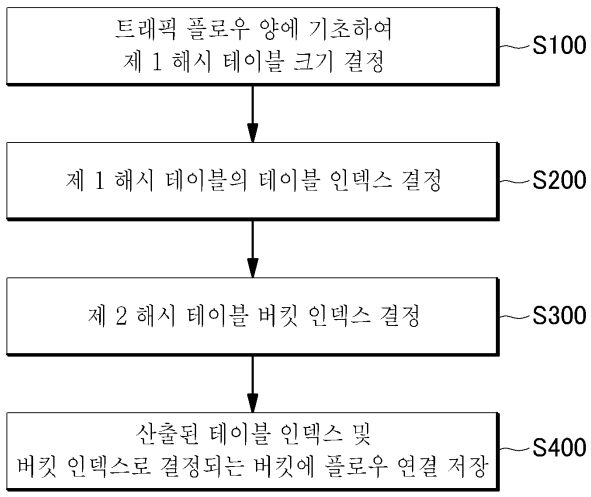
도면2



도면3


플로우 1
플로우 2
플로우 3
제 1 매핑
제 2 매핑
제 1 해시 테이블
제 2 해시 테이블
0
1
2
a-1
b-1
충돌
플로우 1
플로우 2
플로우 3

도면4


트래픽 플로우 양에 기초하여
제 1 해시 테이블 크기 결정
S100
제 1 해시 테이블의 테이블 인덱스 결정
S200
제 2 해시 테이블 버킷 인덱스 결정
S300
산출된 테이블 인덱스 및
버킷 인덱스로 결정되는 버킷에 플로우 연결 저장
S400

도면5

Input : N (트래픽 플로우 양)

Output : a (제 1 해시 테이블 크기)

BEGIN
int FindAlpha(int N)
{
b = 65,536 // 제 2 해시 테이블 크기
int k ;

$2^k \leq \frac{N}{1.2 \times b} \leq 2^{k+1}$ 만족하는 k 산출

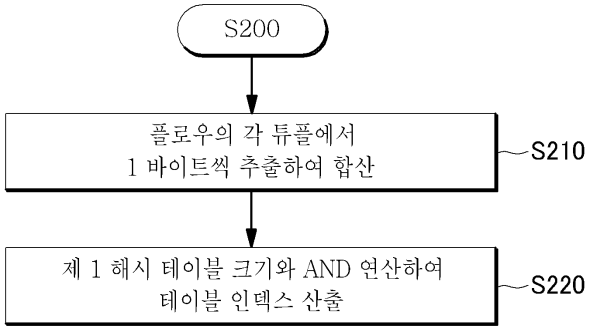
**if** ( $\left| 2^k - \frac{N}{1.2 \times b} \right| \leq \left| 2^{k+1} - \frac{N}{1.2 \times b} \right|$ )

**then** a = $2^{k+1}$

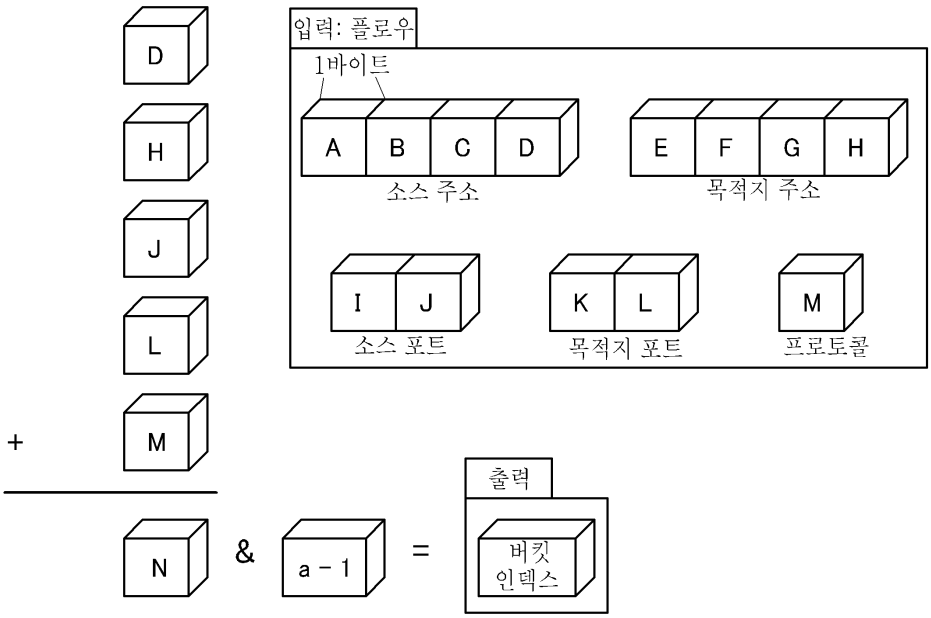
**else** a = $2^k$

return a;
}
END

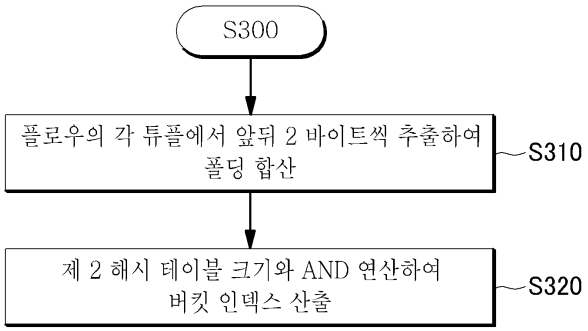
도면6



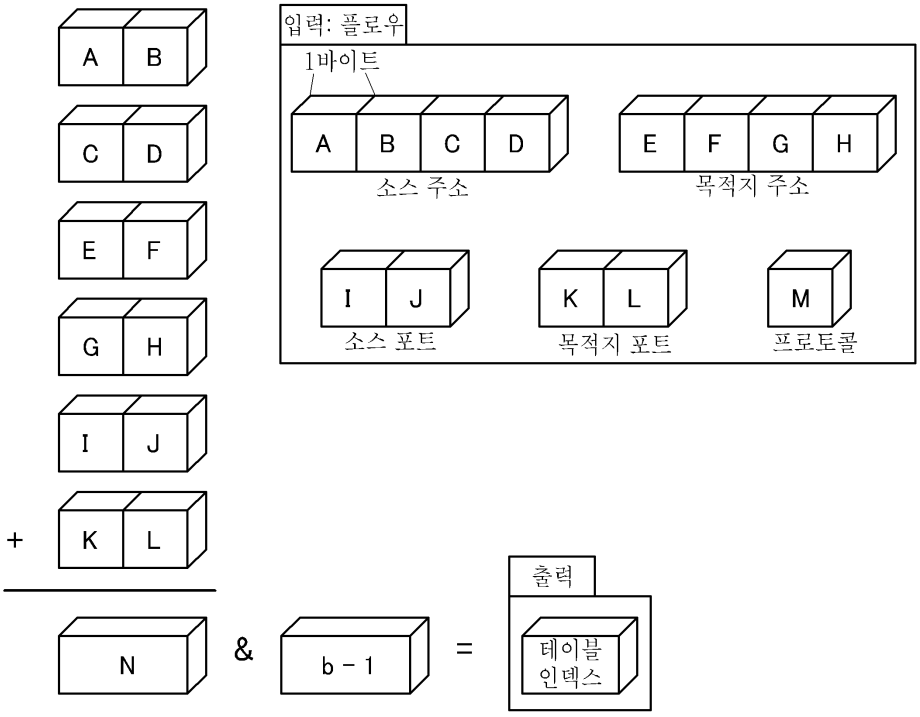
도면7


D
H
J
L
+
M
N
&
a - 1
=
입력: 플로우
1바이트
A
B
C
D
소스 주소
E
F
G
H
목적지 주소
I
J
소스 포트
K
L
목적지 포트
M
프로토콜
출력
버킷
인덱스

도면8


S300
플로우의 각 튜플에서 앞뒤 2 바이트씩 추출하여
폴딩 합산
S310
제 2 해시 테이블 크기와 AND 연산하여
버킷 인덱스 산출
S320

도면9

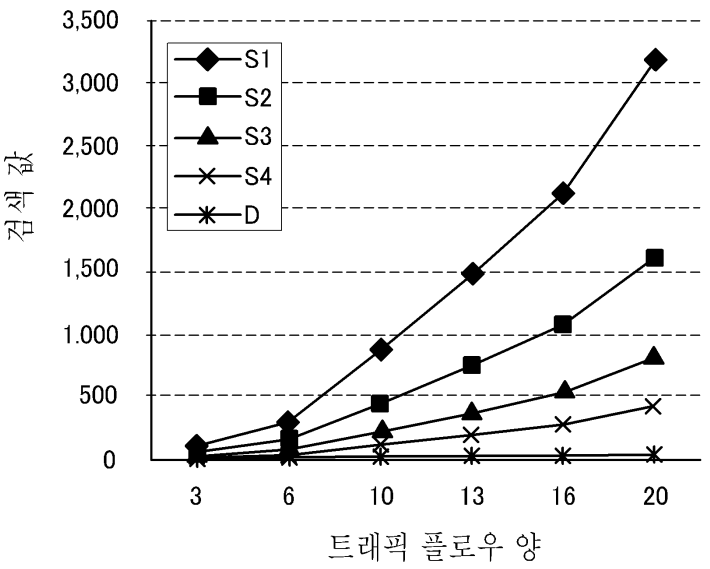

A
B
C
D
E
F
G
H
I
J
+
K
L
N
&
b - 1
=
입력: 플로우
1바이트
A
B
C
D
소스 주소
E
F
G
H
목적지 주소
I
J
소스 포트
K
L
목적지 포트
M
프로토콜
출력
테이블
인덱스

도면10

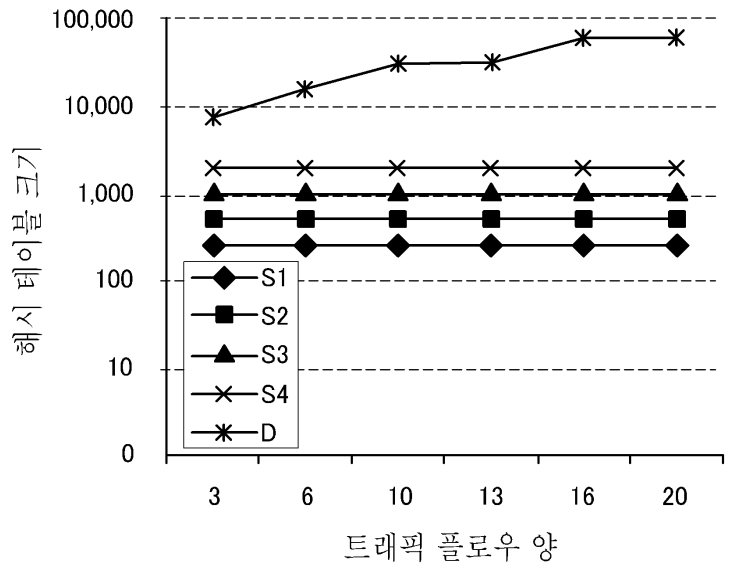
| Hash \ Data | 3 | | 6 | | 10 | | 13 | | 16 | | 20 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | S_v | size | S_v | size | S_v | size | S_v | size | S_v | size | S_v | size |
| S1 | 89 | 0.25 | 291 | 0.25 | 865 | 0.25 | 1,474 | 0.25 | 2,119 | 0.25 | 3,179 | 0.25 |
| S2 | 46 | 0.5 | 148 | 0.5 | 438 | 0.5 | 744 | 0.5 | 1,068 | 0.5 | 1,600 | 0.5 |
| S3 | 25 | 1 | 77 | 1 | 224 | 1 | 379 | 1 | 542 | 1 | 810 | 1 |
| S4 | 14 | 2 | 42 | 2 | 117 | 2 | 196 | 2 | 279 | 2 | 415 | 2 |
| D | 6 | 8 | 11 | 16 | 17 | 32 | 25 | 32 | 25 | 64 | 33 | 64 |

**S1**: 싱글(65,536), **S2**: 싱글(65,536×2), **S3**: 싱글(65,536×4), **S4**: 싱글(65,536×8), **D**: 듀얼(65,536×α), **S_v**: Search_value (단위 : $10^6$)

도면11


3,500
3,000
2,500
2,000
1,500
1.000
500
0
검색 값
S1
S2
S3
S4
D
3
6
10
13
16
20
트래픽 플로우 양

도면12


100,000
10,000
1,000
100
10
0
해시 테이블 크기
S1
S2
S3
S4
D
3
6
10
13
16
20
트래픽 플로우 양