적응적 PMTU 발견 메커니즘을 통한 IPSec 터널 모드에서의 통신 불능 현상 해소에 관한 시뮬레이션 연구

김은성*, 안성진**, 정진욱***, 이도훈***, 윤재우****

A Simulation Study for Resolving Communication Failure in IPSec Tunnel Mode with Adaptive PMTU Discovery Mechanism

Eun Sung Kim, Sung Jin Ahn, Jin Wook Chung, Do Hoon Lee and Jae Woo Youn

Abstract

VPN which cuts down on expense and assures security and reliance, has increased its market shares quickly because the requirement of enterprise on security has increased. But fragmentation may raise communication failure when VPN has been implemented using IPSec. In our paper, we have given careful consideration to various reasons preventing us from communicating stably and have presented the existing solutions about them. Also we have provided adaptive PMTU discovery mechanism to improve the solutions. We have proven a prowess of this mechanism through simulation

Key Words: VPN, IPSec, PMTU, fragmentation

* KISTI 슈퍼컴퓨팅센터
** 성균관대학교 컴퓨터교육과
*** 성균관대학교 정보통신공학부
**** ETRI 부설 국가보안기술연구소

1. 서론

정보화가 급속히 진행되면서 전 세계적으로 보급된 인터넷은 이제 인류의 경제 활동을 비롯한 모든 활동에 있어서 없어서는 안될 주요 기반 구조의 하나로 자리 잡아 가고 있다. 특히 인터넷을 통한 전자상거래의 확산, 개인 간 통신(전자우편)의 확대, 기업 사설 네트워크의 구축 확산 등으로 인터넷을 통한 정보 교환 시 정보 보호의 중요성은 날로 증가하고 있다. 그러나 인터넷에서 사용되고 있는 IP 프로토콜은 단순히 패킷 교환망에서 데이터의 신뢰성 있는 전송만을 염두에 두고 개발한 것이기 때문에 IP 스푸핑(IP 데이터 프로그램의 주소 변조), IP 스니핑(IP 데이터 프로그램 도청)과 같은 보안 취약점이 존재한다. 이러한 문제점을 해결하기 위한 방안으로 IP 계층에서 보안서비스를 제공하는 IPSec(IP Security)이 등장하였다. IPSec은 네트워크 기반의 어떤 서비스나 어플리케이션도 보호할 수 있으며 인증, 무결성, 기밀성, 접근제어, 재전송 공격 방지 등의 다양한 보안 서비스를 제공한다. 현재 여러 단체와 기업들이 IPSec을 지지하고 있으며 향후 인터넷의 표준 보안 프로토콜로 성장이 예상된다[1][2][3].

오늘날 기업에서는 외부 네트워크와의 정보 교환의 필요성과 재택 근무자의 수가 증가함에 따라 사설 네트워크의 규모를 확대하고 있다. 이러한 사설 네트워크는 안전한 정보 교환을 보장하기는 하지만 전용선 설치에 투자해야 하는 비용과 그에 따른 운영 및 관리비용이 커다란 문제가 되고 있다. VPN(Virtual Private Network)은 전용선의 설치 대신 공중 네트워크의 서비스를 이용한 비용 절감 효과를 통하여 상기한 문제를 해결하기 위하여 제안되었고, 현재 많은 관심의 대상이 되고 있다. 예를 들면 원거리 접속의 경우 공중 네트워크를 통한 연결이 두 컴퓨터간의 전용선을 사용하는 것보다 훨씬 저렴하며 모뎀이나 ISDN을 사용하는 통신 방식보다도 저렴하다. 또한 VPN을 사용하여 지사들의 근거리망은 공중 네트워크를 통해 연결되는 기업의 다른 근거리망으로 연결될 수 있고 사용자들은 평상시와 같은 방법으로 접근할 수 있으므로 투명성이 제공된다. 이밖에도 VPN을 이용하면 새로운 ISP 사업자와의 계약을 통하여 손쉽게 네트워크 용량과 범위를 확대할 수 있으므로 시장의 수요와 구조적 변화에 민첩한 대응을 할 수 있고 협력업체와의 새로운 사업 관계를 즉각적으로 맺을 수 있다. 이렇듯이 VPN은 새로운 사설 네트워크 없이 기존의 공중 네트워크를 사용하고, 관리와 운영은 인터넷 서비스 제공업자가 담당하게 됨으로써 적은 운영비용을 들여 넓은 범위의 네트워크를 구성할 수 있도록 해 준다[4][5].

하지만 비용 절감 효과와 보안 문제를 해결하기 위해서 전용선의 설치 대신 공중 네트워크 서비스를 이용한 VPN 구축 시 터널링 프로토콜로서 IPSec을 이용하려 할 때 패킷 단편화 현상으로 인한 통신 불능 현상이 발생할 수 있다.

본 논문에서는 이러한 통신 불능 현상의 원인에 대해서 고찰하고 언급한 원인들에 대한 기존의 해결 방안들을 살펴본 후 개선된 해결책으로서 적응적 PMTU(Path Maximum Transmission Unit) 발견 메커니즘을 제시한다. 또한 이 메커니즘과 기존의 해결 방안을 시뮬레이션을 통해 비교 분석한다.

2장에서는 IPSec 터널 모드의 통신 불능 현상 원인과 기존의 해결 방안들을 살펴보고 3장에서는 이 문제를 해결하기 위해서 적응적 PMTU 발견 메커니즘을 제시한다. 4장에서는 제안된 메커니즘을 시뮬레이션을 통해서 기존의 방법과 비교하여 그 우수성을 증명하고 5장에서 결론을 맺는다.

2. IPSec 터널 모드의 통신 불능 현상

2.1 단편화

2.1.1 단편화 현상

MTU(Maximum Transmission Unit)는 TCP/IP 네트워크 등과 같이 패킷 또는 프레임 기반의 네트워크에서 전송될 수 있는 최대 크기의 패킷

또는 프레임을 가리킨다. 이 값은 링크 계층에 사용되는 프로토콜별로 다른 값을 가진다. 따라서 링크 계층의 상위에 있는 IP 계층은 데이터그램을 전송하기 전에 이 값을 참조하여 이 값보다 더 작은 크기로 데이터그램을 분할하여 링크 계층으로 내려보내야 한다. 이것을 단편화 현상이라고 한다.

이러한 단편화 현상은 다음과 같은 특징을 가진다[6].

① 단편화는 상위 계층 프로토콜이 전송 채널의 특성에 신경 쓸 필요가 없게 한다.
② 단편화는 송신 호스트가 패킷이 거치는 경로에 대한 정보 없이도 다른 MTU를 가지는 여러 경로를 처리할 수 있도록 한다.
③ 단편화는 높은 대역폭 연결에 대한 성능을 최적화할 수 있도록 한다.

2.1.2 단편화 문제점

위에서 언급한 여러 가지 장점에도 불구하고 단편화는 다음과 같은 문제점들로 인해서 가능하면 피해야 하는 것으로 간주된다[6].

① 자원의 비효율적인 사용을 유발한다.
잘못된 패킷 크기는 데이터그램을 전송하는 비용을 크게 증가시킨다. 즉, 분리된 패킷(fragment)마다 추가적인 헤더 정보를 위해 추가적인 대역폭이 필요하고 중간 게이트웨이는 추가적인 라우팅 결정을 하기 위해 계산 오버헤드가 증가할 것이며 수신 호스트는 분리된 패킷들을 재결합(reassembly)시키기 위한 오버헤드가 요구될 것이다.
② 분리된 패킷의 손실은 네트워크의 성능을 격감시킨다.
분리된 IP 패킷들을 재결합시키는 메커니즘에는 문제점이 있다. 즉, 분리된 대부분의 패킷들이 올바르게 수신되었음에도 불구하고 하나의 패킷이라도 손실된다면 상위 계층 프로토콜은 원래 데이터그램의 전체 데이터를 재전송할 것을 요구한다.
③ 효율적인 재결합은 어렵다.
분리된 패킷 중 하나라도 손실된다면 재결합 프로세스가 낮은 성능을 내게 될 가능성이 높다.

2.2 단편화 회피

위와 같은 문제를 해결하기 위해서 PMTU의 개념을 사용한다. PMTU는 송신측으로부터 수신측 사이 경로의 모든 MTU 중 가장 작은 MTU를 말한다. 결국 이 PMTU 크기의 패킷을 사용하면 단편화 없이 송신측으로부터 수신측까지 IP 데이터그램을 보낼 수 있다는 것을 의미한다. 따라서 대부분의 IP 구현이 이를 적용하기 위해서 IP 헤더의 플래그 필드 중 "don't fragment" 비트를 설정하여 보내게 된다. 이렇게 설정시켜서 보내게 되면 중간 라우터에서 단편화가 발생하게 될 때 라우터는 원래의 패킷은 폐기하고 단편화를 발생시킨 링크의 MTU 값을 포함하는 ICMP 에러 메시지(type 3: destination unreachable, code 4: fragmentation needed)를 생성하여 이 메시지를 송신측으로 전송하게 된다. 이 메시지를 받은 송신측은 자신의 PMTU 값을 이 메시지에서 알려온 MTU 값으로 바꾸고 이 값에 맞게 데이터를 분할한 후 패킷을 전송하게 된다[8].

2.3 통신 불능 원인

위에서 설명한 바와 같이 성능 문제를 고려하여 보안 게이트웨이에서도 "don't fragment" 비트를 설정하여 패킷을 전송하게 된다. 따라서 양 종단이 보안 게이트웨이인 터널링 구간에서 데이터를 전송할 때 단편화 현상이 발생하면 ICMP 에러 메시지에 의해 적절한 PMTU를 찾아야 계속적인 통신이 가능하게 되는데 다음과 같은 상황 하에서 통신 불능 현상이 발생하게 된다[9].

① ICMP 에러 메시지를 필터링 하는 방화벽의 존재
중간 라우터는 적절한 ICMP 에러 메시지를

발생시키지만 Firewall-One사 계열 제품을 비롯하여 많은 방화벽 시스템들이 ICMP 에러 메시지를 필터링 하기 때문에 이 메시지가 송신측으로 전송되지 않는다. 따라서 보안 게이트웨이는 이 사실을 알지 못하고 계속 잘못된 크기의 패킷을 보내고 이 패킷은 중간 라우터에서 계속 폐기되게 되어서 통신 불능 현상이 발생하게 된다.

② ICMP 에러 메시지를 생성하지 않는 게이트웨이의 존재
게이트웨이가 ICMP 에러 메시지를 생성하는 것은 RFC 요구 사항이 아니기 때문에 게이트웨이(라우터)가 "don't fragment" 설정된 패킷을 수신 받았을 때 단편화가 발생할 경우에도 ICMP 에러 메시지를 생성하지 않을 수 있다. 이 경우에도 역시 보안 게이트웨이는 계속해서 잘못된 크기의 패킷을 전송하게 되어서 통신 불능 현상이 발생하게 된다.

2.4 기존의 통신 불능 현상 해소 방안

IPSec 터널 모드에서 통신 불능 현상을 해소하기 위해서 다음과 같은 방법들이 현재 이용되고 있다[9].

① ICMP 에러 메시지를 필터링 하는 방화벽의 존재
모든 방화벽이 ICMP 에러 메시지를 필터링 하지 않도록 구성 정보를 변경하도록 한다. 하지만 이러한 방법은 대부분의 사이트들이 사이트의 요구 사항에 적합하게 각자 나름의 기준을 가지고 보안 정책을 운용하고 있기 때문에 이러한 요구를 강요하는 것은 현실적으로 불가능하다고 할 수 있다.

② ICMP 에러 메시지를 생성하지 않는 게이트웨이의 존재
모든 게이트웨이가 ICMP 에러 메시지를 생성하도록 구현을 변경해야 한다. 하지만 이러한 방법은 구현 업체의 설계 원칙에 부합하지 않을 수 있기 때문에 현실적으로 불가능하다고 할 수 있다.

결론적으로 위에서 제시한 어떤 방법도 현실적으로 IPSec 터널 모드에서의 통신 불능 현상을 해소하기 어렵기 때문에 보안 게이트웨이 구현 업체에서는 대부분 보안 게이트웨이에서 "don't fragment" 비트를 해제하여 패킷을 전송하는 방식을 제공하고 있다. 하지만 이렇게 할 경우 단편화가 발생할 수 있으므로 위에서 언급했던 문제가 발생하게 된다. 게다가 보안 게이트웨이는 패킷에 AH나 ESP와 같은 처리를 더하므로 전송중에 분리된 패킷 중 하나라도 손실되게 되면 단편화 시 일반 패킷을 전송할 때 드는 비용보다 훨씬 더 큰 비용을 요구하게 된다. 따라서 이를 해결하기 위한 보다 효율적인 방법이 요구되게 된다.

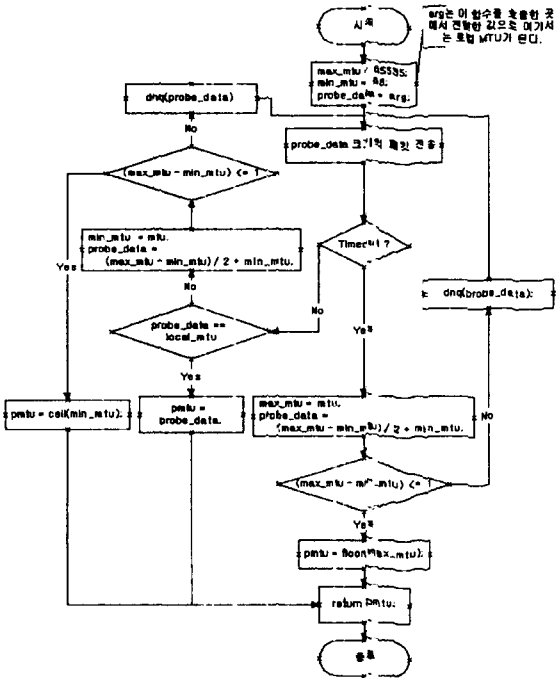
3. 적응적 PMTU 발견 메커니즘

이 메커니즘은 ICMP 에러 메시지가 오지 않는 경우에 즉, PMTU를 발견하기 위해서 기존의 ICMP 메커니즘을 사용할 수 없을 때 기존의 "don't fragment" 비트를 해제하여 패킷을 전송하는 방식의 단점을 해결하기 위해서 개발되었다.

이 메커니즘에서는 PMTU를 찾기 위해서 직접 "don't fragment" 비트를 설정한 다양한 크기의 패킷을 전송하여 적절한 PMTU를 찾는다. 따라서 가능하면 PMTU를 찾기 위해 보내는 패킷의 수를 줄여야 하고 양 종단 보안 게이트웨이에 이를 처리하는 프로세스를 설치해야 한다. 또한 라우팅 정보 변화에 따른 PMTU 변경에 대처하기 위해서 SA (Security Association)의 PMTU age 필드에 저장된 주기에 따라 호출되어 항상 최신의 PMTU 정보를 유지할 수 있도록 한다. 이 적응적 PMTU 발견 메커니즘은 다음과 같은 두 가지 방식으로 구현될 수 있다.

■ Divide-and-Conquer(DnQ) 방식
이 방식은 기존의 DnQ 알고리즘을 이용해서

PMTU를 찾는다. 그 과정은 다음과 같다.
우선 PMTU를 찾기 위해서 초기값으로 로컬 MTU를 사용한다. 이 크기의 패킷을 보냈을 때 목적지 보안 게이트웨이로부터 응답이 오면 이 값을 PMTU로 사용한다. 그러나 정해진 타임아웃 시간 동안 응답이 오지 않으면 이 값의 반 크기의 패킷을 다시 전송한다. 그 후 이에 대한 응답이 오지 않으면 다시 이 값의 반 크기의 패킷을 전송하고 응답이 오면 이 값과 이전에 타임아웃 된 값의 중간값 크기의 패킷을 전송한다. 이와 같은 방식으로 PMTU를 찾을 때까지 계속한다. <그림 1>은 이 알고리즘의 흐름도이다.



<그림 1> DnQ 방식 흐름도

■ Multiplicative-and-Additive(MnA) 방식
이 방식은 TCP의 슬로우 스타트(Slow Start)에서 사용하는 multiplicative increase와 혼잡 회피(Congestion Avoidance)에서 사용하는 additive increase를 혼합한 방식이다[7]. 추가적으로 이 방식에서는 PMTU를 찾는 시간에 가장 많은 영향을 주는 타임아웃 횟수를 가능하면 줄이고 PMTU를 찾는 probe 패킷의 전송 횟수를 줄이기 위해서 가능한 MTU 전체 범위(68 바이트~65535 바이트)를 적절한 범위로 나누어 multiplicative increase와 additive increase를 수행한다. 인터넷상의 공식 최소 MTU값이 68 바이트[8]이기 때문에 범위는 이 값의 배수 중 하나의 숫자가 될 것이다. 이러한 배수들 중에 try&error 방식에 의해 찾아낸 가장 적절한 범위값은 2176 바이트이다.

<표 1>은 범위를 2176 바이트로 정한 근거 자료로서 2176 바이트일 때 PMTU를 찾는데 최소 시간이 걸림을 알 수 있다.

<표 1> 범위에 따른 최대 PMTU 결정 시간

범위	544	1088	2176	4352
범위를 찾기 위한 최대 패킷 전송 횟수	120	60	30	15
재전송 횟수	1	1	1	1
Multiplicative increase 방식의 베이스를 찾기 위한 최대 패킷 전송 횟수	5	6	7	8
재전송 횟수	1	1	1	1
Additive increase를 사용해 PMTU를 찾기 위한 최대 패킷 전송 횟수	8	16	32	64
재전송 횟수	1	1	1	1
총 패킷 전송 횟수	132	82	69	87
총 재전송 횟수	3	3	3	3
PMTU를 찾는 최대시간(s) *RTT: 0.1ms *타임아웃: 3s	22.2	17.2	15.9	17.7

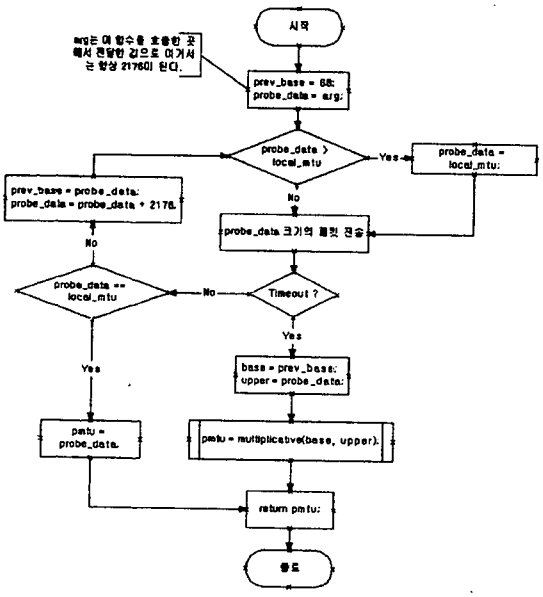
이 방식에 의해서 PMTU를 찾는 과정은 다음과 같다.

우선 PMTU가 어떤 범위에 속하는 가를 찾기 위해서 2176, 4352, 6528, ... 바이트 크기의 패킷을 타임아웃이 발생할 때까지 계속해서 전송한다. 타임아웃이 발생하면 타임아웃이 발생하기 이전의 값을 베이스(base)로 선정한다. 만약 타임아웃 된 값이 2176이라면 68을 베이스로 선정한다.

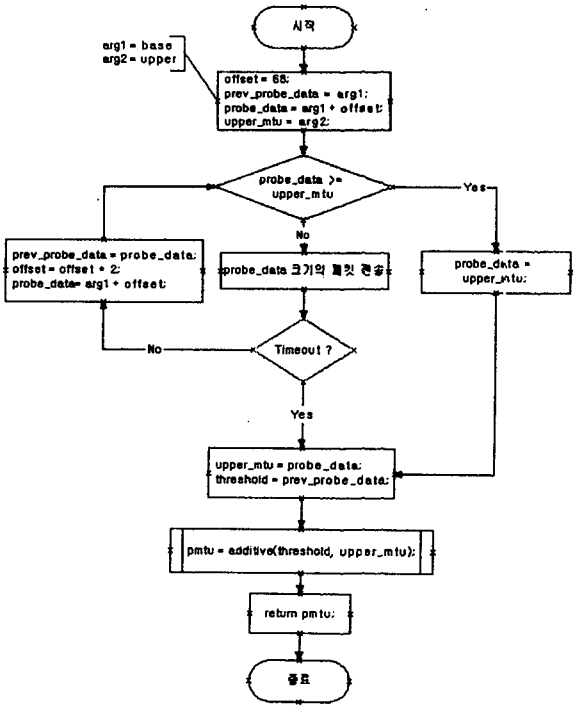
선정된 베이스 값으로부터 multiplicative increase를 수행한다. 즉, base+68, base+68*2, base+68*2*2, ... 크기의 패킷을 타임아웃이 발생할 때까지 계속해서 전송한다. 타임아웃이 발생하면 타임아웃이 발생하기 이전의 값을 베이스로 선정한다.

선정된 베이스 값으로부터 additive increase를 수행한다. 즉, base+68, base+68+68, base+68+68+68, ... 크기의 패킷을 타임아웃이 발생할 때까지 계속해서 전송한다. 타임아웃이 발생하면 타임아웃이 발생하기 이전의 값을 PMTU로 선정한다.

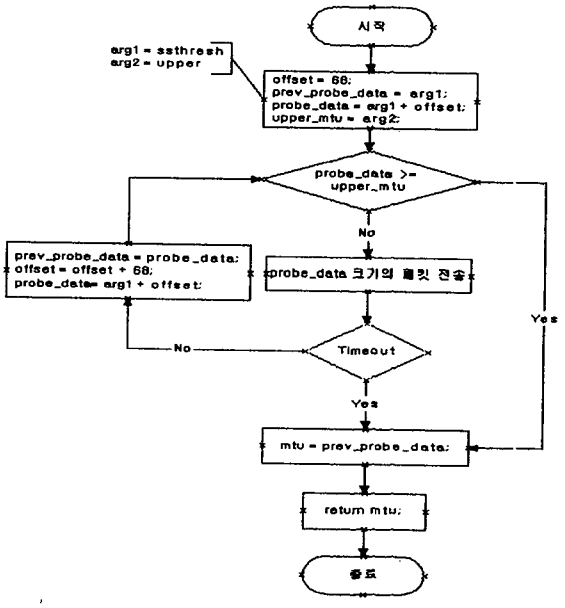
<그림 2>, <그림 3>와 <그림 4>는 이 방식의 흐름도이다.



<그림 2> MnA 방식 흐름도



<그림 3> Multiplicative function 흐름도



<그림 4> Additive Function 흐름도

■ 두 방식 성능 비교

두 방식은 사용된 알고리즘의 특성상 다음과 같은 특성을 가진다. 즉, DnQ 방식은 정확한 PMTU 값을 찾을 수 있지만 재전송이 많이 일어날 수 있고 MnA 방식은 (PMTU - 68) ~ PMTU 사이의 값을 찾지만 재전송 회수는 최대 3번이다.

두 방식의 성능을 비교하기 위해서 각 방식이 적절한 PMTU를 찾는데 걸리는 시간을 살펴보았다. <표 2>는 각 로컬 MTU 환경에서 타임아웃시간이 3s이고 RTT(Round Trip Time)이 100ms라고 가정했을 때 PMTU를 찾는데 걸리는 최대 시간을 정리한 것이다. 여기서 각 로컬 MTU 값은 인터넷 상의 일반적인 MTU 값들이다[8].

<표 2> 로컬 MTU 환경에서의 알고리즘 성능

로컬 MTU (byte)	DnQ 최대 지연시간 (s)	MnA 최대 지연시간 (s)
296	27	6.2
508	30	9.2
512	30	9.2
544	30	9.2
576	30	9.2
1006	33	9.5
1492	36	10
1500	36	10
1536	36	10
2002	36	10
2048	36	10
4352	42	10.1
4464	42	10.1
8166	42	10.3
17914	46	10.5
65535	51	10.5

위의 결과로 미루어 보아 거의 모든 네트워크 환경에서 MnA 방식이 DnQ 방식보다 우수한 성능을 나타냄을 알 수 있다. 이러한 결과가 나타난 이유는 MnA 방식이 성능에 가장 큰 영향을 미치는 재전송 횟수를 줄이기 위해서 적절한 범위를 먼저 찾기 때문이다. MnA 방식은 DnQ 방식보다 최대 5배정도 PMTU를 찾는 시간이 적게 걸린다.

4. 시뮬레이션

4.1 시뮬레이션 목적

본 시뮬레이션은 VINT(Virtual InterNetwork Testbed) 프로젝트에 의해 개발되고 있는 공개 시뮬레이션 툴인 NS2 (Network Simulator 2)를 사용하며 IPSec 터널 모드에서 ICMP 에러 메시지가 도착하지 않는 경우를 대상으로 한다. 이러한 환경에서 현재 일반적으로 사용되는 "don't fragment" 비트를 해제하여 통신하는 방식(앞으로 단편화 방식이라 부르겠다)과 본 논문에서 제시한 적응적 PMTU 발견 메커니즘 중 MnA 방식을 일정 크기의 데이터를 전송하는데 걸리는 지연시간과 사용된 패킷의 수 관점에서 비교 분석한다. 이를 통해 본 논문에서 제시한 방식이 기존의 방식보다 더 우수함을 입증하는데 그 목적이 있다.

4.2 시뮬레이션을 위한 가정

<표 3> 시뮬레이션 가정

항 목	값(ms)
터널 구간 전달 지연시간	50
송신 호스트 패킷 생성 지연시간	2
수신 호스트 지연시간	0
보안 게이트웨이 IPSec 처리 지연시간	7
보안 게이트웨이 단편화 지연시간	10
라우터 라우팅 지연시간	5
라우터 단편화 지연시간	10

<표 3>은 본 시뮬레이션을 수행하기 위해 가정한 사항을 정리한 것이다.

여기서 터널 구간 전달 지연시간은 일반적인 인터넷의 전달 지연 시간 값인 50ms를 취하였다.

송신 호스트 패킷 생성 지연시간은 전체 사용자 데이터를 적절히 패킷화 하여 보내는데 걸리는 지연시간을 의미한다. 수신 호스트 지연시간은 전체 패킷을 받아 원래 데이터를 재결합하는 지연시간을 말하는데 여기서는 패킷이 수신 호스트까지 전달되는 시간만을 고려하므로 이 값을 무시하기로 한다. 보안 게이트웨이 IPSec 처리 지연시간은 패킷에 IPSec 처리를 하는데 걸리는 지연시간을 의미하며 보안 게이트웨이 단편화 지연시간은 보안 게이트웨이에서의 단편화하는데 걸리는 지연시간을 의미한다. 라우터 라우팅 지연시간은 패킷을 라우팅 하는 데 걸리는 지연시간을 말하며 라우터 단편화 지연시간은 라우터에서 단편화하는데 걸리는 지연시간을 말한다.

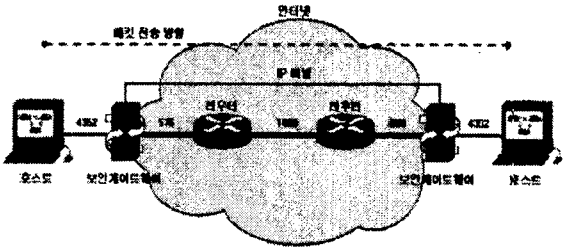
본래 호스트에서 사용자 데이터의 패킷화, 보안 게이트웨이에서의 IPSec 처리와 단편화, 라우터에서의 라우팅과 단편화는 지연시간이라기보다는 자원 사용 비용을 의미하지만 본 시뮬레이션에서는 이들 모두를 지연시간으로 간주하기로 한다. 또한 이들 값들은 부하 정도를 고려하여 임의로 정한 값이다. 임의로 정한 값이지만 시뮬레이션 목적이 정확한 지연시간을 구하기보다는 상대적인 지연시간 차를 구하는 것이 목적이므로 시뮬레이션 목적에 위배되지는 않는다.

4.3 시나리오 I : 단편화가 한번 발생하는 환경

여기서는 단편화가 한번 발생하는 네트워크 환경에서 일정한 크기의 사용자 데이터를 보내는데 걸리는 전송 지연 시간과 총 패킷의 수를 사용자 데이터의 크기를 증가시켜가며 시뮬레이션을 수행하고 이를 통해서 단편화 방식과 MnA 방식의 성능을 비교한다.

4.3.1 네트워크 구성

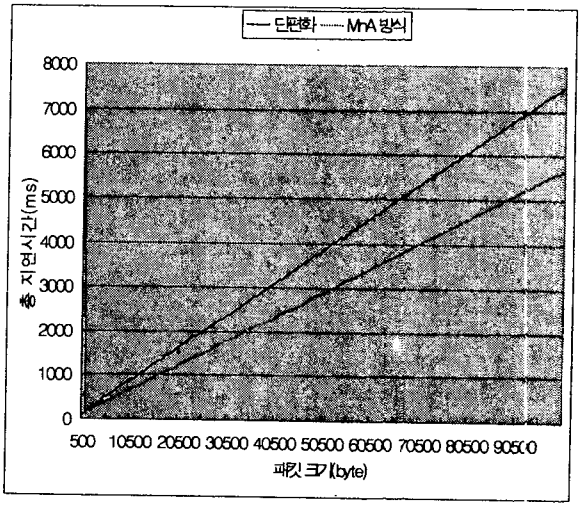
<그림 5>는 시나리오 I의 네트워크 구성도를 나타내고 있다. 단편화 방식으로 전송할 때 단편화는 송신 보안 게이트웨이에서 발생하며, MnA 방식으로 전송할 때 실제 이 네트워크 구성도의 PMTU는 576이지만 이 알고리즘에 의해 발견한 PMTU는 544이고 찾는데 걸린 지연 시간은 6600ms이다.



<그림 5> 시나리오 I 네트워크 구성도

4.3.2 전송 지연시간

<그림 6>은 전송 지연시간에 대한 시뮬레이션 결과를 보여준다. 처음에는 MnA 방식이 더 높은 지연시간을 가지는데 이는 PMTU를 찾기 위해 드는 시간(6600ms) 때문이다. 하지만 한 번 PMTU를 찾으면 라우팅 정보가 변경될 때까지는 다시 PMTU를 찾는 메커니즘을 호출할 필요가 없으므로 MnA 방식이 단편화 방식보다 더 낮은 전송 지연시간을 나타낸다.

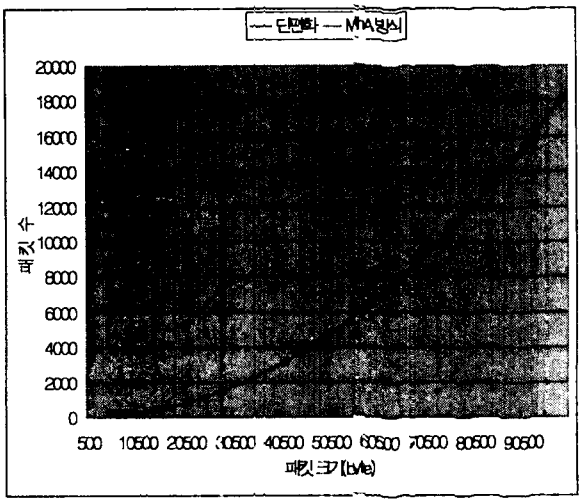


<그림 6> 단편화가 한번 발생하는 상황에서 전송 지연시간 비교

4.3.3 총 전달 패킷 수

<그림 7>은 총 전달 패킷 수에 대한 시뮬레이션 결과를 보여준다. 미세하나마 단편화 방식

이 MnA 방식보다 더 적은 패킷수를 보여주고 있는데 이는 MnA 방식이 단편화 방식에 비해 성능이 떨어진 다기보다는 MnA 방식이 사용자 데이터 크기가 커질수록 상대적으로 적은 PMTU 크기의 많은 패킷을 생성해 내게 되어서 단편화 방식의 패킷 분할 시 발생하는 총 패킷 수를 초과했기 때문이라고 생각된다.



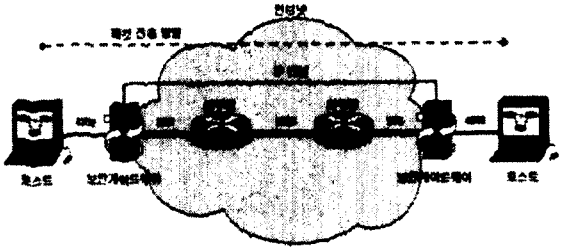
<그림 7> 단편화가 한번 발생하는 상황에서 총 전달 패킷 수 비교

4.4 시나리오II : 단편화가 두번 발생하는 환경

여기서는 단편화가 두 번 발생하는 네트워크 환경에서 일정한 크기의 사용자 데이터를 보내는데 걸리는 전송 지연 시간과 총 패킷의 수를 사용자 데이터를 증가시켜가며 시뮬레이션을 수행하고 이를 통해서 단편화 방식과 MnA 방식의 성능을 비교한다.

4.4.1 네트워크 구성

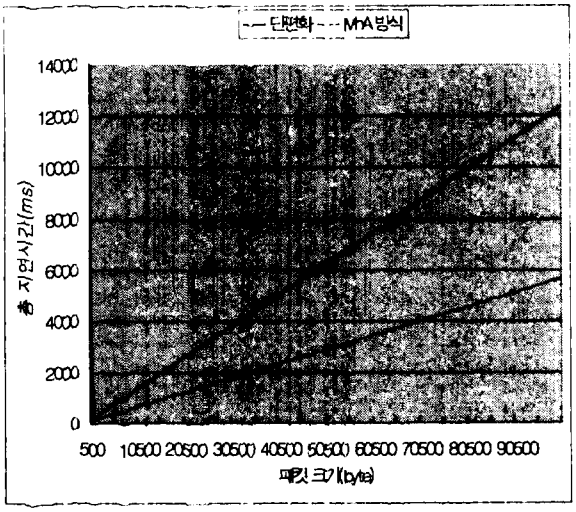
<그림 8>은 시나리오 II의 네트워크 구성도를 나타내고 있다. 단편화 방식으로 전송할 때 단편화는 송신 보안 게이트웨이와 두 번째 라우터에서 발생하며 MnA 방식으로 전송할 때 실제 이 네트워크 구성도의 PMTU는 576이지만 이 알고리즘에 의해 발견한 PMTU는 544이고 찾는데 걸린 지연 시간은 6600ms이다.



<그림 8> 시나리오 II 네트워크 구성도

4.4.2 전송 지연시간

<그림 9>는 전송 지연시간에 대한 시뮬레이션 결과를 보여준다. 처음에 MnA 방식이 더 높은 지연시간을 가지는 이유는 시나리오 I과 같다. 단편화가 두 번 수행됨에 따라 단편화 방식은 시나리오 I에 비해 약 1.5 배의 전송 지연시간을 갖는데 비해 MnA 방식은 시나리오 I과 동일한 전송 지연시간을 갖는다.

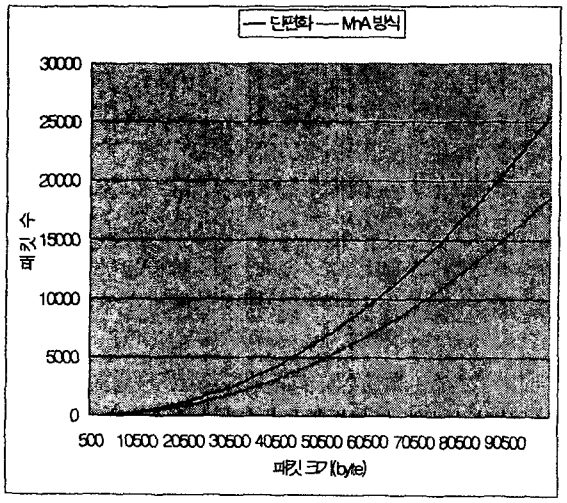


<그림 9> 단편화가 두 번 발생하는 상황에서 전송 지연시간 비교

4.4.3 총 전달 패킷 수

<그림 10>은 총 전달 패킷 수에 대한 시뮬레이션 결과를 보여준다. 시뮬레이션I과는 달리 사용자 데이터 크기가 증가할수록 단편화 방식이

MnA 방식보다 점점 더 많은 패킷수를 보여주고 있다. 이것은 MnA 방식은 PMTU를 찾았기 때문에 단편화 횟수에 관계없이 일정한 패킷 수를 전달하는 반면에 단편화 방식은 단편화 횟수가 증가할수록 더 많은 패킷을 전달해야 한다는 것을 의미한다.



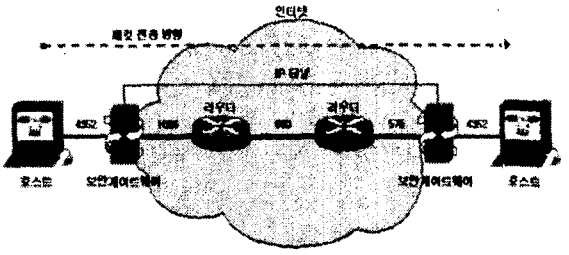
<그림 10> 단편화가 두 번 발생하는 상황에서 총 전달 패킷 수 비교

4.5 시나리오III : 단편화가 세번 발생하는 환경

여기서는 단편화가 세 번 발생하는 네트워크 환경에서 일정한 크기의 사용자 데이터를 보내는데 걸리는 전송 지연 시간과 총 패킷의 수를 사용자 데이터를 증가시켜가며 시뮬레이션을 수행하고 이를 통해서 단편화 방식과 MnA 방식의 성능을 비교한다.

4.5.1 네트워크 구성

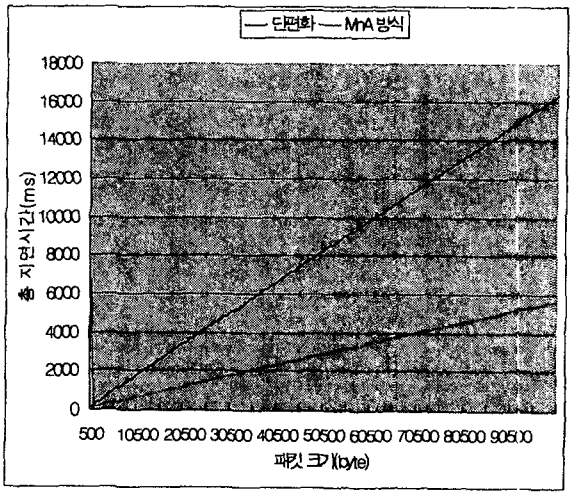
<그림 11>는 시나리오 III의 네트워크 구성도를 나타내고 있다. 단편화 방식으로 전송할 때 단편화는 송신 보안 게이트웨이와 첫 번째, 두 번째 라우터에서 발생하며 MnA 방식으로 전송할 때 실제 이 네트워크 구성도의 PMTU는 576이지만 이 알고리즘에 의해 발견한 PMTU는 544이고 찾는데 걸린 지연 시간은 6600ms이다.



<그림 11> 시나리오 III 네트워크 구성도

4.5.2 전송 지연시간

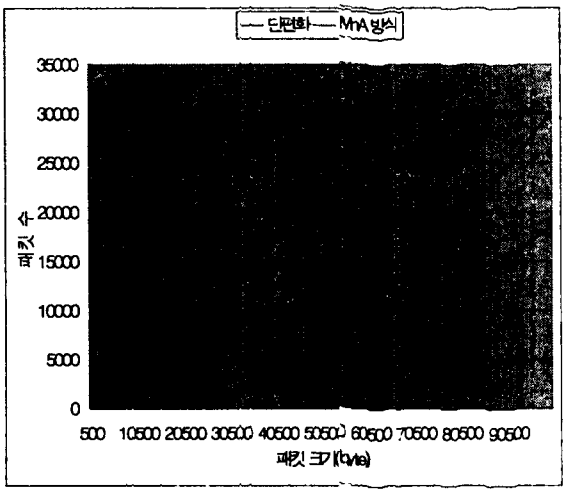
<그림 12>는 전송 지연시간에 대한 시뮬레이션 결과를 보여준다. 처음에 MnA 방식이 더 높은 지연시간을 가지는 이유는 시나리오 I과 같다. 단편화가 세 번 수행됨에 따라 단편화 방식은 시나리오 I에 비해 약 2 배 이상의 전송 지연시간을 갖는데 비해 MnA 방식은 시나리오 I과 동일한 전송 지연시간을 갖는다.



<그림 12> 단편화가 세 번 발생하는 상황에서 전송 지연시간 비교

4.5.3 총 전달 패킷 수

<그림 13>은 총 전달 패킷 수에 대한 시뮬레이션 결과를 보여준다. 시뮬레이션II에서 보다 사용자 데이터 크기가 증가할수록 단편화 방식이 MnA 방식보다 전송해야할 패킷의 수가 더 큰 폭으로 증가했음을 알 수 있다.



<그림 13> 단편화가 세 번 발생하는 상황에서 총 전달 패킷 수 비교

4.6 분석 및 고찰

위의 세 가지 시뮬레이션 결과를 통해서 다음과 같은 사실들을 알 수 있다.

- 지연시간의 증가는 네트워크 자원을 더 많이 소비한다는 것을 의미하고 전달 패킷 수의 증가는 네트워크에 부하를 더 많이 줄뿐만 아니라 패킷을 잃어버릴 가능성이 더 커진다는 것을 의미한다.
- 단편화 방식은 패킷의 네트워크 전달 경로의 단편화 회수에 따라 일정량의 사용자 데이터를 전송하는데 걸리는 지연시간과 패킷의 수가 비례적으로 증가한다.
- MnA 방식은 패킷의 네트워크 전달 경로의 단편화 회수에 관계없이 일정량의 사용자 데이터를 전송하는데 걸리는 지연시간과 패킷의 수가 일정하다.
- MnA 방식의 초기 부하(PMTU를 찾는데 걸리는 시간)는 라우팅 정보가 변경되지 않는 한 단 한번만 수행되면 되므로 시간이 지날수록 단편화 방식보다 지연시간과 패킷 전달 수 측면에서 훨씬 우수한 성능을 나타낸다.

결론적으로 IP 터널링 구간이 ICMP 에러 메시지 전달을 지원하지 않는 네트워크 환경에서는 보안 게이트웨이에 현재 이용되고 있는 단편화 방식을 적용하는 것보다는 본 논문에서 제시한 적응적 PMTU 메커니즘인 MnA 방식을 적용하는 것이 위 시뮬레이션 결과를 통해 보였듯이 네트워크 자원의 낭비를 막을 뿐만 아니라 네트워크의 부하를 줄일 수 있다.

5. 결론

정보 공유를 추구하는 인터넷의 급속한 성장으로 인해 인터넷 사용에 있어서 여러 가지 보안상의 문제점들이 발생하게 되고, 네트워크를 통한 데이터의 공유 또한 보편화되고 활성화됨에 따라 네트워크 보안의 문제와 그를 위한 비용의 문제가 크게 대두되게 되었다. 이런 환경에서 VPN은 기업의 네트워크 구축 시 비용절감 효과를 비롯하여 보안 및 신뢰성을 보장할 수 있는 특징을 갖고 있기 때문에 기업들이 네트워크를 구축하고자 할 때 가장 많이 고려하고 있는 네트워크 구축 형태로서 최근 정보 보호에 대한 기업들의 요구 증가로 인해 시장 확대가 급속히 이루어지고 있다.

본 논문에서는 이러한 VPN을 구현하기 위해서 IPSec을 이용하려 할 때 패킷 단편화 현상으로 인해 발생하는 통신 불능 현상의 원인에 대해서 자세히 고찰해 보았고 이러한 통신 불능 현상을 해결하기 위한 기존의 방법들을 조사하고 새로운 해결 방안으로서 적응적 PMTU 발견 메커니즘을 제시하였다. 또한 이 메커니즘의 우수성을 증명하기 위해서 기존 방식과 시뮬레이션을 통해 비교 분석하였다.

향후 인터넷을 통한 전자상거래의 활성화가 확실히 되는 시점에서 IP에 보안을 지원하는 IPSec의 안정적인 통신을 위한 다양한 시도들은 그 자체로도 매우 의미 있는 일이라 생각된다. 따라서 더욱 안정적인 통신을 위한 다양한 방법에 대한 연구가 필요할 것으로 생각된다.

참고 문헌

[1] Dan Harkins, Naganand Doraswamy, "IP Sec: The New Security Standard for the Internet, Intranets and Virtual Private Networks", Prentice Hall, 1999.

[2] Chris Brenton, "Network Security", SYBEX, 1999.

[3] Daniel Lynch and Marchall Rose, "Internet System Hand book", Addison Wesley, 1993.

[4] Charlie Scott, Paul Wolfe and Mike Erwin, "Virtual Private Network", O'Reilly, 1998.

[5] Steven Brown, "Implementing Virtual Private Networks", McGrow-Hill, 1999.

[6] C. Kent and J. Mogul, "Fragmentation Considered Harmful", SIGCOMM, 1987.

[7] Van Jacobson, "Congestion Avoidance and Control", SIGCOMM, 1988.

[8] J. Mogul, S. Deering, "Path MTU Discovery", RFC 1191, IETF, 1990.

[9] Stephen Kent, Randall Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, IETF, 1998.

[10] Stephen Kent, Randall Atkinson, "IP Authentication Header", RFC 2402, IETF, 1998.

[11] Stephen Kent, Randall Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 2406, IETF, 1998.

[12] Derrell Piper, "The Internet IP Security Domain of Interpretation for ISAKMP", RFC 2407, IETF, 1998.

[13] Douglas Maughan, Mark Schneider, et. "Internet Security Association and Key Management Protocol (ISAKMP)", RFC 2408, IETF, 1998.

[14] D.Harkins, D.Carrel, "The Internet Key Exchange (IKE)", RFC 2409, IETF, 1998.

[15] David D. Clark, "IP Datagram Reasse -mbly Algorithms", RFC 815, Network Information Center, SRI International, July 1982.

[16] S. Bellovin, "Problem Areas for the IP Security Protocols", Proc. Sixth Usenix Security Symp., Usenix Assoc., Berkeley, Calif., 1996.

[17] Perkins. C., "IP Encapsulation within IP", RFC2003, October 1996.

[18] Calhoun, P. et al., "Tunnel Establishment Protocol", Internet Draft, March 1998.

저자소개

김은성

2000 성균관대학교 전기전자및컴퓨터공학부 학사

2002 성균관대학교 전기전자및컴퓨터공학부 석사

2002~현재 한국과학기술정보연구원 슈퍼컴퓨팅센터 연구원

관심 분야 : GRID, 클러스터링, 네트워크 보안, 네트워크 관리

저자소개

안성진

1988 성균관대학교 정보공학과 학사
1990 성균관대학교 정보공학과 석사
1990~1995 한국전자통신연구원 연구전산망 개발실 연구원
1996 정보통신 기술사 자격 취득
1998 성균관대학교 정보공학과 박사
1999~현재 성균관대학교 컴퓨터교육과 조교수
관심 분야 : 네트워크 관리, 트래픽 분석 , Unix 네트워킹

정진욱

1974 성균관대학교 전기공학과 학사
1979 성균관대학교 전자공학과 석사
1991 서울대학교 계산통계학과 박사
1982~1985 한국과학기술 연구소 실장
1981~1982 Racal Milgo Co. 객원연구원
1985~현재 성균관대학교 정보통신공학부 교수
관심 분야 : 컴퓨터 네트워크, 네트워크 관리, 네트워크 보안

이도훈

1989 한양대학교 전자계산학과 학사
1991 한양대학교 전자계산학과 석사
1991~2000 국방과학연구소 선임연구원
2000~현재 국가보안기술연구소 선임연구원
관심 분야 : 컴퓨터 네트워크, 네트워크 및 컴퓨터 보안

윤재우

1983 전북대학교 전자공학과 학사
1985 전북대학교 전자공학과 석사
1988~현재 한국전자통신연구원 책임연구원
관심 분야 : Information Assurance, Crypto-processor Design, Secure TCP/IP Protocols, Key Management Protocols