



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2010년11월23일
(11) 등록번호 10-0996136
(24) 등록일자 2010년11월17일

(51) Int. Cl.

H04L 12/56 (2006.01) H04L 29/02 (2006.01)

(21) 출원번호 10-2009-0056938

(22) 출원일자 2009년06월25일

심사청구일자 2009년06월25일

(56) 선행기술조사문헌

논문1: 정보과학회논문지

논문2: 한국통신학회논문지

논문3: ACM Symposim

논문5: PDCN 2007

전체 청구항 수 : 총 24 항

(73) 특허권자

이화여자대학교 산학협력단

서울 서대문구 대현동 11-1 이화여자대학교내

(72) 발명자

임혜숙

서울특별시 강남구 도곡2동 삼성래미안아파트
108-1102호

(74) 대리인

송해모, 이철희

심사관 : 김세영

(54) 패킷 분류 장치 및 그 방법

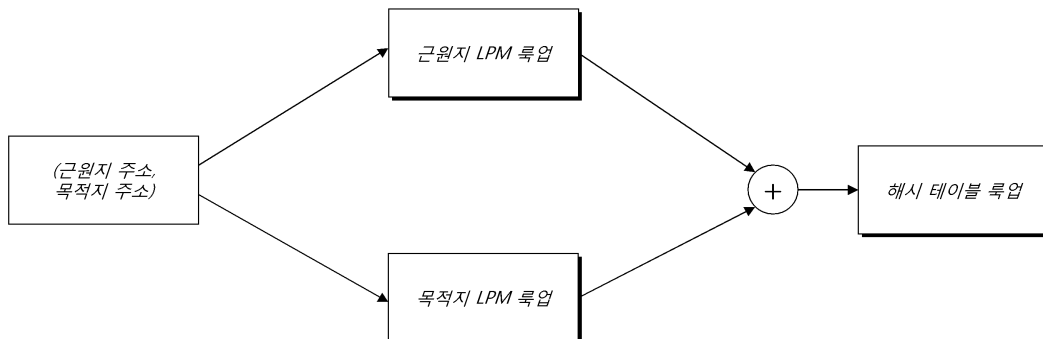
(57) 요약

본 발명의 실시예는 패킷 분류 장치 및 그 방법에 관한 것이다.

본 발명의 실시예는, 블룸 필터를 이용하여 물 켓의 근원지 프리픽스와 목적지 프리픽스의 정보를 블룸 필터에 미리 저장해두고 패킷이 입력되었을 때 각 목적지 블룸 필터 및 근원지 블룸 필터를 각각 병렬로 통과시킴으로써 입력 패킷에 대한 해시 테이블 접근 횟수를 효과적으로 줄여 패킷 분류를 수행하는 것을 특징으로 한다.

본 발명의 실시예에 의하면, 메모리 요구량 및 메모리 접근 횟수를 줄임으로써, 더욱 효율적이고 신속한 패킷 분류를 가능하게 하는 효과가 있다.

대표도 - 도1



특허청구의 범위

청구항 1

입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 프리픽스 입력부;

기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블룸 필터 검색부;

상기 제 1 양성 비트를 토대로 근원지 해시 테이블을 검색하여 상기 제 1 양성 비트 중에서 참 양성으로 판단된 제 1 참 양성 비트를 추출하는 근원지 해시 테이블 검색부;

기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블룸 필터 검색부;

상기 제 2 양성 비트를 토대로 목적지 해시 테이블을 검색하여 상기 제 2 양성 비트 중에서 참 양성으로 판단된 제 2 참 양성 비트를 추출하는 목적지 해시 테이블 검색부; 및

상기 제 1 참 양성 비트 및 상기 제 2 참 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 2

제 1항에 있어서,

상기 근원지 블룸 필터 검색부는,

상기 기 저장된 근원지 프리픽스 길이에 따라 상기 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하는 제 1 비트 추출부;

상기 제 1 비트 각각에 대하여 기 정의된 제 1 코드값 비트 수를 갖는 제 1 코드값을 생성하고, 상기 제 1 코드값으로부터 제 1 해싱 색인 값을 생성하는 제 1 해싱부; 및

상기 제 1 해싱 색인 값에 해당하는 상기 근원지 블룸 필터의 비트벡터를 확인하고, 상기 확인된 비트벡터를 토대로 상기 제 1 비트 중에서 양성으로 판단된 상기 제 1 양성 비트를 추출하는 제 1 양성 비트 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 3

제 2항에 있어서,

상기 제 1 해싱부는,

상기 제 1 비트 각각에 대하여 상기 제 1 코드값 비트 수를 갖는 상기 제 1 코드값을 생성하는 제 1 코드값 생성부; 및

기 정의된 제 1 해싱 색인 값 비트 수만큼의 앞자리 코드값 및 뒷자리 코드값을 상기 제 1 코드값으로부터 추출하고, 상기 앞자리 코드값 및 상기 뒷자리 코드값을 포함하는 상기 제 1 해싱 색인 값을 제 1 해싱 색인 값 생성부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 4

제 3항에 있어서,

상기 근원지 해시 테이블 검색부는,

상기 제 1 양성 비트 각각에 대하여 상기 제 1 해싱부에서 생성된 상기 제 1 해싱 색인 값에 포함된 상기 뒷자리 코드값을 이용해서 상기 근원지 해시 테이블을 검색하여, 상기 제 1 양성 비트 중에서 상기 근원지 해시 테이블의 해당 엔트리에 존재하는 제 1 양성 비트를 상기 제 1 참 양성 비트로서 판단하여 추출하는 것을 특징으로 하는 것을 특징으로 하는 패킷 분류 장치.

청구항 5

제 1항에 있어서,

상기 목적지 블록 필터 검색부는,

상기 기 저장된 목적지 프리픽스 길이에 따라 상기 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하는 제 2 비트 추출부;

상기 제 2 비트 각각에 대하여 기 정의된 제 2 코드값 비트 수를 갖는 제 2 코드값을 생성하고, 상기 제 2 코드값으로부터 제 2 해싱 색인 값을 생성하는 제 2 해싱부; 및

상기 제 2 해싱 색인 값에 해당하는 상기 목적지 블록 필터의 비트벡터를 확인하고, 상기 확인된 비트벡터를 토대로 상기 제 2 비트 중에서 양성으로 판단된 상기 제 2 양성 비트를 추출하는 제 2 양성 비트 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 6

제 5항에 있어서,

상기 제 2 해싱부는,

상기 제 2 비트 각각에 대하여 상기 제 2 코드값 비트 수를 갖는 상기 제 2 코드값을 생성하는 제 2 코드값 생성부; 및

기 정의된 제 2 해싱 색인 값 비트 수만큼의 앞자리 코드값 및 뒷자리 코드값을 상기 제 2 코드값으로부터 추출하고, 상기 앞자리 코드값 및 상기 뒷자리 코드값을 포함하는 상기 제 2 해싱 색인 값을 생성하는 제 2 해싱 색인 값 생성부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 7

제 6항에 있어서,

상기 목적지 해시 테이블 검색부는,

상기 제 2 양성 비트 각각에 대하여 상기 제 2 해싱부에서 생성된 상기 제 2 해싱 색인 값에 포함된 상기 뒷자리 코드값을 이용해서 상기 목적지 해시 테이블을 검색하여, 상기 제 2 양성 비트 중에서 상기 목적지 해시 테이블의 해당 엔트리에 존재하는 제 2 양성 비트를 상기 제 2 참 양성 비트로서 판단하여 추출하는 것을 특징으로 하는 것을 특징으로 하는 패킷 분류 장치.

청구항 8

제 1항에 있어서,

상기 튜플 해시 테이블 검색부는,

상기 제 1 참 양성 비트의 길이 및 상기 제 2 참 양성 비트의 길이를 조합하여 상기 하나 이상의 튜플을 생성하는 튜플 생성부;

상기 하나 이상의 튜플 중에서 상기 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플을 선별하는 튜플 선별부; 및

상기 선별된 튜플에 대하여 상기 튜플 해시 테이블을 검색하여 패킷 분류를 위한 상기 룰을 추출하는 룰 추출부를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 9

제 8항에 있어서,

상기 룰 추출부는,

상기 선별된 튜플의 생성시 이용된 제 1 참 양성 비트 및 제 2 참 양성 비트를 연결하여 연결 비트를 생성하고, 상기 생성된 연결 비트로부터 기 정의된 제 3 코드값 비트 수를 갖는 제 3 코드값을 생성하고, 상기 제 3 코드값으로부터 기 정의된 제 3 해싱 색인 값 비트 수만큼의 뒷자리 코드값을 포함하는 제 3 해싱 색인 값을 생성하는 제 3 해싱부; 및

상기 제 3 해싱 색인 값을 이용하여 상기 튜플 해시 테이블을 검색하여 상기 룰을 추출하는 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 10

제 1항에 있어서,

상기 근원지 블룸 필터 및 상기 목적지 블룸 필터는,

프리픽스 길이의 구분없이 모든 프리픽스에 대하여, 동일한 해싱 함수를 이용하여 얻어진 해싱 색인 값에 해당하는 비트벡터를 특정 값으로 프로그래밍하고, 프로그래밍 시 사용된 프리픽스에 대한 프리픽스 길이는 별도로 저장하는 것을 특징으로 하는 패킷 분류 장치.

청구항 11

입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 프리픽스 입력 단계;

기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블룸 필터 검색 단계;

상기 제 1 양성 비트를 토대로 근원지 해시 테이블을 검색하여 상기 제 1 양성 비트 중에서 참 양성으로 판단된 제 1 참 양성 비트를 추출하는 근원지 해시 테이블 검색 단계;

기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블룸 필터 검색 단계;

상기 제 2 양성 비트를 토대로 목적지 해시 테이블을 검색하여 상기 제 2 양성 비트 중에서 참 양성으로 판단된 제 2 참 양성 비트를 추출하는 목적지 해시 테이블 검색 단계; 및

상기 제 1 참 양성 비트 및 상기 제 2 참 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색 단계

를 포함하는 것을 특징으로 하는 패킷 분류 방법.

청구항 12

입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 프리픽스 입력부;

기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블룸 필터 검색부;

기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블룸 필터 검색부;

상기 제 1 양성 비트 및 상기 제 2 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 블록 필터를 검색하여 양성으로 판단된 양성 튜플을 추출하는 튜플 블록 필터 검색부; 및

상기 양성 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 13

제 12항에 있어서,

상기 근원지 블록 필터 검색부는,

상기 기 저장된 근원지 프리픽스 길이에 따라 상기 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하는 제 1 비트 추출부;

상기 제 1 비트 각각에 대하여 기 정의된 제 1 코드값 비트 수를 갖는 제 1 코드값을 생성하고, 상기 제 1 코드값으로부터 제 1 해싱 색인 값을 생성하는 제 1 해싱부; 및

상기 제 1 해싱 색인 값에 해당하는 상기 근원지 블록 필터의 비트벡터를 확인하고, 상기 확인된 비트벡터를 토대로 상기 제 1 비트 중에서 양성으로 판단된 상기 제 1 양성 비트를 추출하는 제 1 양성 비트 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 14

제 13항에 있어서,

상기 제 1 해싱부는,

상기 제 1 비트 각각에 대하여 상기 제 1 코드값 비트 수를 갖는 상기 제 1 코드값을 생성하는 제 1 코드값 생성부; 및

기 정의된 제 1 해싱 색인 값 비트 수만큼의 앞자리 코드값 및 뒷자리 코드값을 상기 제 1 코드값으로부터 추출하고, 상기 앞자리 코드값 및 상기 뒷자리 코드값을 포함하는 상기 제 1 해싱 색인 값을 제 1 해싱 색인 값 생성부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 15

제 12항에 있어서,

상기 목적지 블록 필터 검색부는,

상기 기 저장된 목적지 프리픽스 길이에 따라 상기 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하는 제 2 비트 추출부;

상기 제 2 비트 각각에 대하여 기 정의된 제 2 코드값 비트 수를 갖는 제 2 코드값을 생성하고, 상기 제 2 코드값으로부터 제 2 해싱 색인 값을 생성하는 제 2 해싱부; 및

상기 제 2 해싱 색인 값에 해당하는 상기 목적지 블록 필터의 비트벡터를 확인하고, 상기 확인된 비트벡터를 토대로 상기 제 2 비트 중에서 양성으로 판단된 상기 제 2 양성 비트를 추출하는 제 2 양성 비트 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 16

제 15항에 있어서,

상기 제 2 해싱부는,

상기 제 2 비트 각각에 대하여 상기 제 2 코드값 비트 수를 갖는 상기 제 2 코드값을 생성하는 제 2 코드값 생

성부; 및

기 정의된 제 2 해싱 색인 값 비트 수만큼의 앞자리 코드값 및 뒷자리 코드값을 상기 제 2 코드값으로부터 추출하고, 상기 앞자리 코드값 및 상기 뒷자리 코드값을 포함하는 상기 제 2 해싱 색인 값을 생성하는 제 2 해싱 색인 값 생성부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 17

제 12항에 있어서,

상기 튜플 블록 필터 검색부는,

상기 제 1 양성 비트의 길이 및 상기 제 2 양성 비트의 길이를 조합하여 상기 하나 이상의 튜플을 생성하는 튜플 생성부;

상기 하나 이상의 튜플 중에서 상기 미리 생성된 룰 셋에 저장된 상기 튜플 리스트에 존재하는 튜플을 선별하는 튜플 선별부; 및

상기 선별된 튜플 각각에 대하여 제 3 해싱 색인 값을 생성하고, 상기 생성된 제 3 해싱 색인 값에 해당하는 상기 튜플 블록 필터의 비트벡터를 확인하며, 상기 확인된 비트벡터를 토대로 각 튜플이 양성인지 음성인지를 판단함으로써, 상기 양성 튜플을 추출하는 양성 튜플 판단 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 18

제 17항에 있어서,

상기 양성 튜플 판단 추출부는,

상기 선별된 튜플 각각에 대하여, 각 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 제 3 연결 비트를 생성하고, 상기 생성된 제 3 연결 비트로부터 기 정의된 제 3 코드값 비트 수를 갖는 제 3 코드값을 생성하며, 상기 생성된 제 3 코드값으로부터 상기 제 3 해싱 색인 값을 생성하는 제 3 해싱부;

상기 생성된 제 3 해싱 색인 값에 해당하는 상기 튜플 블록 필터의 비트벡터를 확인하여, 상기 확인된 비트벡터를 토대로 각 튜플이 양성인지 음성인지를 판단하는 양성 튜플 판단부; 및

상기 선별된 튜플 중에서 양성으로 판단된 튜플을 상기 양성 튜플로서 추출하는 양성 튜플 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 19

제 18항에 있어서,

상기 제 3 해싱부는,

상기 선별된 튜플 각각에 대하여, 각 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 상기 제 3 연결 비트를 생성하는 제 3 연결 비트 생성부;

상기 생성된 제 3 연결 비트로부터 상기 기 정의된 제 3 코드값 비트 수를 갖는 상기 제 3 코드값을 생성하는 제 3 코드값 생성부; 및

기 정의된 제 3 해싱 색인 값 비트 수만큼의 앞자리 코드값 및 뒷자리 코드값을 상기 생성된 제 3 코드값으로부터 추출하고, 상기 앞자리 코드값 및 상기 뒷자리 코드값을 포함하는 상기 제 3 해싱 색인 값을 생성하는 제 3 해싱 색인 값 생성부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 20

제 12항에 있어서,

상기 튜플 해시 테이블 검색부는,

상기 양성 튜플 각각에 대하여, 각 양성 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 제 4 연결 비트를 생성하고, 상기 생성된 제 4 연결 비트로부터 기 정의된 제 4 코드값 비트 수를 갖는 제 4 코드값을 생성하며, 상기 생성된 제 4 코드값으로부터 제 4 해싱 색인 값을 생성하는 제 4 해싱부; 및

상기 제 4 해싱 색인 값이 가리키는 상기 튜플 해시 테이블의 위치에서 패킷 분류를 위한 상기 룰을 추출하는 룰 추출부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 21

제 20항에 있어서,

상기 제 4 해싱부는,

상기 양성 튜플 각각에 대하여, 각 양성 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 상기 제 4 연결 비트를 생성하는 제 4 연결 비트 생성부;

상기 생성된 제 4 연결 비트로부터 상기 기 정의된 제 4 코드값 비트 수를 갖는 상기 제 4 코드값을 생성하는 제 4 코드값 생성부; 및

기 정의된 제 4 해싱 색인 값 비트 수만큼의 뒷자리 코드값을 상기 생성된 제 4 코드값으로부터 추출하고, 상기 뒷자리 코드값을 포함하는 상기 제 4 해싱 색인 값을 생성하는 제 4 해싱 색인 값 생성부

를 포함하는 것을 특징으로 하는 패킷 분류 장치.

청구항 22

제 12항에 있어서,

상기 근원지 블록 필터, 상기 목적지 블록 필터 및 튜플 블록 필터는,

프리픽스 길이의 구분없이 모든 프리픽스에 대하여, 동일한 해싱 함수를 이용하여 얻어진 해싱 색인 값에 해당 하는 비트벡터를 특정 값으로 프로그래밍하고, 프로그래밍 시 사용된 프리픽스에 대한 프리픽스 길이는 별도로 저장하는 것을 특징으로 하는 패킷 분류 장치.

청구항 23

입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 프리픽스 입력 단계;

기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블록 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블록 필터 검색 단계;

기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블록 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블록 필터 검색 단계;

상기 제 1 양성 비트 및 상기 제 2 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 블록 필터를 검색하여 양성으로 판단된 양성 튜플을 추출하는 튜플 블록 필터 검색 단계; 및

상기 양성 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색 단계

를 포함하는 것을 특징으로 하는 패킷 분류 방법.

청구항 24

근원지 프리픽스에 대한 정보를 저장하는 근원지 블록 필터;

목적지 프리픽스에 대한 정보를 저장하는 목적지 블룸 필터; 및

입력 패킷으로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 상기 근원지 블룸 필터 및 상기 목적지 블룸 필터에 각각 병렬로 통과시켜, 상기 입력 패킷에 대한 해시 테이블로의 접근 횟수를 제어하는 제어부

를 포함하는 것을 특징으로 하는 해시 테이블 접근 횟수를 제어하는 패킷 분류 장치.

명세서

발명의 상세한 설명

기술 분야

[0001] 본 발명의 일 실시예는 패킷 분류 장치 및 그 방법에 관한 것이다. 더욱 상세하게는, 메모리 요구량 및 메모리 접근 횟수를 줄임으로써, 더욱 효율적이고 신속한 패킷 분류를 가능하게 하는 패킷 분류 장치 및 그 방법에 관한 것이다.

배경 기술

[0002] 인터넷 사용자 수의 급격한 증가뿐만 아니라, 다양한 애플리케이션의 등장과 이러한 애플리케이션에 대한 사용자들의 수요가 급격히 증가하고 있다. 이러한 애플리케이션들은 인터넷 프로토콜을 기반으로 하고 있으며, 서로 다른 품질 보장(QoS: Quality of Service) 수준을 요구한다. 그러므로 모든 패킷에 대하여 동일한 방식의 최선 지원(BE: Best-Effort) 서비스를 제공하여 애플리케이션의 다양성에 따른 품질을 보장하지 못하던 기존의 라우터들은 다양한 레벨의 품질 보장 서비스를 새로이 제공할 수 있어야 한다.

[0003] 다양한 애플리케이션에 대한 품질 보장을 위해서, 라우터는 입력 패킷의 헤더를 이용하여 입력 패킷이 속하는 클래스를 찾아 주고 각 클래스에 해당하는 적절한 서비스를 제공해 주어야 하는데, 입력 패킷이 속하는 클래스를 찾아주는 작업을 "패킷 분류(Packet Classification)"라고 한다.

[0004] 이러한 패킷 분류는 목적지 IP 주소(Destination IP Address), 근원지 IP 주소(Source IP Address), 목적지 포트 번호(Destination Port Number), 근원지 포트 번호(Source Port Number) 및 프로토콜 정보(Protocol Information)에 해당하는 5개의 필드를 사용하여 이루어진다. 패킷 분류를 위하여 라우터는 미리 정의된 룰(Rule)과 일치하는 룰들을 찾고, 그 중 최우선순위를 갖는 룰을 찾아내는 다차원 검색을 수행하여야 한다.

[0005] 전술한 패킷 분류는 다음과 같은 어려움을 가지고 있다. 첫 번째로, 각 필드에 따라 각각 다른 검색 방법을 사용하면서 이러한 다차원 검색을 동시에 진행하여 모든 필드에 대하여 일치하는 룰을 찾는 복잡한 검색을 요구한다는 점이다. 두 번째로, 검색을 통하여 일치하는 룰의 개수가 한 개 이상일 경우에, 일치하는 모든 룰을 검색하여 그 가운데서 최우선순위의 룰을 찾아내는 과정이 부가적으로 이루어져야 한다. 마지막으로 라우터의 모든 입력 패킷에 대하여 패킷이 입력되는 선속도(wire-speed)로 빠르게 처리해야 한다는 점에서 어려움을 가지고 있다.

[0006] 또한 패킷 분류에 사용되는 5개의 필드는 각각 완전 일치(exact match), 영역 일치(range match), 프리픽스 일치(prefix match)라는 각기 다른 방법을 이용하여 일치 여부를 판단하기 때문에, 검색이 복잡해지고 속도도 느려진다.

[0007] 따라서, 모든 필드를 검색하는 대신, 일부 필드만을 이용한 튜플(tuple) 구조를 정의하여 튜플 공간을 검색할 때 모든 필드를 검색했을 경우와 성능이 동일하다면 튜플 공간을 이용하여 검색하는 것이 보다 효율적이며 이에 대한 연구가 활발히 진행되어 왔다.

[0008] 이렇게 진행되고 있는 튜플 공간을 이용한 패킷 분류 알고리즘은, 튜플 공간을 각 필드 원소의 길이들의 조합으로 정의 내린다. 여기서, 튜플 공간(T)은 5개의 필드를 가지는 벡터로 표현될 수 있는데, 예를 들어, 만약에 $T = (24, 32, 8, 0, 16)$ 이라면 근원지 프리픽스(Source Prefix)의 길이는 24 비트, 목적지 프리픽스(Destination Prefix)의 길이는 32 비트, 근원지 포트 번호는 8 비트의 길이를 갖는다는 등의 정보를 나타내는 5차원 벡터이다. 대개의 경우, 근원지 프리픽스와 목적지 프리픽스의 길이의 조합을 이용하여 2차원 튜플 공간을 정의한다. 그 이유는 두 프리픽스의 길이를 이용하여 조합한 튜플 공간에 대해 각 튜플에 포함되는 룰의 최대 개수가 다른 필드의 조합들보다 상대적으로 적기 때문이다. 프리픽스의 길이들의 조합으로 정의된 튜플은 $33 \times 33 = 1089$ 개의 튜플을 만들어 낼 수 있어, 최악의 경우 1089 개의 튜플을 모두 검색해야 하지만 실제로 대부분의 경우 모든 튜

플들이 룰의 정보를 갖는 것이 아니라 일부 튜플에 룰 정보가 몰려 있기 때문에 룰 정보를 가지지 않는 튜플들을 제거한 후 패킷 분류를 수행한다면 보다 효율적인 검색을 할 수 있다. 이러한 특성 때문에 불필요한 튜플을 제거하는 효율적인 튜플 제거(Tuple Pruning) 알고리즘이 연구되고 있다.

[0009] 이러한 연구중인 패킷 분류를 위한 종래의 튜플 제거 알고리즘의 구조는 도 1에 도시한다. 도 1에 도시된 종래의 튜플 제거 알고리즘의 구조에서는 모든 근원지 프리픽스와 목적지 프리픽스에 대하여 각 프리픽스가 네스팅하고 있는 다른 프리픽스의 길이들을 미리 계산하여 저장한다. 표 1에 간단한 룰 셋의 예를 보였다.

표 1

룰 번호	근원지 프리픽스	목적지 프리픽스	길이에 관한 튜플
R1	1*	10*	(1,2)
R2	00*	11*	(2,2)
R3	*	100*	(0,3)
R4	111*	0*	(3,1)
R5	011*	111*	(3,3)

[0011] 예를 들어, 표 1의 목적지 프리픽스의 예에서, 프리픽스 100*에는 길이 2가 미리 선 계산되어 저장된다. 이는 프리픽스 100*에는 길이가 2인 프리픽스인 10*이 네스팅(Nesting) 되어 있다는 의미이다. 또한 룰 셋(Rule Set)에 존재하는 근원지 프리픽스와 목적지 프리픽스가 각각 가지고 있는 길이에 관한 튜플 리스트(Tuple List)를 미리 작성한다. 표 1의 예에서의 튜플 리스트는, 표 1의 마지막 칼럼(Column)에 보였다.

[0012] 검색과정을 도 1을 참조하여 살펴보면, 입력으로 들어온 패킷의 근원지 IP 주소(Source IP Address)에 대하여 검색을 진행하여 최장길이 일치 프리픽스(LPM: Longest Matching Prefix)를 찾는다. 또한 이러한 근원지 LPM 룩업(Lookup) 과정과 병렬적으로, 목적지 IP 주소(Destination IP Address)에 대해서도 동일한 과정으로서 목적지 LPM 룩업(Lookup) 과정을 진행한다. 위의 과정에서 검색된 최장길이 프리픽스들에 대하여 네스팅되어 있는 길이들이 선 계산되어 저장되어 있으므로, 네스팅 되어 있는 길이들을 서로 교차(Cross-Producting) 하여 튜플을 생성한다. 이 튜플들 중 미리 작성된 튜플 리스트에 속하는 튜플들만을 뽑아내면, 패킷 분류를 위하여 검색되어야 할 튜플 공간이 결정되고, 이 튜플들에 대하여 선형검색을 적용하여 입력된 패킷의 헤더와 일치하는 룰을 찾아낸다.

[0013] 전술한 방식의 종래 튜플 제거 알고리즘은, 빠른 검색 성능을 보이지만, 아래와 같은 몇 가지 문제점을 가진다. 첫째, 모든 근원지 프리픽스와 목적지 프리픽스들의 네스팅 관계를 미리 파악하여 저장하고 있어야 하는 문제점이 있다. 둘째, 근원지 IP 주소와 목적지 IP 주소에 대하여 최장 길이 일치 검색을 진행하여야 하며, 이때 빈번한 메모리 접근이 발생하는 문제점이 있다.

발명의 내용

[0014] 이러한 배경에서, 본 발명의 실시예의 목적은, 메모리 요구량 및 메모리 접근 횟수를 줄여 더욱 효율적이고 신속한 패킷 분류를 가능하게 하는 데 있다.

[0015] 본 발명은, 근원지 프리픽스에 대한 정보를 저장하는 근원지 블룸 필터; 목적지 프리픽스에 대한 정보를 저장하는 목적지 블룸 필터; 및 입력 패킷으로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 상기 근원지 블룸 필터 및 상기 목적지 블룸 필터에 각각 병렬로 통과시켜, 상기 입력 패킷에 대한 해시 테이블로의 접근 횟수를 제어하는 제어부를 포함하는 것을 특징으로 하는 패킷 분류 장치를 제공한다.

[0016] 본 발명의 일 실시예는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력 받는 프리픽스 입력부; 기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블룸 필터 검색부; 상기 제 1 양성 비트를 토대로 근원지 해시 테이블을 검색하여 상기 제 1 양성 비트 중에서 참 양성으로 판단된 제 1 참 양성 비트를 추출하는 근원지 해시 테이블 검색부; 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블룸 필터 검색부; 상기 제 2 양성 비트를 토대로 목적지 해시 테이블을 검색하여 상기 제 2 양성 비트 중에서 참 양성으로 판단된 제 2 참 양성 비트를 추출하는 목적지 해시 테이블 검색부; 및 상기

제 1 참 양성 비트 및 상기 제 2 참 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색부를 포함하는 것을 특징으로 하는 패킷 분류 장치를 제공한다.

[0017] 또한, 본 발명의 일 실시예는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 프리픽스 입력 단계; 기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블룸 필터 검색 단계; 상기 제 1 양성 비트를 토대로 근원지 해시 테이블을 검색하여 상기 제 1 양성 비트 중에서 참 양성으로 판단된 제 1 참 양성 비트를 추출하는 근원지 해시 테이블 검색 단계; 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블룸 필터 검색 단계; 상기 제 2 양성 비트를 토대로 목적지 해시 테이블을 검색하여 상기 제 2 양성 비트 중에서 참 양성으로 판단된 제 2 참 양성 비트를 추출하는 목적지 해시 테이블 검색 단계; 및 상기 제 1 참 양성 비트 및 상기 제 2 참 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색 단계를 포함하는 것을 특징으로 하는 패킷 분류 방법을 제공한다.

[0018] 또한, 본 발명의 다른 실시예는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 프리픽스 입력부; 기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블룸 필터 검색부; 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블룸 필터 검색부; 상기 제 1 양성 비트 및 상기 제 2 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 블룸 필터를 검색하여 양성으로 판단된 양성 튜플을 추출하는 튜플 블룸 필터 검색부; 및 상기 양성 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색부를 포함하는 것을 특징으로 하는 패킷 분류 장치를 제공한다.

[0019] 또한, 본 발명의 다른 실시예는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 프리픽스 입력 단계; 기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 상기 근원지 프리픽스로부터 추출하고, 상기 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 상기 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 근원지 블룸 필터 검색 단계; 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 상기 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 상기 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 목적지 블룸 필터 검색 단계; 상기 제 1 양성 비트 및 상기 제 2 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 상기 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 블룸 필터를 검색하여 양성으로 판단된 양성 튜플을 추출하는 튜플 블룸 필터 검색 단계; 및 상기 양성 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 튜플 해시 테이블 검색 단계를 포함하는 것을 특징으로 하는 패킷 분류 방법을 제공한다.

[0020] 이상에서 설명한 바와 같이 본 발명의 실시예에 의하면, 메모리 요구량 및 메모리 접근 횟수를 줄임으로써, 라우터 등의 패킷 분류 장치에 부하를 주지않으면서 더욱 효율적이고 신속한 패킷 분류를 가능하게 하는 효과가 있다.

[0021] 또한 본 발명의 실시예에 의하면, 전술한 효율적이고 신속한 패킷 분류를 통해, 입력 패킷에 대한 해당 서비스 품질(QoS: Quality of Service)을 더욱 신속히 보장해줄 수 있는 효과도 있다.

발명의 실시를 위한 구체적인 내용

[0022] 본 명세서에서는, 인터넷 라우터 내에서 빠르게 입력되는 패킷을 선 속도로 처리할 수 있는 패킷 분류 방법과 이를 수행하는 패킷 분류 장치를 개시한다.

- [0023] 개시된 패킷 분류 방법은 bloom 필터를 선-필터로 사용하여 불필요한 튜플들을 제거하여 적은 메모리 공간을 차지하면서 적은 메모리 접근 횟수로 패킷을 분류할 수 있는 알고리즘이다. 칩 내부에 저장될 수 있는 적은 메모리 공간을 요구하는 bloom 필터는 필터의 입력 값이 필터에 미리 저장된 정보인지 아닌지를 빠르게 판단하여 불필요한 검색들을 걸러 내어준다. 이러한 bloom 필터의 특성을 이용하여, 룰 셋의 근원지와 목적지 프리픽스의 정보를 bloom 필터에 저장하고 패킷이 입력되었을 때 각 목적지, 근원지 bloom 필터를 병렬로 통과시켜 입력 패킷에 대한 칩 외부의 해시 테이블 접근 횟수를 효과적으로 수를 줄일 수 있도록 한 것이다.
- [0024] 이러한 패킷 분류 방법을 제공하는 패킷 분류 장치는, 근원지 프리픽스에 대한 정보를 저장하는 근원지 bloom 필터와, 목적지 프리픽스에 대한 정보를 저장하는 목적지 bloom 필터와, 입력 패킷으로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 상기 근원지 bloom 필터 및 상기 목적지 bloom 필터에 각각 병렬로 통과시켜, 상기 입력 패킷에 대한 해시 테이블로의 접근 횟수를 제어하는 제어부 등을 포함한다.
- [0025] 이하에서는, 전술한 패킷 분류 방법 및 패킷 분류 장치에 대한 일부 실시예들을 예시적인 도면을 통해 상세하게 설명한다. 각 도면의 구성요소들에 참조부호를 부가함에 있어서, 동일한 구성요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 부호를 가지도록 하고 있음에 유의해야 한다. 또한, 본 발명을 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략한다.
- [0026] 또한, 본 발명의 구성 요소를 설명하는 데 있어서, 제 1, 제 2, 제 3, 제 4, A, B, (a), (b) 등의 용어를 사용할 수 있다. 이러한 용어는 그 구성 요소를 다른 구성 요소와 구별하기 위한 것일 뿐, 그 용어에 의해 해당 구성 요소의 본질이나 차례 또는 순서 등이 한정되지 않는다. 어떤 구성 요소가 다른 구성요소에 "연결", "결합" 또는 "접속"된다고 기재된 경우, 그 구성 요소는 그 다른 구성요소에 직접적으로 연결되거나 또는 접속될 수 있지만, 각 구성 요소 사이에 또 다른 구성 요소가 "연결", "결합" 또는 "접속"될 수도 있다고 이해되어야 할 것이다.
- [0027] **본 발명의 일 실시예 (알고리즘 1)**
- [0028] 도 2는 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에 대한 블록 구성도이다.
- [0029] 도 2를 참조하면, 본 발명의 일 실시예에 따른 패킷 분류 장치(200)는 프리픽스 입력부(210), 근원지 bloom 필터 검색부(220), 근원지 해시 테이블 검색부(230), 목적지 bloom 필터 검색부(240), 목적지 해시 테이블 검색부(250) 및 튜플 해시 테이블 검색부(260) 등을 포함한다.
- [0030] 프리픽스 입력부(210)는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스(Source Prefix) 및 목적지 프리픽스(Destination Prefix)를 입력받는다.
- [0031] 이렇게 입력된 근원지 프리픽스 및 목적지 프리픽스는, 근원지 bloom 필터 검색부(220) 및 목적지 bloom 필터 검색부(240)로 각각 병렬적으로 전달된다.
- [0032] 또한, 입력 패킷의 헤더 필드 정보 내에는, 근원지 IP 주소(Source IP Address), 목적지 IP 주소(Destination IP Address), 근원지 포트 넘버(Source Port Number), 목적지 포트 넘버(Destination Port Number) 및 프로토콜 정보(Protocol Information)에 해당하는 5개의 필드를 포함할 수 있으며, 근원지 IP 주소 및 목적지 IP 주소로부터 근원지 프리픽스 및 목적지 프리픽스를 각각 확인한다.
- [0033] 근원지 bloom 필터 검색부(220)는, 기 저장된 근원지 프리픽스 길이(UL: Unique Length)에 따라 하나 이상의 제 1 비트를 입력된 근원지 프리픽스로부터 추출하고, 추출된 제 1 비트를 토대로 근원지 bloom 필터(Src. BF: Source Bloom Filter)를 검색하여 제 1 비트 중에서 양성(Positive)으로 판단된 제 1 양성 비트를 추출한다.
- [0034] 근원지 해시 테이블 검색부(230)는, 근원지 bloom 필터 검색부(220)에서 추출된 제 1 양성 비트를 토대로 근원지 해시 테이블(Source Hash Table)을 검색하여 제 1 양성 비트 중에서 참 양성(True Positive)으로 판단된 제 1 참 양성 비트를 추출한다.
- [0035] 목적지 bloom 필터 검색부(240)는, 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 입력된 목적지 프리픽스로부터 추출하고, 추출된 제 2 비트를 토대로 목적지 bloom 필터(Dst. BF: Destination Bloom Filter)를 검색하여 제 2 비트 중에서 양성(Positive)으로 판단된 제 2 양성 비트를 추출한다.
- [0036] 목적지 해시 테이블 검색부(250)는, 목적지 bloom 필터 검색부(240)에서 추출된 제 2 양성 비트를 토대로 목적지 해시 테이블(Destination Hash Table)을 검색하여 제 2 양성 비트 중에서 참 양성(True Positive)으로 판단된

제 2 참 양성 비트를 추출한다.

- [0037] 튜플 해시 테이블 검색부(260)는, 근원지 해시 테이블 검색부(230)에서 추출된 제 1 참 양성 비트 및 목적지 해시 테이블 검색부(250)에서 추출된 제 2 참 양성 비트를 토대로 하나 이상의 튜플(Tuple)을 생성하고, 이렇게 하나 이상의 튜플 중에서 미리 생성된 룰 셋(Rule Set)에 저장된 튜플 리스트(Tuple List)에 존재하는 튜플에 대하여 튜플 해시 테이블(Tuple Hash Table)을 검색하여 패킷 분류를 위한 룰을 추출한다.
- [0038] 위에서 언급한 근원지 bloom 필터 및 목적지 bloom 필터 각각은, 모든 프리픽스 길이(모든 근원지 프리픽스 길이 및 모든 목적지 프리픽스 길이)마다 개별적으로 할당되는 '개별 bloom 필터'를 프리픽스 길이(근원지 프리픽스 길이 및 목적지 프리픽스 길이)의 개수만큼 포함하는 것일 수도 있고, 프리픽스 길이의 구분없이 모든 프리픽스에 대하여 동일한 해싱 함수를 이용하여 얻어진 해싱 색인 값에 해당하는 비트벡터를 특정 값으로 프로그래밍하고, 프로그래밍 시 사용된 프리픽스에 대한 프리픽스 길이는 별도로 저장하는 '통합 bloom 필터(All-Length Bloom Filter)'일 수 있다.
- [0039] 이하 설명에서는, 근원지 bloom 필터 및 목적지 bloom 필터 각각은, 개별 bloom 필터를 프리픽스 길이의 개수만큼 포함하는 것에 비해, bloom 필터의 프로그래밍 및 쿼리 과정이 더욱 단순화하고 거짓 양성을 조절하기가 더욱 용이한 통합 bloom 필터를 사용한 것으로 가정하여 설명한다.
- [0040] 이상에서 기술한 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에 포함된 각 구성요소에 대해서는, 도 3 내지 도 8을 참조하여 더욱 상세하게 설명한다.
- [0041] 도 3은 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에서, 근원지 bloom 필터 검색부(220)를 나타낸 도면이다.
- [0042] 도 3을 참조하면, 기술한 근원지 bloom 필터 검색부(220)는, 프리픽스 입력부(210)로부터 근원지 프리픽스를 전달받아, 전달받은 근원지 프리픽스를 이용하여 근원지 bloom 필터를 통과시켜 근원지 bloom 필터의 제 1 양성 비트, 즉 일치 가능성이 있는 길이에 관한 정보를 추출한다. 이때 추출한 제 1 양성 비트는 참 양성(True Positive)인 제 1 참 양성 비트 및 거짓 양성(False Positive)인 제 1 거짓 양성 비트 중 하나 이상을 포함할 수 있다.
- [0043] 이러한 근원지 bloom 필터 검색부(220)는 기술한 바와 같이 근원지 프리픽스를 전달받아 제 1 양성 비트를 추출하기 위해서, 제 1 비트 추출부(310), 제 1 해싱부(320) 및 제 1 양성 비트 추출부(330) 등을 포함한다.
- [0044] 제 1 비트 추출부(310)는, 기 저장된 근원지 프리픽스 길이(예: 근원지 프리픽스 길이(UL: Unique Length)=0,1,2,3 ...)에 따라 하나 이상의 제 1 비트를 입력된 근원지 프리픽스로부터 추출한다.
- [0045] 제 1 해싱부(320)는, 추출된 제 1 비트 각각에 대하여 기 정의된 제 1 코드값 비트 수(예: 8 비트 등)를 갖는 제 1 코드값을 생성하고, 생성된 제 1 코드값으로부터 제 1 해싱 색인 값을 생성한다.
- [0046] 제 1 양성 비트 추출부(330)는, 생성된 제 1 해싱 색인 값에 해당하는 근원지 bloom 필터의 비트벡터(Bit-Vector)를 확인하고, 확인된 비트벡터를 토대로 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출한다. 예를 들어, 제 1 해싱 색인 값 각각에 대하여 확인된 비트벡터가 모두 1인 경우인 제 1 비트를 양성(Positive)으로 판단할 수 있고, 확인된 비트벡터가 하나라도 1이 아닌 경우인 제 1 비트를 음성(Negative)으로 판단할 수 있다.
- [0047] 기술한 제 1 해싱부(320)는 따라오는 도 4를 참조하여 더욱 상세하게 설명한다.
- [0048] 도 4는 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에서, 근원지 bloom 필터 검색부(220)에 포함된 제 1 해싱부(320)를 나타낸 도면이다.
- [0049] 도 4를 참조하면, 제 1 해싱부(320)는, 입력된 근원지 프리픽스로부터 기 저장된 근원지 프리픽스 길이(예: 근원지 프리픽스 길이(UL: Unique Length)=0,1,2,3 ...)만큼 각각 추출된 하나 이상의 제 1 비트 각각에 대하여 제 1 코드값 비트 수(예: 8 비트 등)를 갖는 제 1 코드값을 생성하는 제 1 코드값 생성부(410)와, 기 정의된 제 1 해싱 색인 값 비트 수(예: 3 비트 등)만큼의 앞자리 코드값 및 뒷자리 코드값을 제 1 코드값으로부터 추출하고, 이렇게 추출된 앞자리 코드값 및 뒷자리 코드값을 포함하는 제 1 해싱 색인 값(즉, 2개의 해싱 색인 값)을 제 1 해싱 색인 값 생성부(420) 등을 포함한다.
- [0050] 위에서 언급한 제 1 코드값 생성부(410)는, 미리 정의된 제 1 코드값 비트 수를 갖는 제 1 코드값을 생성하기 위해서, 제 1 코드값 비트 수에 해당하는 CRC 생성기를 이용할 수 있다. 예를 들어, 미리 정의된 제 1 코드값

비트 수가 8 비트인 경우, 8 비트 길이의 제 1 코드값을 생성할 수 있는 CRC-8 생성기를 이용할 수 있다. 이때의 제 1 코드값을 제 1 CRC 코드값이라고도 한다.

- [0051] 전술한 바와 같이, 근원지 블록 필터 검색부(220)에서 양성으로 판단된 제 1 양성 비트를 추출한 이후, 근원지 해시 테이블 검색부(230)는, 이렇게 추출된 제 1 양성 비트가 정말로 참 양성인지를 판단하여 참 양성으로 판단되는 제 1 양성 비트를 제 1 참 양성 비트로서 추출한다. 즉, 근원지 해시 테이블 검색부(230)는 제 1 양성 비트가 정말로 근원지 블록 필터에 저장된 값들인지를 알아보기 위하여 근원지 해시 테이블에 접근하여 검색을 수행한다.
- [0052] 더욱 상세하게 설명하면, 근원지 해시 테이블 검색부(230)는, 제 1 양성 비트 각각에 대하여 제 1 해싱부(320)에서 생성된 제 1 해싱 색인 값에 포함된 뒷자리 코드값을 이용해서 근원지 해시 테이블을 검색하여, 제 1 양성 비트 중에서 근원지 해시 테이블의 해당 엔트리에 존재하는 제 1 양성 비트를 제 1 참 양성 비트로서 판단하여 추출하는 것이다.
- [0053] 전술한 바와 같이, 근원지 프리픽스는, 근원지 블록 필터 검색부(220) 및 근원지 해시 테이블 검색부(230)를 거쳐 제 1 참 양성 비트가 추출된다.
- [0054] 이와 동일한 방식과 병렬적인 방식으로, 목적지 프리픽스도, 목적지 블록 필터 검색부(240) 및 목적지 해시 테이블 검색부(250)를 거쳐 제 2 참 양성 비트가 추출된다. 이를 따라오는 도 5 및 도 6을 참조하여 상세하게 설명한다.
- [0055] 도 5는 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에서, 목적지 블록 필터 검색부(240)를 나타낸 도면이다.
- [0056] 도 5를 참조하면, 목적지 블록 필터 검색부(240)는, 프리픽스 입력부(210)로부터 목적지 프리픽스를 전달받아, 전달받은 목적지 프리픽스를 이용하여 목적지 블록 필터를 통과시켜 목적지 블록 필터의 제 2 양성 비트, 즉 일치 가능성이 있는 길이에 관한 정보를 추출한다. 이때 추출한 제 2 양성 비트는 참 양성인 제 2 참 양성 비트 및 거짓 양성인 제 1 거짓 양성 비트 중 하나 이상을 포함할 수 있다.
- [0057] 이러한 목적지 블록 필터 검색부(240)는 전술한 바와 같이 목적지 프리픽스를 전달받아 제 2 양성 비트를 추출하기 위해서, 제 2 비트 추출부(510), 제 2 해싱부(520) 및 제 2 양성 비트 추출부(530) 등을 포함한다.
- [0058] 제 2 비트 추출부(510)는, 기 저장된 목적지 프리픽스 길이(예: 목적지 프리픽스 길이(UL: Unique Length)=0,1,2,3 ...)에 따라 하나 이상의 제 2 비트를 목적지 프리픽스로부터 추출한다.
- [0059] 제 2 해싱부(520)는, 추출된 제 2 비트 각각에 대하여 기 정의된 제 2 코드값 비트 수(예: 8 비트 등)를 갖는 제 2 코드값을 생성하고, 이렇게 생성된 제 2 코드값으로부터 제 2 해싱 색인 값을 생성한다.
- [0060] 제 2 양성 비트 추출부(530)는, 생성된 제 2 해싱 색인 값에 해당하는 목적지 블록 필터의 비트벡터를 확인하고, 확인된 비트벡터를 토대로 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출한다. 예를 들어, 제 2 해싱 색인 값 각각에 대하여 확인된 비트벡터가 모두 1인 경우인 제 2 비트를 양성으로 판단할 수 있고, 확인된 비트벡터가 하나라도 1이 아닌 경우인 제 2 비트를 음성으로 판단할 수 있다.
- [0061] 전술한 제 2 해싱부(520)는, 따라오는 도 6을 참조하여 더욱 상세하게 설명한다.
- [0062] 도 6은 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에서, 목적지 블록 필터 검색부(240)에 포함된 제 2 해싱부(520)를 나타낸 도면이다.
- [0063] 도 6을 참조하면, 제 2 해싱부(520)는, 입력된 목적지 프리픽스로부터 기 저장된 목적지 프리픽스 길이(예: 목적지 프리픽스 길이(UL: Unique Length)=0,1,2,3 ...)만큼 각각 추출된 제 2 비트 각각에 대하여 제 2 코드값 비트 수(예: 8 비트 등)를 갖는 상기 제 2 코드값을 생성하는 제 2 코드값 생성부(610); 및 기 정의된 제 2 해싱 색인 값 비트 수(예: 3 비트 등)만큼의 앞자리 코드값 및 뒷자리 코드값을 제 2 코드값으로부터 추출하고, 추출된 앞자리 코드값 및 뒷자리 코드값을 포함하는 제 2 해싱 색인 값(즉, 2개의 해싱 색인 값)을 생성하는 제 2 해싱 색인 값 생성부(620) 등을 포함한다.
- [0064] 위에서 언급한 제 2 코드값 생성부(610)는, 미리 정의된 제 2 코드값 비트 수를 갖는 제 2 코드값을 생성하기 위해서, 제 2 코드값 비트 수에 해당하는 CRC 생성기를 이용할 수 있다. 예를 들어, 미리 정의된 제 2 코드값 비트 수가 8 비트인 경우, 8 비트 길이의 제 2 코드값을 생성할 수 있는 CRC-8 생성기를 이용할 수 있다. 이때의 제 2 코드값을 제 2 CRC 코드값이라고도 한다.

- [0065] 전술한 바와 같이, 목적지 블록 필터 검색부(240)에서 양성으로 판단된 제 2 양성 비트를 추출한 이후, 목적지 해시 테이블 검색부(250)는, 이렇게 추출된 제 2 양성 비트가 정말로 참 양성인지를 판단하여 참 양성으로 판단되는 제 2 양성 비트를 제 2 참 양성 비트로서 추출한다. 즉, 목적지 해시 테이블 검색부(250)는 제 2 양성 비트가 정말로 목적지 블록 필터에 저장된 값들인지를 알아보기 위하여 목적지 해시 테이블에 접근하여 검색을 수행한다.
- [0066] 더욱 상세하게 설명하면, 목적지 해시 테이블 검색부(250)는, 제 2 양성 비트 각각에 대하여 제 2 해싱부(520)에서 생성된 제 2 해싱 색인 값에 포함된 뒷자리 코드값을 이용해서 목적지 해시 테이블을 검색하여, 제 2 양성 비트 중에서 목적지 해시 테이블의 해당 엔트리에 존재하는 제 2 양성 비트를 제 2 참 양성 비트로서 판단하여 추출하는 것이다.
- [0067] 전술한 바와 같이, 근원지 블록 필터 검색부(220)/근원지 해시 테이블 검색부(230)와, 목적지 블록 필터 검색부(240)/목적지 해시 테이블 검색부(250)가 병렬적으로 수행되어, 제 1 참 양성 비트와 제 2 참 양성 비트를 추출하면, 전술한 튜플 해시 테이블 검색부(260)는, 제 1 참 양성 비트와 제 2 참 양성 비트를 이용하여 생성 가능한 모든 튜플을 만들어내고 생성된 튜플 가운데 룰 셋에 존재하는 튜플 리스트와 비교하여 튜플 리스트에 존재하는 튜플에 대해서만 튜플 해시 테이블을 검색함으로써, 패킷 분류를 위한 룰을 추출해낸다. 이러한 튜플 해시 테이블 검색부(260)는 이하 도 7을 참조하여 상세하게 설명한다.
- [0068] 도 7은 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에서, 튜플 해시 테이블 검색부(260)를 나타낸 도면이다.
- [0069] 도 7을 참조하면, 튜플 해시 테이블 검색부(260)는, 제 1 참 양성 비트의 길이 및 제 2 참 양성 비트의 길이를 교차하여 조합함으로써 하나 이상의 튜플을 생성하는 튜플 생성부(710); 이렇게 생성된 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하지 않는 튜플은 제거하고 존재하는 튜플만을 선별하는 튜플 선별부(720); 및 선별된 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 룰 추출부(730) 등을 포함한다.
- [0070] 전술한 룰 추출부(730)는, 이하 도 8을 참조하여 더욱 상세하게 설명한다.
- [0071] 도 8은 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에서, 튜플 해시 테이블 검색부(260)에 포함된 룰 추출부(730)를 나타낸 도면이다.
- [0072] 도 8을 참조하면, 이러한 룰 추출부(730)는, 튜플 선별부(720)에서 선별된 튜플의 생성시 이용된 제 1 참 양성 비트 및 제 2 참 양성 비트를 연결하여 연결 비트를 생성하고, 이렇게 생성된 연결 비트로부터 기 정의된 제 3 코드값 비트 수(예: 16 비트 등)를 갖는 제 3 코드값을 생성하고, 생성된 제 3 코드값으로부터 기 정의된 제 3 해싱 색인 값 비트 수(예: 3 비트) 만큼의 뒷자리 코드값을 포함하는 제 3 해싱 색인 값을 생성하는 제 3 해싱부(810); 및 생성된 제 3 해싱 색인 값을 이용하여 튜플 해시 테이블을 검색하여 룰을 추출하는 추출부(820) 등을 포함한다. 이렇게 추출된 룰은 입력 패킷과 비교되어 일치한다.
- [0073] 전술한 제 3 해싱부(810)는, 미리 정의된 제 3 코드값 비트 수를 갖는 제 3 코드값을 생성하기 위해서, 제 3 코드값 비트 수에 해당하는 CRC 생성기를 이용할 수 있다. 예를 들어, 미리 정의된 제 3 코드값 비트 수가 16 비트인 경우, 16 비트 길이의 제 3 코드값을 생성할 수 있는 CRC-16 생성기를 이용할 수 있다. 이때의 제 3 코드값을 제 3 CRC 코드값이라고도 한다.
- [0074] 이상에서 전술한 본 발명의 일 실시예에 따른 패킷 분류 장치(200)가 수행하는 패킷 분류 방법은 이하 도 9을 참조하여 설명한다.
- [0075] 도 9는 본 발명의 일 실시예에 따른 패킷 분류 방법에 대한 흐름도이다.
- [0076] 도 9를 참조하면, 본 발명의 일 실시예에 따른 패킷 분류 방법은, 프리픽스 입력 단계(S900), 근원지 블록 필터 검색 단계(S902), 근원지 해시 테이블 검색 단계(S904), 목적지 블록 필터 검색 단계(S906), 목적지 해시 테이블 검색 단계(S908) 및 튜플 해시 테이블 검색 단계(S910) 등을 포함한다.
- [0077] 프리픽스 입력 단계(S900)는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 단계이다.
- [0078] 근원지 블록 필터 검색 단계(S902)는, 기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 근원지 프리픽스로부터 추출하고, 제 1 비트를 토대로 근원지 블록 필터를 검색하여 제 1 비트 중에서 양성으로 판단된

제 1 양성 비트를 추출하는 단계이다.

- [0079] 근원지 해시 테이블 검색 단계(S904)는, 제 1 양성 비트를 토대로 근원지 해시 테이블을 검색하여 제 1 양성 비트 중에서 참 양성으로 판단된 제 1 참 양성 비트를 추출하는 단계이다.
- [0080] 목적지 블룸 필터 검색 단계(S906)는, 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 목적지 프리픽스로부터 추출하고, 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 단계이다.
- [0081] 목적지 해시 테이블 검색 단계(S908)는, 제 2 양성 비트를 토대로 목적지 해시 테이블을 검색하여 제 2 양성 비트 중에서 참 양성으로 판단된 제 2 참 양성 비트를 추출하는 단계이다.
- [0082] 튜플 해시 테이블 검색 단계(S910)는, 제 1 참 양성 비트 및 제 2 참 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 단계이다.
- [0083] 이상에서는 본 발명의 일 실시예에 따른 패킷 분류 장치(200) 및 패킷 분류 방법을 설명하였으며, 이러한 설명을 위해, 제 1 비트, 제 1 코드값, 제 1 해싱 색인 값, 제 1 양성 비트, 제 1 참 양성 비트, 제 2 비트, 제 2 코드값, 제 2 해싱 색인 값, 제 2 양성 비트, 제 2 참 양성 비트, 제 3 코드값, 제 3 해싱 색인 값 등과 같이 제 1, 제 2, 제 3 등의 용어를 사용하였다. 이때 이러한 용어는, 설명의 편의를 위해, 그 구성 요소 또는 정보를 다른 구성 요소 또는 정보와 구별하기 위한 것일 뿐, 그 용어에 의해 해당 구성 요소 또는 정보의 본질이나 차례 또는 순서 등이 한정되지 않는다.
- [0084] 이하에서는, 전술한 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에 의해 제공되는 패킷 분류 방법을 따라 오는 도 10 내지 도 14를 참조하여, 예시적으로 설명한다. 다만, 이하에서의 예시적인 설명에서는, 제 1 비트, 제 1 코드값, 제 1 해싱 색인 값, 제 1 양성 비트, 제 1 참 양성 비트, 제 2 비트, 제 2 코드값, 제 2 해싱 색인 값, 제 2 양성 비트, 제 2 참 양성 비트, 제 3 코드값, 제 3 해싱 색인 값 등과 같이 제 1, 제 2, 제 3 등의 전술한 바 있는 용어에 대하여, 그 용어가 의미하는 바가 동일한 경우, 제 1, 제 2, 제 3 등의 용어를 사용하지 않는다.
- [0085] 이하에서는 전술한 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에 의해 제공되는 패킷 분류 방법은, 블룸 필터를 이용하여 보다 효율적으로 불필요한 튜플들을 제거하여 패킷 분류를 하기 위한 패킷 분류 알고리즘으로서, 이러한 패킷 분류 알고리즘은 도 1에 보인 종래의 튜플 제거 알고리즘의 최장길이 일치 검색과정을 블룸 필터를 사용하여 수행하는 알고리즘이다.
- [0086] 전술한 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에 의해 제공되는 패킷 분류 방법에서는, 최장길이 일치(LPM: Longest Prefix Matching) 과정을 블룸 필터를 사용하여 대체하고, 주어진 룰 셋의 특정 필드에 존재하는 모든 길이에 대하여 순차적으로 블룸 필터(근원지 블룸 필터 및 목적지 블룸 필터)에 접근하여 일치 가능한 길이들을 모으므로, 존재하는 모든 프리픽스들에 대하여 네스팅 관계를 미리 선계산하여 저장할 필요가 없다.
- [0087] 표 1의 룰 셋에 대하여 프로그래밍 된 본 발명의 일 실시예에 따른 패킷 분류 알고리즘을 도 10에 보였다. 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서 사용하는 모든 블룸 필터(근원지 블룸 필터, 목적지 블룸 필터)는 통합 블룸 필터를 사용하였다.
- [0088] 먼저, 본 발명의 일 실시예에 따른 패킷 분류 알고리즘의 "프로그래밍 과정"을 표 1의 예를 이용하여 설명한다. 근원지 블룸 필터 및 목적지 블룸 필터를 프로그래밍하기 전에 근원지 블룸 필터 및 목적지 블룸 필터 각각의 모든 비트-벡터의 값을 0으로 초기화한다.
- [0089] 표 1에서 서로 다른 근원지 프리픽스는 *, 1*, 00*, 111*, 011* 이다. 블룸 필터를 프로그래밍하기 위하여, 이 근원지 프리픽스를 코드값 생성기에 입력되어 기 정의된 코드값 비트 수만큼의 코드값을 생성한다. 여기서, 코드값 생성기는, CRC 생성기가 이용될 수 있으며, 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서 일 예로서 이용되는 CRC-8 생성기의 구조를 도 11에서 예시적으로 나타낸다. 도 11에 보인 CRC-8 생성기(생성기)에 근원지 프리픽스를 입력하여 8-비트의 코드값을 얻어낸다. 각 근원지 프리픽스를 CRC-8 해싱 함수에 넣어 얻어진 코드값 및 근원지 블룸 필터를 위한 해싱 색인 값(예: 코드값의 처음 3-비트와 마지막 3-비트), 그리고 (근원지) 해시 테이블 색인 값(예: 코드값의 마지막 3-비트)를 표 2에 보였다. 표 2에서는 도 11에 보인 CRC-8 생성기가 11111100으로 초기화되어 있음을 가정하였다. 또한 룰 집합에 존재하는 근원지 프리픽스의 길이에 관한 집합 {0,1,2,3}을 저장한다. 여기서 *의 경우에는 와일드 카드(Wild Card)로 분류되어 모든 근원지 프리픽스

와 일치하므로 프로그래밍시 제외되지만 그 길이 정보인 0은 저장된다.

표 2

[0090]

개별 프리픽스		CRC-8 코드값	블룸 필터의 해싱 색인 값	해시 테이블의 해싱 색인 값
근원지 필터	1*	11001111	6,7	7
	00*	00111111	1,7	7
	011*	11110110	7,6	6
	111*	10011010	4,2	2
목적지 필터	10*	11100111	7,7	7
	11*	01010110	2,6	6
	100*	11110011	7,3	3
	0*	01111110	3,6	6
	111*	10011010	4,2	2

[0091]

이와 동일하게, 목적지 프리픽스들 10*, 11*, 100*, 0*, 111*을 이용하여 목적지 블룸 필터를 프로그래밍하고
를 집합에 존재하는 목적지 프리픽스의 길이들, 즉 {1,2,3}을 저장한 후 목적지 프리픽스의 (목적지) 해시 테이블
을 프로그래밍한다.

[0092]

도 12는 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서의 근원지 블룸 필터 및 목적지 블룸 필터를 예시
적으로 나타내고, 도 13은 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서의 근원지 해시 테이블의 구조
(a) 및 목적지 해시 테이블의 구조(b)를 예시적으로 나타낸다. 이와 같은 과정을 거쳐 프로그래밍 된 블룸 필터
(근원지 블룸 필터, 목적지 블룸 필터)와 해시 테이블(근원지 해시 테이블, 목적지 해시 테이블)을 앞서 언급
한 도 10에 전체적으로 나타내었다.

[0093]

주어진 룰 들을 튜플 해시 테이블에 "저장"하는 방법은 근원지 프리픽스 길이와 목적지 프리픽스 길이를 가지고
튜플을 만드는데, 근원지 프리픽스 길이 및 목적지 프리픽스 길이 각각이 나타내는 길이만큼 근원지 프리픽스
및 목적지 프리픽스를 떼어내고, 떼어낸 근원지 프리픽스의 뒤에 떼어낸 목적지 프리픽스를 연결한다. 이 과정
에서 얻어진 문자열(즉, 연결 비트)을 CRC-16의 입력 값으로 하여 튜플 해시 테이블을 위한 해싱 색인 값을 얻
는다. 여기서 CRC-16의 초기값은 1111110011111100으로 가정하였다. 주어진 룰 셋에 대한 튜플 해시 테이블에
대한 해싱 색인 값을 표 3에 보였다. 표 3에 보인 튜플 해시 테이블의 해싱 색인 값에 의하여 저장된 튜플 해시
테이블 또한 앞서 설명한 도 10에 포함하였다.

표 3

[0094]

룰	튜플	연결 프리픽스	CRC 코드값	해시 테이블의 해싱 색인 값
R1	(1,2)	110	0011110010011001	1
R2	(2,2)	0011	1000100111000010	2
R3	(0,3)	100	1111111010011101	5
R4	(3,1)	1110	0101110001001000	0
R5	(3,3)	011111	0111110000111101	5

[0095]

표 3을 참조하면, 튜플 해시 테이블의 해싱 색인 값이 가리키는 튜플 해시 테이블 위치에 입력 값의 모든 정보,
즉 룰 넘버, 근원지 프리픽스, 목적지 프리픽스, 근원지 프리픽스 길이, 목적지 프리픽스 길이, 프로토콜 타입,
근원지 포트 범위, 목적지 포트 범위 등을 저장한다.

[0096]

즉, 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서는 해시 테이블의 색인을 위해 룰의 처음 두 필드인 근
원지 프리픽스와 목적지 프리픽스만이 사용되어, 룰의 모든 필드들은 처음 두 필드에 의하여 구하여진 해시 테
이블의 색인이 가리키는 곳에 저장되므로, 근원지 포트 번호와 목적지 포트 번호 필드는 다른 데이터 구조로의
변환의 필요 없이 있는 그대로 저장될 수 있는 장점이 있다. 도 14에서는 본 발명의 일 실시예에 따른 패킷 분
류 알고리즘에서의 튜플 해시 테이블의 구조를 도시한다. 또한 튜플의 길이 종류에 대한 정보를 별도의 리스트
에 저장한다.

[0097]

본 발명의 일 실시예에 따른 패킷 분류 방법에서 "검색 과정"은 다음의 몇 단계를 거친다. 첫째 단계로, 입력된

패킷에 대하여 근원지 주소필드와 목적지 주소필드의 정보를 각각의 블록 필터(근원지 블록 필터 및 목적지 블록 필터)를 병렬적으로 통과시켜 블록 필터의 양성 정보, 즉 일치 가능성이 있는 길이에 관한 정보를 얻어낸다. 이때 주어진 룰 셋에 존재하는 필드별 길이 리스트와 비교하여, 존재하는 길이들에 대하여 블록 필터에서의 쿼리를 진행한다. 이 결과는 거짓 양성을 포함하는 결과이다. 두 번째 단계로, 해시 테이블에 접근하여 거짓 양성을 제거한 참 양성의 결과, 즉 주어진 입력과 일치하는 프리픽스들의 길이 정보를 얻어내는 과정이 수행된다. 세 번째 단계로, 근원지 주소에 대하여 얻은 일치 길이 정보와 목적지 주소에 대하여 얻은 일치 길이 정보들을 서로 교차하여 튜플을 만든다. 네 번째 단계로, 세 번째 단계에서 생성된 튜플들 가운데 룰셋에 저장된 튜플들의 리스트와 비교하여 존재하지 않는 튜플들은 검색에서 제외시키고 나머지 튜플들에 대하여 튜플 해시 테이블 검색을 수행하여 최우선 순위의 룰을 찾아낸다.

[0098] 본 발명의 일 실시예에 따른 패킷 분류 방법에서 검색 과정을 예를 들어 설명하면 다음과 같다. 검색 과정의 예시적인 설명을 위해, 근원지 프리픽스가 "11101101"이고, 목적지 프리픽스가 "00011001"인 패킷이 입력으로 들어오는 것을 가정한다.

[0099] 각각의 프리픽스(근원지 프리픽스 및 목적지 프리픽스)에 대하여 근원지 블록 필터와 목적지 블록 필터를 병렬로 검색한다.

[0100] 근원지 블록 필터에 대한 검색의 경우, 미리 저장된 근원지 프리픽스 길이는 {0, 1, 2, 3}이므로, 길이 0을 제외하고 나머지 길이들에 대해 순차적으로 검색한다. 길이 0의 경우는 모든 입력과 일치하는 근원지 프리픽스이므로, 항상 블록 필터를 통과함을 가정한다. 길이 1에 해당하는 근원지 프리픽스(즉, 추출된 제 1 비트)인 "1*"을 CRC-8의 입력 값으로 넣으면 결과값인 코드값(즉, 제 1 코드값)은 "11001111"이고 해싱 색인 값(즉, 제 1 해싱 색인 값)은 "110(십진수로는 '6'임)"와 "111(십진수로는 '7'임)"이다. 이렇게 생성된 해싱 색인 값(즉, 제 1 해싱 색인 값)을 이용하여 근원지 블록 필터의 6번째와 7번째 비트벡터를 확인하면, 확인된 비트벡터가 모두 "1"이므로 양성의 결과를 얻는다. 이와 마찬가지로 길이 2에 해당하는 "11*" (즉, 제 1 비트)를 CRC-8 생성기에 입력하여 얻어진 CRC 코드값(즉, 제 1 코드값)은 "01010110"이므로 근원지 블록 필터의 해싱 색인 값(즉, 제 1 해싱 색인 값)은 "010(십진수: 2)"와 "110(십진수: 6)"이 얻어지고, 이를 이용하여 도 12의 근원지 블록 필터를 검색하면 양성임을 알 수 있다. 또한 길이 3에 해당하는 "111*" (즉, 제 1 비트)을 CRC-8 생성기에 입력하여 얻어진 코드값(즉, 제 1 코드값)은 "10011010"이고 이로부터 생성된 해싱 색인 값(즉, 제 1 해싱 색인 값)은 4, 2이고, 이러한 해싱 색인 값을 이용하여 근원지 블록 필터의 해당 비트벡터를 확인하면, 양성의 결과를 얻는다. 즉, 근원지 프리픽스 "11101101"로부터 미리 저장된 근원지 프리픽스 길이(1,2,3)만큼 추출된 제 1 비트인 "1*", "11*", "111*"는, 근원지 블록 필터 검색 과정을 거쳐 모두 양성인 것으로 판단되어 제 1 양성 비트로 추출된다.

[0101] 전술한 바와 같이, 근원지 블록 필터 검색 과정을 통해 제 1 양성 비트를 추출한 이후, 근원지 블록 필터의 검색 과정에서 양성으로 판단된 결과 값들(즉, 제 1 양성 비트)이 정말로 근원지 블록 필터에 저장된 값들인지를 알아보기 위하여 근원지 해시 테이블에 접근하여 근원지 해시 테이블 검색 과정을 수행한다.

[0102] 우선, 길이 1의 근원지 프리픽스인 "1*" (즉, 제 1 양성 비트)에 해당하는 CRC 코드 값인 "11001111"의 마지막 3-비트(즉, 기 정의된 제 1 해싱 색인 값 비트 수(3 비트)만큼의 뒷자리 코드값)인 "111(십진수: 7)"을 사용하여 근원지 해시 테이블의 7번지를 검색하면, "1*"는 참 양성임을 알 수 있다.

[0103] 길이 2의 근원지 프리픽스인 "11*" (즉, 제 1 양성 비트)의 CRC 코드 값인 "01010110"에서 얻어진 마지막 3-비트(즉, 기 정의된 제 1 해싱 색인 값 비트(3비트)만큼의 뒷자리 코드값)인 "110(십진수: 6)"을 해싱 색인 값으로 하여 이에 해당하는 근원지 해시 테이블의 엔트리에는 "11*"가 저장되어 있지 않으므로, "11*"는 거짓 양성이다.

[0104] 또한 길이 3의 근원지 프리픽스인 "111*" (즉, 제 1 양성 비트)의 코드 값인 "10011010"에서 얻어진 마지막 3-비트(즉, 뒷자리 코드값)인 "010(십진수: 2)"을 해싱 색인 값으로 하여 이에 해당하는 근원지 해시 테이블의 엔트리에는 "111*"가 저장되어 있으므로, "111*"는 참 양성임을 알 수 있다. 따라서, 근원지 해시 테이블 검색 과정을 거쳐, 제 1 양성 비트인 "1*", "11*", "111*" 중에서 "1*", "111*"가 제 1 참 양성 비트인 것으로 추출되고, 길이 0의 경우는 모든 입력과 일치하는 근원지 프리픽스이므로, 항상 블록 필터를 통과하기 때문에, 최종적인 제 1 참 양성 비트는 "*", "1*", "111*"로서 추출된다. 그러므로 주어진 입력에 일치하는 근원지 프리픽스에는 길이 0, 1, 3이 존재하는 것으로 결론을 짓는다.

[0105] 목적지 블록 필터 검색 과정의 경우, 위의 근원지 블록 필터 검색 과정과 동일하게 수행된다. 우선, 입력된 목

적지 프리픽스에 대하여, 미리 저장된 목적지 프리픽스 길이 {1, 2, 3}에 해당하는 목적지 프리픽스 "0*", "00*", "000*" (즉, 제 2 비트)를 추출하고, 추출된 "0*", "00*", "000*"를 CRC-8 생성기를 통과시켜 각각의 코드값(즉, 제 2 코드값)을 생성하고 이로부터 해싱 색인 값(즉, 제 2 해싱 색인 값)을 얻는다. "0*" (즉, 제 2 비트)에 해당하는 CRC 코드값(즉, 제 2 코드값)은 "01111110"이고, 목적지 블룸 필터의 해싱 색인 값(즉, 제 2 해싱 색인 값)은 "3", "6"이므로 도 12의 목적지 블룸 필터로부터 양성임을 알 수 있다. "00*" (즉, 제 2 비트)의 코드값(즉, 제 2 코드값)은 "00111111"이고 목적지 블룸 필터의 해싱 색인 값(즉, 제 2 해싱 색인 값)은 "001 (십진수: 1)", "111(십진수: 7)"인데, 이 중에서 해싱 색인 값 "001(십진수: 1)"에 해당하는 목적지 블룸 필터의 비트벡터가 "0"이므로 "00*"는 음성임을 알 수 있다. "000*" (즉, 제 2 비트)에 해당하는 CRC 코드값(즉, 제 2 코드값)은 "10011111"이고 이로부터 얻어진 해싱 색인 값(즉, 제 2 해싱 색인 값)은 "100(십진수: 4)", "111(십진수: 7)"이며, 이를 이용하여 목적지 블룸 필터의 해당 비트벡터를 검색하면 양성임을 알 수 있다. 즉, 목적지 프리픽스 "00011001"로부터 미리 저장된 목적지 프리픽스 길이(1,2,3)만큼 추출된 제 2 비트인 "0*", "00*", "000*"는, 목적지 블룸 필터 검색 과정을 거쳐, "0*" 및 "000*"는 양성이고, "00*"는 음성인 것으로 판단되어, 제 2 양성 비트로서 "0*" 및 "000*" 가 추출된다.

[0106] 이렇게 추출된 제 2 양성 비트에 대하여, 제 2 양성 비트인 "0*", "000*"를 다시 목적지 해시 테이블에 접근시켜 참 양성인지를 판단하는 목적지 해시 테이블 검색 과정을 수행한다.

[0107] "0*" (즉, 제 2 양성 비트)에 해당하는 코드값인 "01111110"의 마지막 3-비트(즉, 기 정의된 제 2 해싱 색인 값 비트(3비트)만큼의 뒷자리 코드값)인 "110(십진수: 6)"을 해싱 색인 값으로 사용하여 목적지 해시 테이블의 엔트리에 접근하면 "0*"가 저장되어 있는 것을 확인할 수 있다. 이로 인해, "0*" (즉, 제 2 양성 비트)는 참 양성인 것으로 판단되어 제 2 참 양성 비트로 추출된다. 이와 마찬가지로, 나머지 제 2 양성 비트인 "000*"에 해당하는 코드값인 "10011111"의 마지막 3-비트(즉, 기 정의된 제 2 해싱 색인 값 비트(3비트)만큼의 뒷자리 코드값)인 "111(십진수: 7)"을 해싱 색인 값으로 사용하여 목적지 해시 테이블의 엔트리를 검색하면 "000*"가 저장되어 있지 않으므로, 거짓 양성임을 알 수 있다. 따라서, 목적지 해시 테이블 검색 과정을 거쳐, 제 2 양성 비트인 "0*" 및 "000*" 중에서 "0*"는 참 양성이고 "000*"는 거짓 양성으로 판단되어 "0*"가 제 2 참 양성 비트로 추출된다. 그러므로 주어진 입력과 일치하는 목적지 프리픽스는 길이 1인 프리픽스가 있음으로 결론을 짓는다.

[0108] 다음 단계로, 근원지 해시 테이블 검색 과정 및 목적지 해시 테이블 검색 과정을 수행한 이후, 얻어진 근원지 프리픽스 및 목적지 프리픽스의 참 양성 결과 값(즉, 제 1 참 양성 비트와 제 2 참 양성 비트)을 가지고 생성 가능한 모든 튜플을 만들어낸다. 이렇게 튜플들 가운데 룰셋에 존재하는 튜플 리스트와 비교하여 리스트에 존재하는 튜플에 대해서만 튜플 해시 테이블 검색을 수행한다. 주어진 예에서는, 입력된 근원지 IP 주소에 대하여 길이 0, 1 및 3의 프리픽스가 존재하고, 입력된 목적지 IP 주소에 대하여 길이 1의 프리픽스가 존재하는 것으로 결론이 났으므로, (0,1), (1,1), (3,1) 튜플이 생성된다. 그러나 튜플 (0,1)과 (1,1)은 표 1에 주어진 룰 셋에 존재하지 않는 튜플이므로, 튜플 해시 테이블에서의 검색을 진행할 필요가 없고, (3,1) 튜플의 경우에는 표 1에 주어진 룰 셋에 존재하는 튜플이므로 튜플 해시 테이블에서의 검색을 진행하여야 한다. (3,1) 튜플을 위해 근원지 프리픽스의 첫 세 비트인 "111"과 목적지 프리픽스의 첫 비트인 "0"을 연결하여 "1110*" (즉, 연결 비트)을 생성하고, 이렇게 생성된 "1110*" (즉, 연결 비트)를 CRC-16에 통과시켜 얻은 CRC 코드값(즉, 제 3 코드값)인 "000101110001001000"을 얻는다. 이렇게 얻어진 CRC 코드값(즉, 제 3 코드값)으로부터 마지막 3-비트(즉, 기 정의된 제 3 해싱 색인 값 비트 수(3 비트) 만큼의 뒷자리 코드값)인 "000(십진수: 0)"를 제 3 해싱 색인 값으로 생성하고, 이렇게 생성한 제 3 해싱 색인 값을 사용하여 튜플 해시 테이블을 검색하면 룰로서 R4가 검색되고, 입력된 패킷은 룰 R4와 비교되어 일치함을 알 수 있다.

[0109] 이상에서 예시적으로 전술한 본 발명의 일 실시예에 따른 튜플 제거를 통한 패킷 분류 알고리즘은, 기존의 튜플 제거 알고리즘과 비교하여, 각 프리픽스(근원지 프리픽스 및 목적지 프리픽스)가 네스팅하고 있는 다른 프리픽스의 길이들을 선계산 할 필요가 없는 장점이 있으나, 각 필드별로 입력과 일치하는 길이에 관한 참 양성 결과 값을 얻기 위하여 해시 메모리(즉, 근원지 해시 테이블 및 목적지 해시 테이블)에 접근해야 한다.

[0110] 따라서, 이하에서는, 각 필드별로 별도의 해시 메모리(즉, 근원지 해시 테이블 및 목적지 해시 테이블)를 사용하지 않고 거짓-양성의 결과까지를 포함하여 튜플 리스트를 만들어 내되, 별도의 튜플 블룸 필터를 추가하여, 튜플 제거의 단계를 다시 한번 거치는 알고리즘을 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘으로서, 따라오는 도 15 내지 도 26을 참조하여 기술한다.

- [0111] **본 발명의 다른 실시예 (알고리즘 2)**
- [0112] 도 15는 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에 대한 블록 구성도이다.
- [0113] 도 15를 참조하면, 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)는 프리픽스 입력부(1510), 근원지 블룸 필터 검색부(1520), 목적지 블룸 필터 검색부(1530), 튜플 블룸 필터 검색부(1540) 및 튜플 해시 테이블 검색부(1550) 등을 포함한다.
- [0114] 프리픽스 입력부(1510)는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는다.
- [0115] 프리픽스 입력부(210)는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스(Source Prefix) 및 목적지 프리픽스(Destination Prefix)를 입력받는다.
- [0116] 이렇게 입력된 근원지 프리픽스 및 목적지 프리픽스는, 근원지 블룸 필터 검색부(1520) 및 목적지 블룸 필터 검색부(1530)로 각각 병렬적으로 전달된다.
- [0117] 또한, 입력 패킷의 헤더 필드 정보 내에는, 근원지 IP 주소(Source IP Address), 목적지 IP 주소(Destination IP Address), 근원지 포트 넘버(Source Port Number), 목적지 포트 넘버(Destination Port Number) 및 프로토콜 정보(Protocol Information)에 해당하는 5개의 필드를 포함할 수 있으며, 근원지 IP 주소 및 목적지 IP 주소로부터 근원지 프리픽스 및 목적지 프리픽스를 각각 확인한다.
- [0118] 근원지 블룸 필터 검색부(1520)는, 기 저장된 근원지 프리픽스 길이(UL: Unique Length)에 따라 하나 이상의 제 1 비트를 입력된 근원지 프리픽스로부터 추출하고, 추출된 제 1 비트를 토대로 근원지 블룸 필터(Src. BF: Source Bloom Filter)를 검색하여 제 1 비트 중에서 양성(Positive)으로 판단된 제 1 양성 비트를 추출한다.
- [0119] 목적지 블룸 필터 검색부(240)는, 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 입력된 목적지 프리픽스로부터 추출하고, 추출된 제 2 비트를 토대로 목적지 블룸 필터(Dst. BF: Destination Bloom Filter)를 검색하여 제 2 비트 중에서 양성(Positive)으로 판단된 제 2 양성 비트를 추출한다.
- [0120] 튜플 블룸 필터 검색부(1540)는, 제 1 양성 비트 및 상기 제 2 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 블룸 필터를 검색하여 양성으로 판단된 양성 튜플을 추출한다.
- [0121] 튜플 해시 테이블 검색부(1550)는, 양성 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출한다.
- [0122] 위에서 언급한 근원지 블룸 필터, 목적지 블룸 필터 및 튜플 블룸 필터 각각은, 모든 프리픽스 길이(모든 근원지 프리픽스 길이 및 모든 목적지 프리픽스 길이)마다 개별적으로 할당되는 '개별 블룸 필터'를 프리픽스 길이(근원지 프리픽스 길이 및 목적지 프리픽스 길이)의 개수만큼 포함하는 것일 수도 있고, 프리픽스 길이의 구분 없이 모든 프리픽스에 대하여 동일한 해싱 함수를 이용하여 얻어진 해싱 색인 값에 해당하는 비트벡터를 특정 값으로 프로그래밍하고, 프로그래밍 시 사용된 프리픽스에 대한 프리픽스 길이는 별도로 저장하는 '통합 블룸 필터(All-Length Bloom Filter)'일 수 있다.
- [0123] 이하 설명에서는, 근원지 블룸 필터, 목적지 블룸 필터 및 튜플 블룸 필터 각각은, 개별 블룸 필터를 프리픽스 길이의 개수만큼 포함하는 것에 비해, 블룸 필터의 프로그래밍 및 쿼리 과정이 더욱 단순화하고 거짓 양성을 조절하기가 더욱 용이한 통합 블룸 필터를 사용한 것으로 가정하여 설명한다.
- [0124] 이상에서 전술한 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에 포함된 각 구성요소에 대해서는, 도 16 내지 도 24를 참조하여 더욱 상세하게 설명한다.
- [0125] 도 16은 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 근원지 블룸 필터 검색부(1520)를 나타낸 도면이다.
- [0126] 도 16을 참조하면, 전술한 근원지 블룸 필터 검색부(1520)는, 프리픽스 입력부(1510)로부터 근원지 프리픽스를 전달받아, 전달받은 근원지 프리픽스를 이용하여 근원지 블룸 필터를 통과시켜 근원지 블룸 필터의 제 1 양성 비트, 즉 일치 가능성이 있는 길이에 관한 정보를 추출한다. 이때 추출한 제 1 양성 비트는 참 양성(True Positive)인 제 1 참 양성 비트 및 거짓 양성(False Positive)인 제 1 거짓 양성 비트 중 하나 이상을 포함할 수 있다.

- [0127] 이러한 근원지 블록 필터 검색부(1520)는 전술한 바와 같이 근원지 프리픽스를 전달받아 제 1 양성 비트를 추출하기 위해서, 제 1 비트 추출부(1610), 제 1 해싱부(1620) 및 제 1 양성 비트 추출부(1630) 등을 포함한다.
- [0128] 제 1 비트 추출부(1610)는, 기 저장된 근원지 프리픽스 길이(예: 근원지 프리픽스 길이(UL: Unique Length)=0,1,2,3 ...)에 따라 하나 이상의 제 1 비트를 입력된 근원지 프리픽스로부터 추출한다.
- [0129] 제 1 해싱부(1620)는, 추출된 제 1 비트 각각에 대하여 기 정의된 제 1 코드값 비트 수(예: 8 비트 등)를 갖는 제 1 코드값을 생성하고, 생성된 제 1 코드값으로부터 제 1 해싱 색인 값을 생성한다.
- [0130] 제 1 양성 비트 추출부(1630)는, 생성된 제 1 해싱 색인 값에 해당하는 근원지 블록 필터의 비트벡터(Bit-Vector)를 확인하고, 확인된 비트벡터를 토대로 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출한다. 예를 들어, 제 1 해싱 색인 값 각각에 대하여 확인된 비트벡터가 모두 1인 경우인 제 1 비트를 양성(Positive)으로 판단할 수 있고, 확인된 비트벡터가 하나라도 1이 아닌 경우인 제 1 비트를 음성(Negative)으로 판단할 수 있다.
- [0131] 전술한 제 1 해싱부(1620)는 따라오는 도 17을 참조하여 더욱 상세하게 설명한다.
- [0132] 도 17은 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 근원지 블록 필터 검색부(1520)에 포함된 제 1 해싱부(1620)를 나타낸 도면이다.
- [0133] 도 17을 참조하면, 제 1 해싱부(1620)는, 입력된 근원지 프리픽스로부터 기 저장된 근원지 프리픽스 길이(예: 근원지 프리픽스 길이(UL: Unique Length)= 0,1,2,3 ...)만큼 각각 추출된 하나 이상의 제 1 비트 각각에 대하여 제 1 코드값 비트 수(예: 8 비트 등)를 갖는 제 1 코드값을 생성하는 제 1 코드값 생성부(1710)와, 기 정의된 제 1 해싱 색인 값 비트 수(예: 3 비트 등)만큼의 앞자리 코드값 및 뒷자리 코드값을 제 1 코드값으로부터 추출하고, 이렇게 추출된 앞자리 코드값 및 뒷자리 코드값을 포함하는 제 1 해싱 색인 값(즉, 2개의 해싱 색인 값)을 제 1 해싱 색인 값 생성부(1720) 등을 포함한다.
- [0134] 위에서 언급한 제 1 코드값 생성부(1710)는, 미리 정의된 제 1 코드값 비트 수를 갖는 제 1 코드값을 생성하기 위해서, 제 1 코드값 비트 수에 해당하는 CRC 생성기를 이용할 수 있다. 예를 들어, 미리 정의된 제 1 코드값 비트 수가 8 비트인 경우, 8 비트 길이의 제 1 코드값을 생성할 수 있는 CRC-8 생성기를 이용할 수 있다. 이때의 제 1 코드값을 제 1 CRC 코드값이라고도 한다.
- [0135] 전술한 바와 같이, 근원지 프리픽스는, 근원지 블록 필터 검색부(1520)를 거쳐 제 1 양성 비트가 추출된다.
- [0136] 이와 동일한 방식과 병렬적인 방식으로, 목적지 프리픽스도, 목적지 블록 필터 검색부(1530)를 거쳐 제 2 양성 비트가 추출된다. 이를 따라오는 도 18 및 도 19를 참조하여 상세하게 설명한다.
- [0137] 도 18은 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 목적지 블록 필터 검색부(1530)를 나타낸 도면이다.
- [0138] 도 18을 참조하면, 목적지 블록 필터 검색부(1530)는, 프리픽스 입력부(1510)로부터 목적지 프리픽스를 전달받아, 전달받은 목적지 프리픽스를 이용하여 목적지 블록 필터를 통과시켜 목적지 블록 필터의 제 2 양성 비트, 즉 일치 가능성이 있는 길이에 관한 정보를 추출한다. 이때 추출한 제 2 양성 비트는 참 양성인 제 2 참 양성 비트 및 거짓 양성인 제 1 거짓 양성 비트 중 하나 이상을 포함할 수 있다.
- [0139] 이러한 목적지 블록 필터 검색부(1530)는 전술한 바와 같이 목적지 프리픽스를 전달받아 제 2 양성 비트를 추출하기 위해서, 제 2 비트 추출부(1810), 제 2 해싱부(1820) 및 제 2 양성 비트 추출부(1830) 등을 포함한다.
- [0140] 제 2 비트 추출부(1810)는, 기 저장된 목적지 프리픽스 길이(예: 목적지 프리픽스 길이(UL: Unique Length)=0,1,2,3 ...)에 따라 하나 이상의 제 2 비트를 목적지 프리픽스로부터 추출한다.
- [0141] 제 2 해싱부(1820)는, 추출된 제 2 비트 각각에 대하여 기 정의된 제 2 코드값 비트 수(예: 8 비트 등)를 갖는 제 2 코드값을 생성하고, 이렇게 생성된 제 2 코드값으로부터 제 2 해싱 색인 값을 생성한다.
- [0142] 제 2 양성 비트 추출부(1830)는, 생성된 제 2 해싱 색인 값에 해당하는 목적지 블록 필터의 비트벡터를 확인하고, 확인된 비트벡터를 토대로 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출한다. 예를 들어, 제 2 해싱 색인 값 각각에 대하여 확인된 비트벡터가 모두 1인 경우인 제 2 비트를 양성으로 판단할 수 있고, 확인된 비트벡터가 하나라도 1이 아닌 경우인 제 2 비트를 음성으로 판단할 수 있다.
- [0143] 전술한 제 2 해싱부(1820)는, 따라오는 도 19를 참조하여 더욱 상세하게 설명한다.

- [0144] 도 19는 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 목적지 블록 필터 검색부(150)에 포함된 제 2 해싱부(1820)를 나타낸 도면이다.
- [0145] 도 6을 참조하면, 제 2 해싱부(520)는, 입력된 목적지 프리픽스로부터 기 저장된 목적지 프리픽스 길이(예: 목적지 프리픽스 길이(UL: Unique Length)= 0,1,2,3 ...)만큼 각각 추출된 제 2 비트 각각에 대하여 제 2 코드값 비트 수(예: 8 비트 등)를 갖는 상기 제 2 코드값을 생성하는 제 2 코드값 생성부(1910); 및 기 정의된 제 2 해싱 색인 값 비트 수(예: 3 비트 등)만큼의 앞자리 코드값 및 뒷자리 코드값을 제 2 코드값으로부터 추출하고, 추출된 앞자리 코드값 및 뒷자리 코드값을 포함하는 제 2 해싱 색인 값(즉, 2개의 해싱 색인 값)을 생성하는 제 2 해싱 색인 값 생성부(1920) 등을 포함한다.
- [0146] 위에서 언급한 제 2 코드값 생성부(1910)는, 미리 정의된 제 2 코드값 비트 수를 갖는 제 2 코드값을 생성하기 위해서, 제 2 코드값 비트 수에 해당하는 CRC 생성기를 이용할 수 있다. 예를 들어, 미리 정의된 제 2 코드값 비트 수가 8 비트인 경우, 8 비트 길이의 제 2 코드값을 생성할 수 있는 CRC-8 생성기를 이용할 수 있다. 이때의 제 2 코드값을 제 2 CRC 코드값이라고도 한다.
- [0147] 도 20은 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 튜플 블록 필터 검색부(1540)를 나타낸 도면이다.
- [0148] 도 20을 참조하면, 전술한 튜플 블록 필터 검색부(1540)는, 근원지 블록 필터 검색부(1520)에서 추출된 제 1 양성 비트의 길이 및 목적지 블록 필터 검색부(1530)에서 추출된 제 2 양성 비트의 길이를 조합하여 하나 이상의 튜플을 생성하는 튜플 생성부(2010); 이렇게 생성된 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하지 않는 튜플은 제거하고 존재하는 튜플만을 선별하는 튜플 선별부(2020); 및 이렇게 선별된 튜플 각각에 대하여 제 3 해싱 색인 값을 생성하고, 생성된 제 3 해싱 색인 값에 해당하는 튜플 블록 필터의 비트 벡터를 확인하며, 확인된 비트벡터를 토대로 각 튜플이 양성인지 음성인지를 판단함으로써, 양성 튜플을 추출하는 양성 튜플 판단 추출부(2030) 등을 포함한다.
- [0149] 전술한 바와 같이, 튜플 블록 필터 검색부(1540)에 포함된 튜플 생성부(2010)에서 생성된 하나 이상의 튜플은, 근원지 블록 필터 검색부(1520)에서 추출된 제 1 양성 비트의 길이 및 목적지 블록 필터 검색부(1530)에서 추출된 제 2 양성 비트의 길이를 조합하여 생성되었기 때문에, 거짓 양성인 제 1 양성 비트 및 제 2 양성 비트도 포함되어 있을 수 있다.
- [0150] 이와 같이 생성된 튜플에 거짓 양성도 포함되어 있을 수 있는 점은, 도 2에 도시된 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에서의 튜플 생성과 비교될 수 있다.
- [0151] 도 2에 도시된 본 발명의 일 실시예에 따른 패킷 분류 장치(200)에 포함된 튜플 해시 테이블 검색부(260)에 포함된 튜플 생성부(710)에서 생성하는 튜플은, 근원지 해시 테이블 검색 과정과 목적지 해시 테이블 검색 과정을 거친 이후, 이로부터 추출된 참 양성인 제 1 양성 비트(제 1 참 양성 비트)의 길이 및 참 양성인 제 2 양성 비트(제 2 참 양성 비트)의 길이를 조합하여 생성되기 때문에, 거짓 양성인 제 1 양성 비트 및 제 2 양성 비트를 포함하지 않고, 참 양성인 제 1 양성 비트(제 1 참 양성 비트) 및 참 양성인 제 2 양성 비트(제 2 참 양성 비트)만을 포함한다. 즉, 튜플 블록 필터 검색부(1540)에 포함된 튜플 생성부(2010)에서 생성된 하나 이상의 튜플은, 근원지 해시 테이블 검색 과정과 목적지 해시 테이블 검색 과정을 거치지 않고 생성되기 때문에, 참 양성뿐만 아니라 거짓 양성인 제 1 양성 비트 및 제 2 양성 비트도 포함되어 있을 수 있는 것이다.
- [0152] 도 21은 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 튜플 블록 필터 검색부(1540)에 포함된 양성 튜플 판단 추출부(2030)를 나타낸 도면이다.
- [0153] 도 21을 참조하면, 양성 튜플 판단 추출부(2030)는, 앞선 튜플 선별부(2020)에서 선별된 튜플 각각에 대하여, 선별된 튜플 중 어느 튜플을 이용하여 튜플 해시 테이블 검색을 수행할지를 결정하여 불필요한 튜플 해시 테이블 검색을 방지하기 위하여, 선별된 모든 튜플 중에서 튜플 해시 테이블 검색을 수행할 양성 튜플과 튜플 해시 테이블 검색을 수행하지 않을 음성 튜플을 분별하여 판단함으로써 양성 튜플만을 추출하는 선 필터링 과정을 수행한다. 이를 위해, 양성 튜플 판단 추출부(2030)는, 제 3 해싱부(2110), 양성 튜플 판단부(2120) 및 양성 튜플 추출부(2130) 등을 포함한다.
- [0154] 제 3 해싱부(2110)는, 앞선 튜플 선별부(2020)에서 선별된 튜플 각각에 대하여, 각 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 제 3 연결 비트를 생성하고, 이렇게 생성된 제 3 연결 비트로부터 기 정의된 제 3 코드값 비트 수(예: 16 비트 등)를 갖는 제 3 코드값을 생성하며, 생성된 제 3 코드값으로부터 제

3 해싱 색인 값을 생성한다.

- [0155] 양성 튜플 판단부(2120)는, 생성된 제 3 해싱 색인 값에 해당하는 튜플 블록 필터의 비트벡터를 확인하여, 확인된 비트벡터를 토대로 각 튜플이 양성인지 음성인지를 판단한다.
- [0156] 양성 튜플 추출부(2130)는, 선별된 튜플 중에서 양성으로 판단된 튜플을 튜플 해시 테이블에 접근하기 위한 양성 튜플로서 추출한다.
- [0157] 전술한 제 3 해싱부(2110)는 따라오는 도 22를 참조하여 더욱 상세하게 설명한다.
- [0158] 도 22는 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 튜플 블록 필터 검색부(1540)에 포함된 양성 튜플 판단 추출부(2030)에 포함된 제 3 해싱부(2110)를 나타낸 도면이다.
- [0159] 도 22를 참조하면, 이러한 제 3 해싱부(2110)는, 튜플 선별부(2020)에서 선별된 튜플 각각에 대하여, 각 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 제 3 연결 비트를 생성하는 제 3 연결 비트 생성부(2210); 생성된 제 3 연결 비트로부터 기 정의된 제 3 코드값 비트 수(예: 16 비트)를 갖는 제 3 코드값을 생성하는 제 3 코드값 생성부(2220); 및 기 정의된 제 3 해싱 색인 값 비트 수(예: 4 비트)만큼의 앞자리 코드값 및 뒷자리 코드값을 생성된 제 3 코드값으로부터 추출하고, 이렇게 추출된 앞자리 코드값(예: 4 비트) 및 뒷자리 코드값(예: 4 비트)을 포함하는 제 3 해싱 색인 값(즉, 2개의 해싱 색인 값)을 생성하는 제 3 해싱 색인 값 생성부(2230) 등을 포함한다.
- [0160] 전술한 제 3 해싱부(2110)에 포함된 제 3 코드값 생성부(2220)는, 미리 정의된 제 3 코드값 비트 수를 갖는 제 3 코드값을 생성하기 위해서, 제 3 코드값 비트 수에 해당하는 CRC 생성기를 이용할 수 있다. 예를 들어, 미리 정의된 제 3 코드값 비트 수가 16 비트인 경우, 16 비트 길이의 제 3 코드값을 생성할 수 있는 CRC-16 생성기를 이용할 수 있다. 이때의 제 3 코드값을 제 3 CRC 코드값이라고도 한다.
- [0161] 도 23은 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 튜플 해시 테이블 검색부(1550)를 나타낸 도면이다.
- [0162] 도 23에 도시된 튜플 해시 테이블 검색부(1550)는, 앞에서도 전술한 바와 같이, 양성 튜플 각각에 대하여 튜플 해시 테이블의 접근을 통해 검색을 수행하여 패킷 분류를 위한 물을 추출한다.
- [0163] 이를 위해, 이러한 튜플 해시 테이블 검색부(1550)는, 양성 튜플 각각에 대하여, 각 양성 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 제 4 연결 비트를 생성하고, 이렇게 생성된 제 4 연결 비트로부터 기 정의된 제 4 코드값 비트 수(예: 16 비트)를 갖는 제 4 코드값을 생성하며, 이렇게 생성된 제 4 코드값으로부터 제 4 해싱 색인 값을 생성하는 제 4 해싱부(2310); 및 생성된 제 4 해싱 색인 값이 가리키는 튜플 해시 테이블의 위치에서 패킷 분류를 위한 물을 추출하는 물 추출부(2320) 등을 포함한다.
- [0164] 전술한 제 4 해싱부(2310)는 따라오는 도 24를 참조하여 더욱 상세하게 설명한다.
- [0165] 도 24는 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)에서, 튜플 해시 테이블 검색부(1550)에 포함된 제 4 해싱부(2310)를 나타낸 도면이다.
- [0166] 도 24를 참조하면, 전술한 제 4 해싱부(2310)는, 양성 튜플 각각에 대하여, 각 양성 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트를 연결하여 제 4 연결 비트를 생성하는 제 4 연결 비트 생성부(2410); 생성된 제 4 연결 비트로부터 기 정의된 제 4 코드값 비트 수(예: 16 비트)를 갖는 제 4 코드값을 생성하는 제 4 코드값 생성부(2420); 및 기 정의된 제 4 해싱 색인 값 비트 수(예: 3 비트)만큼의 뒷자리 코드값을 생성된 제 4 코드값으로부터 추출하고, 뒷자리 코드값을 포함하는 제 4 해싱 색인 값을 생성하는 제 4 해싱 색인 값 생성부(2430) 등을 포함한다.
- [0167] 전술한 제 4 해싱부(2310)에 포함된 제 4 코드값 생성부(2420)는, 미리 정의된 제 4 코드값 비트 수를 갖는 제 4 코드값을 생성하기 위해서, 제 4 코드값 비트 수에 해당하는 CRC 생성기를 이용할 수 있다. 예를 들어, 미리 정의된 제 4 코드값 비트 수가 16 비트인 경우, 16 비트 길이의 제 4 코드값을 생성할 수 있는 CRC-16 생성기를 이용할 수 있다. 이때의 제 4 코드값을 제 4 CRC 코드값이라고도 한다. 또한, 제 4 코드값 생성부(2420)는, 각 양성 튜플의 생성시 이용된 제 1 양성 비트 및 제 2 양성 비트가 연결된 제 4 연결 비트로부터 제 4 코드값을 생성할 때, 동일한 제 1 양성 비트 및 제 2 양성 비트를 이용하여 미리 생성된 제 3 코드값을 이용할 수 있다.
- [0168] 이상에서 전술한 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)가 제공하는 패킷 분류 방법의 흐름도는 도 25에 도시된다.

- [0169] 도 25를 참조하면, 본 발명의 다른 실시예에 따른 패킷 분류 장치(1500)가 제공하는 패킷 분류 방법은, 프리픽스 입력 단계(S2500), 근원지 블룸 필터 검색 단계(S2502), 목적지 블룸 필터 검색 단계(S2504), 튜플 블룸 필터 검색 단계(S2506) 및 튜플 해시 테이블 검색 단계(S2508) 등을 포함한다.
- [0170] 프리픽스 입력 단계(S2500)는, 입력 패킷의 헤더 필드 정보로부터 확인된 근원지 프리픽스 및 목적지 프리픽스를 입력받는 단계이다.
- [0171] 근원지 블룸 필터 검색 단계(S2502)는, 기 저장된 근원지 프리픽스 길이에 따라 하나 이상의 제 1 비트를 근원지 프리픽스로부터 추출하고, 제 1 비트를 토대로 근원지 블룸 필터를 검색하여 제 1 비트 중에서 양성으로 판단된 제 1 양성 비트를 추출하는 단계이다.
- [0172] 목적지 블룸 필터 검색 단계(S2504)는, 기 저장된 목적지 프리픽스 길이에 따라 하나 이상의 제 2 비트를 상기 목적지 프리픽스로부터 추출하고, 제 2 비트를 토대로 목적지 블룸 필터를 검색하여 제 2 비트 중에서 양성으로 판단된 제 2 양성 비트를 추출하는 단계이다.
- [0173] 튜플 블룸 필터 검색 단계(S2506)는, 제 1 양성 비트 및 상기 제 2 양성 비트를 토대로 하나 이상의 튜플을 생성하고, 하나 이상의 튜플 중에서 미리 생성된 룰 셋에 저장된 튜플 리스트에 존재하는 튜플에 대하여 튜플 블룸 필터를 검색하여 양성으로 판단된 양성 튜플을 추출하는 단계이다.
- [0174] 튜플 해시 테이블 검색 단계(S2508)는, 양성 튜플에 대하여 튜플 해시 테이블을 검색하여 패킷 분류를 위한 룰을 추출하는 단계이다.
- [0175] 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘은, 목적지 프리픽스와 근원지의 프리픽스 필드를 사용하여 2차원 패킷 분류알고리즘을 제안하는 데 있어, 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에는 있는 근원지 해시 테이블 검색 과정과 목적지 해시 테이블 검색 과정은 없애고, 즉, 근원지 해시 테이블과 목적지 해시 테이블을 없애고, 튜플 블룸 필터를 추가한 구조이다. 이러한 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘의 구조를 도식화하면 도 26과 같다.
- [0176] 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘은, 입력된 근원지 프리픽스와 목적지 프리픽스에 대한 해시 테이블 검색을 수행하지 않으므로, 즉, 참 양성만을 추출하여 튜플을 생성하지 않기 때문에, 근원지 블룸 필터와 목적지 블룸 필터에서 생성된 참 양성뿐만 아니라 거짓 양성의 결과 또한 교차하여 더 많은 튜플을 생성할 수도 있다. 생성된 튜플에 포함될 수 있는 거짓 양성의 결과는, 추가된 튜플 블룸 필터로부터 필터링 될 수 있다.
- [0177] 근원지 블룸 필터 및 목적지 블룸 필터에 대한 "프로그래밍 과정"은, 전술한 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서의 프로그래밍 과정과 동일하게 이루어진다. 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서는 없는 튜플 블룸 필터에 대한 "프로그래밍 과정"은 다음과 같다.
- [0178] 주어진 룰 셋에서 각각의 룰에 대하여, 근원지 프리픽스 길이와 목적지 프리픽스 길이에 해당하는 프리픽스들을 추출하여 연결하여 연결 비트(즉, 제 3 연결 비트)를 생성한 이후, CRC-16 생성기를 통과시켜 튜플 블룸 필터의 해싱 색인 값(즉, 제 3 해싱 색인 값)을 얻어내고, 이렇게 얻어진 튜플 블룸 필터의 해싱 색인 값에 해당하는 비트벡터 값을 1로 프로그래밍을 한다.
- [0179] 알고리즘 2를 위한 CRC-16 결과 값, 블룸 필터의 색인 값, 해시 테이블 색인 값에 대한 정보를 표 4에 나타내었다. 여기서 CRC-16의 초기값은 1111110011111100으로 가정하였으며, 튜플 블룸 필터의 해싱 색인 값(즉, '제 3 해싱 색인 값')으로는 CRC-16 코드의 처음과 마지막 4-비트(즉, 기 정의된 제 3 해싱 색인 값 비트 수인 4 비트 길이의 앞자리 코드값과 뒷자리 코드값)를 사용하고, 튜플 해시 테이블의 해싱 색인 값(즉, '제 4 해싱 색인 값')으로는 마지막 3-비트(즉, 기 정의된 제 4 해싱 색인 값 비트 수인 3 비트 길이의 뒷자리 코드값)를 사용하였다.

표 4

[0180]	룰	튜플	연결 프리픽스	CRC-16 코드값	제 3 해싱 색인 값	제 4 해싱 색인 값
	R1	(1,2)	110	0011110010011001	3,9	1
	R2	(2,2)	0011	1000100111000010	8,2	2
	R3	(0,3)	100	1111111010011101	15,13	5

R4	(3,1)	1110	0101110001001000	5,8	0
R5	(3,3)	011111	0111110000111101	7,13	5

- [0181] 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘에서의 튜플 해시 테이블의 구조는, 도 14에 도시된 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서의 튜플 해시 테이블의 구조와 동일하다. 앞서 언급한 도 26에 표 4에 의하여 프로그래밍 된 튜플 블록 필터 및 튜플 해시 테이블의 전체적인 구조를 볼 수 있다.
- [0182] 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘에서의 "쿼리 과정"을 살펴보면 다음과 같다.
- [0183] 주어진 입력에 대해 근원지 블록 필터와 목적지 블록 필터를 검색하여 일치 가능한 길이의 리스트를 뽑아내는 과정은, 전술한 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서와 동일하다. 이러한 근원지 블록 필터 검색 과정과 목적지 블록 필터 검색 과정을 병렬적으로 수행한 이후, 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서는 근원지 해시 테이블 및 목적지 해시 테이블에 각각 접근하여 참 양성 길이를 걸러내었으나, 즉 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서는, 근원지 블록 필터 검색 과정과 목적지 블록 필터 검색 과정을 병렬적으로 수행한 이후, 근원지 해시 테이블 검색 과정 및 목적지 해시 테이블 검색 과정을 병렬적으로 각각 이어서 수행하여 제 1 참 양성 비트 및 제 2 참 양성 비트를 각각 추출해냈으나, 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘에서는 해시 테이블(근원지 해시 테이블 및 목적지 해시 테이블)에 접근하지 않고, 일치 가능성이 있는 길이의 집합을 서로 교차하여 튜플을 만든다. 이렇게 생성된 튜플들에 대하여 주어진 룰 셋의 튜플 리스트와 비교하여 튜플 리스트에 존재하지 않는 튜플들을 제거하고 나머지 튜플에 대하여 튜플 블록 필터 검색 과정을 수행하는데, 각 튜플에서 정의된 근원지 프리픽스 길이 및 목적지 프리픽스 길이에 해당하는 길이만큼의 프리픽스들을 취하여 CRC-16에 입력하여 튜플 블록 필터에 대한 해싱 색인 값(즉, 제 3 해싱 색인 값)을 얻는다. 이 해싱 색인 값들을 이용하여 튜플 블록 필터를 검색하여 양성으로 판단된 튜플들에 한하여 외부의 튜플 해시 테이블에 접근하여 입력된 패킷 값과 비교하여 일치하는 최우선순위의 룰을 찾아낸다.
- [0184] 이와 같은 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘에 대한 검색 과정을 예를 들어서 다음과 같다. 표 1의 룰 셋에 대해, 본 발명의 일 실시예에 따른 패킷 분류 알고리즘에서의 예시와 동일한 입력 값(근원지 프리픽스: "11101101", 목적지 프리픽스: "00011001")을 가정한다.
- [0185] 이러한 근원지 프리픽스 및 목적지 프리픽스가 입력되면 각각에 대하여, 근원지 블록 필터 검색 과정 및 목적지 블록 필터 검색 과정을 수행하기 위하여, 근원지 프리픽스 및 목적지 프리픽스를 각각 근원지 블록 필터와 목적지 블록 필터를 병렬로 통과시켜 양성 결과값들(즉, 제 1 양성 비트 및 제 2 양성 비트)을 얻는다.
- [0186] 본 발명의 다른 실시예에 따른 패킷 분류 알고리즘에서는, 근원지 프리픽스 및 목적지 프리픽스 각각에 대한 근원지 해시 테이블 검색 과정 및 목적지 해시 테이블 검색 과정이 생략되었으므로 거짓-양성을 걸러낼 수 없다. 따라서, 근원지 블록 필터 및 목적지 블록 필터 각각의 거짓-양성을 포함한 양성 결과값(즉, 제 1 양성 비트 및 제 2 양성 비트)을 가지고 교차하여 튜플의 조합을 만들어 내면, 생성된 튜플(UT: Unique Tuple)은, 도 26을 참조하면, (0,1), (0,3), (1,1), (1,3), (2,1), (2,3), (3,1) 및 (3,3)을 포함한다.
- [0187] 교차 생성과정을 통하여 생성한 튜플 중에서 미리 저장된 튜플 리스트에 존재하는 (0,3), (3,1) 및 (3,3)에 대해서만 튜플 블록 필터를 이용하여 검색한다.
- [0188] 먼저, (0,3) 튜플인 "000*"를 CRC-16의 입력으로 사용하여 얻은 CRC-16 코드값(즉, 제 3 코드값)인 "1001111110011111"로부터 기 정의된 제 3 해싱 색인 값 비트 수(여기서는, 4 비트)만큼의 처음 4-비트(즉, 앞자리 코드값) 및 마지막 4-비트(즉, 뒷자리 코드값)를 제 3 해싱 색인 값으로서 생성한다. 즉, 앞자리 코드값인 "1001(십진수로는 '9'임)"와 뒷자리 코드값인 "1111(십진수로는 '15'임)"를 포함하는 제 3 해싱 색인 값이 생성된다. 이렇게 생성된 제 3 해싱 색인 값(9, 15)을 사용하여, 도 26에 주어진 튜플 블록 필터를 검색하면 (0,3) 튜플인 "000*"는 양성 튜플임을 알 수 있다. 다음으로, (3,1) 튜플인 "1110*"에 대한 CRC-16 코드값(즉, 제 3 코드값)인 "0101110001001000"로부터 제 3 해싱 색인 값인 5, 8을 얻고 이 또한 양성임을 알 수 있다. 마지막으로 (3,3) 튜플인 "111000*"에 대한 CRC-16 코드값(즉, 제 3 코드값)인 "0001011100010010"로부터 얻어진 제 3 해싱 색인 값인 1과 2를 사용하여 튜플 블록 필터를 검색하면 이는 음성임을 알 수 있다. 따라서, 튜플 블록 필터 검색 과정을 통해, (0,3), (3,1) 및 (3,3) 튜플 중에서 (0,3) 및 (3,1) 튜플이 양성 튜플인 것으로 판단되어 추출된다.
- [0189] 이렇게 튜플 블록 필터에서 음성으로 선언된 값들에 대해서는 튜플 해시 테이블 검색 과정을 수행하지 않아도 되므로, 양성 값을 가지는 양성 튜플(즉, (0,3) 튜플 및 (3,1) 튜플)에 대해서만 튜플 해시 테이블에 접근하여

튜플 해시 테이블 검색 과정을 수행한다. 이때, 해당 튜플에 대하여 CRC-16 코드의 마지막 3-비트를 튜플 해시 테이블의 해싱 색인 값(즉, 제 4 해싱 색인 값)으로 사용한다.

[0190] 먼저, 첫 번째 양성 튜플인 (0,3)의 경우에는, (0,3) 튜플인 "000*"를 CRC-16의 입력으로 사용하여 얻은 CRC-16 코드값(즉, 제 4 코드값)인 "1001111110011111"로부터 기 정의된 제 4 해싱 색인 값 비트 수(여기서는, 3 비트)만큼의 마지막 3-비트(즉, 뒷자리 코드값)인 "111(십진수로 '7'임)"를 제 4 해싱 색인 값으로서 생성하는데, 튜플 해시 테이블에 대한 제 4 해싱 색인 값이 "111(십진수로 '7'임)"이므로 해당 제 4 해싱 색인 값이 가리키는 튜플 해시 테이블 위치에 찾아가 검색하는데, 이 경우 해당 제 4 해싱 색인 값인 "111(십진수로 '7'임)"에 일치하는 룰이 존재하지 않으므로 튜플 블록 필터의 거짓-양성으로 인한 불필요한 검색이었음을 알 수 있다.

[0191] 다음으로, 나머지 양성 튜플인 (3,1)의 경우에는, (3,1) 튜플인 "1110*"에 대한 CRC-16 코드값(즉, 제 4 코드값)인 "0101110001001000"로부터 기 정의된 제 4 해싱 색인 값 비트 수(여기서는, 3 비트)만큼의 마지막 3-비트(즉, 뒷자리 코드값)인 "000(십진수로 '0'임)"를 제 4 해싱 색인 값으로서 생성하는데, 이렇게 생성된 튜플 해시 테이블의 제 4 해싱 색인 값인 "000(십진수로 '0'임)"에 해당하는 튜플 해시 테이블의 위치로 가서 입력 패킷 값과 엔트리에 저장된 값을 비교하면 입력 값과 일치하는 것을 확인할 수 있고, 이때의 일치 결과는 "R4"이다.

[0192] 이상에서는, 튜플 제거 기법을 통한 패킷 분류 알고리즘의 실시예로서 2가지 실시예를 기술하였다. 즉, 본 발명의 일 실시예에 따른 패킷 분류 알고리즘(이하, "알고리즘 1"이라 칭함)과, 발명의 다른 실시예에 따른 패킷 분류 알고리즘(이하, "알고리즘 2"라 칭함)을 설명하였다. 아래에서는 패킷 분류 알고리즘으로서 전술한 2가지 알고리즘(알고리즘 1과 알고리즘 2)를 비교 설명한다.

[0193] 이상에서 전술한 알고리즘 2는, 알고리즘 1에 포함된 근원지 해시 테이블 검색 과정과 목적지 해시 테이블 검색 과정을 생략하고, 대신 근원지 블록 필터 및 목적지 블록 필터 각각에서 얻어진 거짓 양성 결과값을 포함할 수도 있는 제 1 양성 비트 및 제 2 양성 비트를 이용하여 튜플을 만들게 됨으로써, 다음 단계에서 불필요한 검색을 수행할 수 있는 단점을 지니게 된다.

[0194] 하지만, 전체적인 메모리 접근 횟수의 관점에서 외부 메모리 접근 횟수(즉, 튜플 해시 테이블 검색 과정시, 튜플 해시 테이블의 접근 횟수)를 비교해보면, 알고리즘 1의 경우, 근원지 해시 테이블, 목적지 해시 테이블 및 튜플 해시 테이블에 모두 접근해야 한다. 동일하게 주어진 예시에서, 알고리즘 1은, 근원지 해시 테이블에 대하여 세 번, 목적지 해시 테이블에 대하여 두 번, 튜플 해시 테이블에 대하여 한 번의 해시 테이블 검색을 수행하게 되고, 모두 합하여 여섯 번의 메모리 접근(해시 테이블의 접근)을 요구한다. 반면, 알고리즘 2의 경우에는, 튜플 해시 테이블만 접근하기 때문에, 동일하게 주어진 예시에서, 두 번의 튜플 해시 테이블 접근이 요구되어, 제안된 알고리즘 1보다 적은 메모리 접근 횟수(해시 테이블의 접근 회수)를 가질 수 있다는 장점을 가진다. 또한 메모리 요구량에서도 블록 필터 자체는 해시 테이블에 비하여 매우 적은 메모리 용량을 차지하기 때문에, 튜플 블록 필터를 추가로 가지는 대신 근원지 해시 테이블 및 목적지 해시 테이블의 접근이 없는 알고리즘 2가 튜플 블록 필터를 검색하지 않는 대신 근원지 해시 테이블 및 목적지 해시 테이블의 접근을 요구하는 알고리즘 1에 비하여 적은 메모리 공간을 소요한다는 장점이 있다.

[0195] 이하에서는, 튜플 제거 기법을 통한 패킷 분류 알고리즘의 2가지 실시예인 알고리즘 1과 알고리즘 2에 대하여, 패킷 분류 성능, 메모리 접근 횟수 및 메모리 사용량 등에 대한 성능 실험을 수행한 결과를 설명한다.

[0196] 성능 평가 실험

[0197] 라우터에서 패킷 분류를 하는데 걸리는 시간은 메모리 접근 횟수에 비례하기 때문에 빠른 패킷 분류를 위해서는 가능한 메모리 접근 횟수를 줄이는 것이 중요하다. 일반적으로 패킷 분류 속도의 측정시 평균메모리 접근횟수와 최대 메모리 접근횟수를 가지고 성능을 평가한다. 본 발명에서 제안하는 알고리즘 1 및 알고리즘 2의 성능을 평가하기 위하여 클래스 벤치에서 제공하며 실제 라우터에 사용되는 룰 특성을 반영한 데이터 베이스 ACL1k, ACL5k, FW1k, FW5k, IPC1k, IPC5k 룰 셋을 이용하여 시뮬레이션을 수행하였다.

[0198] 표 5는 ACL5k, FW5k, IPC5k 룰 셋이 가지는 룰의 개수와 입력의 개수, 각 블록 필터의 크기에 따른 최대 메모리 접근횟수와 평균메모리 접근 횟수를 나타낸 것이다.

표 5

[0199]

물 셋 타입	물 개 수	입력 개수	알고 리즘	블록 필터 크기	최대 메모리		접근 횟수		평균 메모리		접근 횟수	
					근원지	목적지	튜플	합	근원지	목적지	튜플	합
ACL5k	4660	13980	알고 리즘 1	4N	7	13	16	31	4.09	6.52	6.87	17.48
				8N	5	9	16	26	3.16	4.47	6.87	14.51
				16N	4	6	16	25	3.01	3.43	6.87	13.31
				32N	4	7	16	24	3.02	3.36	6.87	13.25
			알고 리즘 2	4N	-	-	8	8	-	-	4.28	4.28
				8N	-	-	8	8	-	-	4.01	4.01
				16N	-	-	6	6	-	-	3.91	3.91
				32N	-	-	6	6	-	-	3.91	3.91
IPC5k	4468	20610	알고 리즘 1	4N	15	16	20	43	8.53	9.26	15.29	33.08
				8N	9	11	20	33	4.28	5.45	15.29	25.02
				16N	9	8	20	31	3.78	4.90	15.29	23.97
				32N	5	7	20	30	3.02	4.65	15.29	22.96
			알고 리즘 2	4N	-	-	27	27	-	-	14.72	14.72
				8N	-	-	15	15	-	-	9.86	9.86
				16N	-	-	14	14	-	-	9.47	9.47
				32N	-	-	14	14	-	-	9.34	9.34
FW5k	3067	9201	알고 리즘 1	4N	10	12	9	30	4.26	7.45	8.16	19.87
				8N	4	9	9	20	2.07	3.98	8.16	14.21
				16N	3	6	9	16	2.04	2.80	8.16	13.00
				32N	3	4	9	15	2.02	2.04	8.16	12.21
			알고 리즘 2	4N	-	-	14	14	-	-	8.07	8.07
				8N	-	-	10	10	-	-	6.91	6.91
				16N	-	-	10	10	-	-	6.87	6.87
				32N	-	-	9	9	-	-	6.87	6.87

[0200]

여기서 N은 근원지 블록 필터와 목적지 블록 필터가 저장하는 서로 다른 프리픽스의 개수보다 크거나 같으면서 가장 작은 2의 제곱수를 나타내고, 튜플 블록 필터의 경우에는 저장하는 서로 다른 튜플의 개수보다 크거나 같으면서 가장 작은 2의 제곱수를 나타낸다. FW5k의 경우, 물 셋내의 서로 다른 근원지 프리픽스의 개수가 67개이므로, 근원지 블록 필터의 크기 $N=2^{B1}=2^7=128$ (여기서, B1는 $\log_2 67$ 이상인 정수이므로 7이 됨)이 된다. 이와 마찬가지로, 서로 다른 목적지 프리픽스의 개수가 505개이므로, 목적지 블록 필터의 크기 $N=2^{B2}=2^9=512$ (여기서, B2는 $\log_2 505$ 이상인 정수이므로 9가 됨)이고, 튜플 블록 필터의 경우는 $N=2^{B3}=2^{11}=2048$ (여기서, B3는 $\log_2 1116$ 이상인 정수이므로 11이 됨)이다.

[0201]

알고리즘 1 및 알고리즘 2의 성능 평가를 간단하게 하기 위하여, 모든 블록 필터의 크기를 같은 비율로 증가시키며 실험하였다. 실험을 통하여 블록 필터의 크기가 커질수록 거짓-양성의 개수가 줄어들면서 음성 결과가 증가하는 블록 필터의 기본 특성을 확인할 수 있다. 이러한 블록 필터의 특성 덕분에 블록 필터의 크기를 가능한 크게 키우면, 칩 외부에 존재하는 해시 테이블에 대한 최대 메모리 접근 횟수와 평균 메모리 접근 횟수를 크게 줄일 수 있기 때문에, 더욱 빠른 검색을 수행할 수 있음을 알 수 있다. 표 5를 참조하면, 제안하는 알고리즘 2의 경우 블록 필터 크기 32N을 사용하면 평균 4~9번의 메모리 접근으로 패킷 분류가 가능함을 알 수 있다.

[0202]

진술한 알고리즘 1의 경우에는 근원지 블록 필터를 통과 후 거짓-양성 결과를 제거하기 위해 근원지 해시 테이블에 접근한다. 이와 마찬가지로 목적지 블록 필터를 통과 후 목적지 해시 테이블에 접근하여 거짓-양성 결과를 제거한다. 근원지 해시 테이블과 목적지 해시 테이블을 통과한 결과값(즉, 제 1 참 양성 비트 및 제 2 참 양성 비트)을 교차하여 만든 튜플들을 이용하여 튜플 해시 테이블에 접근하여 일치하는 최우선 순위 물을 찾아낸다. 그러므로, 알고리즘 1에서의 메모리 접근 횟수는 근원지 해시 테이블, 목적지 해시 테이블 및 튜플 해시 테이블의 접근 횟수를 합한 값과 같다.

[0203] 전술한 알고리즘 2의 경우에는 근원지 블록 필터와 목적지 블록 필터를 통과한 후, 거짓-양성의 결과를 포함한 양성의 결과값들(즉, 제 1 양성 비트 및 제 2 양성 비트)을 교차하여 얻어진 튜플들을 튜플 블록 필터를 통과시켜 이 과정에서 양성의 결과로 얻어지는 결과값들(즉, 양성 튜플)에 대해서만 튜플 해시 테이블 접근을 하게 된다. 알고리즘 2에서는 근원지와 목적지에 대하여 근원지 해시 테이블 및 목적지 해시 테이블에 접근하지 않으므로, 알고리즘 1보다 적은 수의 메모리 접근 횟수를 가지게 된다.

[0204] 도 27은 본 발명의 일 실시예에 따른 패킷 분류 알고리즘(알고리즘 1)의 실험 결과로서, 근원지 블록 필터/목적지 블록 필터 크기별 음성 결과 비율에 대한 실험 결과 그래프이다. 도 27을 참조하면, 근원지 블록 필터 및 목적지 블록 필터 각각의 크기가 커짐에 따라 음성 결과의 비율이 커짐을 알 수 있다. 음성의 결과 값에 대해서는 검색을 더 이상 진행하지 않아도 되므로 음성의 결과가 증가한다면 불필요한 메모리 접근 횟수를 줄일 수 있다.

[0205] 도 28은 본 발명의 일 실시예에 따른 패킷 분류 알고리즘(알고리즘 1)의 실험 결과로서, 근원지 블록 필터/목적지 블록 필터 크기에 따른 해시 테이블 접근 횟수에 대한 실험 결과 그래프이다. 도 28을 참조하면, 제안된 알고리즘 1에서 근원지 블록 필터 및 목적지 블록 필터의 크기가 커짐에 따라, 음성의 결과가 증가하고, 그에 따라 해시 테이블에 대한 전체적인 접근 횟수가 급격히 줄어들어 검색 성능이 좋아지는 것을 확인할 수 있다.

[0206] 표 6과 표 7에는 ACL5k, IPC5k, FW5k 를 셋의 블록 필터를 위한 메모리 사용량과 해시 테이블을 위한 메모리 사용량(Kbytes)을 나타내었다.

표 6

[0207]

룰 셋 타입	룰 개수	알고리즘	블록 필터 크기	블록 필터 크기			
				근원지	목적지	튜플	합
ACL5k	4660	알고리즘 1	4N	0.512	0.512	-	1.024
			8N	1.024	1.024	-	2.048
			16N	2.048	2.048	-	4.096
			32N	4.096	4.096	-	8.192
		알고리즘 2	4N	0.512	0.512	2.048	3.072
			8N	1.024	1.024	4.096	6.144
			16N	2.048	2.048	8.192	12.288
			32N	4.096	4.096	16.384	24.576
IPC5k	4468	알고리즘 1	4N	0.064	0.256	-	0.32
			8N	0.128	0.512	-	0.64
			16N	0.256	1.024	-	1.28
			32N	0.512	2.048	-	2.56
		알고리즘 2	4N	0.064	0.256	2.048	2.368
			8N	0.128	0.512	4.096	4.736
			16N	0.256	1.024	8.192	9.472
			32N	0.512	2.048	16.384	18.944
FW5k	3067	알고리즘 1	4N	0.064	0.256	-	0.32
			8N	0.128	0.512	-	0.64
			16N	0.256	1.024	-	1.28
			32N	0.512	2.048	-	2.56
		알고리즘 2	4N	0.064	0.256	1.024	1.344
			8N	0.128	0.512	2.048	2.688
			16N	0.256	1.024	4.096	5.376
			32N	0.512	2.048	8.192	10.752

표 7

[0208]

룰 셋 타입	룰 개수	알고리즘	해시 테이블 크기			
			근원지	목적지	튜플	합

ACL5k	4660	알고리즘 1	6.27	6.27	163.84	176.38
		알고리즘 2	-	-	163.84	163.84
IPC5k	4468	알고리즘 1	0.78	3.14	163.84	167.76
		알고리즘 2	-	-	163.84	163.84
FW5k	3067	알고리즘 1	0.78	3.14	81.92	85.84
		알고리즘 2	-	-	81.92	81.92

[0209] 각 블록 필터의 크기는 룰 셋 안의 서로 다른 프리픽스의 수보다 크거나 같으면서 가장 작은 2의 제곱수에 비례한다. 표에서 보여주는 바와 같이 블록 필터의 크기는 매우 작아 칩 내부에 저장할 수 있으며, 블록 필터의 기능을 향상시키기 위하여 4N, 8N, 16N 32N으로 늘려도 적은 메모리 공간을 차지하므로 전체 메모리의 사용량에 영향을 미치지 않는 것을 볼 수 있다. 칩 외부에 존재하는 근원지와 목적지 해시 테이블의 크기는 룰 셋에 존재하는 서로 다른 프리픽스의 수보다 크거나 같으면서 가장 작은 2의 제곱수이다. 튜플 해시 테이블의 크기는 룰 셋에 존재하는 룰의 수보다 크거나 같으면서 가장 작은 2의 제곱수이다.

[0210] 표 8 및 표 9에는 각 룰 셋에 대하여, 종래의 패킷 분류 알고리즘들(HSP-BST, H-trie, Set-Pruning Trie, AQT, PQT, BV)과 본 발명의 실시예로서 개시되는 2가지 알고리즘(알고리즘 1 및 알고리즘 2)의 평균 메모리 접근 횟수와 최대 메모리 접근 횟수를 비교하여 나타내었다. 종래의 알고리즘과 비교해보면, 적은 메모리 공간을 차지하는 블록 필터를 이용하여 불필요한 검색을 제거한 제안된 알고리즘(알고리즘 1 및 알고리즘 2)의 메모리 접근 횟수가 현저히 적은 것을 확인할 수 있다. 최대 메모리 접근 횟수의 경우에도 모든 룰 셋에서 우수한 검색 성능을 보이고 있다는 것을 알 수 있다.

표 8

[0211]

알고리즘 룰		알고리즘 1	알고리즘 2	HSP-BST	H-trie	set-pruning Trie	AQT	PQT	BV
ACL	1k	10.3	2.9	13.8	77.2	67.6	38.6	35.6	66.0
	5k	13.2	3.9	17.8	84.0	67.6	50.1	59.6	64.1
IPC	1k	12.6	1.5	13.3	71.9	50.9	94.5	73.6	63.6
	5k	23.0	9.3	17.9	85.6	59.8	344.8	202.1	151.9
FW	1k	12.0	5.4	16.2	52.1	39.4	369.3	197.9	196.6
	5k	12.2	6.9	24.7	67.5	52.2	660.5	571.1	738.8

표 9

[0212]

알고리즘 룰		알고리즘 1	알고리즘 2	HSP-BST	H-trie	AQT	PQT	BV
ACL	1k	14	4	68	124	64	75	68
	5k	14	6	68	177	94	113	76
IPC	1k	14	4	68	128	119	106	80
	5k	21	13	70	192	415	2950	230
FW	1k	12	7	69	117	444	293	318
	5k	14	9	69	146	1193	999	1044

표 10

[0213]

알고리즘 룰		알고리즘 1	알고리즘 2	HSP-BST	H-trie	set-pruning Trie	AQT	PQT	BV
ACL	1k	24.0	20.5	27.9	102.5	138.6	56.4	29.9	153.3
	5k	176.4	163.8	285.1	509.0	990.9	200.2	145.6	2793.0
IPC	1k	25.2	20.5	184.5	157.2	655.7	71.2	30.9	154.3
	5k	167.8	163.8	438.2	246.8	620.4	234.3	139.9	2351
FW	1k	22.4	20.5	163.6	41.9	219.0	35.2	27.2	111.9
	5k	85.8	81.9	293.8	109.9	491.8	479.8	136.0	2340

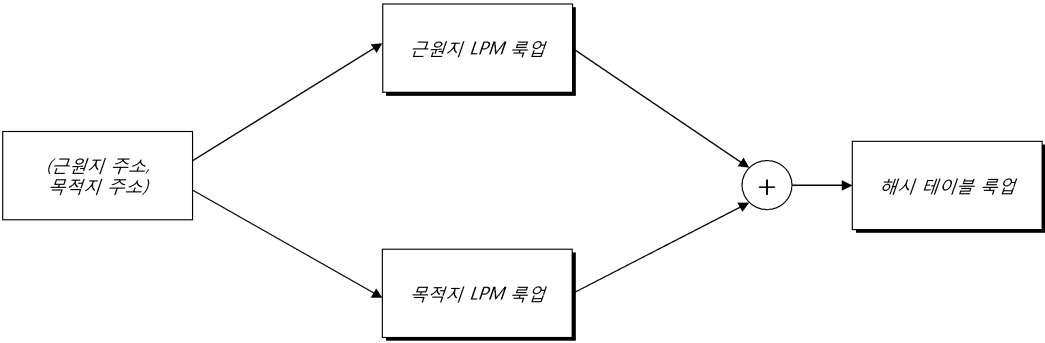
- [0214] 상기 표 10은 메모리 요구량(Kbytes)을 비교한 표이다. 다른 알고리즘과 비교하였을 때 제안하는 알고리즘은 상대적으로 적은 메모리를 요구하는 것을 알 수 있으며, 가장 적은 메모리를 요구하는 HSP-BST 구조보다도 더 적은 메모리를 요구하는 것을 확인할 수 있다. 메모리 접근 횟수와 메모리 요구량에 대한 실험결과를 살펴보았을 때, 본 발명에서 실시예로서 개시되는 알고리즘 1과 알고리즘 2는, 기존의 존재하는 패킷 분류 알고리즘들에 비해 메모리 요구량과 메모리 접근 횟수 모두 다른 구조보다 좋은 성능을 보이는 것을 알 수 있다.
- [0215] 다시 말해, 이상에서 설명한 바와 같이 본 발명의 실시예에 의하면, 메모리 요구량 및 메모리 접근 횟수를 줄임으로써, 라우터 등의 패킷 분류 장치에 부하를 주지않으면서 더욱 효율적이고 신속한 패킷 분류를 가능하게 하는 효과가 있다.
- [0216] 또한 본 발명의 실시예에 의하면, 전술한 효율적이고 신속한 패킷 분류를 통해, 입력 패킷에 대한 해당 서비스 품질(QoS: Quality of Service)을 더욱 신속히 보장해줄 수 있는 효과도 있다.
- [0217] 이상에서, 본 발명의 실시예를 구성하는 모든 구성 요소들이 하나로 결합되거나 결합되어 동작하는 것으로 설명되었다고 해서, 본 발명이 반드시 이러한 실시예에 한정되는 것은 아니다. 즉, 본 발명의 목적 범위 안에서라면, 그 모든 구성 요소들이 하나 이상으로 선택적으로 결합하여 동작할 수도 있다. 또한, 그 모든 구성 요소들이 각각 하나의 독립적인 하드웨어로 구현될 수 있지만, 각 구성 요소들의 그 일부 또는 전부가 선택적으로 조합되어 하나 또는 복수 개의 하드웨어에서 조합된 일부 또는 전부의 기능을 수행하는 프로그램 모듈을 갖는 컴퓨터 프로그램으로서 구현될 수도 있다. 그 컴퓨터 프로그램을 구성하는 코드들 및 코드 세그먼트들은 본 발명의 기술 분야의 당업자에 의해 용이하게 추론될 수 있을 것이다. 이러한 컴퓨터 프로그램은 컴퓨터가 읽을 수 있는 저장매체(Computer Readable Media)에 저장되어 컴퓨터에 의하여 읽혀지고 실행됨으로써, 본 발명의 실시예를 구현할 수 있다. 컴퓨터 프로그램의 저장매체로서는 자기 기록매체, 광 기록매체, 캐리어 웨이브 매체 등이 포함될 수 있다.
- [0218] 또한, 이상에서 기재된 "포함하다", "구성하다" 또는 "가지다" 등의 용어는, 특별히 반대되는 기재가 없는 한, 해당 구성 요소가 내재될 수 있음을 의미하는 것이므로, 다른 구성 요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있는 것으로 해석되어야 한다. 기술적이거나 과학적인 용어를 포함한 모든 용어들은, 다르게 정의되지 않는 한, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 사전에 정의된 용어와 같이 일반적으로 사용되는 용어들은 관련 기술의 문맥 상의 의미와 일치하는 것으로 해석되어야 하며, 본 발명에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0219] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 발명에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 본 발명의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

도면의 간단한 설명

- [0220] 도 1은 패킷 분류를 위한 종래의 튜플 제거 알고리즘 구조를 나타낸 도면,
- [0221] 도 2는 본 발명의 일 실시예에 따른 패킷 분류 장치에 대한 블록 구성도,
- [0222] 도 3은 본 발명의 일 실시예에 따른 패킷 분류 장치에서, 근원지 블룸 필터 검색부를 나타낸 도면,
- [0223] 도 4는 본 발명의 일 실시예에 따른 패킷 분류 장치에서, 근원지 블룸 필터 검색부에 포함된 제 1 해싱부를 나타낸 도면,
- [0224] 도 5는 본 발명의 일 실시예에 따른 패킷 분류 장치에서, 목적지 블룸 필터 검색부를 나타낸 도면,
- [0225] 도 6은 본 발명의 일 실시예에 따른 패킷 분류 장치에서, 목적지 블룸 필터 검색부에 포함된 제 2 해싱부를 나타낸 도면,
- [0226] 도 7은 본 발명의 일 실시예에 따른 패킷 분류 장치에서, 튜플 해시 테이블 검색부를 나타낸 도면,
- [0227] 도 8은 본 발명의 일 실시예에 따른 패킷 분류 장치에서, 튜플 해시 테이블 검색부에 포함된 룰 추출부를 나타

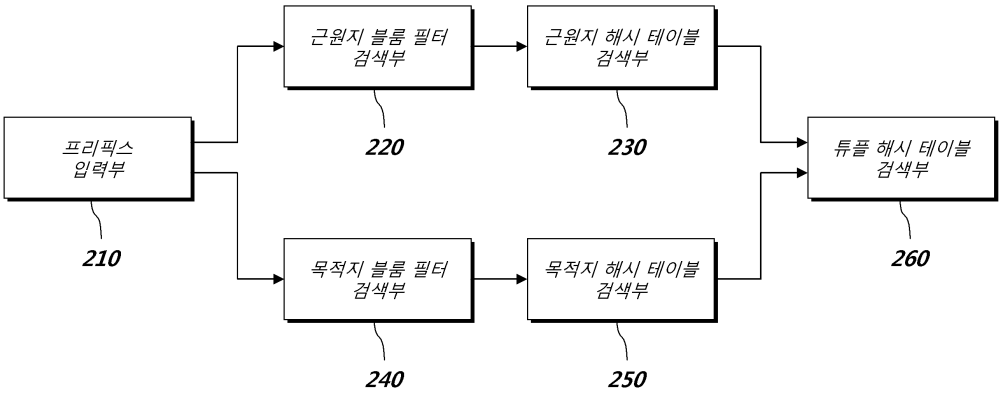
도면

도면1

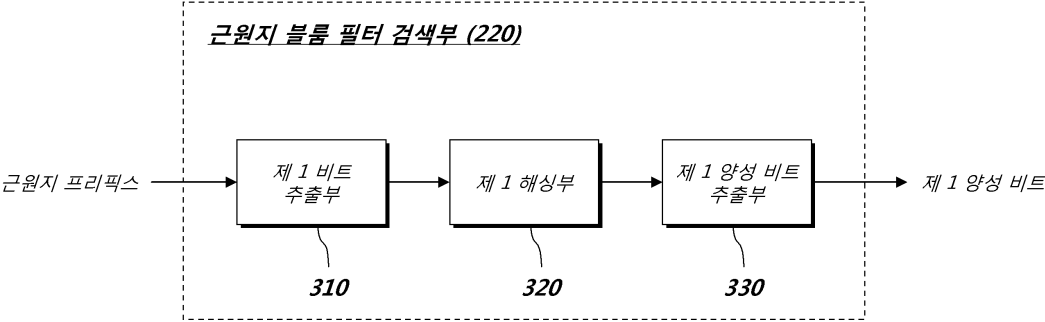


도면2

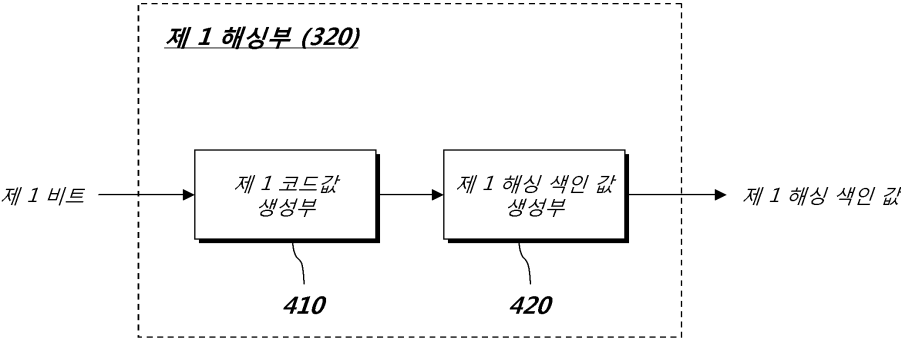
패킷 분류 장치 (200)



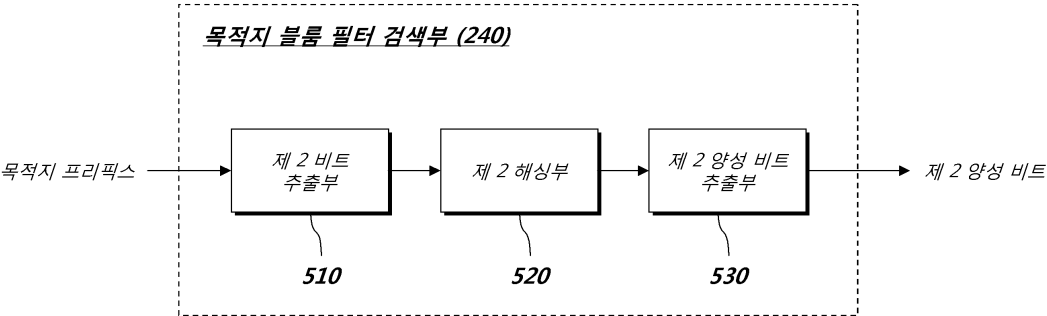
도면3



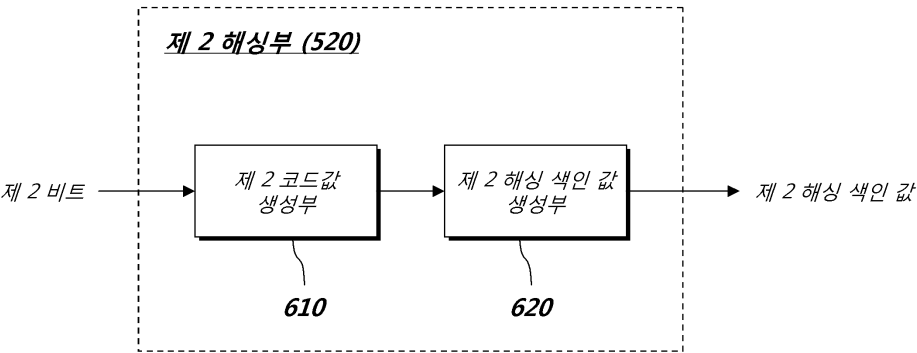
도면4



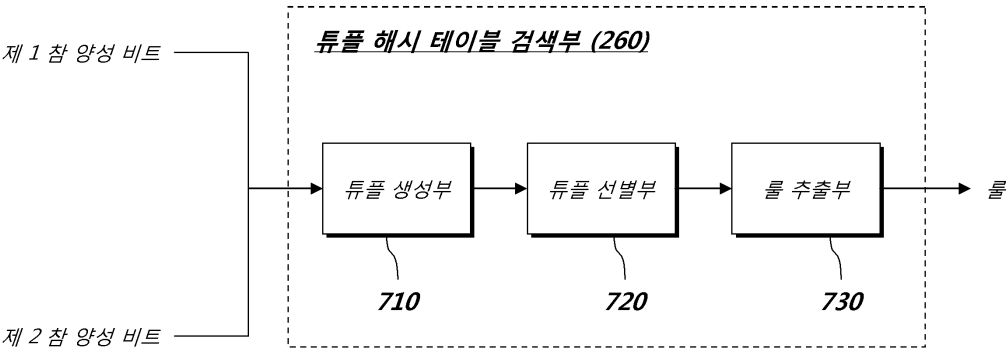
도면5



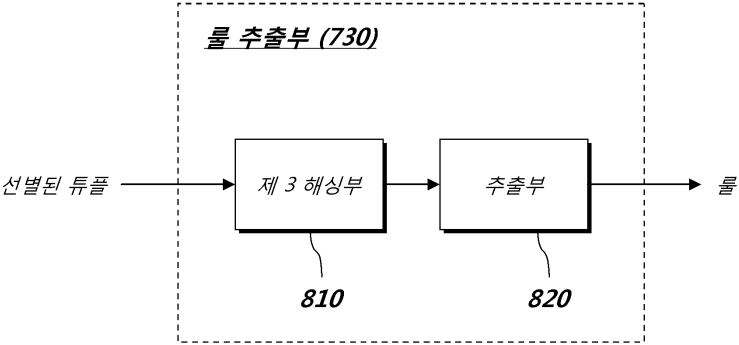
도면6



도면7

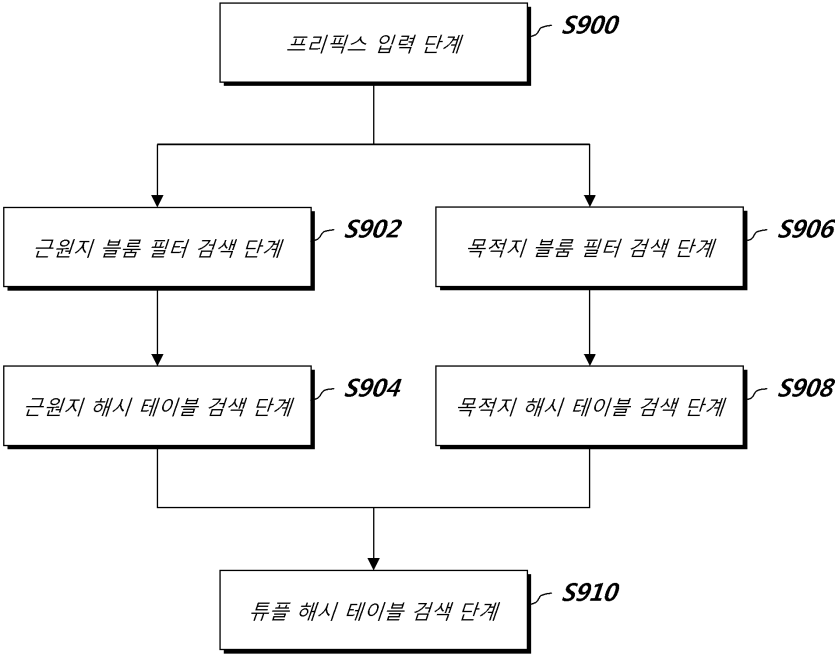


도면8

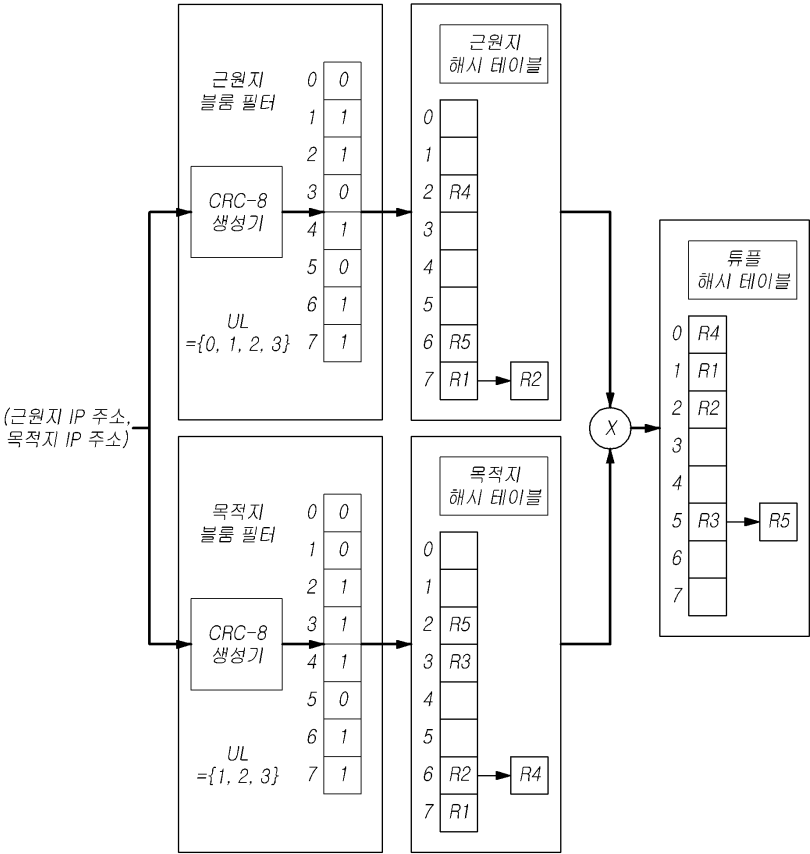


도면9

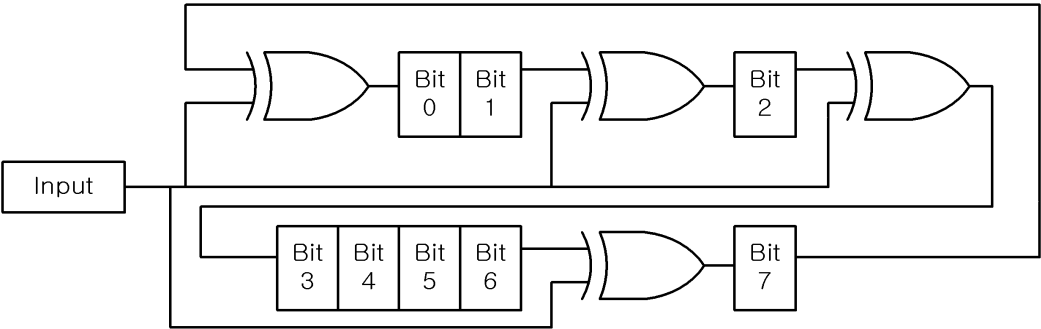
패킷 분류 방법



도면10



도면11



도면12

근원지 블록 필터 :

0	1	2	3	4	5	6	7
0	1	1	0	1	0	1	1

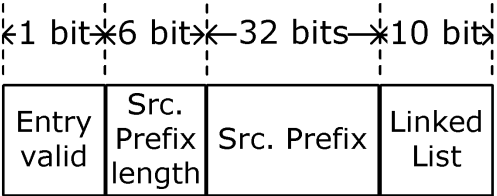
UL : {0,1,2,3}

목적지 블록 필터 :

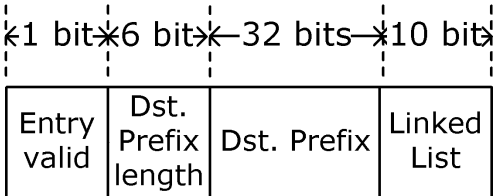
0	1	2	3	4	5	6	7
0	0	1	1	1	0	1	1

UL : {1,2,3}

도면13



(a)



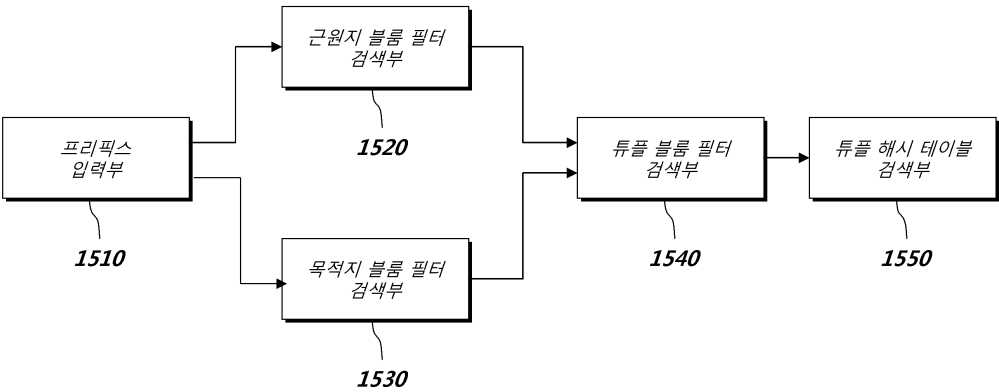
(b)

도면14

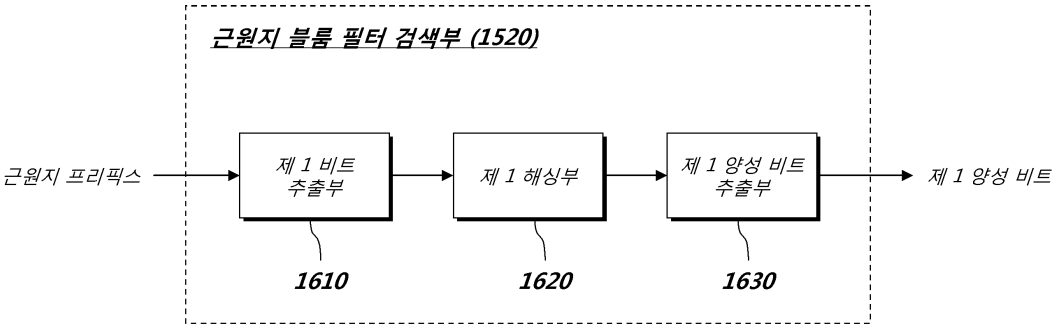
1 bit	6 bit	32 bits	6 bit	32 bits	16 bits	16 bits	16 bits	16 bits	1 bit	8 bit	10 bit
Entry valid	Src. Prefix length	Src. Prefix	Dst. Prefix length	Dst. Prefix	Source Port Start	Source Port End	Dst. Port Start	Dst. Port End	Protocol wild	Protocol type	Linked List

도면15

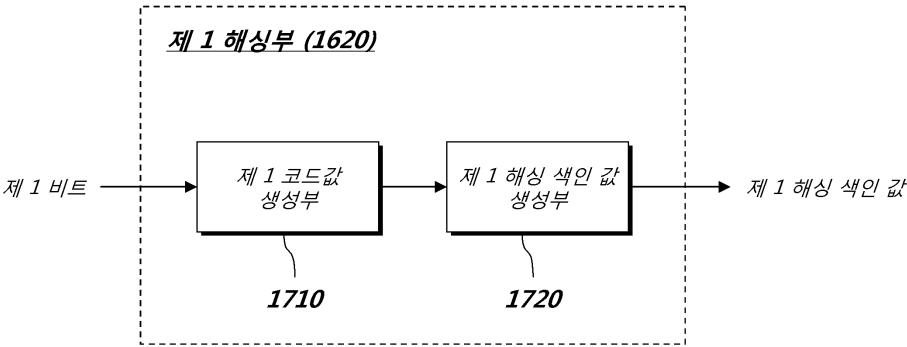
패킷 분류 장치 (1500)



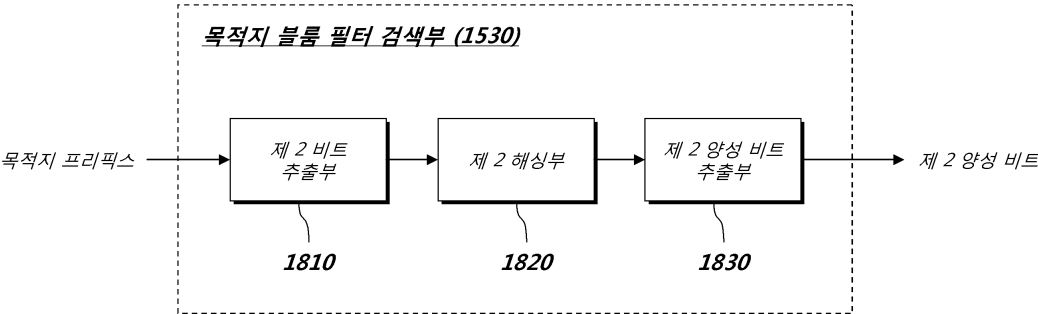
도면16



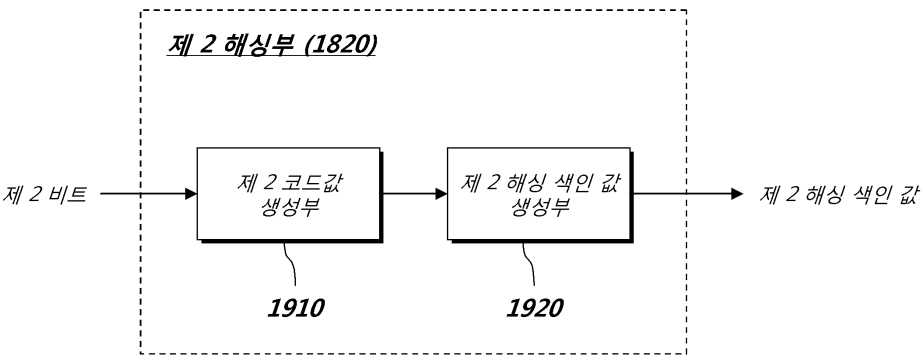
도면17



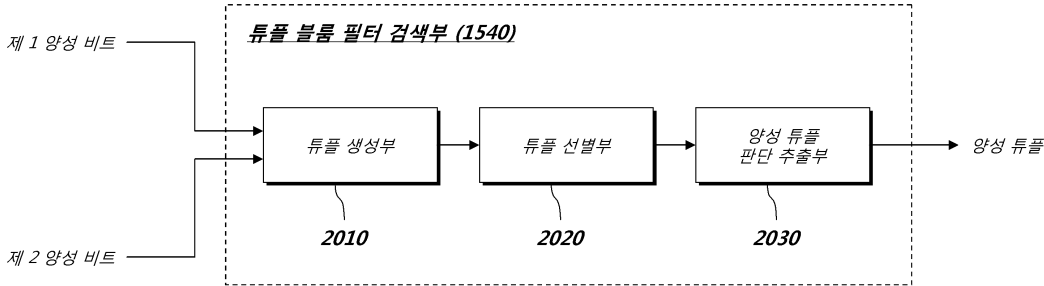
도면18



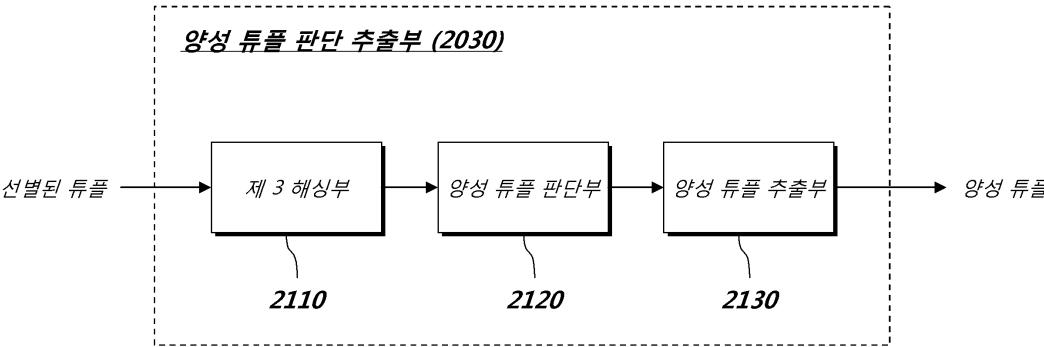
도면19



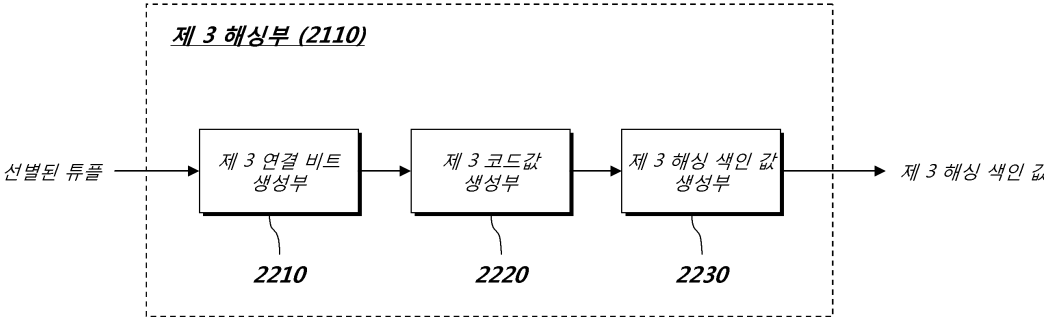
도면20



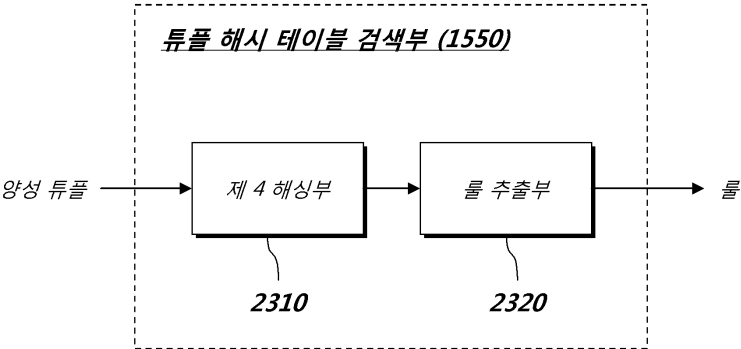
도면21



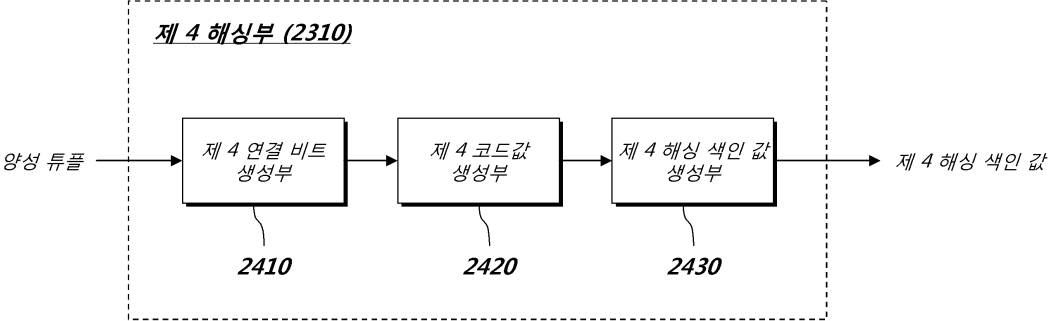
도면22



도면23

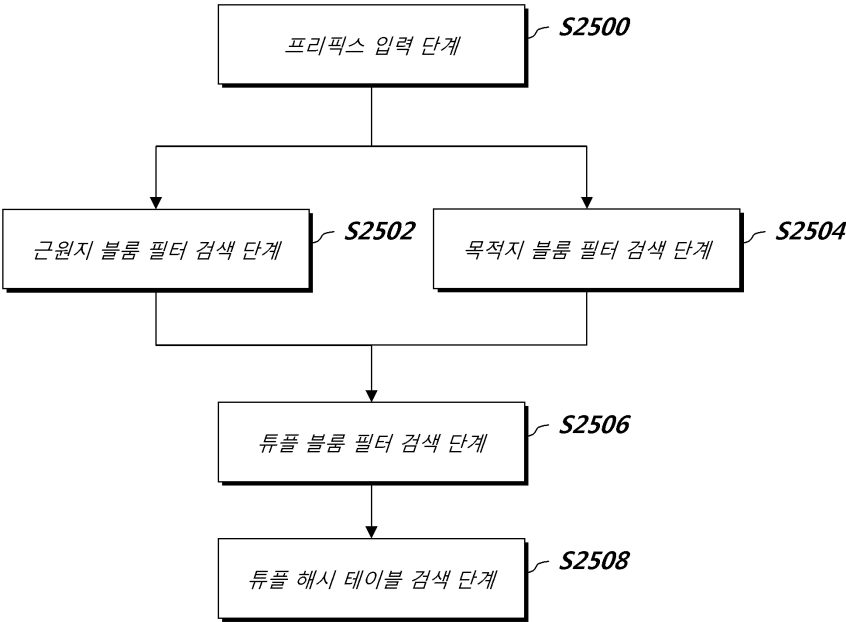


도면24

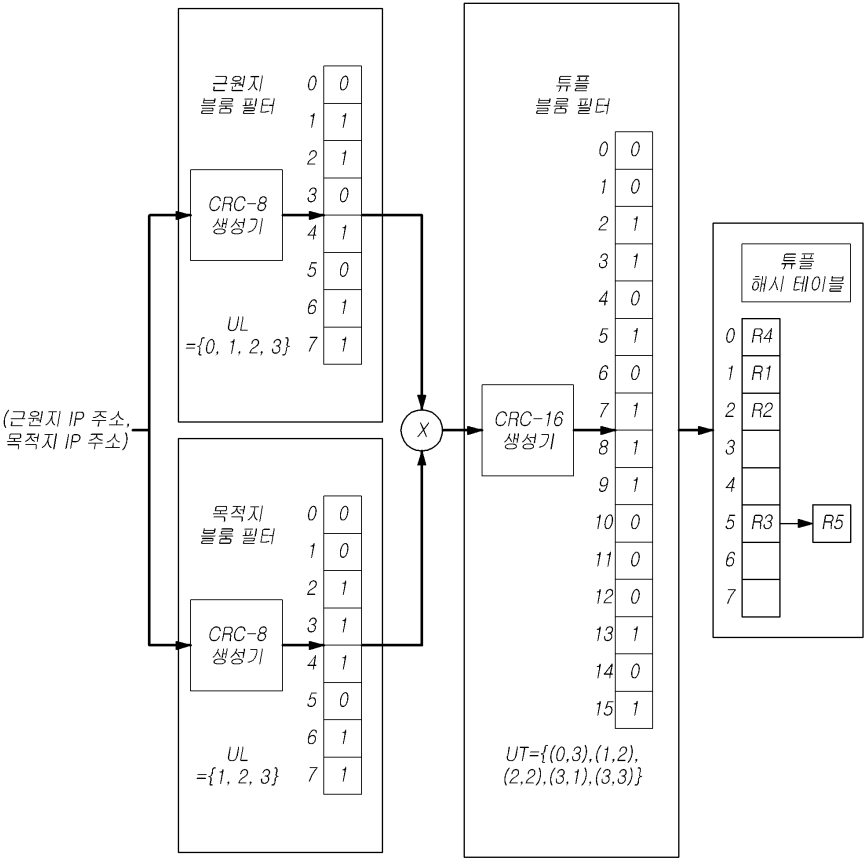


도면25

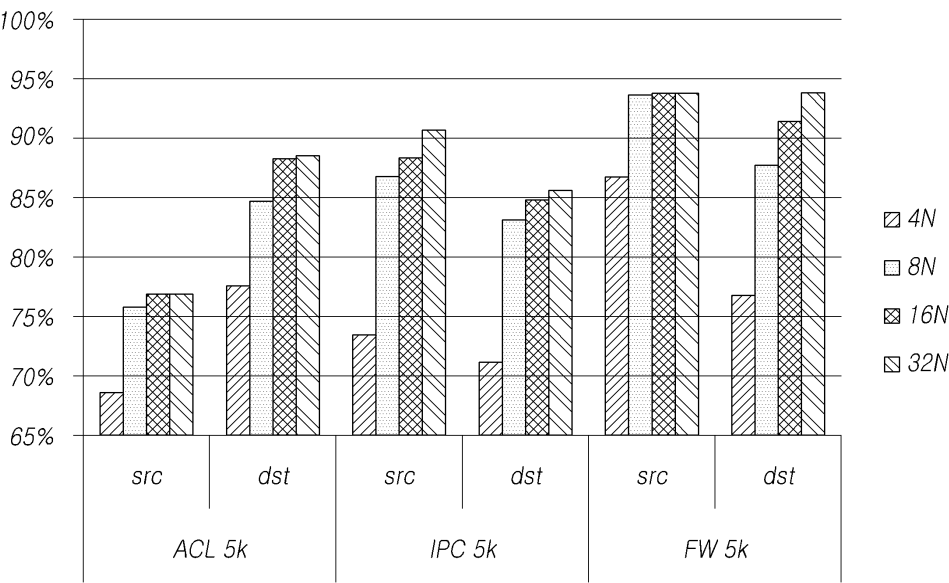
패킷 분류 방법



도면26



도면27



도면28

