H.323을 지원하는 Application Proxy Server의 설계

오영선 · 이현태

목원대학교

Design of Application Proxy Server for H.323 Protocol

Young-Seon Oh · Hyeun-Tae Lee

Mokwon University

E-mail : htlee@mokwon.ac.kr

요 약

본 논문에서는 응용 계층에서 수행되는 Proxy Server방식을 이용하여 Firewall이 있는 환경에서 H.323기반 서비스를 지원할 수 있는 기술적 방안을 연구하였다. H.323 프로토콜 서비스 시나리오와 기존의 Firewall방식을 연구하여 H.323 응용서비스 제공하기 위해서 어떠한 문제를 해결해야할 것인지를 연구하고 기존에 설치된 많은 Firewall 환경에 적용 가능한 H.323 응용서비스를 위한 Application Proxy Server 구조를 제안하였다.

ABSTRACT

This paper provides the solution for supporting H.323 services through firewalls with proxy server. The paper provides a survey of H.323 protocol, and framework for firewalls. The paper also discusses the issues of H.323 and proxy server - why H.323 is hard for firewalls, and service scenarios with H.323 terminals and proxy server. Finally, the paper propose an application proxy server architecture for supporting H.323 protocol.

Ⅰ. 서 론

인터넷전화 혹은 IP(Internet Protocol) telephony, Voice over IP(VoIP)란 말 그대로 인터넷을 이용하여 전화 혹은 음성서비스를 제공하는 기술이다. 인터넷 트래픽의 급격한 증가와 인터넷서비스 보급의 대중화로 인해 인터넷을 통해 음성전화 서비스를 제공하는 인터넷전화 기술이 전세계 통신사업자들의 주요 관심사가 되고 있다.

H.323은 인터넷전화와 관련된 가장 많이 사용하는 프로토콜이다. 최근 H.323기반의 응용이 많이 개발되어 사용되고 있는데 Intranet의 보안을 위하여 Firewall이 설치되어있는 환경에서는 서비스가 제공되기 어렵다. H.323은 복잡하고, 동적인 포트들을 사용하고, 복수의 UDP stream들을 포함하고 있어 Firewall이 있는 환경에 H.323응용을 지원하기 위해서는 기존의 Firewall의 보완이 필요하다.

본 논문에서는 응용 계층에서 수행되는 Proxy Server방식을 이용하여 Firewall이 있는 환경에서 H.323을 지원할 수 있는 기술적 방안을 연구하였고, 그 적용방법을 제안하고 있다.

본 논문에서는 기존의 Firewall방식에서 H.323 응용서비스 제공하기 위해 어떠한 문제를 해결해야할 것인지를 연구하였고 Firewall 구성방식에서 보안기능 제공 특성이 우수한 Application level Firewall방식을 이용하여 H.323 응용서비스 제공을 위한 서비스 제공 방안을 제안하였다.

마지막으로, Proxy Server를 이용한 가능한 구성방식을 연구하여 장단점을 비교 연구한 다음 기존에 설치된 많은 Firewall환경에 적용 가능한 Proxy Server 시스템을 설계하였다.

Ⅱ. Firewall환경에서의 H.323 서비스 제공의 문제점 분석

1. H.323 신호 프로토콜

H.323 호처리 기능은 RAS 프로토콜에 의한 게이트키퍼(Gatekeeper)와의 등록절차, H.225.0에 의한 Call Signaling절차, H.245를 통한 미디어 채널의 능력정보의 교환 및 협상 등의 세가지의 기본 절차 기능으로 나누어진다[1].

게이트키퍼는 도메인 구성에서 선택사양이지만

사용할 것을 권고하고 있다. 게이트키퍼가 있는 경우 호 설정에서 호출 요구 단말이 종단 단말과 직접 통화하는 직접 호 신호 제어(direct call signaling routing)의 선택이 가능하다. 이때 게이트키퍼는 주소변환기능, 연결에 필요한 대역폭 할당 기능을 수행한다[1][2].

신호모델과 관계없이 H.245 신호 절차는 단말간에 직접 수행된다. 가능한 H.323 신호모델은 조합의 수가 매우 많다. 그러나 크게는 직접 경로 모델과 게이트 키퍼 경유 경로도 기본 구성으로 설명할 수 있다.

H.323의 신호모델은 다음과 같은 5단계로 구성된다.

· 단계 A : Call Setup (호 설정) - 호 제어 절차에 따라 H.225.0 에 정의된 호 제어 메시지들을 이용하여 호 설정이 수행된다. 대역폭 예약요청은 가능한 초기 단계에서 이루어져야 한다[3].

· 단계 B : 초기통신과 능력교환 - 일단 양측이 호 설정 메시지들을 교환하고 나면, 단말들은 H.245 제어채널을 설립해야 한다. H.245 제어채널 상에서 능력교환과 매체 채널 개방을 위한 H.245 절차들이 수행된다. Connect가 도착하지 않거나 Release Complete을 수신하면 H.245 제어채널을 폐쇄되어야 한다[4].

· 단계 C : 오디오와 비디오 통신 확립 - 능력교환과 주종 정에 이어서 다양한 정보 스트림들을 위한 논리채널 개방을 위한 H.245 절차들이 수행되어야 한다. H.245 논리채널들을 통해 오디오와 비디오 스트림은 Unreliable 프로토콜을 데이터 스트림의 경우는 Reliable 프로토콜을 이용한다.

· 단계 D : 호 서비스 - 대역폭 변경, 단말의 상태 결정

· 단계 E : Call termination(호 종료) - 어떤 단말이라도 다음 절차에 따라 호를 종료할 수 있다

① 완전한 화상이 끝에서 비디오의 전송을 중단하고, 비디오를 위한 모든 논리 채널들을 폐쇄한다.

② 데이터의 전송을 중단하고 데이터를 위한 모든 채널들을 폐쇄한다.

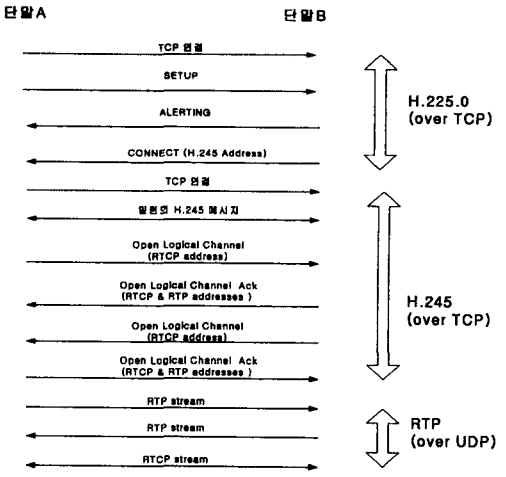
③ 오디오의 전송을 중단하고 오디오를 위한 모든 논리채널들 폐쇄한다

④ H.245 endSessionCommand 메시지를 H.245 제어채널로 전송하고 H.245메시지 전송을 중단해야한다.

⑤ H.245 endSessionCommand 메시지를 기다리다가 수신하면 H.245 제어채널을 폐쇄해야한다.

⑥ 호 signalling 채널이 개방되어 있으면 메시지가 Release Complete 전송되고 채널이 폐쇄되어야 한다.

<그림-1>은 게이트키퍼를 고려하지 않을 경우 단말간에 신호절차를 개략적으로 보여주고 있다.



<그림-1> H.323 기본 신호 절차

단말 A가 단말 B에게 전화를 하는 경우의 두 개의 H.323 종단점들의 연결을 설정하기 위해 사용하는 단계, H.323 종단점의 동작을 제어하고 있는 end-to-end의 제어 메시지들 교환하기 위해 사용하는 단계, 실시간 오디오/비디오의 end-to-end 전달 서비스를 제공하는 단계이다. 단말 A가 H.323을 위한 알려진 port 1720번으로 TCP연결을 한 후에 이 연결을 통하여 H.225.0 프로토콜의 SETUP 메시지를 전송한다[3].

단말 B는 ALETERING 메시지를 보낸 후 연결수락을 나타내는 CONNECT 메시지를 보내면서 H.245연결에 사용할 port(1024보다 큰 번호로 동적으로 할당)번호를 전달한다. H.245 절차를 위한 TCP연결이 설정되면 H.245 프로토콜 기능은 사용할 코덱 등 상호통신에 필요한 호 파라메터 협상을 수행한다. 또한 H.245는 채널통신에 사용할 UDP연결을 결정하게 된다. 사용할 음성(그리고 비디오) 코덱과 관련 파라메터의 협상이 이루어지면 H.245 세션은 Open Logical Channel Segment를 수행하여 특정미디어를 위한 발신측 RTCP주소와 port 번호를 전송하고 수신측에서 RTP와 RTCP 주소 port번호를 응답하다. H.323에서는 각 Logical Channel이 단방향이므로 양방향통신을 위해서는 각각의 방향으로 설정되어야 하므로 2개의 Logical Channel이 설정하게 된다. 결과적으로 보면 음성통화를 위해서 H.323에서 필요한 port는 TCP port 2개와 UDP port 2개가 기본적으로 필요하게 된다. 논리채널이 개방될 것임을 표시하는 Open Logical Channel 메시지(RCP Address)를 전송한다. 이 논리채널을 허락하고자 하는 상대는 Open Logical Channel Ack(RTCP & RTP Address)를 이용해 논리채널의 개방에 동의함을 알린다. UDP상에서 RTP Stream을 송수신한다[5].

2. Firewall 환경

Firewall 시스템의 기본 목표는 네트워크 사용자에게 투명성을 보장하지 않아 사용자에게 약간의 제약을 주더라도 위험지대를 줄이려는 적극적인 보안 대책을 제공하려는데 있고, 다른 사용자로부터 네트워크를 보호하는 것으로 중요한 데이터를 정당하지 않은 사용자가 접근하는 것을 막고 정당한 사용자가 네트워크 자원을 방해없이 접근하도록 하는 것이다[6].

Firewall은 모든 트래픽을 감시하여 완전히 안전한 트래픽만을 전달한다. Firewall을 구현하는 기술에는 Packet Filtering, Application Level Proxy, Circuit -Level Proxy 등이 있다.

Packet Filtering은 가장 간단한 형태의 Firewall의 기능으로 IP헤더와 상위 프로토콜(TCP, UDP, ICMP 등) 헤더의 정보(Source/Destination IP Address, Port Number)만을 이용해 미리 설정된 엑세스 제어 규칙에 따라 해당 패킷의 통과여부를 결정한다. Packet Filtering은 최소한의 엑세스 규칙만을 적용시켜 처리하기 때문에 가장 빠른 속도를 낼 수 있고 구현이 간단하며 사용자에겐 투명성이 보장된다는 장점이 있다. 그러나 제한된 정보를 바탕으로 하는 간단한 엑세스 규칙에 따라 접속허가 기능만을 제공하기 때문에 일단 Packet Filtering을 통과한 위험한 서비스에 대해서는 내부와 외부망 사이에 접속이 이루어져 내부 자원을 효과적으로 보호해야 하는 Firewall의 가장 기본적인 기능면에서는 많은 취약점을 가진다.

Application Level Proxy[7]는 특정 응용 서비스에 대해 내부망과 외부망을 연결시켜 주는 중간 매개자 역할을 하는 것으로 클라이언트/서버 간의 직접적인 접속 대신 클라이언트를 대신해서 서버와 접속하여 클라이언트/서버 사이의 통신을 중개해 준다. Application Level Proxy와 다른 종류의 Firewall 간의 가장 큰 차이는 Application Proxy는 각각의 응용 프로토콜을 Application Proxy로 구현한다는 것이다. 따라서 응용프로토콜에 대한 완전한 이해를 바탕으로 한 만큼 보다 정교한 엑세스제어가 가능하고 그에 따른 상세한 로깅이 가능하다. 또한 Application Proxy는 외부에 알려진 것은 단지 Proxy 서버뿐이므로 내부망의 시스템 구성(IP주소들)을 외부로부터 완전히 숨길 수 있다. 그러나 특정 응용 서비스마다 Proxy 서버가 필요하므로 다양한 인터넷 서비스를 제공하기 위해서는 모든 서비스에 대한 Proxy를 구현해야 한다. 따라서, 새로운 인터넷 서비스가 생길 경우는 상당한 시간이 지나야만 해당 Proxy가 제공될 수 있다. 또한 대부분의 Application Gateway Firewall들은 사용자 인증 기능을 제공하는 반면 이 경우 매 응용마다 인증이 일어나야 하므로 사용자가 불편을 느낄 수 있으며 대부분의 경우 클라이언트 소프트웨어나 사용법에 변경이 요구되며 클라이언트와 서버사이의 모든 트래픽을 분석하고 재전송해 주어야 하므로 다른 기술에 비해 성능은 훨씬 떨어진다.

Circuit Level Proxy는 Application Proxy와 유사한 기능을 제공하지만 Application 계층이 아니라 Session 계층에서 동작한다. 즉, Circuit Level Proxy는 Session 단위로 클라이언트/서버 간의 가상의 circuit을 형성하여 데이터를 전송해 준다. 즉, Application Proxy는 개별응용마다 개발되어야 하는 전용 Proxy이지만 Circuit Proxy는 응용서비스와 무관한 세션계층에서 동작하므로 대부분의 프로토콜을 자동으로 지원해주는 일반적인 Proxy이다. 반면 응용 프로토콜을 해석하지 않기 때문에 특정 응용기반의 엑세스 제어 및 로깅 기능은 제공하지는 못한다.

3. Firewall환경에서 H.323 서비스의 문제점

H.323이 Firewall을 통과하기 어려운 이유는 H.323은 복잡하고 동적인 포트들을 사용하고 복수의 UDP stream들을 포함하기 때문이다.

앞절에서 H.323 신호프로토콜 동작을 분석해보면 H.323의 호(call)는 동시에 서로 다른 다수의 연결로 이루어지고, 연결 중 적어도 2개의 TCP 연결이 필요하며 음성회의 경우 최고 4개까지의 UDP 연결이 필요하다. 이외의 연결들은 모두 동적인 포트를 사용하고 있다. 그리고, 대부분의 제어정보는 ASN.1로 코드화된다. 따라서 주소정보를 갖기 위해서는 코드화를 이해해야 하므로 매우 복잡하고 어렵다. 따라서 Firewall이 있는 환경에 H.323응용을 지원하기 위해서는 기존의 Firewall의 보완이 필요하다.

III. Firewall 환경에서의 H.323 서비스 수용방안

앞장에서 Firewall 환경에서의 H.323지원의 어려운 점을 분석하였다. 본 논문에서는 Firewall 보안특성을 최대한 지원하고 H.323 서비스를 지원하는 방안으로 Application Proxy로 구분한 H.323 Proxy Server방식을 이용한 H.323 서비스 수용방안을 연구한다.

기능적으로 보면 Proxy Server는 Client측 기능과 Server측 기능을 모두 구현해야 한다. 그러나 H.323의 모든 기능을 구현할 필요는 없다. 다만 시나리오 분석 결과를 통하여 메시지를 분석하고 재구성하여 전달하거나 채널연결을 위한 동적 포트와 연결관계를 관리할 수 있는 최소한의 기능의 구현이 필요하다.

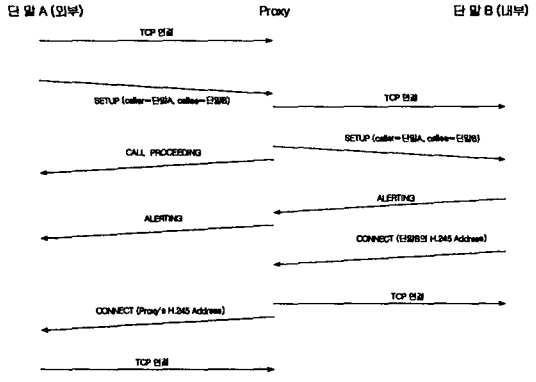
H.323 메시지 상호동작에서 Proxy가 삽입된 경우의 서비스 시나리오를 설계하였고 H.323을 지원하는 Proxy Server가 삽입된 환경에서 Proxy서비스 제공을 위한 관련 H.323 PDU와 주요동작을 설계하였다.

1. H.323 Proxy 서비스 시나리오 설계

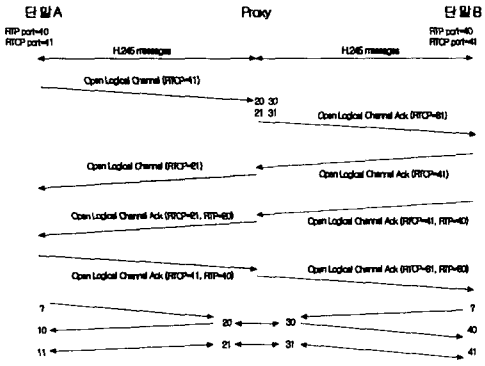
Proxy Server가 개입된 구성에서 H.225.0 메시지

전달 절차는 <그림-2>와 같다.

단말 A에서 단말 B로 통신하고자 할 때 Proxy Server가 개입하여 대신 연결을 중개한다. 단말 A에서 호 설정 메시지가 오면 Proxy Server가 대신 단말 B로 보내고 응답대기메시지나 H.245주소(연결 port 번호)등도 Proxy Server가 단말 A에게 전해준다.



<그림-2> Proxy가 삽입될 때, H.225.0 호처리 절차



<그림-3> H.245 세션을 위한 Proxy 연결절차

<그림-3>은 H.245 세션을 위한 Proxy Server를 통한 변경절차를 나타낸다.

<그림-3>은 설명을 위해서 임의의 port번호를 표시하였으나 실제로는 1024이상의 port번호를 사용한다. Proxy 서비스 제공을 위한 각 H.323 관련 PDU 및 세부 동작 절차는 다음 제2절에서 기술한다.

2. H.323 Proxy Server구현을 위한 PDU 및 동작

H.323서비스 수용방안을 연구하기 위하여 관련되는 PDU를 분석하였다. <표-1>은 Proxy 서비스를 위한 H.225.0의 주요 PDU 주요동작을 나타낸다[8].

PDU	시나리오	주 요 동 작
SETUP	이미 알려진 call signaling channel (1720TCP port)을 통하여 새로운 호출요구 전달	1.PROCEEDING 메시지를 발신측으로 보냄. 2.호가 외부호인지, 내부호인지를 판단. 3.SETUP메시지의UUIE.destCallSignaling Address가 존재하고 유효한지 검사하고, 유효하면 6단계를 수행. 4.만일, 존재하지 않거나 유효하지 않으면 SETUP메시지의 UUIE.calleeAliasName의 aliaslist를 읽고 DNS를 이용하여 목적지주소를 찾아낸다. 만약, 이 절차가 실패하고 내부에서 외부로의 호이고 DNS name이 있으면 Proxy Server 자신이 아닌지 확인한다. 유효한 DNS 이름을 네트워크 주소로 변환하고 user part를 가지고 있고 remote ExtensionAlias가없으면 outgoing Setup 메시지에 대하여 user part의 remoteExtensionAlias를 만든다. 만약 유효주소가 결정되면 6단계를 수행 5.호출이 외부에서 내부호이고 remote-ExtensionAlias가 있으면 이것을 DNS(혹은 Gatekeeper 등)를 이용하여 IP주소로 변환한다. 6.목적지 IP주소가 결정되면 목적지 주소로 TCP 연결을 설정한다. 7.새로 수정하여 생성한 SETUP메시지를 전달한다.
CONNET	착신측이 발신측으로 전송	1.만약 UUIE.H245Address가 존재하면 UUIE.H245Address,UUIE.destinationInfo의 정보를 새로운 값으로 갱신한다. 2. PDU를 전달한다.
Release Complete	양측에서 전달 가능	1.Release는 모든 H.245와 RTP채널을 위한 자원을 돌려준다. 2.PDU를 전달한다. 3.H.225.0 연결과 관련된 자원을 돌려주고 연결을 해제한다.
ALERTING	발신측에서 전송	1. 고친 CRV값을 전달한다.
이외의 PDU		1. 고친 CRV값을 전달한다.

<표-1> Proxy와 관련된 H.225.0 PDU

PDU	시나리오	주 요 동 작
Open Logical Channel	송신에서 채널개설을 요청함	1.만약 dataType이 오디오 혹은 비디오이면 RTCP와 RTP를 위한 포트를 할당한다. 이때 RTP포트는 짝수이고 RTCP는 하나 많은 수의 홀수를 할당한다. 2.전달할 PDU의 RTCP 주소는 Proxy 주소도 바꾼다. 3.PDU를 전달한다.
Open Logical Channel Ack	수신측에서 채널개방 요구를 허락	1.채널이 알려진 채널이면 내부와 외부 포트간의 매핑을 갱신하고, RTP와 RTCP 세션에 대하여 패킷전달(forwarding)을 작 동한다. 2.RTP와 RTCP의 주소를 Proxy 주소로 교환한 다음 3.PDU를 전달한다.
Open Logical Channel Reject	수신측에서 채널개방요구를 거절	1.알려진 채널에 대해 모든 포트 자원을 해제한다. 2.PDU를 전달한다.
Close Logical Channel	한쪽측에서 설정된 채널의 해제 요구	1.알려진 채널에 대하여 RTP와 RTCP 포트의 패킷전달을 중지한다. 2.PDU를 전달한다.
End Session	한쪽측에서 설정된 모든 채널과 호를 해제요청	1.연결된 모든 채널에 대해 RTP와 RTCP포트의 패킷 전달을 중단하고 모든 UDP 포트 자원을 해제한다. 2.PDU를 전달한다. 3.H.245 연결과 관련한 자원을 해제하고 연결을 종료한다. 4.양측으로 Release Complete 메시지를 전송하고 연결을 종료한다.
이 외 PDU		1.PDU를 전달한다.

<표-2> Proxy와 관련된 H.245 PDU

3. H.323 Proxy Server 방식의 서비스 제공을 위한 H.323 응용 고려사항

제안한 H.323 Proxy Server 방식의 서비스 제공을 위하여 일반적인 H.323 단말에서 다음 사항을 고려하여야 한다.

1) Proxy Server를 지원하려면 H.323 단말기능에서 적어도 Proxy 관련 정보를 입력할 수 있는 이용자 인터페이스를 제공해야 한다.

2) 착신 단말이 Proxy Server 뒤에 존재할 때 SETUP 메시지의 destCallSignalingAddress 혹은 destinationAddress는 원격 Proxy Server의 주소를 갖고 있고 TheremoteExtensionAlias 영역은 연결한 착신 단말의 정보(H.323 alias 혹은 E.164 주소)를 포함해야 한다(Proxy Server는 이 정보로부터 해당 착신단말의 IP 주소를 찾는다.). 따라서, H.323 단말 응용프로그램은 해당 영역을 이용자 적절하게 채울 수 있도록 하여야 한다.

3) 착신 단말이 동일한 Intranet(Proxy 안쪽)에 존재할 때 Proxy Server를 이용하여 연결할 것인지 직접 연결할 것인지를 결정할 수 있어야 한다.

IV. Proxy Server 시스템의 설계

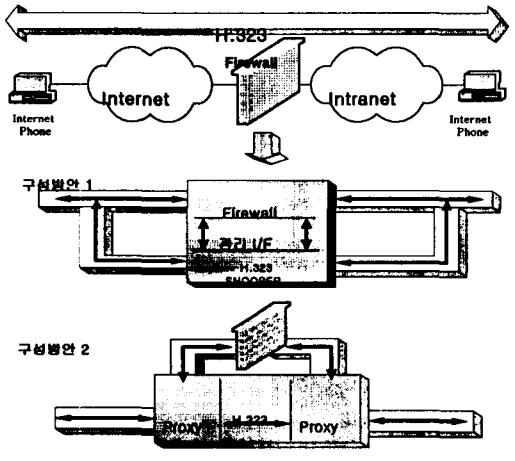
1. 시스템 구조 설계

H.323 서비스 제공을 위한 Proxy Server는 다음 2가지 기능을 수행하여야 한다.

(1) H.323과 H.245 프로토콜 기능(양쪽에 대한 상대 기능)의 수행을 다룰 수 있어야 한다.

(2) 데이터 패킷은 쌍방간에 가능한한 고속으로 전달해야 한다. 병목이 되지 않기 위해서는 고속 수행환경이 필요하고 어느 정도의 보안기능을 제공하여야 한다.

앞에서 Proxy Server 방식을 통한 H.323 서비스 제공방안을 제시하였다. 본 장에서는 앞에서 제시한 서비스 시나리오를 만족할 수 있는 Proxy Server의 구현방안을 연구하였다.



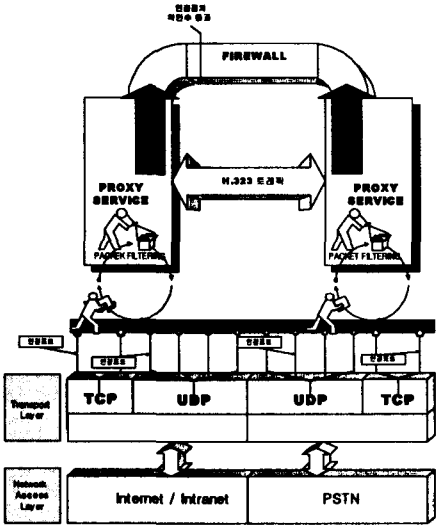
<그림-4> H.323을 지원하는 Application Proxy Server

<그림-4>는 일반적으로 Intranet의 보안을 위해 기존에 Firewall이 설치된 환경을 고려하고 이러한상황에서 Proxy Server 기능을 구현하는 방안을 제시하고 있다. 기능적으로 Firewall 기능과 H.323 Proxy 기능을 수행하는 Proxy Server로 나누어 볼 때 H.323을 지원하는 Proxy Server의 구현은 어떻게 기존의 Firewall을 H.323 트래픽이 통과할 수 있도록 구성하는가에 대한 구성방식의 선택이다.

본 논문에서는 <그림-4>에서와 같이 2가지 구성방식을 고려하였다.

구성방안 ①은 Proxy Server가 H.323 Proxy 절차를 모니터하고 동적으로 사용되는 포트를 Firewall이 통과시킬 수 있도록 관리 I/F를 통하여 제어하는 방식이다. 이 경우 Proxy Server는 Proxy 기능 즉, 메시지를 재구성하여 전달하는 등의 기능을 수행하지 않을 수도 있다. 즉, 단순하게 H.323 메시지의 모니터 기능만 가진다. 이 때, 단순 H.323 모니터 기능만을 수행할 경우는 H.323 동작을 해결하여 해당 TCP나 UDP port를 동적으로 적절하게 열어주어 동작한다. 따라서, 이 경우는 H.323 단말간의 트래픽이 투명하게 전달할 수 있어 Ⅲ. 3절에서와 같은 기존 H.323 응용 단말의 수정이 필요하지 않다. 그러나, 기존 Firewall기능과 I/F를 가지며 같은 시스템이 구현되어야 하므로 시스템에 의존적이고 구현이 어려운 문제점을 갖고 있다.

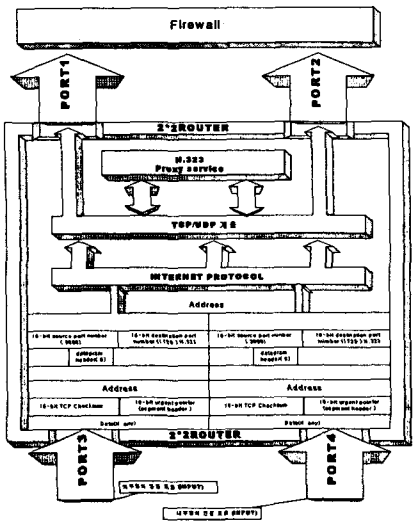
구성방안 ②는 Proxy Server가 H.323 Traffic에 대하여 Firewall을 통과하지 않고 직접 바이패스하여 전달하는 구성방안이다. 이 때는 Proxy Server가 Proxy 기능을 모두 수행하고 Channel stream의 전달에도 관여한다. 또, Proxy Server를 위한 별도의 시스템을 구성하므로 기존의 Firewall 시스템을 이용할 수 있어 Firewall 시스템에 의존적이 아닌 장점을 가진다. 그러나, 앞에서 살펴본 바와 같이 기존 H.323 단말에 다소 보안이 필요하고 H.323 stream 전달에 관여하므로 성능문제를 발생시킬 수 있다.



<그림-5> Application level Proxy Server 프로토콜 스택

<그림-5>는 구성방안 ②의 경우에 Proxy Service를 위한 프로토콜 스택을 나타낸다. Proxy Service 기

능은 H.323의 특정 포트에 대한 서비스에 대하여 Proxy 기능을 수행하고 다른 일반 트래픽은 Firewall 기능을 통하여 정상적으로 제어된다.



<그림-6> Proxy Server 시스템의 구성도

<그림-6>은 Proxy Server 시스템의 구성도이다. 설계한 Proxy Server는 4개의 Network I/F를 가진 4port 시스템으로 구성한다.

<그림-6>에서는 port 3는 외부망에서 port 4는 내부망에 접속되고 port1과 port2는 Firewall 시스템에 접속한다.

V. 결 론

본 논문에서는 H.323 프로토콜의 신호절차를 분석하고 Firewall 구성방식을 연구하여 Firewall이 설치된 환경에서의 H.323 기반 응용 서비스를 제공하기 위한 방안을 제안하였다.

Firewall의 보안 특성을 최대한 지원하고 H.323 서비스를 제공하는 방안으로 Application Proxy방식의 H.323 Proxy Server를 이용하여 H.323 서비스 수용 방안을 제안하였다. 이를 위하여 H.323 메시지 상호동작에서 Proxy가 삽입된 경우의 서비스 시나리오 설계를 하였고 Proxy 서비스 제공을 위한 관련 H.323 PDU와 주요동작을 설계하였다.

본 논문에서는 Proxy Server 구현 방안으로 Proxy Server가 H.323 Traffic에 대하여 Firewall을 통과하지 않고 직접 바이패스하여 전달하는 구성 방안을 제안하였다. 이 경우, Proxy Server가 Proxy 기능을 모두 수행하고 Channel stream의 전달에도 관여한다. 또한, Proxy Server를 위한 별도의 시스템을 구성하므로 기존의 Firewall 시스템을 이용할 수 있어 Firewall 시스템에 의존적이 아닌 장점을 가진다. 이 Proxy Server 시스템은 4개의 Network I/F를 가진 4port 시스템으로 구성하였다.

본 논문에서 제안한 4port를 갖는 Proxy Server는 H.323 트래픽 처리 관점에서 제안되었으나 앞으로 고급화/고성능화된 Layer4/5 Switch의 실현을 위한 플랫폼 시스템으로 개선될 수 있을 것이다.

참고문헌

[1] H.323 Packet Based Multimedia Communication Systems, ITU-T Recommendation, 1998.

[2] Bernard Pagurek (Carleton University), "Management of Advanced Services in H.323 Internet Protocol Telephony," IEEE infocom 2000.

[3] H.225.0, Call Signaling Protocols and Media Stream Packetization for Packet Based Multimedia Communications Systems, ITU-T Recommendation, 1998.

[4] H.245, Control Protocol for Multimedia Communication, ITU-T Recommendation, 1998.

[5] Charles Kalmanek (AT&T Labs -Research), "DOSA: An Architecture for Providing Robust IP Telephony Service," IEEE infocom 2000.

[6] D.Brent Chapman and Elizabeth Zwicky, *Building Internet Firewalls*, O'reilly Edition, 1995.

[7] RFC 1919, "Classical Versus Transparent IP Proxies,"
http://sunsite.auc.dk/RFC/rfc/rfc1919.html

[8] "H.323 and Firewalls,"
http://support.intel.com/support/videophone/trial21/h323_wpr.htm