보안 예측기능이 있는 IPv4/IPv6 변환기 설계

장성만* · 길민욱** · 이 극*
*한남대학교 컴퓨터공학과 · **문경대학교 인터넷정보계열

요 약

통신망에서 Internet Protocol version 4(IPv4)에서 IPv6로의 전환은 필수적이나 실제 상용망에서는 적용되지는 않고 있고 현재 실험망으로만 가동되고 있다. 본 논문에서는 Internet Protocol version 4(IPv4)의 한계로 인해 등장한 새로운 프로토콜인 IPv6로의 변환이 필요한 시점에서 IPv6에 대하여 살펴보고, 보안 문제를 일으킬 수 있는 프로토콜의 요소를 사전에 예측하게 함으로써 IPv6 망으로의 전환시 유·무선 환경에 있어 안전한 전환을 위한 사전 시험 기능을 제시한다.

Design of IPv4/IPv6 Converter with Security Prediction Capability

Sung-man Jang* · Min-Wook Kil** · Geuk Lee*

ABSTRACT

It is necessary to change internt protocl from version 4 (IPv4) to version 6 (IPv6). A converter from IPv4 to IPv6 is working in a laboratory not in practice. In this paper, we review internet protocol version6 (IPv6) and design IPv4 to IPv6 converter. The IPv4 to IPv6 converter also has security prediction capability so as to deal with security problem when IPv4 packet is converted to IPv6 packet.

Ⅰ. 서 론

현재의 IPv4는 RFC791에서 정의된 프로토콜이며 다음과 같은 큰 특징을 갖는다.

첫째, 통합된 주소(universal address)의 특징을 가진다. 즉 네트워크 인터페이스는 유일한 32비트 길이의 유일한 주소를 가지게된다. 그러나 기존의 IPv4 체계에서는 클래스 단위의 주소 할당 방식으로 인해 할당되고 남은 IP 주소(address)가 낭비되고 있다.

둘째, 비연결형 서비스이다. 이 특징은 패킷 전송시 제어를 매우 단순하게 한다.

셋째, 최선의 노력형(best effort) 서비스이다. 비연결형인 대신에 패킷의 전달 비율이나 소요시간 등에 최선의 노력을 하지만, 전송에 대한 어떠한 보장도 하지 않는다.

현재의 IPv4는 폭발적으로 증가하고 있는 인터넷 접속기반 서비스의 증가로주소 고갈 문제에 직면하고 있다[1]. 이는 인터넷 사용자의 증가와 인터넷 접속이 기존의 환경 뿐만 아니라 무선인터넷과 이동기기가 인터넷 접속이 가능한 형태로 발달함에 따른 주소 고갈 문제를 야기하고 있다. 또 하나의 문제는 기업환경, 개인, 공공 서비스 등이 인터넷 접속기반을 추구하고 있다.

현재는 이러한 주소 고갈 문제를 해결하기 위해 CIDR(Class Inter-Domain Routing), NAT(Network Address Translation), DHCP(Dynamic Host Configuration Protocol) 등을 도입하였으나 단기적인 해결책에 그치고 있다[2]. IPv4는 인터넷 기반의 전자상거래에 따른 보안 문제, 프로토콜 자체의 문제점에 의한 보안 문제, 그리고 기존의 공중전화망 또한 인터넷망과의 통합쪽되면서 실시간 데이터 전송을 요하는 음성, 화상 서비스에서의 QoS 보장 문제를 가지고 있다. 이러한 문제를 해결하기 위한 차세대인터넷으로의 이전에 대해 고려할 필요가 대두되고 있다. 따라서 새로운 방식의 프로토콜의 필요성이 제기되고 있다.

IPv4의 주소 고갈 문제를 비롯한 문제를 해결하고, 차세대 인터넷을 위한 다양한 기반 기술을 제공하기 위한 새로운 주소지정 방식은 다음과 같은 특성을 요구한다[1].

첫째, 주소공간 고갈 문제를 해결하기 위한 더욱 큰 주소 비트 수를 가진다. 이를 위해 클래스 기반 방식이 아닌 CIDR[2]를 사용하는 유연한 계층적 주소 구성이 필요하다.

둘째, 차세대 인터넷을 위한 라우팅의 효율성, 보안기능, 이동성 지원, 서비스의 질 보장 등 다양한 기반 기술 제공을 제공하여야한다.

위와 같은 특성을 달성하기 위해 등장한 것이 IPv6(Internet Protocol Version 6)[3]이다. 즉 IPv6은 기존의 IPv4와 같은 기능을 제공하면서, 더욱 향상된 기능 즉 향상된 라우팅, 보안, 높은 수행성을 제공하는 형태로 개발되었다.

IPv4와 IPv6간의 연동호환성 문제에 있어서는 IPv6는 IPv4와 연동되지 않으며, IPv6를 사용하기 위해선 모든 네트워크 장치내의 소프트웨어에 대한 수정이 필요하다. 또한 현재 거의 대부분이 IPv4 방식으로 작동하고 있으므로 상당기간 IPv6/IPv4가 상호 공존할것으로 예상된다. 결과적으로 IPv6로 기존의 호스트와 네트워크를 운영하기 위해서는 응용 프로그램이 계속해서 수행될 수 있도록 하는 IPv4/IPv6 변환 메커니즘이 필요하다[4].

본 논문에서는 IP 프로토콜 변환에 대한 설계 및 표준 RFC문서를 바탕으로 IPv6 프로토콜에 대한 시험 절차를 제안하였다. 다양한 시험 환경을 정의하였다. 시험 절차의 목적은 표준문서를 기반으로 상이한 조건에서의 실행을 통해 IPv6와의 상

호 운용성을 향상시키고, 각종 응용들에 대한 적응성을 시험하는 것에 목적으로 한다. 이러한 시험 절차를 기본으로 하여 IPv4/IPv6 변환시 프로토콜 보안 예측 기능을 갖는 변환 시스템을 설계하고자 한다. 시험 시나리오(testing scenario)의 목적은 표준문서를 기반으로 상이한 조건에서의 실행을 통해 IPv6와의 상호 운용성을 향상시키는데 있다.

II. IPv4/IPv6 프로토콜

2.1 IPv4 구조

이 절에서는 IPv4 헤더를 설명하고, 주소체계에 대하여 살펴본다.

2.1.1 IPv4 헤더 및 주소체계

IP는 신뢰성이 없는 비연결형 전송 서비스를 제공한다. IPv4 헤더의 각 필드에 대해 살펴보면, 다음 (그림1)과 같다.

<table>
<tr><td>0 4</td><td>8</td><td>16</td><td colspan="2">31</td></tr>
<tr><td>Ver</td><td>H-Len</td><td>TOS</td><td colspan="2">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td>Fragment offset</td></tr>
<tr><td colspan="2">TTL</td><td>Protocol</td><td colspan="2">Header Checksum</td></tr>
<tr><td colspan="5">Source Address</td></tr>
<tr><td colspan="5">Destination Address</td></tr>
<tr><td colspan="5">Option</td></tr>
<tr><td colspan="5">data</td></tr>
</table>

(그림 1) IPv4 헤더필드와 IPv4데이터그램

인터넷에 접속하기 위해서는 고유의 총 32비트의 인터넷 주소가 인터페이스 단위로 할당 되어야 한다. 이러한 주소 유형에는 유니캐스트(unicast), 멀티캐스트(multicast), 브로트캐스트(broadcast) 등 세가지가 있다. IPv4의 주소 공간은 클래스 단위로 나누어져 있다. 이와 같은 전통적인 주소클래스 기반외에 CIDR 주소 마스크를 사용하기도 한다.

2.2 IPv6의 구조

2.2.1 IPv6 헤더 및 주소체계

IPv6는 IPv4를 이은 IP의 새로운 버전을 말하며, 다음과 같은 특징을 갖는다.

① 확장된 주소 공간(128비트의 주소를 사용)과 유니캐스트, 멀티캐스트 외에 애니케스트(anycast) 주소 개념을 도입하였다.

② 기본 헤더 길이는 40바이트이지만 전체 필드 수를 12개에서 8개로 단순화시켜 기본적인 처리 속도를 향상하는 효과를 가지고 왔다.

③ 확장 헤더(extension header)의 정의로 패킷의 대부분은 전달되는 경로상의 라우터상에서 처리될 필요가 없으므로 포워딩의 효율이 높아졌다.

④ IPv6은 인증, 데이터 무결성, 기밀 유지 기능을 지원하기 위해 확장헤더를 정의한다.

IPv6 패킷은 기본헤더(basic header)와 확장헤더(extension header) 그리고 페이로드(payload)로 구성되어 있다. 또한 IPv6의 기본 헤더 구조는 아래 (그림2)와 같다.

<table>
<tr><td>0 4</td><td>12</td><td colspan="3">31</td></tr>
<tr><td>Ver</td><td>Traffic Class</td><td colspan="3">Flow Label</td></tr>
<tr><td colspan="2">Payload Length</td><td>Next Header</td><td colspan="2">Hop Limit</td></tr>
<tr><td colspan="5">Source Address(128Bbts)</td></tr>
<tr><td colspan="5">Destination Address(128Bbts)</td></tr>
</table>

(그림 2) IPv6 기본 헤더 구조

IPv6 패킷 포맷은 아래와 같다.

- traffic class : IPv4의 서비스 유형(TOS : Type Of Service)필드와 같은 기능 수행
- flow label : 송신 노드에서 라우터에 의해 특별히 다루어 지기를 요청하는 패킷 흐름을 지정하는 식별자
- payload length : 패킷 전체 길이 중에서 IPv6 기본헤더를 제외한 길이를 나타냄
- next header : IPv6 기본헤더 바로 다음에 위치하는 확장헤더의 종류를 나타냄
- hop limit : IPv4의 TTL 필드와 같은 역할, 전달이 허용되는 최대 홉(hop)수를 나타낸다.

IPv6 주소는 128비트로 인터페이스들과 인터페이스 집합을 식별한다.

IPv6에서는 유니캐스트, 애니캐스트와 브로드캐스트를 대신하여 멀티캐스트가 쓰인다. IPv6주소를 문자열로 표현한 일반적인 형태는 16비트 8개의 필드로 주소를 나타낸다. 이 IPv6주소 유형과 서브넷은 이진 프리픽스(prefix)를 통해 구별하며, 물리적인 인터페이스는 인터페이스(interface) ID로 구분한다.

2.2.2 ICMPv4 / ICMPv6

에러와 정보 메시지 전달을 위한 목적으로 ICMPv4[5] / ICMPv6[6]를 사용한다.

주요 특징을 살펴보면 IP패킷에 대한 응답으로 경로상의 라우터나 목적지 노드가 응답하는데 사용되며, 기존 ICMPv4에서 독립적으로 존재하는 ARP, RARP의 기능을 통합하고 있다. 에러 메시지는 유형필드의 최상위 비트가 0이며, 정보 메시지는 최상위 비트가 1인 것으로 구분할 수 있다. 코드 필드는 유형 값에 따라 세부 사항을 분류하기 위해 사용된다. 일반적인 ICMP의 메시지 형식은 다음 (그림 3)과 같다.

0 7	8 15	16 31
Type	Code	Checksum
Type 과 Code에 따른 내용		

(그림 3) ICMP 메시지 형식

2.3 IPv4/IPv6 비교

위에서 살펴본 IPv4와 IPv6 프로토콜 및 헤더의 차이는 다음 (표 1), (표 2)과 같이 정리될 수 있다.

(표 1) IPv4와 IPv6 프로토콜 비교

	IPv4	IPv6
주소 길이	32bits	128bits
Flow Labels	정의되지 않음	정의됨
Header Checksum	체크섬을 계산	정의되지 않음
단편화 정보	모든 데이터 그램에 정보 삽입	확장헤더로지정
패킷헤더	기본 20바이트(가변적)	40바이트 고정
주소할당 방법	클래스기반, CIDR	CIDR기반 계층적 할당
주소 유형	unicast, multicast, broadcast	unicast, multicast, anycast

(표 2) IPv4와 IPv6 프로토콜의 헤더의 차이

IPv4 header filed	IPv6 header filed
Version	Version
Header Length	정의되지 않음 Flow Label
TOS	Traffic Class
Total Length	Payload Length
Identification Flags Fragment offset	확장헤더로 별도 정의됨 (Fragment header)
TTL	Hop Limit
Protocol	Next Header
Header Checksum	정의되지 않음
32bits Source Address Destination Address	128bits Source Address Destination Address
Option	확장헤더

Ⅲ. IPv4 / IPv6 변환 프로토콜의 설계

3.1 IPv4/IPv6 프로토콜의 변환

프로토콜 변환은 두 개의 IP 프로토콜 필드에 대응하여 구성할 수 있다.

2절에서 살펴보았듯이 IPv6은 IPv4를 삽입할 수 있는 두 가지 형태의 주소를 제공한다. 이 주소방식을 바탕으로 하여 변환의 기본적인 전략은 원래의 IP 헤더를 제거하고 다른 IP 헤더로 대치하는 것이다.

(1) IP 헤더 변환

IPv6과 IPv4는 매우 유사하지만 생략되었거나 다른 크기, 또는 다른 의미를 가지는 필드가 존재한다. 따라서 IP 변환은 직접적으로 복사하거나 변환하거나 무시하거나 또는 IP 헤더내의 필드들을 기본값으로 설정한다. 각 필드에 대응하는 변환은 비교적 단순하다. IPv4의 체크섬 필드는 IPv6에서 IPv4로 변환될 때 사용되며, IPv4에서 IPv6로 변환될 때는 무시된다. IPv4의 총길이(total length) 필드는 IPv4의 헤더길이를 포함하지만 IPv6의 페이로드 길이(payload length) 필드는 헤더길이를 포함하지 않으며, 페이로드 길이 필드는 패킷의 단편화 여부에 따라 다르게 계산되어야 한다. 단편화 되었다면, IPv6의 단편화 헤더(fragment header) 길이만큼을 더해 주어야 한다. TTL, Hop Limit 필드는 각각 필드에서 1을 뺀 값을 직접 복사한다. 프로토콜(protocol) 필드는 직접 복사되며, 만일 상위 프로토콜이 ICMP이면 ICMP 메시지 변환을 수행한다. IPv6 단편화 헤더(fragment header)를 제외하고 IPv6 확장헤더(extension header)와 IPv4 옵션(option) 필드는 누락된다. IPv4 TOS(Type Of Service)와 IPv6 트래픽 클래스(traffic class) 필드와 플루 라벨(flow label) 필드는 적절한 대응이 존재하지 않으므로 무시된다.

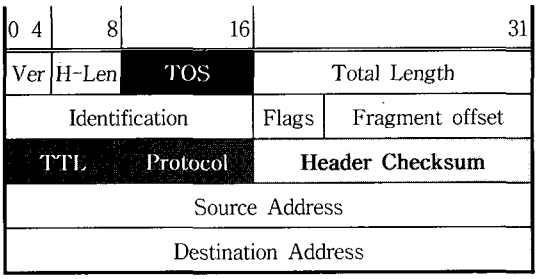
만약 변환시 단편화(fragment)된 패킷을 수신하였을 경우에는, IPv4와 IPv6의 단편화 필드(fragmentation filed)들간에 직접적으로 복사한다. 다만 차이점은 IPv6에서 32비트의 단편화 헤더(fragment header)를 확장헤더로 정의하였기 때문에 단편화 식별 필드의 크기가 다르다는 것이다.

(2) ICMP 헤더 변환

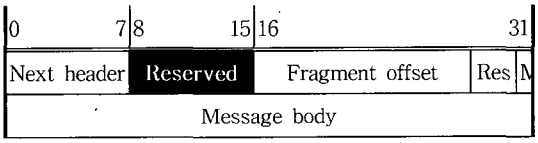
ICMP 헤더 구조형식은 ICMPv4, ICMPv6 양쪽에서 유사하게 나타난다. 양쪽 ICMP 메시지와 에러에는 목적지 접근 불가(destination unreachable) 에러 메시지, 너무 큰 패킷(packet too big) 에러 메시지, 시간 초과(time exceeded) 에러 메시지, 매개 변수(parameter problem) 에러 메시지, 에코 요청(echo requet) 정보 메시지, 에코 응답(echo reply) 정보 메시지 등이 있다. 일반적으로 ICMP 유형과 코드 필드를 변환 시켜서 ICMP 메시지에 대한 변환을 한다. ICMP 에러 메시지는 에러를 유발한 패킷의 IP 헤더를 포함하여 변환을 시도한다.

너무 큰 패킷(packet too big) 에러 메시지가 도착한 경우에는 최대전송단위(MTU : Maximum Transmission Unit)을 변환기가 조정할 필요가 있음을 말한다. 그리고 매개 변수(parameter problem) 에러 메시지는 확장헤더에서 문제가 발생한 경우 에러를 유발한 IP 헤더내의 정확한 필드를 가리키도록 조정해야 한다. ICMP 에러 메시지는 에러를 일으킨 IP 헤더와 정확한 값으로 조절할 수 있는 데이터의 종류만큼의 에러 메시지를 가지고 있다. 이러한 ICMP 메시지들은 송신될 때 정상적인 IP 헤더처럼 변환될 필요가 있다. 즉

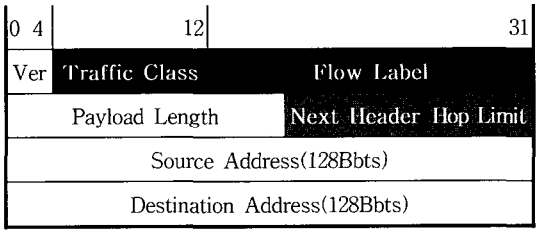
ICMP 에러 메시지를 포함하는 IP 패킷의 변환은 라우터 또는 수신자에서 반복적으로 이루어 져야 한다. 아래 (그림 4)는 위와 같은 내용을 설명한다. 또한 IPv6은 패킷의 단편화(fragment)에 대해 별도의 확장헤더를 정의하여 단편화 정보를 저장한다.



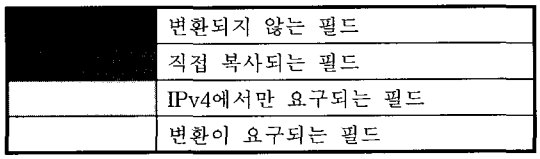
(a) IPv4 헤더



(b) IPv6 fragmentation Header



(c) IPv6 기본 Header 구조



(d) 각 필드 특성

(그림 4) 프로토콜 변환 필드

(3) IPv4-to-IPv6, IPv4-to-IPv6 프로토콜 변환 기능부의 동작 절차

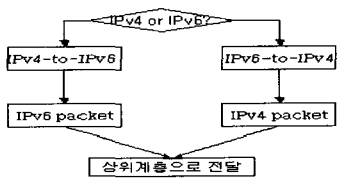
3.2.2에 기술된 IP 프로토콜 및 ICMP 메시지 변환 기본 설계를 토대로 각 단계별 동작의 절차를 기술한다. 아래 (그림 5)은 IPv6헤더의 각 필드를 나타내는 구조체이다. 변환시에는 이 구조체에 맞도록 변환한다.

```
struct ip6_hdr
 union
  struct ip6_hdrctl
   uint32_t    ip6_un1_flow;   /*  24  bits  of
                                  flow-ID */
   uint16_t ip6_un1_plen;/* payload length */
   uint8_t  ip6_un1_nxt;   /* next header */
   uint8_t  ip6_un1_hlim;   /* hop limit */
   ip6_un1;
   uint8_t ip6_un2_vfc;
/*4bits version, 4bits priority*/
ip6_ctlun;
   struct    in6_addr    ip6_src;/*source    address
                        */
    struct   in6_addr    ip6_dst;   /*   destination
                         address */
  ;
```

(그림 5) IPv6 헤더 구조체

본 논문에서는 기본적인 IPv4/IPv6 전환 메커니즘-IPv4/IPv6 듀얼 스택, IPv6-in-IPv4 터널링-중에서 호스트와 라우터에서 두 IP 프로토콜을 모두 지원하는 방식인 IPv4/IPv6 듀얼 스택에 대하여 변환 모듈을 적용한다.

아래 (그림 6)은 변환 모듈의 구조를 도식화한 것이다. 각 목적지에 전달된 패킷이 IPv4인지 IPv6인지를 구별하게되며 각 호스트가 사용하고 있는 프로토콜 스택에 따라 적절한 변환을 처리하게 된다.



(그림 6) 변환 모듈구조

다음의 (표 3), (표 4)는 설계된 변환 모듈에서 양쪽 IP간에 변환되는 필드를 나타낸다.

(표 3)IPv4-to-IPv6

	IPv6
4->6	Version
TOS필드 복사	Traffic Class
정의되지 않음 0 으로 설정	Flow Label
	Payload Length
	Fragment header Option(확장헤더)
TTL 값에서 1을 뺀값을 복사	Hop Limit
Protocol 필드값 복사	Next Header
IPv4-mapped address 사용	Source Address
IPv4-mapped address 사용	Destination Address

(표 4) IPv6-to-IPv4

IPv4	
Version	6->4
H-Len	옵션이 없는 경우 기본값 5로 설정
TOS	Traffic Class필드 복사
Total Length	
Identification	0으로 설정
Flags	1=MF 0=DF
Fragment offset	0으로 설정
TTL	Hop Limit값에서 1을 뺀값을 복사
Protocol	Next Header값 복사
Header Checksum	
Source Address	IPv6 Source Address의 하위 32비트 복사
Destination Address	IPv6 Destination Address의 하위 32비트 복사

변환 모듈과 IP 헤더구조를 기반으로한 변환 기능 동작 순서는 다음과 같다. 각각의 필드 값은 위의 표를 참고한다. 다음 헤더(next header) 필드의 값은 상위 계층의 프로토콜을 구분하기 위해 처리 과정의 맨 마지막에 변환된다.

인터넷 프로토콜(Internet Protocol) 변환 순서

첫 번째, version 필드 변환

두 번째, traffic class 필드 변환

세 번째, flow label 필드 변환

네 번째, payload length 필드 변환

다섯 번째, hop limit 필드 변환

여섯 번째, source address 필드 변환

일곱 번째, destination address 필드 변환

여덟 번째, next header 필드 변환

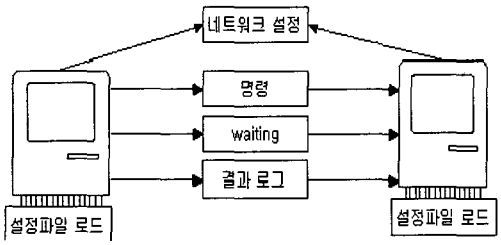
Ⅳ. 비정상 프로토콜의 생성 모듈

인터넷 어플리케이션 클라이언트 테스트는 클라이언트에서의 IPv4-to-IPv6, IPv6-to-IPv4에대한 처리를 테스트하면 됨으로 두 개의 IP를 처리할 수 있는 듀얼 스택이 요구된다. 본 테스트 시나리오에서는 기존의 IPv4 네트워크상의 호스트에 IPv6을 지원하도록 리눅스(커널 2.4.X은 기본적으로 지원) 윈도우 2000(서비스 팩 버전1) 등의 각 운영체제의 커널 설정을 필요로 한다. 이를 위해 윈도우2000 시스템에 IPv6 프로토콜을 설치하였다.

테스팅을 위한 시나리오는 RFC2460(Internet Protocol, Version 6 (IPv6)Specification 및 RFC2463(Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification)을 기초로 구성한다.

각 호스트에는 환경 설정 파일, 네트워크 설정 파일, 그리고 테스팅을 실행할 명령어 파일이 필요하다. 환경 설정 파일은 호스트의로그 파일, 헤더파일 위치 등 내부 설정을 위하여 사용되며, 네트워크 설정파일은 IP 통신을 위하여 가상의 네트

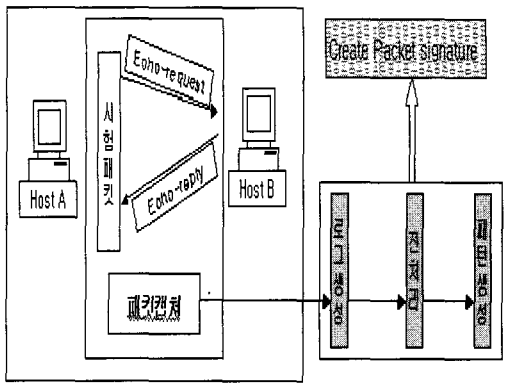
워크 토폴로지를 구성한다. 마지막으로 명령어 파일은 앞의 두 파일이 설정되고 호스트간의 동기화 및 명령 전송으로 실제로 각 환경에 따른 테스팅을 실행하기 위한 파일이다. 아래 (그림 7)과 같은 기본 구조를 가진다. 여기서 IP프로토콜의 로그 파일 생성시 각 로그 파일의 타임 스탬프 확인을 위한 데이터 전송시간을 일치시켜 준다.



(그림 7) 실행과정

4.1 시험 시스템의 구조

앞 부분에서 제시한 각 고려사항 및 설정 사항을 기반으로 하여, (그림 14)과 같이 시스템을 구성하였다. 호스트들 간의 통신은 시험패킷을 통해 이루어진다. 시험 패킷은 시험을 위해 각 헤더에 대한 조작을 통해 사전에 정의된 것이며, 시험의 목적별로 달리하여야 한다. echo 메시지를 통해 패킷의 전달을 확인할 수 있으며, 오류를 검출할 수 있으며, 호스트간 패킷을 캡쳐하기 위한 모듈을 가진다. 이러한 전달된 패킷에 따라 각 시스템은 로그를 기록하게 하며, 각종 패킷의 자극에 시스템이 반응한 패턴을 생성하고, 취약점을 예측하기 위해서는 필요한 정보를 추출하기 위한 전처리 과정이 필수적이다. 추출된 정보를 바탕으로 패턴을 생성하여, 최종적으로 패킷의 자극에 대한 반응을 형성하게 한다.



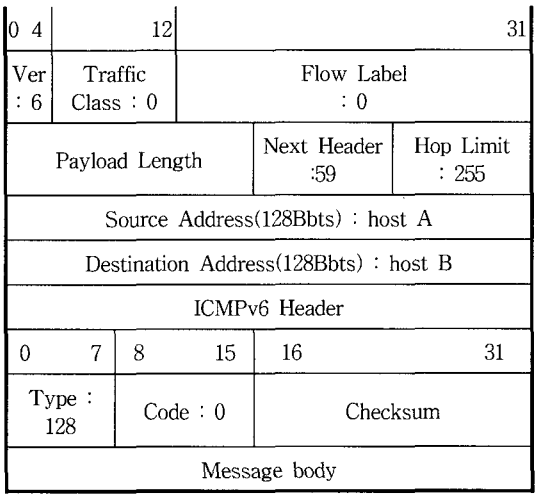
(그림 8) system work flow

4.2 비정상 IP 프로토콜의 생성

기본적으로 다음처럼 패킷을 초기화 한다. IPv6 헤더를 (그림 9)처럼 설정하며, 응답을 받기위한 ICMP 메시지를 (그림 10)같이 설정한다. next header 필드는 코드 59로 설정하여 IPv6 기본헤더만을 생성하도록한다. 이더넷 상에서 각 메시지에 대한 ICMP echo request 정보 메시지를 생성하기 위해 ICMP 메시지의 유형을 128로 설정한다.

<table>
<tr><td>0 4</td><td>12</td><td colspan="2">31</td></tr>
<tr><td>Ver
: 6</td><td>Traffic
Class : 0</td><td colspan="2">Flow Label
: 0</td></tr>
<tr><td colspan="2">Payload Length</td><td>Next Header
:59</td><td>Hop Limit
: 255</td></tr>
<tr><td colspan="4">Source Address(128Bbts) : host A</td></tr>
<tr><td colspan="4">Destination Address(128Bbts) : host B</td></tr>
</table>

(그림 9) 기본 패킷 설정



(그림 10) ICMP 메시지 설정 패킷

'next header'필드IPv6 기본 헤더 다음에 위치하는 헤더의 종류를 지시한다. 아래의 (표 5)는 이러한 'next header' 필드의 코드값을 나타낸다.

(표 5)'next header' 코드값

0	Hop-by-hop Option Header
2	Internet Control Message Protocol
4	Internet Protocol
17	Transmission Control Protocol
41	Internet Protocol Version 6
43	Routing Header
44	Fragment Header
45	Inter-domain Routing Protocol
46	Resource Reservation Protocol
50	Encapsulating Security Payload
58	Internet Control Message Protocol Version 6
59	No Next Header
60	Destination Option Header

다음에 나타나는 (그림 11~18)은 생성된 비정상적인 패킷을 나타낸다. 이를 바탕으로 IPv6프로토콜 스택을 사용하는 시스템에서의 나타날 수 있는 보안 문제를 예측할 수 있다. 각각의 패킷에 대한 검증사항은 목표로 기술하였다. 각 목표의 하단에는 비정상적인 패킷에 대응되는 RFC 문서의 해당 부분은 별도로 표기하였다. 생성된 비정상 패킷은 (표 6)과 같다.

(표 60 생성된 비정상 패킷

(그림11) 알려지지 않은 'next header'패킷
(그림12) 확장헤더가 중복된 패킷
(그림13) 'payload length' 필드가 0인 패킷
(그림14) 확장헤더 처리 순서 검증 패킷
(그림15)ICMP 에러 메시지 처리 검증 패킷
(그림16) 여러개의 옵션처리 검증 패킷
(그림17) 단편화된 패킷 처리 검증
(그림18) 단편화된 패킷을 모두 수신하지 못한 경우 처리 검증

(1) 알려지지 않은 'next header'패킷

목표 : (그림 11)과 같이 노드가 알려지지 않은 'next header'패킷을 버리는지 검증하며, 패킷의 발신자에게 'ICMPv6 parameter problem' 메시지를 전송하는지를 검증한다.

이와 관련된 RFP는 다음과 같다.

-RFC-2460-4. IPv6 Extension Headers

-RFC-2463-3.4 Parameter Problem Message

Packet 형식
IPv6 Header Next Header : 138(unknown)

(그림 11) 알려지지 않은 'next header'패킷

각 노드는 'next header'를 처리하여야 하지만, 이처럼 알려지지 않은 값에 대해서는 패킷을 버려야한다. 'ICMPv6 parameter problem' 메시지를 'Code 1'로 설정하여 발신자에게 응답을 한다.

(2) 확장헤더가 중복된 패킷

목표 : (그림 12)와 같이 노드가 IP 헤더 이외의 확장헤더에서 'next header'필드가 0으로 설정된 패킷을 버리는지 검증하며, 패킷의 발신자에게 'ICMPv6 Parameter Problem' 메시지를 전송하는지를 검증한다.

Packet 형식
IPv6 Header Next Header : 0
Hop-by-hop option Header Next Header : 0 Header Ext Len(확장헤더 길이) : 0 Option
Hop-by-hop option Header Next Header : 59 Header Ext Len(확장헤더 길이) : 0 Option

(그림 12) 확장헤더가 중복된 패킷

기본헤더 다음의 확장헤더는 'destination option header'를 제외하고는 한 번만 나타나야한다. 만약 'hop-by-hop option header'의 'next header'필드가 0이면 다음에 다시 'hop-by-hop option header' 헤더가 나타나게 된다. 이러한 패킷은 버려져야한다.

(3) 'payload length' 필드가 0인 패킷

목표 : (그림 13)은 각 노드가 'payload length' 필드가 0이고, 'next header'필드를 59로 설정된 패킷을 수신한 경우 더 이상 패킷을 생성하지 않는 것을 검증한다.

이와 관련된 RFP는 다음과 같다.

-RFC-2460-3. IPv6 Header Format

-RFC-2460-4. IPv6 Extension Headers

Packet 형식
IPv6 Header Payload Length : 0 Next Header : 59

(그림 13) 'payload length' 필드가 0인 패킷

'payload length'는 헤더 이후의 나머지 패킷에 대한 길이를 표시한다. 이 필드를 0으로 설정한 후에는 'hop-by-hop options header'를 사용한다. 이 필드가 패킷의 끝을 나타내고 'next header'필드가 59로 설정된 패킷은 반드시 무시하여야 한다.

(4) 확장헤더 처리 순서 검증 패킷

목표 : (그림 14)는 IPv6 패킷의 헤더와 옵션을 올바른 순서로 적절히 처리하는지를 검증한다.

이와 관련된 RFP는 다음과 같다.

-RFC-2460-4.1 Extension Header Order

4.2 Options

-RFC-2463-3.4 Parameter Problem Message

4.1 Echo Request Message

IPv6 노드는 확장헤더가 한 패킷 내에서 몇 개가 있든, 어떤 순서로 있든 처리하려 노력해야 한다. 한 가지 예외로 'hop-by-hop option header'는 IPv6 기본헤더 바로 뒤에 위치해야 한다. 확장헤더내의 일련의 옵션들은 헤더에 나타난 순서대로 처리되어져야 한다. 각 옵션들은 'option type' 필드(8bits) 값에 의해 식별되며 그렇지 않은 경우 최소한 최상위 두 개의 비트에 의해 구별되어 처리되도록 구현해야 한다. 이러한 'option type' 필드를 만나는 경우에 대하여 처리를 실행함으로써 패킷내의 확장헤더의 순서대로 처리 되는지 검증할 수 있다. (option :135 는 이진수로 변환시 상위 두 비트가 10이며, 이 패킷을 제거하고 'ICMP

parameter problem' 메시지를 전송한다.

Packet 형식
IPv6 Header Payload Length : 37 Next Header : 60
Destination Option Header Next Header : 60 Destination Option Header Option :135 Next Header : 44
Fragment Header Next Header : 58 Fragment Offset : 0 More Fragment flag : 1
ICMPv6 Echo Request

(그림 14) 확장헤더 처리 순서 검증 패킷

(5) ICMP 에러 메시지 처리 검증 패킷

목표 : (그림 15)는 'option type' 필드 최상위 두 비트의 값에 따른 올바른 처리 여부를 검증한다.

이와 관련된 RFP는 다음과 같다.

-RFC-2460-4.2 Options

-RFC-2463-3. ICMPv6 Error Messages

-RFC-2463-4. ICMPv6 Informational Messages

아래 〈표7〉는 ICMPv6 'option type' 필드 최상위 두 비트의 값을 나타낸다.

(표 7) ICMPv6 'option type' 필드

00	이 옵션은 건너뛰고 헤더 처리를 계속한다.
01	이 패킷은 제거한다.
10	패킷을 제거하고, ICMP Parameter Problem전송
11	패킷을 제거하고, 목적지 주소가 멀티캐스트 주소가 아닌 경우에만 ICMP Parameter Problem전송

Packet 형식 1	Packet 형식 2
IPv6 Header Next Header : 60	IPv6 Header Next Header : 60
Destination Option Header Next Header : 58 Header Ext Len(확장헤더 길이) : 0 Option :135(10)	Destination Option Header Next Header : 58 Header Ext Len(확장헤더 길이) : 0 Option :199(11)
ICMPv6 Echo Request	ICMPv6 Echo Request

(그림 15) ICMP 에러 메시지 처리 검증 패킷

각 옵션들은 'option type' 필드(8bits)의 값에 따라 식별되어야 한다. 그러나 8비트를 통해 노드가 식별하지 못하는 경우 최상위 두 비트의 값에 따라 IPv6 노드는 각각의 행동을 선택하여 조치하여야만 한다.

(6) 여러개의 옵션처리 검증 패킷

목표 : (그림 16)은 하나의 헤더안에 여러개의 옵션이 나타날 경우 순서대로 처리되는지를 검증한다.

이와 관련된 RFP는 다음과 같다.

-RFC-2460-4.2 Options

-RFC-2463-3.4 Parameter Problem Message

Packet 형식
IPv6 Header Next Header : 60
Destination Option Header Next Header : 58 Header Ext Len(확장헤더 길이) : 3 Option 7:(00) Option 71:(01) Option 135:(10) Option : 199(11)
ICMPv6 Echo Request

(그림 16) 여러개의 옵션처리 검증 패킷

(7) 단편화된 패킷 처리 검증

목표 : (그림 17)은 노드가 송신 주소와 발신주소를 이용하여 패킷이 단편화와 단편 ID를 구별할 수 있는 지를 검증한다.

이와 관련된 RFP는 다음과 같다.

-RFC-2460-4.5 Fragment Heade

-RFC-2463-3. ICMPv6 Error Messages

-RFC-2463-4. ICMPv6 Informational Messages

Packet 형식
IPv6 Header Next Header : 44 Source address : host A
Fragment Header Next Header : 58 Fragment offset : 0 More Fragment flag : 1
ICMPv6 Echo Request

(그림 17) 단편화된 패킷 처리 검증

'fragment header'는 송신자가 'path-MTU'보다 큰 패킷을 목적지로 보내는데 사용된다. 원래의 패킷은 송신지와 목적지 주소가 같고, 단편화 ID를 가진 단편화된 패킷들로부터만 재 조립되어질 수 있다.

(8) 단편화된 패킷을 모두 수신하지 못한 경우 처리 검증

목표 : (그림 18)은 단편화된 패킷의 일부가 일정 시간동안 도착하지 않을 경우에 적절한 처리를 검증한다.

이와 관련된 RFP는 다음과 같다.

-RFC-2460-4.5 Fragment Header

-RFC-2463-3. ICMPv6 Error Messages

-RFC-2463-4. ICMPv6 Informational Messages

만약 단편화된 패킷을 모두 수신하지 못한경우, 일정시간이 경과 후에는 재조립과정이 일어나지 않으며 패킷은 폐기되어야한다. 이 경우 ICMPv6 시간초과(Time exceeded) 에러 메시지는 'Code 1'로 설정하여 응답한다.

Packet 형식
IPv6 Header Next Header : 44
Fragment Header Next Header : 58 Fragment offset : 0 More Fragment flag : 1
ICMPv6 Echo Request

(그림 18) 단편화된 패킷을 모두 수신하지 못한 경우 처리 검증

인터넷을 통한 불법 사용자 및 해커의 침입으로 인한 정보의 손실, 파괴, 변조 등의 피해 발생 외부로부터 내부망을 보호하기 위한 네트웍 구성요소 중의 하나로써 시험 절차를 기준으로 보안 요구 사항을 반영하여 방화벽 (firewall) 에 대한 보안 정책 수립, 구축 및 응용 테스트를 통한 침입 차단 기능 제공할 수 있다. 이와 같은 비정상 패킷에 대한 반응을 시험함으로써 두가지 프로토콜 (IPv4,IPv6)이 상호 운영되는 환경에서 IPv6 헤더에 대한 반응을 예측할 수 있다.

인터넷의 확장으로 부가되는 각종 서비스로 인한 기업의 전산 의존도가 증가하고 있다. 시험 시나리오는 내부 자원의 안전한 외부 제공 및 외부 불법침입의 차단 기능에 대한 규칙으로 활용가능하다.

본 연구결과를 활용하면 외부의 불법 침입으로부터 내부의 정보자산을 보호하며 외부로부터 유해정보 유입을 차단하기 위한 정책과 이를 지원하는 내부 네트워크를 보호하기 위해 외부에서의 불

법적인 트래픽 유입을 막고, 허가되고 인증된 트래픽만을 허용하려는 적극적인 방어 대책을 구축할 수 있다.

Ⅵ. 결 론

기존의 IPv4는 인터넷 접속자 증가로 인한 주소 고갈, 보안 강화의 필요성, 비실시간성적인 특성으로 인해 실시간성을 요하는 데이터에 대한 서비스의 질(QoS) 문제 등의 한계로 인해 새로운 인터넷 프로토콜의 필요성의 대두되었다. 이로 인해 새로운 인터넷 프로토콜인 IPv6이 등장하게 되었다. IPv6이 IPv4를 대치하는 프로토콜이지만 헤더 구조상의 차이로 인해 자연스럽게 통신이 이루어지지 않는다. 따라서 IPv6과 IPv4는 자연스럽게 호환되지 않는다. 현재 대부분의 호스트가 IPv4를 사용하고 있으며 IPv6와 IPv4는 상당기간 상호 공존할 것이며 점진적인 전환이 이루어질 것이다. 이러한 전환의 과정 중에는 IPv4/IPv6 듀얼 스택의 망, 혹은 IPv6전용, 혹은 때에 따라서는 IPv4가 남아있는 상황도 가정할 수 있다.

본 논문에서는 IPv4/IPv6이 혼재하는 상황을 고려하여 IP 프로토콜의 변환기를 설계하였다. IP의 헤더에 대하여 변환을 시도함으로써 호스트의 프로토콜 스택에 대한 수정을 고려하지 않아도 된다. IP 헤더 변환기의 설계는 RFC 표준 문서의 사양을 그대로 적용하였다. 본 논문에서는 실제 망에서 두 개의 인터넷 프로토콜을 적용하기 이전에 프로토콜 변환 혹은 생성시 발생할 수 있는 비정상적인 IP 헤더패킷을 생성하게 설계하였다. 생성된 비정상 패킷에 대한 반응을 고찰하여 실제 비정상 패킷의 유입에 대한 반응을 예측할 수 있게 하였다. 사전에 정의된 패킷은 시험 환경별로 적합성에 따라 구성이 가능하다는 장점을 가진다. 패킷에 대한 동작을 사전에 살펴봄으로써 각각의 패킷에 대한 결과를 추정할 수 있으며, 새로운 인터넷 환경에 대한 적응성 및 효과적인 전환 전략을 네트워크 구성 시에 반영 할 수 있다.

참 고 문 헌

[1] 김용진 외3, 차세대 인터넷 프로토콜 IPv6, 다성출판사, 2002.

[2] V. Fuller, BARRNet, T. Li, J. Yu, MERIT, K. Varadhan, Classless Inter-Domain Routing (CIDR) : an Address Assignment and Aggregation Strategy, RFC1519, September 1993.

[3] S. Deering, R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, RFC2460, December 1998.

[4] R. Gilligan, E. Nordmark, Transition Mechanisms for IPv6 Hosts and Routers, RFC1933, April 1996.

[5] J. Postel, Internet Control Message Pprotocol, RFC792, September 1981.

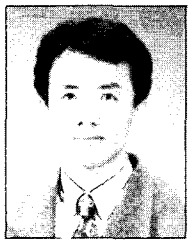
[6] A. Conta, S. Deering, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification, RFC2463, December 1998.

장 성 만

2000년 한남대학교 수학과 졸업(이학사)

2001년~현재 한남대학교 대학원 컴퓨터공학과 석사과정

관심분야 : 정보보호, 정보 시스템 보안 및 위험 분석



길 민 욱

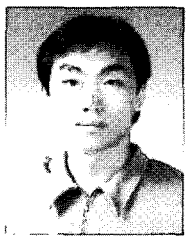
1989년 한남대학교 전자계산공학과 공학사

1991년 한남대학교 전자계산공학과 공학석사

2000년 한남대학교 전자계산공학과 공학박사

1997년~현재 문경대학 인터넷정보계열 조교수

관심분야 : 정보보호, 보안시스템, 인공지능, 음성인식, 멀티미디어, 유전자 알고리즘



이 극

1983 경북대학교 전자공학과(전산모듈) 공학사

1986년 서울대학교 컴퓨터공학과 공학석사

1993년 서울대학교 컴퓨터공학과 공학박사

1988년~현재 한남대학교 정보통신멀티미디어학부 컴퓨터공학전공 교수

2001년~현재 한남대학교 부설 정보보호응용기술연구소 소장

관심분야 : 정보보호, 보안시스템, 인공지능, 멀티미디어, 생체인식