

# VoIP 에서 Q-SIP 와 COPS-ODRA 를 통한 정책결정과 QoS 지원

조규철\*, 한기준\*\*

\*경북대학교 컴퓨터공학과

\*\*경북대학교 컴퓨터공학과

\*e-mail : k5435@netopia.knu.ac.kr

## 요 약

인터넷의 발달로 인하여 기존 전화망을 인터넷 망으로 적은 비용으로 사용하고자 VoIP 가 연구 되기 시작하였다. VoIP 의 기존전화망과의 품질의 차이에도 불구하고 대폭적인 통화비용의 절감과 다양한 이점으로 많은 연구가 되고 있다. 이에 통화품질의 향상을 위하여 QoS 에 대한 연구가 활발히 진행되고 있다. 여기서는 QoS 를 지원하는 Q-SIP 서버와 Policy Based 의 COPS 를 이용하여 Pre-COPS update message 와 Reservation COPS cache 로 정책결정과 QoS 를 지원하고자 한다.

## 1. 서론

지금 인터넷의 사용은 급격하게 대중화되고 있다. 인터넷이 아직은 일반전화기와 별개의 서비스로 일반 사용자에게 인식되고 있지만 기존 전화망과의 연동에 관심과 이용이 늘고 있다. 이를 이용하고자 VoIP(Voice over IP)가 거론되고 있다. 이는 네트워크 표준인 IP 를 이용하여 Data 뿐만 아니라 음성도 함께 전송하는 통신기술로서 Internet 과의 통합에 의한 다양한 제품 및 부가 서비스를 창출하고 있다. 지금 이에 대한 각종 표준안들이 제안되고 확정되고 있는 단계이다. 일례로서 IP telephony 서비스는 현재 network 에서 Best-effort 를 지원하나 차세대 통신망에서는 다양한 방식의 QoS 지원에 대한 연구가 되고 있다.

이 논문에서는 VoIP 의 QoS 를 지원하고자 하는 방법에 대한 제안을 할 것이다. 기존에 제시된 Q-SIP 을 이용한 call signaling 의 방법 Policy based Network 에서 COPS 를 응용하여 QoS 를 제공하는 방식을 제안하고자 한다.

## 2. 관련연구

### 2.1 VoIP 와 Call signaling protocol : SIP

VoIP 는 인터넷상에서 음성서비스를 제공하는 기술로서, 음성신호를 디지털화하고 압축한 후 IP 패킷화하여 인터넷상에서 전달케 한다. 초기에는 실

험적인 수준이었으나 현재는 데이터망의 사용에 따른 기존 전화망과의 QoS 의 차이에도 불구하고, 통화비용의 대폭적인 절감외에 기존 인프라의 사용과 관리의 편리성, 오디오/데이터 통합에 의한 다양한 응용서비스 제공 등의 효과들로 인해 그 이용이 늘어나고 있다.[1]

제어 및 신호 프로토콜로는 H.323 과 SIP 가 주로 사용되고 있으며, 여기서는 IETF 에서 제시한 SIP 프로토콜만을 다루려고 한다. SIP 은 VoIP 를 위한 호 세션 설정, 유지 및 해제 등 관리에 대한 전반적인 절차를 다루고 있다. 특히 SIP 는 H.323 과 달리 클라이언트-서버기반 프로토콜로서 호 시도자가 상대방을 호출하여 세션에 참가시키는 형태로 전개된다.

### 2.2.1 SIP 의 QoS 지원 : Q-SIP

Q-SIP 의 기본은 모든 SIP User Agent 들이 입출력을 그들의 도메인에 있는 default SIP proxy 서버를 사용하는 것이다. 즉 SIP 서버는 User Agent 사이의 message 교환 장치에 포함된다. 이때 SIP 서버에서 QoS 를 지원하고자 확장된 것이 Q-SIP 서버이다. 물론 QoS 를 지원하고 전형적인 SIP message 도 처리한다. Q-SIP 의 기본구조는 그림 1 과 같다. 송신측이나 수신측에서 SIP 서버나 User Agent 중 한 개의 요소는 분명 QoS 를 지원하는 모델로 이루어져야 한다. [2]

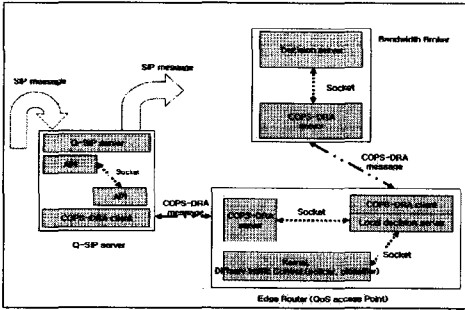


그림 1 Q-SIP 과 정책결정의 flow

2.2.2 Q-SIP 의 적용방법

Q-SIP 의 시나리오를 본다면 2 가지를 거론하고 있다. 먼저 unidirectional 예약 방식과 bidirectional 예약 방식으로 나눌 수 있다.

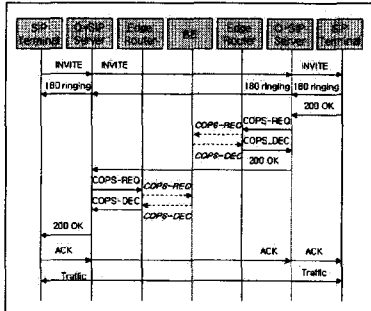


그림 2 unidirectional QoS mode (Q-SIP)

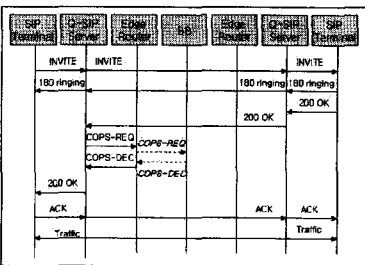


그림 3 bidirectional QoS mode(Q-SIP)

그림 2 와 같이 unidirectional 예약 방식에서 송신 측 Q-SIP 서버는 수신 측 Q-SIP 서버가 callee-to-callee 흐름을 위해 자원을 예약해 두는 동안 caller-to-callee 의 트래픽 흐름을 위해 예약을 체결한다. 이 두가지 예약은 양방향 흐름이 요구 된다. 그림 3 에서의 bidirectional 예약 방식에서는 SIP 서버가 stateful Q-SIP 를 지원하여 송신 측 Q-SIP 서버에 기록된 state 를 이용하여 양방향모두의 자원 예약을 수행한다.

다음 절에서는 Q-SIP 서버가 정책을 설정하기 위

하여 Policy network 에서 Policy information 을 전달 하기 위한 COPS 프로토콜을 설명한다.

2.3.1 COPS 의 정의

COPS(Common Open Policy Service)는 라우터들과 스위치들이 Network Policy Server 로부터 QoS Policy information 을 얻을 수 있는 프로토콜이다. 서비스 우선 순위에 따른 트래픽 자원 배정 등의 전반적인 서비스 품질(QoS) 사항으로서 PDP(Policy decision point)와 PEP(Policy enforcement point)간의 네트워크 정책 정보 교환을 위한 표준 프로토콜이다. PDP 는 음성, 대량 데이터, 비디오, 화상회의 등 트래픽 최우선 순위 지정에 대한 정책 사항을 다루는 네트워크 서버이고, PEP 는 네트워크 트래픽 이동에 대한 정책 선택 수행을 담당하는 라우터 혹은 제 3 계층 스위치들이다.[3]

IP QoS 환경하에서 Policy control 을 지원하기 위해서 DiffServ 와 IntServ 를 응용해 PDP, PEP 에서 Policy information 을 전송해서 VoIP 에서 QoS 를 제공하고자 한다. 여기서 Network Device 와 Policy Server 와의 연결은 COPS 프로토콜이 기본으로 사용 된다. QoS 를 제공하는 주체에 따라서

Outsourcing model 과 Provisioning model 로서 구별 된다. Outsourcing model 은 실시간으로 발생하는 요구에 매년 정책 결정을 PDP 에서 한다. 이를 PEP 에 통보한다. Provisioning model 은 사전에 decision message 를 PDP 로 보내어 PEP 에서 자원 관리 매커니즘을 설정한다. 즉 사전에 PDP 에 정해진 정책에 의하여 PEP 에서 결정한다. DiffServ QoS 에서는 신호 방식 매커니즘과 admission control framework 의 정의가 필요로 한다. DiffServ 의 admission control server 는 일반적으로 Bandwidth Broker(BB) 가 담당한다. 모든 Edge 라우터들은 중앙 BB 와 자원 할당 요구를 교환한다.[4]

2.3.2 COPS 프로토콜의 응용

이 절에서는 dynamic DiffServ QoS 를 지원하기 위한 COPS 의 확장을 이야기 하려고 한다. Client 의 형태에 따라 Outsourcing model 을 적용한

Outsourcing DiffServ Resource Allocation(COPS-ODRA)와 Provisioning model 을 적용한 COPS-PR 이 있다. 그림 4 는 COPS-ODRA 모델 적용 예이다. 우선 COPS-ODRA 는 항상 PEP 는 input 에서

output 으로 가기까지 QoS 설정을 PDP/BB 에게 문의한다. 확장성을 위하여 개별적인 flow state 를 PDP/BB 에 저장하지 않는다. 포괄적 state 가 PDP/BB 에 저장된다. Edge 라우터는 PEP 로서 작동하고 PDP/BB 와 상호 작용한다.[5]

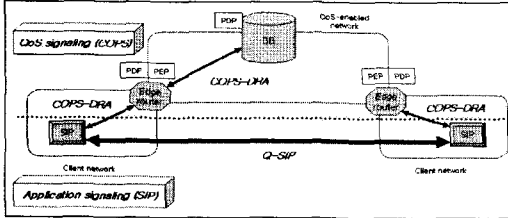


그림 4 COPS-ODRA 의 flow

본논문에서는 호설정시 state 정보를 집합체로서 다루어 확장성이 좋고 request 의 수가 많아져도 처리가 효율적인 COPS-ODRA 를 기준으로 전개를 하려고 한다.

### 3. Q-SIP COPS 와 Q-SIP 을 이용한 QoS 지원 방안

#### 3.1 기존 방식의 문제점

COPS-ODRA 의 방식에서 Q-SIP 프로토콜의 흐름과 COPS 프로토콜의 흐름(그림 1)에서 Q-SIP 서버에 SIP message 가 오면 Q-SIP 서버는 COPS 프로토콜의 절차에 의해서 Edge 라우터와 BB 를 거쳐서 자원예약이 이루어진다. 자원예약이 될 때까지 SIP message 는 Q-SIP 서버에서 대기한다. 반드시 QoS 를 지원하기 위해서 Edge 라우터와 BB 와 정책설정이 이루어져야 한다. 이 방법에서 SIP message 의 대기나 COPS 프로토콜의 처리시간이 소요되는 문제가 있다.

다음으로 unidirectional 예약 방식의 경우 COPS 프로토콜의 흐름에서 항상 BB 를 이용하여 정책결정이 이루어지므로 SIP message 의 딜레이와 COPS 프로토콜의 처리시간이 소요되며 또한 bidirectional 예약 방식에서도 송신측 Q-SIP 서버에 stateful 방식에서 저장된 자료를 이용하여 COPS 의 설정이 이루어집니다. 이때 상태가 저장된 양이 적거나와 최근 상태의 자원상태를 반영하지는 못하고 적절한 QoS 의 설정이 어렵다.

#### 3.2 COPS ODRA 와 Q-SIP 을 응용한 방안

위의 여러 가지 단점으로 인하여 3 가지 시나리오를 제시하고자 한다. 첫번째는 Q-SIP 서버에서 COPS 프로토콜을 이용한 정책 결정에서

Reservation COPS cache 를 생성하여 COPS 프로토콜을 전형적인 정책결정단계를 거치지 않고 정책결정 Cache 를 통한 정책결정 방법이다. 둘째로는 unidirectional 예약 방식에서 Edge 라우터에 BB 와의 COPS 설정을 호설정 때마다 정하여 지는 것이 아니라 Pre-COPS update message 를 두어서 송수신측의 Q-SIP 서버나 Edge 라우터의 Cache 를 갱신해 주는 역할을 하는 것이다. 물론 bidirectional 예약 방식에서는 송신측의 COPS 설정시 stateful 의 상태를 이용하므로 이때 cache 를 이용한다면 좀더 적절한 QoS 설정이 가능하다. 물론 이 message 는 호설정과는 동기적으로 발생하지 않고 적절한 주기로 발생하는 것이다. 마지막으로 이 두 방식의 조합하여 VoIP 에서 최대한의 효율을 높인 QoS 설정을 제시한다.

#### 3.3.1 Reservation COPS Cache 의 적용

SIP 에서 QoS 를 지원을 위하여 COPS-ODRA 를 사용할 경우 그림 1 과 같은 구조로 되어 있다. 우선 초기에 호설정 매커니즘에서는 Edge 라우터와 BB 에서 COPS 프로토콜을 이용하여 QoS 가 협상이 된다. 이때 Q-SIP 서버쪽에 Reservation COPS Cache 를 설정을 하면 호설정시 COPS 를 이용한 기존의 정책결정방법이 아닌 cache 를 통해 정책결정을 하고 QoS 설정이 이루어진다.

그림 5 와 같이 Edge 라우터와 BB 를 통하지 않고 정책결정이 된다. 만약 cache 에 존재하지 않을 경우 일반적인 Q-SIP 프로토콜의 호설정과 같은 순서를 가진다. 여기에서 Cache 를 이용하는 시간이 소모가 되나 Q-SIP 서버에 있거나 같은 네트워크에 존재하므로 Cache 를 이용한 정책결정 시간은 Edge 라우터나 BB 와의 정책결정을 위한 COPS 프로토콜 이용보다는 작은 시간이 소요된다. 이때 Reservation COPS cache DB 에는 COPS 프로토콜을 사용한 정책결정과 같은 정보들을 저장한다.

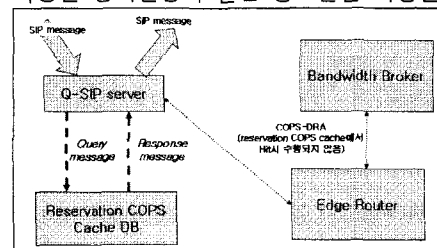


그림 5 Reservation COPS Cache DB 설정

QoS 의 설정을 위한 COPS 프로토콜의 이용 없이 Q-SIP server 에서 Cache 를 통한 Q-SIP 헤더를 포함하여 QoS 의 설정이 이루어진다.[5]

3.2.2 Pre-COPS update message 의 적용

Q-SIP 의 두 가지 예약 방식(unidirectional, bidirectional)에서 Edge 라우터는 호설정시에 항상 BB 와의 COPS 프로토콜을 이용하여 이루어진다. 이 경우 Edge 라우터가 호설정시에만 COPS 의 설정이 이루어지는 것이 아니라 Edge 라우터에서 Pre-COPS update message 를 생성하여 SIP server 나 BB 에 주기적으로 임시적으로 COPS 설정을 하여 Cache 에 포함된 자료들을 갱신하는 것이다. 즉 한 번 설정된 QoS 의 설정은 갱신을 유지하면서 계속 유지하는 것이다. 이는 호설정과는 비동기적으로 발생한다.

3.2.3 Reservation COPS Cache 와 Pre-COPS update message 의 응용

위에 제시한 두 가지 방법을 혼용하여 사용할 경우를 제시하여 본다. 송신측 Q-SIP 서버에서 Cache 를 내장하거나 Cache 서버를 두어 이를 활용하고자 한다. 우선 Cache 에서는 이전에 설정된 호설정의 state 를 저장한다. 발생빈도를 이용한 cache 알고리즘으로 생성을 하고 이 갱신은 호설정시와 Pre-COPS update message 를 이용하여 갱신한다.

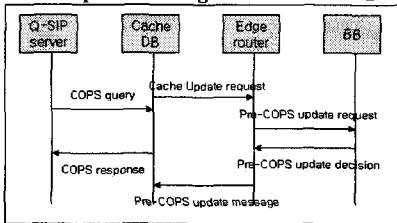


그림 6 Reservation COPS cache 와 Pre-COPS update 의 응용

그림 6 은 위의 두 가지 방식을 조합한 것이다. Q-SIP 서버에서 발생하는 COPS Query 는 호설정시에 정책결정을 위해 질의를 하는 것이다. Cache DB 에서는 주기적인 갱신을 위하여 cache 정보의 update 가 필요한 경우와 특정한 주기로 Edge 라우터에 cache update request message 를 보내고 Edge 라우터에서는 BB 와의 COPS 프로토콜을 이용하여 정책 결정을 한다. 물론 이 정보는 Cache DB 에 보내어 cache 가 갱신된다. 이 Cache update 가 이루어지는 때는 네트워크의 상태가 idle 상태일때 발생하여 네

트워크의 부하를 줄일 수 있다.

물론 전형적인 Q-SIP 설정시에도 정해진 정책결정을 cache DB 에 보내어 갱신한다.

전형적인 QoS 를 위한 COPS 프로토콜을 통한 정책결정은 더 많은 자원 낭비를 발생할 수 있지만 이를 Reservation COPS Cache 와 Pre-COPS update message 를 이용하여 줄여보자는 것이다. 그리고 Cache DB 의 갱신은 SIP 서버의 호설정과 필요한 경우 비동기적으로 발생한다. 이 두 가지 방식을 이용하면 외부 네트워크와의 연결을 줄이고 효율적인 QoS 의 설정이 이루어질 것이다.

4. 결론

VoIP 에서 많이 사용되어지는 호 설정 프로토콜은 H.323 과 SIP 이다. 많이 사용되고 있는 H.323 의 많은 단점으로 SIP 프로토콜에 대한 지속적인 연구가 이루어지고 있다. 여기에 QoS 를 제공하기 위한 방법으로 COPS-ODRA 와 Q-SIP 의 응용방법으로 Reservation COPS cache 와 Pre-COPS update message 를 제시하였다. 이 두 가지 방법을 혼용하여 cache 의 hit 율을 높인다면 QoS 설정단계에서 복잡한 절차를 간소화 할 것으로 기대된다.

제시한 방법, 즉 VoIP 의 SIP 을 통한 호설정시 QoS 를 지원하는 것으로 지금보다는 좀 더 간소화된 QoS 지원의 호설정이 이루어지고, QoS 를 지원하는 VoIP 를 통한 응용 또한 사용자 위주의 환경으로 개발될 수 있을 것이다.

참고문헌

[1] Kaist 정보통신경영연구소 “인터넷 기술세계” 2001.10(page 209-223)  
 [2] Luca Veltri et al “SIP Extension for QoS support” <draft-veltri-sip-qsip-01.txt>, Apr. 2003  
 [3] Stefan Salano et al “QoS Control by Means of COPS to support SIP-Based Application” IEEE network, Mar. 2002  
 [4] J.Boyle et al., “COPS Usage for RSVP.” IETF FRC 2749, Jan 2000.  
 [5]s. Salsano “COPS Usage for DiffServ Resource Allocation (COPS-DRA),” <draft-salsano-cops-dra-00.txt>, Oct, 2001