http://dx.doi.org/10.3745/KIPSTC.2012.19C.3.173

# 블룸필터를 사용하는 두 보안기법에 대한 메시지 길이의 효율성에 대하여

맹 영 재[†] · 강 전 일[††] · 양 대 헌[†††] · 이 경 희[††††]


## 요 약

블룸필터를 이용하면 다수의 MAC을 표현하기 위해 요구되는 메시지의 길이를 줄일 수 있다고 주장하는 두 논문이 최근에 발표되었다. 하지만 이 기법들은 보안성을 고려하지 않고 메시지의 길이만 비교한 것으로 분석되었다. MAC은 보안을 목적으로 하는 코드이기 때문에 다수의 MAC과 이들을 표현한 블룸필터가 동등한 보안수준을 가지도록 하고 메시지의 길이를 비교해야 한다. 이 논문에서는 블룸필터와 압축블룸필터, 그리고 다수의 MAC이 동등한 보안수준을 가질 때의 메시지 길이를 분석하여 보인다.


**키워드 : 압축블룸필터, 블룸필터, MAC, 공간효율**

# On Message Length Efficiency of Two Security Schemes using Bloom Filter

YoungJae Maeng[†] · Jeonil Kang[††] · DaeHun Nyang[†††] · KyungHee Lee[††††]


## ABSTRACT

Recent two security schemes showed that a bloom filter can reduce a message length required for representing multiple MACs. The schemes, however, made message length comparison without considering security level. Since the MAC is intended for security, it is important to let multiple MACs and the bloom filter have the same level of security for making message length comparison. In this paper, we analyze the message length efficiency of bloom filter, compressed bloom filter and multiple MACs, letting them have the same security level.


**Keywords : Compressed Bloom Filter, Bloom Filter, MAC, Space Efficiency**

## 1. 서 론

다수 MAC(Message Authentication Code)의 길이를 줄이기 위한 목적으로 블룸필터를 이용한 기법이 두 편 발표되었다. 정석재 등은 차량네트워크 연구 중 하나인 RAISE[5]에서 사용된 다수의 MAC을 블룸필터에 표현한 결과로 메시지의 길이를 줄였으며[1], 최임성 등이 소개한 센서네트워크에서의 방송형 인증 기법 또한 블룸필터를 이용하여 다수 MAC 전송에 필요한 통신량을 줄였다[2].

하지만 메시지의 길이를 줄이는 것이 목적이라면 길이가 짧은 MAC을 사용하는 것도 방법이 될 수 있다. 즉, 블룸필터를 사용하지 않아도 메시지의 길이를 줄일 수 있다. 이때, 그 두 접근방법의 메시지 길이를 비교할 때의 비교기준은 보안성이다. 블룸필터의 긍정오류와 다수 MAC이 동등한 보안수준을 유지할 때의 메시지 길이를 비교해야 한다. 하지만 위의 두 논문에서는 그러한 기준을 설정하지 않고 비교하였다. 특히, [1]에서 보인 메시지 길이 비교는 공평하지 않다(그림 1). 메시지의 길이는 그 구조체가 가지는 공간효율에 따라 정해진다.

이 논문에서는 다수 MAC을 하나의 블룸필터에 표현한 방법과 MAC을 잘라 사용하는 방법이 동등한 보안수준을 가질 때의 공간효율을 비교하였다. 이에 더해서 압축블룸필터 또한 함께 비교하였다. 2장에서 블룸필터의 긍정오류와 MAC의 약한 충돌 저항성에 소개하고 3장에서는 블룸필터와 압축블룸필터 그리고 MAC이 동등한 보안수준을 유지할 때의 공간효율 및 장단점을 비교분석하고 4장에서는 결과를 보인다.

※ 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2010-0013254).
† 준 회 원 : 인하대학교 컴퓨터정보공학과 박사과정
†† 준 회 원 : 인하대학교 정보공학과 박사과정
††† 정 회 원 : 인하대학교 컴퓨터정보공학과 부교수
†††† 종신회원 : 수원대학교 전자공학과 교수(교신저자)
논문접수 : 2012년 1월 13일
수 정 일 : 1차 2012년 5월 16일
심사완료 : 2012년 5월 18일

## 2. 블룸필터와 MAC의 보안성

### 2.1 블룸필터의 긍정오류

블룸필터(Bloom filter)는 1970년 Burton Howard Bloom이 고안한 통계적 특성을 갖는 데이터 구조의 일종으로써, 밀집되지 않은 데이터를 저장하고 검색하기 좋은 구조를 가지고 있다. 하지만 블룸필터의 구조상 긍정오류(false positive, 이하 $f$)를 가지기 때문에 이에 대한 대비책이 마련되어 있거나 그러지 않아도 무방한 경우(필터링이나 검색 등)에 주로 사용되고 있다.

블룸필터의 비트열 크기 $m$과 데이터의 개수 $n$은 이를 사용할 시스템을 설계할 때 정해진다. 입력 데이터에 대한 해시값은 $m$에서 임의적인 위치를 가리키며, 이때 사용할 해시함수의 개수는 가장 낮은 $f$(가장 높은 보안성)를 가지고자 할 때 $k=(-\ln p)(m/n)$와 같이 정한다. $p$는 $m$에서의 어느 한 비트가 '0'일 확률이다. 최대 복잡도를 가지는 블룸필터를 가정하면 절반이 '0'이 되어야 하므로 $p=1/2$라 했을 때, $k=(m/n)\ln 2$와 같이 정할 수 있다. $n$개의 데이터가 $k$번씩 $m$에 모두 표현된 이후에 어떤 비트가 '0'일 확률 $p$는 아래와 같다.

$$p=(1-\frac{1}{m})^{kn}\approx e^{-kn/m} \tag{1}$$

이와 관련하여, 의도하지 않은 데이터에 대한 $k$개의 해시값이 가리키는 위치가 충돌로 인해 모두 '1'로 설정되어 있을 확률, 즉 $f_{bf}$는 아래와 같다.

$$f_{bf}=\left(1-\left[1-\frac{1}{m}\right]^{kn}\right)^k\approx\left(1-e^{-kn/m}\right)^k=(1-p)^k \tag{2}$$

실제로는 '1'이 독립적으로 표현되지 않을 수 있기 때문에 엄밀히 따지면 $m$상의 어떤 비트가 '1'일 확률이 $1-p$은 아니지만 실용적인 환경에서는 이는 무시해도 무방하다[3].

### 2.2 $n$개의 MAC과 약한 충돌 저항성 (second-preimage resistance)

해시함수나 블록암호기법을 통해 구현될 수 있는 MAC은 메시지의 무결성(그리고 때에 따라 인증)을 제공한다. 길이가 $s$인 MAC이 가지는 보안성은 $1/2^s$이며, 의도하지 않은 데이터의 MAC이 $n$개의 MAC 중 하나와 충돌할 확률은 $f_{mac}=n/2^s$이다. 이 확률은 $n$개의 개체가 속하는 공간 전체에 대해 충돌여부를 평가하는 것이기 때문에, 블룸필터의 $f$와 같다고 할 수 있다. 따라서, 블룸필터의 $f$와 $n$개의 MAC이 가지는 약한 충돌 저항성은 보안성 측면에서 동등한 조건이며, 메시지 길이를 비교하기 위한 기준이 될 수 있다.

### 2.3 압축블룸필터의 긍정오류

Michael Mitzenmacher가 소개한 압축블룸필터[3]는 블룸필터에 산술 부호(Arithmetic Coding)를 적용한 것으로, 최종적으로 전송할 데이터의 크기를 줄이기 위한 목적으로 제안되었다. 최적 $k$를 사용하는 블룸필터는 모든 자리의 비트가 대략 1/2의 확률로 '1'인 비트열이고, 이미 최대 복잡도를 가지기 때문에 압축으로 인한 이득은 얻기 힘들다. 압축블룸필터는 원하는 압축률을 얻어내기 위해 의도적으로 $p$가 높아지도록 $1\le k<(m/n)\ln 2$ 와 같이 설정하며, 이때 최적 $k$를 사용했을 때와 동등한 $f$를 가지기 위해서는 블룸필터의 길이 $m$이 늘어나게 된다. 이때의 $m$에 binary entropy function $H(p)=-p\log_2 p-(1-p)\log_2(1-p)$을 적용한 $z=mH(p)$는, 일반블룸필터에 비해 결과적으로 약 5~15%정도 짧아지는 효과가 있다. 압축블룸필터의 긍정오류 $f_{cbf}$는 $k$에 $z=mH(p)$을 적용한 $(1-p)^{-(z\ln p/nH(p))}$와 같다.

## 3. 동등한 보안성상에서의 공간효율 분석

### 3.1 블룸필터와 MAC의 최대 복잡도 비교

128비트의 출력을 갖는 해시함수에 최대 복잡도를 갖는 128비트의 입력이 주어졌을 때, 이 해시함수의 결과물에서 잃어버린 복잡도는 해시함수가 이상적인 경우 0이다. 128비트의 입력에 대해 최대 복잡도를 가지는 블룸필터는 그 크기의 절반이 1로 세팅되어야 하므로, ${}_lC_{l/2}>2^{128}$인 최소 $l$이다. 여기서 $l=132$가 된다.

132비트의 크기의 블룸필터에 추가적인 데이터를 표시하는 경우에는 다른 데이터와 충돌을 일으키게 되고 블룸필터에 66개보다 많지만 132개보다 작은 수의 비트가 1로 세팅될 것이다. 이때의 블룸필터는 최대 복잡도를 잃게 되고 최대 복잡도를 유지하기 위해서는 블룸필터의 크기가 더 커져야한다. 데이터의 개수를 $n$, 해시의 출력의 크기를 $s$라 할 때, 블룸필터가 해시 함수보다 더 큰 복잡도를 유지할 수 있다면

$$2^{sn}<{}_{sn}C_{sn/2}=\frac{(sn)!}{((sn/2)!)^2} \tag{3}$$

를 만족하는 최소한의 $n$과 $s$이 있어야 한다. ${}_{sn}C_{sn/2}$는 스털링 근사(Stirling's approximation)[4]에 따라

$$\sqrt{2\pi}\,n^{n+1/2}e^{-n}\le n!\le en^{n+1/2}e^{-n} \tag{4}$$

$$\sqrt{2\pi}\,(sn)^{sn+1/2}e^{-sn}\le (sn)!\le e(sn)^{sn+1/2}e^{-sn} \tag{5}$$

$$2\pi(sn/2)^{sn+1}e^{-sn}\le((sn/2)!)^2\le e^2(sn/2)^{sn+1}e^{-sn} \tag{6}$$

$$\sqrt{\frac{2}{\pi sn}} \times 2^{sn} = \frac{\sqrt{2\pi}(sn)^{sn+1/2}}{2\pi(sn/2)^{sn+1}} \leq {}_{sn}C_{sn/2} \leq \frac{(sn)^{sn+1/2}}{e(sn/2)^{sn+1}} = \frac{1}{e}\sqrt{\frac{4}{sn}} \times 2^{sn} \quad (7)$$

과 같다. 이를 계산하면 $0 < \sqrt{\frac{2}{\pi sn}} < 0.798$, $0 < \frac{1}{e}\sqrt{\frac{4}{sn}} < 0.368$ 이고, 모든 양의 정수 $s$과 $n$에 대해서 ${}_{sn}C_{sn/2} < 2^{sn}$ 이므로 식(3)을 만족하는 $s$과 $n$은 존재하지 않는다. 이 계산 결과는 블룸필터가 해시함수 대신 써서 정보를 압축하여 저장할 수는 없음을 보여준다.

### 3.2 동등한 긍정오류에서의 공간효율 비교

3.1장과 같은 결과는 블룸필터와 해시 함수의 효율성 분석은 사용되는 구조체나 출력 데이터의 크기가 아니라 긍정오류를 기준으로 분석해야 함을 보여준다. 블룸필터의 긍정오류는 구조체 전체 $m$에 대하여 발생되지만 해시 함수의 경우 각각의 고정된 길이를 갖는 출력물 $s$에 대해서 개별적으로 발생되기 때문이다. 따라서 두 방법의 긍정오류를 고정시키고 블룸필터가 MAC보다 더 공간효율적이기 위한 조건($m < ns$)을 분석하여 보았다.

블룸필터의 긍정오류 $f_{bf}$를 최소화하기 위한 $k$는 $(m/n)\ln 2$, 최대 복잡도를 위한 $p$는 1/2이므로, $f_{bf} = (1/2)^{(m/n)\ln 2}$와 같고, $m = -n\ln f_{bf}/(\ln 2)^2$과 같다. MAC이 갖는 긍정오류의 경우 $f_{mac} = n/2^s$와 같고 블룸필터가 해시함수보다 더 효율적이어야 하므로, $f_{bf} = f_{mac}$로 둘 때,

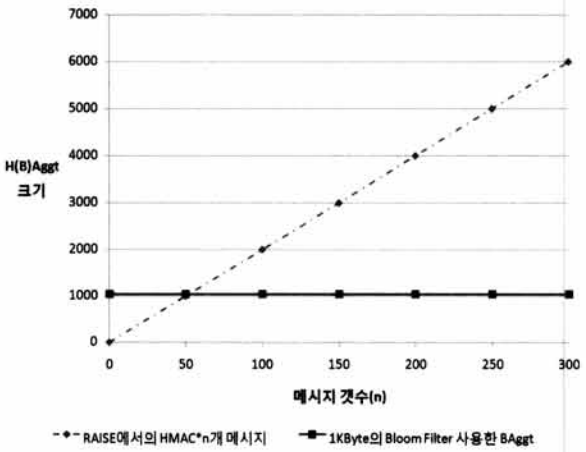
$$m = -\frac{n\ln f_{bf}}{(\ln 2)^2} = -\frac{n\ln f_{mac}}{(\ln 2)^2} = -\frac{n\ln n/2^s}{(\ln 2)^2} < ns \quad (8)$$

를 동시에 만족하는 $n$과 $s$를 찾아야 한다. 식을 전개하면

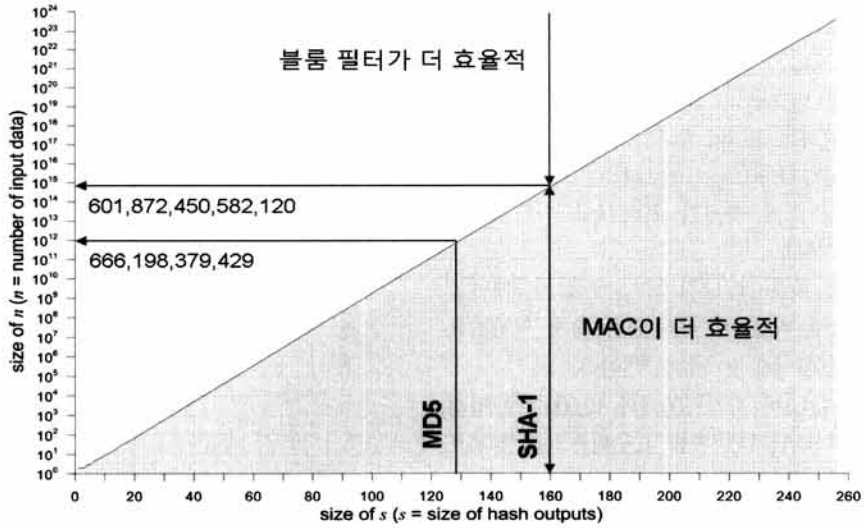
$$n > e^{s(\ln 2(1-\ln 2))} \approx 2^{0.3068 \times s} \quad (9)$$

와 같다. 이 식을 그래프로 표현하면 (그림 2)와 같다.

(그림 2)는 해시 함수의 출력크기가 $s$로 주어졌을 때, 주어진 선분보다 위쪽의 데이터의 개수 $n$이 들어오면 블룸필터(또는 압축블룸필터)가 더 효율적이라는 것이고, 반대라면 MAC이 더 효율적이라는 것을 의미한다. 예를 들어, MD5와 같은 해시 함수를 이용한 MAC의 긍정오류 수준을 갖기 위해서 블룸 필터는 $2^{40}$개 정도의 데이터가 있어야 한다. 이는 대략, 16TB 정도이다.




(그림 1) [1]에서 보인 메시지 개수 n에 따른 메시지 길이 비교




(그림 2) 동등한 $f$를 기준으로 한 MAC과 블룸필터의 효율성 분석

〈표 1〉 주어진 $f$에 대한 $bpe$계산식

| $\frac{m}{n}=-\frac{\ln f_{bf}}{(\ln 2)^2}$ | $s=\left\lceil -\lg\frac{f_{mac}}{n}\right\rceil$ | $\frac{z}{n}=-\frac{H(p)\log_{(1-p)}f_{dbf}}{\ln p}$ |
|---|---|---|
| 블룸필터 | 다수의 MAC | 압축블룸필터 |

〈표 2〉 $n=10{,}000$에 대한 $bpe$ 비교

| $s$ | 17 | 17 | 16 | 16 | 17 | 18 | 16 | 25 | 25 | 25 | 26 | 25 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $m/n$ | 7 | 4 | 46 | 12.6 | 14 | 92 | 8 | 93 | 37.5 | 28 | 48 | 16 |
| $z/n$ | 3.962 | 4 | 6.891 | 7.582 | 7.923 | 7.923 | 8 | 13.815 | 14.666 | 15.846 | 15.829 | 16 |
| $k$ | 1 | 3 | 1 | 2 | 2 | 1 | 6 | 2 | 3 | 4 | 3 | 11 |
| $f$ | 0.133 | 0.147 | 0.0215 | 0.0216 | 0.0177 | 0.0108 | 0.0216 | 0.000453 | 0.000454 | 0.000314 | 0.000222 | 0.000459 |

### 3.3 동등한 긍정오류와 데이터 개수에 따른 크기비교

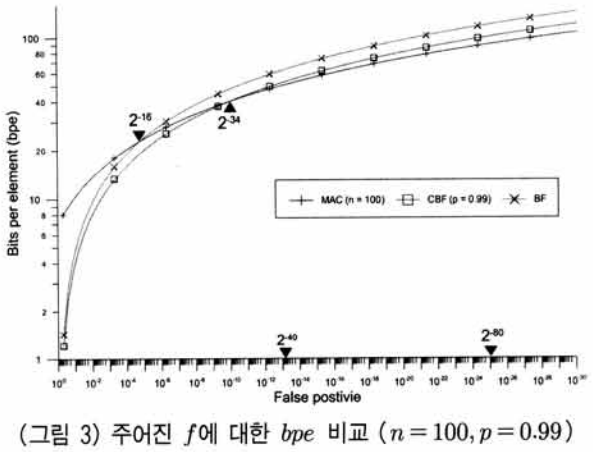
이 장에서는 데이터 하나를 표시하기 위해 요구되는 비트수(bits per element, 이하 $bpe$)를 비교한다. 이때 유의해야 할 점은 동등한 $f$를 기준으로 삼아야 한다는 것이다. 각 방법의 $f$계산식은 2장에 소개되어 있으며 이 식을 $bpe$에 대해 전개하면 <표 1>과 같다.

MAC의 경우, $bpe$에 영향을 주는 요소는 $f$외에도 $n$이 있다. 입력되는 데이터 개수가 $f$를 선형적으로 증가시키고 결국 $s$의 크기 또한 증가시키는 것을 알 수 있다. 압축블룸필터의 경우, $bpe$에 영향을 주는 요소는 $f$와 $p$이다. 확률 $p$가 양극에 가까울수록 $H(p)$는 높은 압축률을 보이고 $bpe$를 낮추는 효과가 있다. 이론적으로는, 그러한 $p$를 만들기 위해 최소 $k=1$ 즉, $n$개의 비트만을 '1'로 표현하도록 하고 주어진 $f$에 대한 블룸필터의 크기를 $m\approx n/f$와 같이 정할 수 있다. 하지만 이렇게 정한 $m$은 너무 커서 현실적이지 않게 된다.

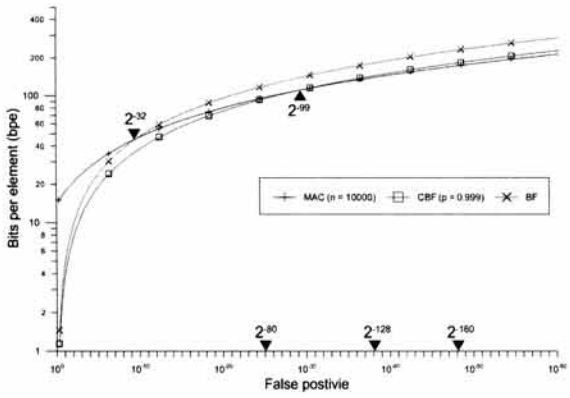
<표 1>의 $bpe$계산식을 참고하여 $n$을 고정한 상태에서 $f$가 낮아짐에 따라 요구되는 각 방법의 $bpe$를 계산하고 비교하여 보았다. 낮은 $bpe$는 높은 공간 효율을 의미한다. <표 2>는 주어진 $f$를 만족하기 위한 $s$를 [3]의 연구에서 보인 수치와 비교한 결과이다. 이 결과의 $s$는 모든 경우에서 $z/n$보다 크게 나타났다($m/n$이 $s$보다 더 높게 나타난 경우는 $H(p)$의 효율을 크게 가지기 위해 $k$를 의도적으로 작게 설정하였기 때문이다).

다만, 이러한 결과는 제공되는 $f$가 낮은 수준(표 2에서는 최소 $f$가 $2^{-12}$정도임)에 머물 때 나타난다. 현재 일반적으로 사용되는 MAC의 보안성은 $2^{-128}$~$2^{-160}$이다.

(그림 3)과 (그림 4)는 $n$이 각각 100개와 10,000개로 정해졌을 때, 목표하는 $f$에 도달하기 위해 요구되는 $bpe$를 <표 1>의 계산식에 대입한 결과이다. 입력의 개수가 100일 때, MAC은 $2^{-16}$이상의 $f$에서 블룸필터보다 $bpe$가 낮았고, $2^{-34}$이상의 $f$에서는 $p=0.99$인 압축블룸필터보다 $bpe$가 낮았다(참고로, $p=1/2$이면 $z/n=m/n$이다.). 이러한 현상은 10,000개의 입력에 대해서도 비슷하게 나타났고 특히, 충분히 높은 $p=0.999$가 설정된 압축블룸필터라도 $2^{-99}$이상의 $f$에서는 MAC의 공간효율이 더 높은 것으로 분석되었다. <표 2>, (그림 3)과 같은 결과가 발생하는 이유는 $n$이 $bpe$에 영향을 주는 정도가 다르기 때문이다. 블룸필터에서는 $n$이 구조체 전체에 대해 일정하게 영향을 주는 반면에, MAC의 경우에는 $n$이 MAC 각각에 대해 독립적으로 영향을 준다. 그렇기 때문에 MAC에서 $f$가 높을 때는 $n$의 영향을 비교적 크게 받아 블룸필터보다 요구되는 $bpe$가 크지만 그 영향이 선형적인 수준에 머물기 때문에 $f$가 낮아짐에 따라 그 영향이 적어져 이후에는 블룸필터보다 요구되는 $bpe$가 작아지는 것으로 해석된다. 이 분석은 잘 알려진 MD5나 SHA1 수준의 보안성이 요구될 때, 블룸필터나 압축블룸필터는 MAC보다 공간효율이 높지 않다는 것을 보여준다. [1,2]에서는 이렇게 동등한 수준의 보안성이 제공될 때의 공간효율을 비교한 것이 아니기 때문에 공평하게 분석하였다고 보기 어렵다.




(그림 3) 주어진 $f$에 대한 $bpe$ 비교 ($n=100, p=0.99$)




(그림 4) 주어진 $f$에 대한 $bpe$ 비교
($n=10000, p=0.999$)

### 3.4 응용환경 고려사항

$n$과 $f$가 정해져있는 상황에서는 블룸필터와 다수의 MAC 중 효율이 좋은 것을 선택하여 사용하면 된다. 하지만 $n$이 정해지지 않은 동적인 상황에서, $n$이 구조체 전체에 영향을 미치는 블룸필터는 유연성이 떨어진다. 반면에 MAC은 $n$이 $f$에 영향을 주기는 하지만 선형적인 수준에 머물고, 서로 다른 입력에 대해 독립적인 구조를 가지기 때문에 비교적 유연성이 좋다. 또한 일반적으로 받아들여지는 MAC의 보안성을 고려하면 블룸필터보다는 MAC을 사용하는 것이 공간효율 면에서도 좋은 결과를 보인다.

한편, 압축블룸필터는 응용환경에 대한 제약이 추가된다. 압축블룸필터에서 데이터를 확인하기 위해서는 압축해제과정을 거쳐야 하고 압축이 해제된 블룸필터는 높은 $p$(=높은 압축률)를 가질수록 큰 공간을 요구하게 된다. 즉 MAC과 같은 보안성을 가지는 압축블룸필터는, 압축 해제된 블룸필터를 표현하기 위한 메모리의 크기를 고려해야 하고, 인코딩 및 디코딩으로 인한 비용 또한 고려해야 한다. 하지만 그러한 노력에 비해 공간효율이 높게 나타나지는 않기 때문에 무결성을 검증하기 위한 목적으로 압축블룸필터가 MAC을 대체하기에는 무리가 있어 보인다.

## 4. 결 론

이 논문에서는 블룸필터, 압축블룸필터 그리고 MAC이 동등한 보안성을 가질 때의 공간효율에 대해서 분석하여 보았다. 일반적으로 받아들여지는 MAC의 보안성 수준에서는 블룸필터나 압축블룸필터보다 MAC을 사용하는 것이 더 효율적이며, 요구되는 보안성이 낮은 경우에는 블룸필터나 압축블룸필터의 공간효율이 더 좋은 것으로 분석되었다. 따라서 블룸필터를 보안과 관련된 응용 환경에 사용할 때에는 효율성과 보안성에 대해 주의 깊은 고려가 요구된다.

## 참 고 문 헌

[1] 정석재, 유영준, 백정하, 이동훈, "차량 밀집환경에서 안전하고 효율적인 V2V 메시지 인증 기법", 한국정보보호학회논문지, 제 20권 제 4호, pp.41-52, 2010년 8월.

[2] 최임성, 김진, 김광조, "무선센서 네트워크에서 안전하고 효율적인 방송형 인증 기법 연구", 한국정보보호학회 동계학술대회, 제 19권 제 2호, pp.206-213, 2009년 12월.

[3] M. Mitzenmacher, "Compressed Bloom filters," in IEEE/ACM Trans. Networking, Vol.10, Oct., 2002, pp.604-612.

[4] E. T. Whittaker and G. N. Watson, "A Course in Modern Analysis," 4th edition, Cambridge University Press, ISBN 0-521-58807-3, 1963.

[5] C. Zhang, X. Ling, and P-H. Ho, "RAISE: An Efficient RSU-aided Message Authentication Scheme in Vehicular Communication Networks," in Proc. IEEE ICC 2008, Beijing, China, pp.1451-1457, May, 2008.

**맹 영 재**

e-mail : brendig@isrl.kr
2004년 2월 대덕대학 정보보안.해킹과 (전문학사)
2006년 8월 인하대학교 컴퓨터공학과(학사)
2008년 8월 인하대학교 정보통신대학원 (공학석사)
2008년 9월~현 재 인하대학교 컴퓨터정보공학과 박사과정
관심분야 : 무선 센서 네트워크, 웹 보안, 사용자 인증, 금융 보안

**강 전 일**

e-mail : dreamx@isrl.kr
2003년 2월 인하대학교 컴퓨터공학과(학사)
2006년 2월 인하대학교 정보통신대학원 (공학석사)
2006년 3월~현 재 인하대학교 정보공학과 박사과정
관심분야 : RFID 보안, 생체 인식 보안, 무선 센서 네트워크, 무선 인터넷 보안, 웹 인증 보안

**양 대 헌**

e-mail : nyang@inha.ac.kr
1994년 2월 한국과학기술원 과학기술대학 전기 및 전자 공학과(학사)
1996년 2월 연세대학교 컴퓨터과학과 (공학석사)
2000년 8월 연세대학교 컴퓨터과학과 (공학박사)
2000년 9월~2003년 2월 한국전자통신연구원 정보보호연구본부 선임연구원
2003년 2월~현 재 인하대학교 컴퓨터정보공학과 부교수
관심분야 : 암호이론, 암호프로토콜, 인증프로토콜, 무선 인터넷 보안


## 이 경 희

e-mail : khlee@suwon.ac.kr

1989년 서울대학교 식품영양학과(학사)

1993년 연세대학교 전산과학과(학사)

1998년 연세대학교 컴퓨터과학과 (공학석사)

2004년 연세대학교 컴퓨터과학과 (공학박사)

1993년 1월~1996년 5월 LG소프트(주) 연구원

2000년 12월~2005년 2월 한국전자통신연구원 선임연구원

2005년 3월~현 재 수원대학교 전자공학과 교수

관심분야 : 영상처리, 컴퓨터비젼, 인공지능, 패턴인식, 생체인식, 얼굴인식, 다중생체인식