논문 2004-41TC-9-2

IPSec 보안서버의 성능분석 모델

(IPSec Security Server Performance Analysis Model)

윤 연 상*, 이 선 영*, 박 진 섭*, 권 순 열*, 김 용 대**, 양 상 운***, 장 태 주***, 유 영 갑**

(Yeonsang Yun, Seonyoung Lee, Jinsub Park, Soonyoul Kwon, Youngdae Kim, Sangwoon Yang, Taejoo Chang, and Younggap You)

요 약

본 논문에서는 IPSec 가속기를 보안서버에 장착하였을 경우의 성능분석모델을 제안하였다. 제안된 보안서버는 M/M/1 시스템으로 모델링하였으며 트래픽 로드는 포아송분포를 이용하였다. 보안서버의 성능변수를 통합하여 디코딩지연이라고 정의하였으며 IPSec 가속기인 BCM5820의 실측 결과와 비교하여 15%정도의 차이를 갖는 디코딩지연을 추출하였다. 디코딩 지연을 제안된 성능분석모델에 대입하여 시뮬레이션 하였을 경우 보안연결은 BCM5820의 발표된 성능의 75%의 처리량을 보였다. 그리고 데이터전달은 발표된 성능의 각각 3.125%(패킷크기 64byte), 14.28%(패킷크기 1024byte)의 처리량을 보였다.

Abstract

This paper proposes a performance analysis model of security servers comprising IPSec accelerators. The proposed model is based on a M/M/1 queueing system with traffic load of Poisson distribution. The decoding delay has been defined to cover parameters characterizing hardware of security servers. Decoding delay values of a commercial IPSec accelerator are extracted yielding less than 15% differences from measured data. The extracted data are used to simulate the server system with the proposed model. The simulated performance of the cryptographic processor BCM5820 is around 75% of the published claimed level. The performance degradation of 3.125% and 14.28% are observed for 64byte packets and 1024byte packets, respectively.

Keywords : IPSec accelerator, Security server, Server performance estimation

I. 서 론

1990년대 이후 네트워크 시장의 증가에 따라 데이터 보안의 중요도가 높아지고 있다. 이미 SSL이나 IPSec 등의 보안 프로토콜을 통하여 VPN과 같은 보안네트워크의 사용이 일반화되고 있다. 하지만 최근 이러한 암호화 어플리케이션을 수행하는 과정에서 CPU의 95% 이상의 자원을 소모한다는 연구결과가 발표되었다[1].

이를 해결하기 위해 보안 어플리케이션 수행 시 CPU를 대신하는 별도의 암호화 프로세서가 제작되고 있다[2]. 현재 상용되고 있는 PCI 인터페이스 기반의 IPSec 가속기가 이에 해당된다.

이러한 IPSec 가속기의 성능은 크게 RSA와 같은 공개키 연산과 3DES와 같은 대칭키 연산의 처리속도로 대표된다. 일례로 Broadcom사의 BCM5820의 경우 RSA-1024bit의 키 생성 속도는 초당 800회 3DES+SHA1의 처리속도는 300Mbps의 성능을 발표하고 있다[3]. 하지만 이상의 발표된 성능은 IPSec 가속기를 서버에 장착했을 경우의 성능이 아닌 장치 내부에서의 처리속도를 나타낸다[4]. 더욱이 대부분의 장치들이 IPSec의 주요 기능인 보안연결과 데이터전달 중 후자의 성능만을 측정해 놓았기 때문에 보안서버의 보안연결 회수에 관한 성능을 확인하기 힘들다. 실제로 BCM5820을 장착한 네트워크 서버의 3DES+SHA1의 처리속도를 측정한 결과 발표된 성능의 50%를 약간 웃도는 결과가 발생하

* 학생회원, ** 정회원, 충북대학교 정보통신공학과
(Department of Information & Communication Engineering, Chungbuk Nat'l University)
*** 정회원, ETRI 부설 국가보안기술연구소
(National Security Research Institute)
접수일자: 2004년1월19일, 수정완료일:2004년9월8일

였다[5]. 이 측정 결과는 네트워크속도 및 서버의 사양을 고려할 때 최적의 테스트 환경에서 얻어진 결과이다. 이 점을 미루어 볼 때, IPSec 가속기를 네트워크 서버에 장착했을 경우 발표된 성능을 기대하기란 힘든 상황이다.

사용자들이 자신들의 서버에 적합한 가속기를 선택할 수 있도록 IPSec 가속기를 실제 네트워크 서버에 장착하였을 경우의 성능분석이 이루어져야 한다. 그렇지만 성능분석을 위해 요구되는 모든 사용자들의 서버와 네트워크 입력발생기 그리고 측정 장비를 이용하기 위해서는 많은 비용과 시간이 요구된다. 본 논문에서는 모델링을 이용한 시뮬레이션 기법을 사용하여 효율적으로 보안서버의 성능을 분석하였다. 시뮬레이션의 모델링은 대기행렬이론에 근거하였다. 대기행렬이론은 수식적으로 시스템의 처리율과 응답시간을 계산할 수 있는 해법을 제공한다. 대기행렬이론을 이용한 서버모델의 경우 정확한 모델링과 파라미터를 바탕으로 실측 결과와 비교하여 상당히 높은 정확도를 보이고 있다[6]. 시뮬레이션 툴로 사용된 Anylogic4.5는 기존의 COVERS 시뮬레이션 툴의 후속 버전으로 중간 사양의 워크스테이션(PIV 1.7GHz, 512MB RAM)에서 50,000개의 서로 다른 객체를 동시에 프로세싱 할 수 있는 테스트 환경을 지원한다[7],[13]. 본 논문의 구성은 우선 Ⅱ장에서 제안된 보안서버의 모델을 설명하였고, Ⅲ장에서는 시뮬레이션을 통한 보안서버 성능분석 결과를 논하였다. 마지막으로 Ⅳ장은 결론을 기술하였다.

Ⅱ. 보안서버 모델

본 장에서는 보안서버의 구체적인 모델링 결과를 설명하였다. 먼저 보안 어플리케이션의 처리를 위한 트랜젝션과 이에 따른 큐(queue) 모델의 적용을 통하여 서버의 응답시간을 도출해내는 과정을 설명하였다. 그리고 본 논문에서 보안서버의 성능분석을 위하여 사용한 포아송분포의 적합성을 검증하였다.

1. 보안서버의 트랜젝션에 따른 큐 모델

IPSec에서 보안 어플리케이션은 데이터의 암호화 및 복호화의 처리를 담당한다. IPSec은 단순히 데이터의 기밀성만을 보장하기 위하여 우선적으로 클라이언트와 서버 간 상호인증과정을 포함한다. 이 과정을 보안연결(Security Association)이라 한다. 데이터전달과정은 보안연결이 완료된 후에 가능하게 된다. 즉, IPSec은 데이터의 암 복호화를 수행하는 데이터전달 트랜젝션과 송수신자 상호간의 보안연결을 설정하는 보안연결 트랜젝션으로 구분된다[8].

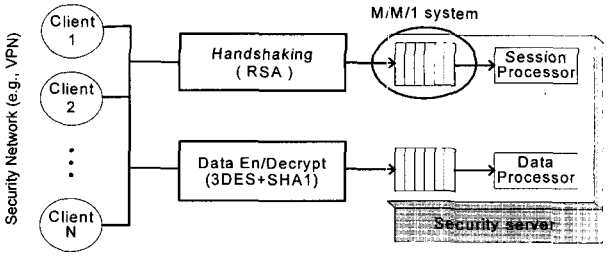


그림 1. 제안된 보안서버 모델
Fig. 1. Proposed security server model.

보안서버의 트랜젝션에 따른 큐 모델을 그림 1에 나타내었다. 보안서버로 입력되는 트래픽은 보안연결 또는 데이터전달의 목적을 갖는다. 각각의 트래픽 로드를 보안서버는 보안연결프로세서와 데이터프로세서로 분기하여 처리하게 된다. 보안서버는 각각의 네트워크 인터페이스에 해당하는 입력 단에 큐를 배치하여 M/M/1 시스템을 구성하였다. M/M/1 시스템에서 서버로 입력되는 서비스의 도착률(λ)과 프로세서의 처리율(μ)을 이용하여 클라이언트가 서비스를 받기 위해 큐 안에서 대기하는 시간은 식 (1)과 같이 계산된다[9]. 식 (1)은 일반적인 M/M/1 시스템에서 서비스들이 큐 내에서 대기하는 평균시간을 의미한다. 서버 전체의 응답시간은 Little의 법칙에 따라 큐 내에서의 대기시간과 프로세서에서의 처리시간을 합한 값이며 식 (2)와 같이 계산된다[9].

$$T_q = \frac{\lambda}{\mu(\mu-\lambda)} \quad (1)$$

$$T_s = \frac{1}{\mu} + \frac{\lambda}{\mu(\mu-\lambda)} \quad (2)$$

2. 트래픽 로드

포아송분포는 네트워크 트래픽을 모델링하기에 적합한 분포로 알려져 왔다. 하지만 1995년 Paxon과 Floyd의 연구 결과에 따르면 실제의 네트워크가 포아송분포와는 차이가 있음을 밝혔다. 그리고 보안연결을 위한 네트워크 트래픽의 경우 10^3초의 시간대에서는 포아송분포로 관찰됨을 설명하고 있다[10]. 이는 보안서버의 보안연결을 위한 트래픽 로드가 포아송분포로 모델링하기에 적합함을 증명한다.

보안서버의 데이터전달 트래픽 로드 역시 포아송분포로 모델링하기에 적합한 패턴을 따른다. 일반적인 서버는 보안연결 과정을 거치지 않고 데이터를 전송한다.

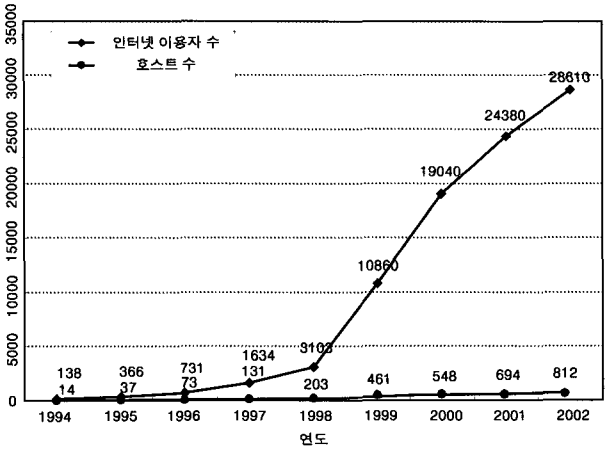


그림 2. 인터넷 이용율 증가 현황[11]
Fig. 2. Development of internet utilization[11].

이 때 발생되는 네트워크 트래픽의 패턴을 분석하면 포아송분포로 모델링하기엔 무리가 있다. 하지만 보안서버의 경우 반드시 한명의 클라이언트는 서버와 1개 이상의 보안세션이 연결되어야 하므로 데이터전달 트래픽은 보안연결 트래픽과 종속적인 관계에 있다. 만약 모든 클라이언트가 보안연결 당 정해진 크기의 파일전송만을 요구한다면 데이터전달시 트래픽 로드는 보안연결시의 트래픽 로드와 같은 패턴으로 나타날 것이다. 특히 보안서버의 경우 보안연결 당 정해진 크기의 파일을 요구하는 경우가 대부분이다. 대표적인 예로 인터넷 뱅킹의 경우 클라이언트는 잔액조회 및 계좌이체등과 같은 동일한 작업을 요구하게 된다.

현재의 네트워크는 Paxson과 Floyd의 연구당시에 비하여 네트워크 속도 및 인터넷 사용자수의 획기적인 증가 추세에 있다. 보안연결 트래픽이 10^3초의 시간대에 이르면 포아송분포를 따른다는 연구결과는 현재의 사정에 맞게 수정되어야 한다. 우선 인터넷 이용자 및 호스트의 증가추세를 그림 2에서 나타내었다. 1995년 당시에 비하여 2002년의 인터넷 이용자는 78배가 증가하였다. 반면 호스트의 증가는 인터넷 사용자의 증가에 비해 적은 증가추세를 보였다. 1995년도의 경우 인터넷 사용자에 대한 호스트 수의 비는 10.11%이고 2002년의 경우 2.84%를 기록했다. 즉, 서비스를 요구하는 클라이언트의 수가 서버의 수에 비하여 크게 증가되었음을 확인할 수 있다. 서버사용자 수와 호스트 수의 비율증가는 식 (3)과 같다.

$$UHR = \frac{10.11}{2.84} \times 78 = 277.67 \quad (3)$$
$$(User\ Host\ Ratio)$$

Ethernet LAN의 경우 1995년 당시의 네트워크 속도와 현재를 비교하면 100배의 속도 발전이 이루어졌다[12]. 일반적으로 인터넷 전용선을 사용하는 사용자가 증가됨에 따라 1995년 당시 주로 사용했던 56kbps 규격의 모뎀과 비교하여 현재는 최소 Mbps 단위의 네트워크 대역폭으로 발전하였다. 결론적으로 현재의 네트워크 속도는 1995년과 비교하여 최소 20배 증가하였음을 알 수 있다. 식 (3)의 UHR 증가량과 네트워크 속도증가율을 모두 고려하면 포아송분포의 단위시간은 5000배 ($\simeq 277.67 \times 20$) 이상 감소하여야 하지만 최악조건을 감안하여 안정적인 수치인 500배로 결정하였다.

그림 3는 제안된 포아송분포의 생성과정을 나타낸다. 그림 3a는 식 (4)의 포아송확률분포함수를 이용하여 패킷이 입력될 확률분포를 추출한 결과이다.

$$P_n(t) = \frac{e^{-\lambda t}(\lambda t)^n}{n!} \quad n = 0, 1, 2, \ldots \quad (4)$$

여기서 λt의 값은 단위시간(unit time)당 입력되어질 패킷의 평균을 의미한다. 본 실험에서는 Paxson과 Flo-yd의 포아송분포와 같은 조건인 평균 57패킷을 입력하였다. 그림 3a는 단위시간당 입력되는 패킷의 수에 따른 입력확률을 의미한다. 실제 네트워크상과 같이 모델링하기 위하여 시간에 따른 패킷분포로 변환하여야 한다. 그림 3b는 그 중간과정으로 해당 평균패킷이 얼마의 시간동안 입력되는지를 이산시간대로 분포시킨 결과이다. 그리고 각각의 평균패킷이 어느 시점에 입력될지 모르는 상황이므로 시간순서를 상호치환 하였다.

이 과정은 각각의 입력확률이 적어도 한번은 입력되어야 하므로 정규분포(Gaussian distribution)가 아닌 균등분포(uniform distribution)를 따른다. 그 결과 그림 3c와 같이 제안된 포아송분포를 생성하였다. 그림 3d는 Paxon과 Floyd 의 연구에서 발생시킨 포아송분포이며 당시의 보안연결 네트워크 트래픽과 유사한 패턴을 보인다는 것이 증명되었다[10]. 이 분포는 평균 57패킷입력 (λt=57)을 갖고 단위시간은 5초이다. 그림 3c는 그림 3d와 비교할 때 λt의 값은 동일하지만 단위시간은 5초에서 10ms로 500배 감소하였다.

III. 성능분석

본 장은 시뮬레이션을 위한 테스트 환경과 보안서버의 성능분석 시 고려해야할 파라미터를 정의하였다. 특히 본 논문에서 제안한 디코딩지연을 설명하였고 시

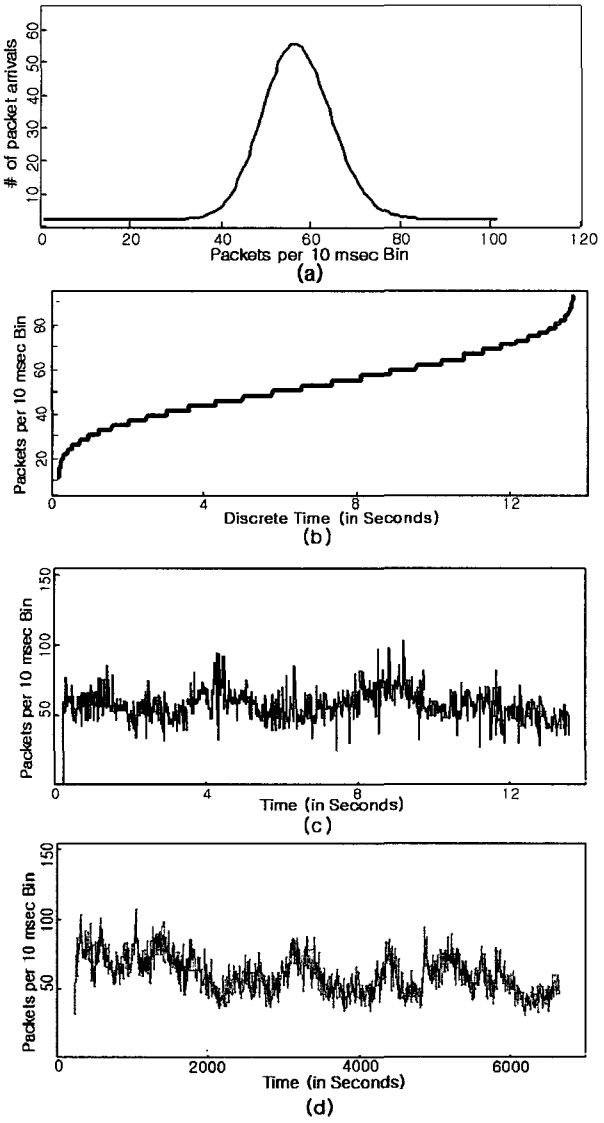


그림 3. 제안된 포아송분포 생성과정:
(a)포아송 확률분포; (b)이산시간별 패킷분포;
(c)제안된 포아송분포; (d)Paxon과 Floyd의 포아송분포

Fig. 3. Process of proposed Poisson distribution:
(a) the Poisson probability distribution (b) packet distribution in discrete time domain (c) proposed Poisson distribution and (d) the Poisson distribu-tion of Paxon and Floyd.

뮬레이션을 통하여 디코딩지연을 추출하는 과정을 설명하였다. 또한 제안된 모델의 시뮬레이션을 통해 IPSec 가속기를 보안서버에 장착하였을 경우의 처리량과 응답시간을 측정하였다.

1. 테스트 환경 및 성능변수

장비를 이용한 측정 시 테스트 환경은 그림 4와 같이 네트워크 상에서 클라이언트 수를 증가시키면서 입력되

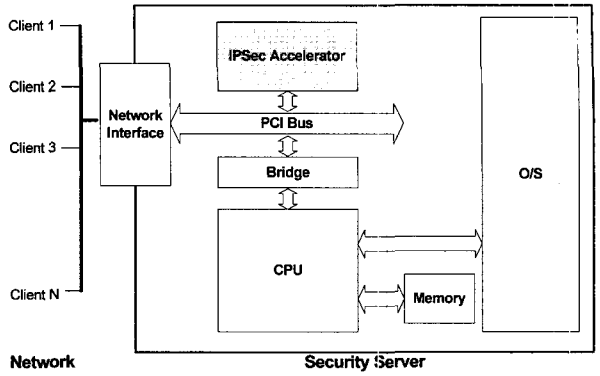


그림 4. 실측 시 사용되는 테스트 환경

Fig. 4. Test bed for parameter measurement.

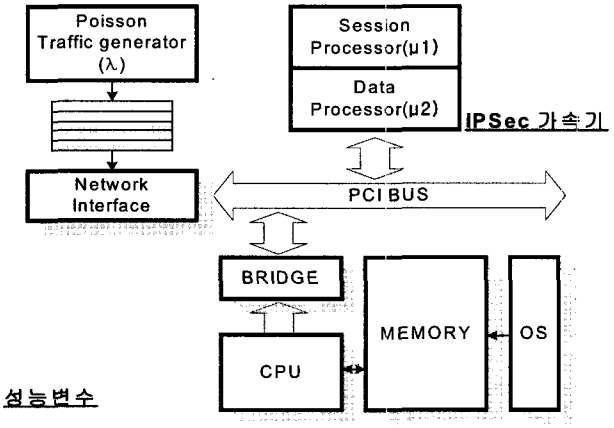


그림 5. 제안된 테스트 환경

Fig. 5. Proposed test bed.

는 요청과 출력되는 서비스를 측정하게끔 설계된다. 제안된 모델을 적용하기 위한 테스트 환경은 그림 5와 같다. IPSec 가속기는 M/M/1 시스템의 프로세서로 네트워크 입력은 포아송분포로 그리고 보안서버의 운영체제 및 하드웨어는 성능변수(performance variables)로 각각 모델링되었다.

성능변수는 IPSec 가속기 내의 암호화 모듈로 데이터를 공급하는 과정에서 저해요소를 뜻한다. 즉, 네트워크 대역폭이 100Mbps와 10Mbps일 경우 서버로 입력되는 데이터양은 계산상 10배의 차이를 보인다. 만약 전자의 네트워크에서 보안서버의 데이터 암호화 처리량이 100Mbps였다면 후자는 10Mbps에 불과할 것이다. 이는 네트워크 대역폭에 따라 IPSec 가속기로의 데이터 공급에 영향을 주는 성능변수가 된다. 그 밖의 성능변수는 CPU와 메모리의 처리속도, 네트워크 어댑터, 운영체제, 패킷크기 등이 있다.

그림 5에서 보이고 있는 성능변수들을 통합하여 디코딩지연(decoding delay)으로 정의하였다. 디코딩지연은 보안서버가 패킷 한 개를 처리하기 위해 최초 네트

표 1. 실측과정에서 이용된 보안서버사양[5]
Table 1. Specification of the security server[5].

구 분	Performance variable	value
네트워크	Bandwidth	1Gbps(host-to-host)
가속기	IPSec 가속기	BCM5820
하드웨어	CPU	1GHz Intel P3 processor
	Memory	256MB PC133 SDRAM
	Hard drive	10GB WDP IDE
	Network adapter	Intel PRO/1000 F
	Mother board	Supermicro 370DE6 Server Works ServersetIII HE-SL chipset with daul PCI buses
운영체제	OS	OpenBSD 3.0

표 2. 제안된 시뮬레이션 모델
Table 2. Proposed simulation model.

구 분	Performance variable	Value
네트워크	Bandwidth (1Gbps)	DistrPoisson.sample(0.4)
	File Size	50kByte
가속기	BCM5820	Queue model
하드웨어	CPU	Decoding delay (ms)
	Memory	
	Hard drive	
	Network adapter	
	Mother board	
운영체제	OS	
시뮬레이션 반복회수	Simulation times	100 times

워크 입력으로부터 성능변수들을 거쳐 IPSec 가속기까지 패킷을 전달하는데 소요되는 시간이다.

2. 시뮬레이션을 통한 디코딩지연 값의 결정

Miltchev의 실험 시 사용된 장비들의 사양은 표 1과 같고 본 논문에서는 이를 표 2와 같이 모델링하였다. BCM5820의 데이터암호화 속도는 300Mbps이므로 50 kbyte파일을 처리하는 시간은 1.333ms로 계산된다. 패킷크기가 64byte인 경우 50kbyte 파일은 총 782개의 패킷으로 나누어 전송된다. 패킷의 개수만큼 디코딩지연이 발생하고 총 디코딩지연과 데이터 처리시간을 합하면 해당 패킷(782개)을 모두 처리하는 데 소비되는 총 처리시간($1/\mu$)이 계산된다. 패킷크기에 따른 총 처리시간을 표 3에서 정리하였다.

적당한 디코딩지연 값 구하기 위하여 0.01ms~10ms 사이에서 값을 0.01ms 간격으로 변화시키며 시뮬레이션을 반복하였다. 그 결과 디코딩지연을 0.5ms로 설정하였을 때, 그림 6과 같이 실측 결과(Miltchev)와 비교하여 가장 유사한 결과를 얻었다.

두 결과의 차이의 평균은 15.4%이다. 패킷의 크기가 8192바이트 이하의 경우 평균 13.3%의 차이만을 보였다. 시뮬레이션 결과는 패킷크기가 작을수록 실측 결과와 적은 차이를 보였다. 현재 네트워크상의 패킷크기는 메시지 전송기준의 경우 최대 1500byte, 미디어 스트리밍기준은 820byte 패킷크기가 가장 많은 분포를 보이고 있다[14].

표 3. 패킷 크기에 따른 처리시간
Table 3. Processing times per packet size.

(단위 : ms)

Packet Size (byte)	디코딩 회수 (#)	디코딩 지연	총 디코딩지연	데이터 처리시간	총 처리시간 ($1/\mu$)
64	782	0.5	391.0	1.333	392.333
128	391		195.5		196.833
256	196		98.0		99.333
512	98		49.0		50.333
1024	49		24.5		25.833
2048	25		12.5		13.833
4096	13		6.5		7.833
8192	7		3.5		4.833
16384	4		2.0		3.333
32768	2		1.0		2.333
65536	0		0.0		1.333

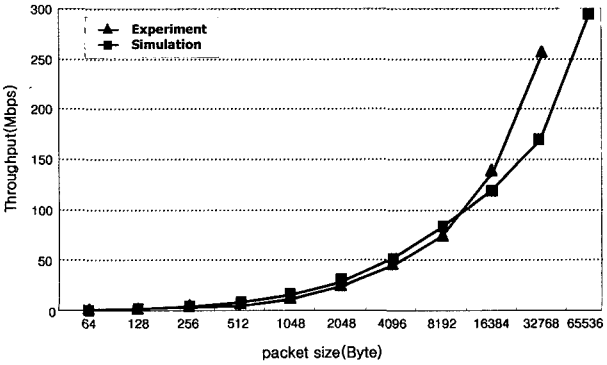


그림 6. Miltchev와 제안된 모델과의 성능분석결과 비교
Fig. 6. Comparison of Miltchev and proposed model.

표 4. 보안서버의 성능분석 기준
Table 4. Metrics of security server performances.

	처리량	응답시간
보 안 연 결	connection/s	ms
데이터 전달	files/s	ms

표 5. 보안서버의 성능 시뮬레이션 파라미터
Table 5. Simulation parameter.

	Security Association		File transfer	
Throughput	metric	connection/s	metric	file transfer/s
	output	Processing time -RSA1024bit key setup	output	Processing time -10kB file transfer
	process time	RSA 1.25	process time	3DES+SHA1 3.08
		TDD* 0.5		TDD 78.5
Response time	metric	ms	metric	ms
	output	Queue size	output	Queue size
Simulation step		1		
Average step (unit time)		1000		
Repetition times		100 times		

*TDD : total decoding delay

3. 보안서버 성능분석 결과

제안된 시뮬레이션 모델을 이용하여 클라이언트 수를 증가시키며 보안연결과 파일전달 각각에 대하여 시뮬레이션 하였다. 보안서버의 성능분석 기준을 표 4에서 정리하였다.

제안된 시뮬레이션 모델의 입력 파라미터는 표 5와 같다. 보안연결의 경우 패킷크기에 관계없이 클라이언트의 연결요청패킷 1개만 입력된다. 즉, 디코딩지연이 1회만 발생하게 된다. 보안연결 처리시간(1.25ms)과 디코딩지연(0.5ms)을 합한 결과가 보안연결 트랜잭션의 총 처리시간이 된다. 파일전달의 경우 10kbyte 파일을 64byte 패킷크기로 전송하기 위한 총 디코딩회수는 157회이며 총 디코딩지연은 78.5ms로 계산된다.

그림 7은 BCM5820을 보안서버에 장착하였을 경우 보안연결 트래픽 입력에 따른 처리량을 나타낸다. 이상적인 결과는 디코딩지연을 고려하지 않고 보안연결 처리시간(1.25ms)만을 모델에 적용한 결과이다. 클라이언트의 수가 128명이 되는 시점에서 처리량이 더 이상 증가하지 않았다. 이 시점은 보안서버의 부하를 의미하며 패킷손실이 예상된다. 디코딩지연을 고려했을 경우 클라이언트의 수가 64명이 되는 시점에서 보안서버의 부하가 발생하였다. 처리량은 600connection/s로 발표된 성능의 75%만을 나타내었다. 그림 8은 보안연결 트래픽 로드에 따른 응답시간의 결과이다. 이상적인 경우 클라이언트의 동시 접속 수가 64명일 경우까지 처리시간인 1.25ms가 응답시간이 된다. 클라이언트의 수가 128명으로 증가되면 평균 2500ms의 응답시간을 갖는다. 디코딩지연을 고려하였을 경우 클라이언트의 수가 64명인 시점부터 응답시간이 증가하였고 128명일 경우 평균 3500ms의 응답시간을 기록했다.

데이터전달 트랜잭션의 경우 입력되는 파일크기는 10kbyte로 하였다. 10kbyte 파일크기는 SPECweb99의 표준 웹페이지규격(standard web page size)을 참조하였다[15]. 패킷크기는 64byte와 1024byte로 나누어 시뮬레이션 하였다. 그림 7의 결과에서도 확인하였듯이 64byte 패킷크기는 최악의 조건에 해당된다. 1024byte 패킷크기는 현재 네트워크상에서의 메시지 전송기준 크기에 해당한다. 10kbyte 파일은 64byte의 패킷크기로 157회 입력되며 각각의 패킷은 0.5ms의 디코딩지연을 갖는다. 따라서 표 3의 계산방식에 의해 10kbyte의 파일을 모두 처리하는데 소비되는 총 처리시간은 81.58ms로 계산되었다. 그리고 1024byte 패킷의 경우에는 5.79

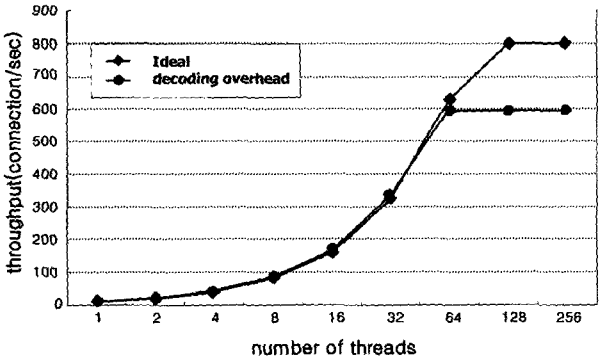


그림 7. 보안연결 처리량
Fig. 7. Throughput of security association.

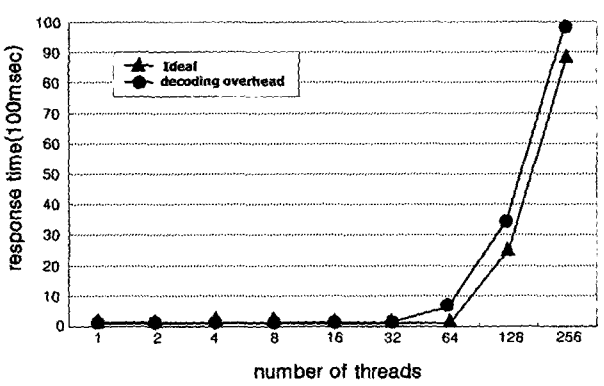


그림 8. 보안연결 응답시간
Fig. 8. Response time of security association.

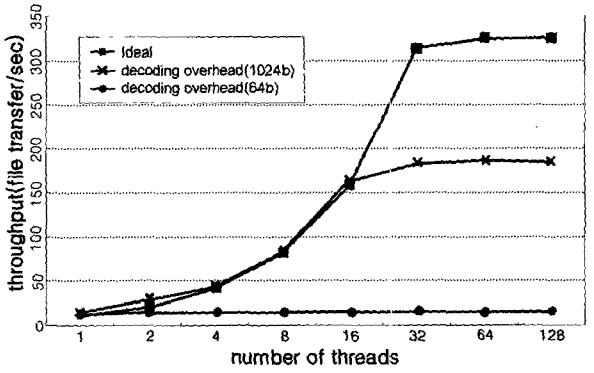


그림 9. 데이터전달 처리량
Fig. 9. Throughput of data transfer.

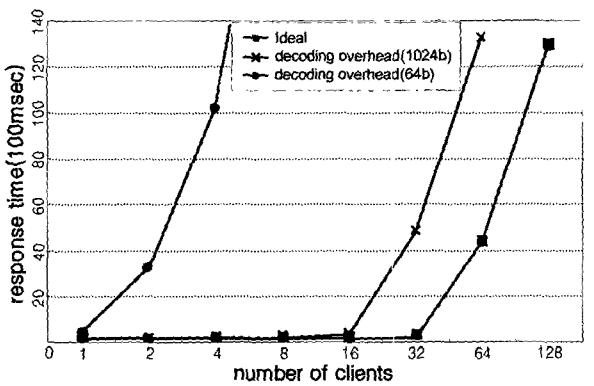


그림 10. 데이터전달 응답시간
Fig. 10. Response time of data transfer.

ms로 계산되었다. 그림 9는 BCM5820 장착서버의 데이터전달 트래픽 로드에 따른 처리량의 결과이다. 이상적인 경우의 최대 처리량은 약 320files/s(25.6Mbps)를 기록하였다. 이는 패킷크기가 64byte일 경우의 발표된 성능(26Mbps)과 같다. 하지만. 디코딩지연을 고려하였을 경우 처리량은 10files/s로 발표된 성능의 3.125%를 나타내었다. 디코딩지연을 고려하고 패킷크기를 1024byte로 설정하였을 경우 처리량은 최대 180files/s(14.4Mbps)를 기록하였다. 이는 BCM5820의 발표된 성능(101Mbps)의 14.28%에 해당한다.

BCM5820 장착서버의 데이터전달 트래픽 로드에 따른 응답시간의 그래프를 그림 10에서 보였다. 64byte 패킷크기 입력 시 이상적인 경우에는 최대 64명의 클라이언트까지 프로세서의 부하가 발생되지 않았다. 디코딩지연을 고려하였을 경우에는 2명의 클라이언트부터 3000ms가 넘는 응답시간을 기록했다. 1024byte 패킷크기를 입력하였을 경우는 16명의 클라이언트까지 부하를 나타내지 않았다.

IV. 결 론

제안된 보안서버 모델을 이용한 시뮬레이션 결과 IPSec 가속의 발표된 성능은 서버에 장착하였을 경우 현저히 낮아진다는 결론을 내릴 수 있다. BCM5820의 경우 보안연결은 발표된 성능의 75%만을 나타내었다. 그리고 데이터전달은 64byte 패킷크기에서 3.125%, 1024byte 패킷크기에서 14.28%의 성능만을 나타내었다. IPSec에서 병목현상의 원인으로 분류되어 왔던 공개키 연산은 최대 32명의 클라이언트까지 동시 처리가 가능하지만 대칭키 연산의 경우 16명의 클라이언트까지 동시 처리가 가능하였다. 이는 보안서버의 디코딩지연이 암호화 프로세서의 성능보다 전체적인 서버의 성능에 더 큰 변수로 작용한다는 사실을 증명한다. 제안된 성능분석을 통해 얻어진 결과를 미루어 볼 때, 보안서버의 성능향상을 위해서는 IPSec 가속기의 발표된 성능 이전에 디코딩지연을 감소시키기 위한 고속회로 설계가 중요하다.

참 고 문 헌

[1] M. Merkow and J. Breithaupt, The Complete Guide to Internet Security, AMACOM, 2000.

[2] M. McLoone and J.V. McCanny, "A single-chip IPSec cryptographic processor," IEEE Workshop on Signal Processing Systems. pp. 133-138, Oct. 2002.

[3] Broadcom Co., BCM5820 Product Brief, http://www.broadcom.com/collateral/pb/5820-PB04-R.pdf.

[4] Broadcom Co., "Comparising the performance of Broadcom IPSec boards," http://www.broadcom.com/collateral/wp/IPSEC-WP100-RDS.pdf.

[5] S. Miltchev and S. Ioannidis, "A study of the relative costs of network secutity protocols," In Proceedings of USENIX Annual Technical Conf., Freenix Track, pp. 41-48, June 2002.

[6] I. Cao and M. Anderson, "Web server performan-ce modeling using an M/G/1/K*PS queue," 10th Int'l. Conf. on Telecommunications, vol. 2, pp. 1501-1506, Feb. 2003.

[7] A.V. Borshchev and Y.G. Karpov, "Systems modeling, simulation and analysis using COVE-RS active objects," IEEE Workshop on Enginee-ring of Computer Based Systems (ECBS '97), pp. 220-227, Mar 1997.

[8] S. Ken, Security Architecture for the Internet Protocol, http://www.ietf.org/internetdrafts/draft-ietf-ipsec-rfc2401bis-00.txt

[9] 이호우, 대기행렬이론-확률과정론적 분석, 시그마프레스, 1998.

[10] V. Paxson and S. Floyd, "The failure of Poisson modeling," IEEE/ACM Trans on Networking, vol. 3, pp. 226-244, June 1995.

[11] 한국전산원, 2002 국가정보화백서

[12] 윤문길, "인터넷 접속기술," http://mslab.hau.ac.kr/it_02/4.ppt.

[13] XJ Technologies, Anylogic4.5 Product Overview, http://www.xjtek.com.

[14] C. Fraleigh and S. Moon, "Packet-level traffic measurements from the SPRINT IP backbone," IEEE Journal of Network, vol. 17, pp. 6-16, Nov. 2003.

[15] SPEC Co,, "Standard Web Page Size in SPEC web 99," http://www.spec.org/web96/workload.html

저 자 소 개

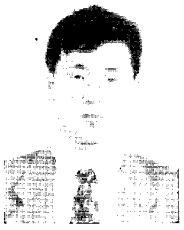
윤 연 상(학생회원)
2004년 충북대학교 전기전자 공학부 학사 졸업.
2004년 현재 충북대학교 정보통신 공학과 석사과정 재학중
<주관심분야: 디지털 회로 설계, 암호시스템>

이 선 영(학생회원)
2003년 충북대학교 전기전자 공학부 학사 졸업.
2004년 현재 충북대학교 정보통신 공학과 석사과정 재학중
<주관심분야: 디지털회로 설계, 정보보호>

박 진 섭(학생회원)
2004년 충북대학교 전기전자 공학부 학사 졸업.
2004년 현재 충북대학교 정보통신 공학과 석사과정 재학중
<주관심분야: 디지털 회로 설계, 암호시스템>

권 순 열(학생회원)
2003년 충북대학교 전기전자 공학부 학사 졸업.
2004년 현재 충북대학교 정보통신 공학과 석사과정 재학중
<주관심분야: Computer arithmetic, ASIC>

김 용 대(정회원)
1990년 충북대학교 정보통신 공학과 학사 졸업.
1993년 충북대학교 컴퓨터공학과 석사 졸업
1989년~1998년 신흥기술연구소 팀장
2000년~현재 충북대학교 정보통신공학과 박사과정
<주관심분야: Computer arithmetic, ASIC 설계, 암호시스템>



양 상 운(정회원)
1992년 충북대학교 정보통신 공학과 학사 졸업.
1998년 충북대학교 정보통신 공학과 석사 졸업
1992년~2000년 국방과학연구소 연구원
2000년~현재 한국전자통신연구소 부설 국가보안기술연구소 선임연구원
<주관심분야: 암호프로세서 설계, Computer Arithmetic, 정보보호, 반도체>

장 태 주(정회원)
1982년 울산대학교 전기공학과 학사 졸업
1990년 한국과학기술원 전기 및 전자 공학과 석사 졸업
1998년 한국과학기술원 전기 및 전자공학과 공학박사 졸업
1982년~2000년 국방과학연구소 선임연구원
2000년~현 재 한국전자통신연구원 부설 국가보안기술연구소 책임연구원
<주관심분야: 암호프로세서 설계, 정보보호, 통계학적 신호처리>

유 영 갑(정회원)
1975년 서강대학교 전자공학과 학사 졸업
1975~1979년 국방과학연구소 연구원
1981년 Univ.of Michigan, Ann Arbor 전기전산학과 석사 졸업
1986년 Univ.of Michigan, Ann Arbor 전기전산학과 공학박사 졸업
1986년~1988년 금성반도체 (주) 책임 연구원
1993년~1994년 아리조나 대학교 객원교수
1998년~2000년 오레곤 주립대학교 교환교수
1988년~현재 충북대학교 정보통신공학과 교수
<주관심분야: VLSI 설계 및 테스트, 고속 인쇄회로 설계, 암호학>