

SCTP 표준기술 분석 및 전망

Analysis and Prospect of Stream Control Transmission Protocol

고석주(S.J. Koh)

정희영(H.Y. Jung)

민재홍(J.H. Min)

박기식(K.S. Park)

표준기반연구팀 선임연구원

표준기반연구팀 선임연구원

표준기반연구팀 책임연구원, 팀장

표준연구센터 책임연구원, 센터장

최근 SCTP(Stream Control Transmission Protocol)는 TCP/UDP 이후의 차세대 수송계층 프로토콜로서 주목 받고 있다. SCTP는 기존 TCP 및 UDP의 문제점을 극복하도록 설계되었으며 특히 multi-streaming 및 multi-homing 특성을 제공한다. 본 고에서는 SCTP 프로토콜의 기본 특징에 대하여 알아보고, 현재 논의중인 확장작업의 주요 골자를 살펴본다.

I. 서론

인터넷 실시간 멀티미디어 응용 서비스의 다양한 요구사항에도 불구하고, 그 동안 인터넷에서의 수송계층 프로토콜은 TCP(Transmission Control Protocol) 및 UDP(User Datagram Protocol)로 제한되어 왔다. TCP 및 UDP 프로토콜은 기존의 전기통신망에서 제공되던 전화서비스를 비롯하여 새로이 개발되는 고도의 실시간 멀티미디어 응용 서비스 제공에는 부적합하다. 이러한 배경으로 IETF에서는 2000년 10월에 SCTP(Stream Control Transmission Protocol)을 RFC 2960으로 제정하였다. 초기 SCTP 개발 목적은 인터넷 망에서의 전화서비스 신호메시지 전달에 있었으나, 최근에는 범용(general-purpose) 수송계층 프로토콜로서 다양한 종류의 응용서비스로 적용 범위를 확대하고 있다.

SCTP는 UDP의 메시지 지향(message-oriented) 특성과 TCP의 연결형(connection-oriented) 및 신뢰성(reliability) 특성을 조합한 프로토콜이다. 이 외에도 SCTP는 멀티스트리밍(multi-streaming)

및 멀티호밍(multi-homing) 특성을 제공한다. 멀티스트리밍 특성을 통해 TCP의 HOL(Head-of-Line) 블로킹 문제를 해결하고 있으며, 멀티호밍 특성을 통해, IP 경로 장애에 대한 복구(fail-over) 기능을 제공한다. 또한 세션 초기화 및 종료단계에서 기존 TCP의 문제점으로 지적되던 “TCP-SYN 공격” 문제 및 “half-closed state” 문제 등을 해결하고 있다. SCTP는 IP 패킷에 표시되는 프로토콜 식별번호(protocol ID)로서 UDP(6), TDP(17)처럼 132번을 부여 받았다.

SCTP 수송계층 프로토콜은 VoIP(Voice over IP) 신호전달 외에도, 실시간 다중 미디어를 전송하는 웹 응용, AAA(Authentication, Authorization, Accounting) 등의 고도의 신뢰성을 요구하는 보안 응용 등을 위해 사용될 수 있으며, 특히 최근 이슈화되고 있는 all-IP 기반 이동통신망에서 핸드오버 기능을 용이하게 제공할 수 있을 것으로 전망된다.

현재 IETF에서는 SCTP 기본규격 제정작업을 마치고, 구현, 검증 및 추가 기능 확장작업을 진행중이다. SCTP 구현작업은 미국, 독일을 비롯한 세계

여러 나라에서 진행중이며, 일부 구현 코드가 공개되어 있다. 현재 활발하게 논의중인 SCTP 추가 확장작업으로는 “부분 신뢰성(partial reliability)” 제공 기능 및 “동적인 IP 주소 재구성” 기능 등이 있다.

본 고에서는 SCTP 프로토콜의 기본 특징에 대하여 알아보고, 현재 논의중인 추가 확장작업에 대하여 살펴본다.

II. SCTP 개요

SCTP 표준개발은 1998년에 SIGTRAN WG에서 시작되었다. SIGTRAN WG에서는 IP 망에서의 VoIP 시그널링(예: SS7) 전송을 위해 적합한 새로운 전송 프로토콜 표준개발을 주요 목표로 하였다. 신호정보의 특성상 고도의 실시간성 및 신뢰전송을 제공하는 수송계층 프로토콜을 필요로 하였다. 특히, 수송계층에서의 전송경로 장애를 대비한 대체 경로(alternate path) 확보 기능을 필요로 하였으며, 하나의 세션을 통해 여러 호처리(call processing) 메시지가 전송되는 ‘multi-streaming’ 기능도 요구되었다. 위와 같은 기능은 전통적인 TCP로서는 지원되기 어렵다는 인식 하에, SIGTRAN WG은 새로운 수송계층 프로토콜, SCTP를 개발하기에 이른다.

SCTP는 2000년 10월 SIGTRAN WG에서 RFC 2960으로 제정되었으며[1]-[5], 이후 보완[6]-[8], 구현 및 검증 관련[9]-[12] 후속 표준 작업이 계속되고 있다. 이후 SCTP를 단순히 신호 전달 기능이 아닌, TCP 이후의 차세대 수송계층 프로토콜로 개발하려는 취지 하에, 관련 확장 작업은 Transport Area WG에서 진행되고 있다[13],[14].

현재, SCTP 구현 및 확장 작업은 미국, 독일 등의 여러 곳에서 진행중이다. 특히, SCTP의 활용은 다중 스트림 특성을 갖는 멀티미디어 응용, 실시간 신뢰전송을 요구하는 응용 및 이동 단말의 핸드오버(handover) 기능 제공을 위해 사용되는 방안이 적극 검토되고 있으며, 관련 표준개발 작업이 진행중이다.

1. SCTP 아키텍처

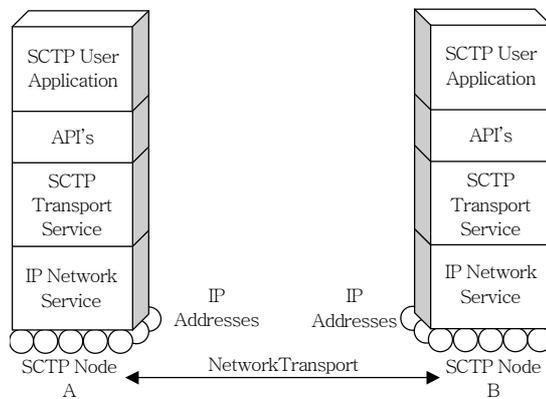
가. 프로토콜 구조

SCTP는 기존의 UDP 및 TCP에 비해 많은 장점을 제공한다. SCTP는 UDP의 데이터그램 지향성과 TCP의 순서화(sequencing) 및 신뢰성 특징을 결합한 것으로 볼 수 있다. 추가적으로 SCTP는 멀티호밍(multi-homing) 환경에서 다중스트림 및 메시지 지향성 전달 특성을 갖는다.

(그림 1)에서 보여지듯이, SCTP는 응용 계층과 네트워크 계층 사이에 위치한다. SCTP는 SCTP peers 간에 응용 데이터들 APIs(Application Programming Interfaces)로 전달 받아 IP 망을 통해 전송하는 기능을 수행한다. 각 SCTP 단말은 하나의 SCTP 세션에서 여러 개의 IP 주소를 사용할 수 있다.

SCTP는 두 지점간에 메시지 전송을 위해 다중경로(multiple path) 및 다중스트림 기능을 사용한다. 두 SCTP 지점간의 SCTP 연결을 “SCTP association”이라 한다. 특정 SCTP association에서의 전송 절차는 크게 연결 초기화(initiation), 데이터 전송(data transfer) 및 연결종료(shutdown) 단계로 나뉘어진다.

SCTP 패킷 혹은 PDU(Protocol Data Unit)은 하나의 헤더(header)와 여러 개의 chunks로 구성되며, 각 chunk는 제어 정보 혹은 응용데이터를 포



(그림 1) SCTP 프로토콜 구조

함한다. SCTP 스트림에서 각 데이터는 순서적으로 전달되며, 멀티호밍 특성을 활용하여 망장애(network failure)에 대비한 복구 기능을 수행할 수 있다.

SCTP의 기본 특징은 다음과 같다.

1) 세션 인증 및 SACK 메커니즘

SCTP 세션 초기화 과정에서, SCTP endpoint 간에 COOKIE를 사용하여 인증을 위한 key 정보를 교환한다. COOKIE 정보는 MAC(Message Authentication Codes) 코드를 통해 생성되며 DoS (Denial of Service) 공격 방지를 위해 사용된다.

또한 세션 도중에 데이터 수신여부 및 재전송요구 등을 나타내기 위해 수신자는 SACK(Selective Acknowledgement) chunk를 송신자에게 보낸다.

2) 경로 선정 및 감시(Path Selection and Monitoring)

각 사용자는 하나의 SCTP 세션에 여러 IP 주소를 사용할 수 있으며, 세션 중에는 primary 주소가 지정되어 데이터 전송에 사용되며, 나머지 alternate 주소는 재전송 혹은 primary 주소가 failure된 경우에 사용된다. 이를 위해, SCTP는 주기적으로 가용 IP 주소의 연결 상태를 검사하여, active 혹은 inactive 상태 정보를 기록한다.

3) 흐름 및 혼잡 제어(Flow and Congestion Control)

SCTP 흐름제어(flow control)는 association 별로 수행되는 반면, 혼잡제어(congestion control)는 전송경로 별로 수행된다. 흐름제어를 위해 각 SCTP endpoint는 TCP처럼 수신윈도를 송신자에게 알려주며, 혼잡제어를 위해 congestion 윈도 크기를 적절히 조절한다.

나. SCTP 특징

기본적으로 SCTP는 TCP에 비해 멀티스트리밍

및 멀티호밍 특성을 제공하며, 또한 세션 초기화, 데이터 전송 및 세션 종료에서도 TCP와 다른 특징을 제공한다.

1) 멀티호밍

SCTP의 멀티호밍 특성은 SCTP 세션이 여러 개의 IP 주소를 동시에 사용할 수 있도록 하며, 세션 도중 네트워크 장애가 발생하는 경우 대체경로(혹은 대체 IP 주소)를 통해 세션이 유지되도록 한다. 세션 초기화 단계에서, 각 SCTP peers는 가용한 IP 주소 목록을 교환하며, 이 중에 primary IP 주소를 선정해 준다. 상대방 SCTP는 primary 주소를 수신주소로 하여 데이터를 전송하게 된다. SCTP 경로관리에 의해 primary IP 주소에 이상이 있다고 판단되는 경우, SCTP는 다른 대체 IP 주소로 데이터 전송을 계속한다.

2) 멀티스트리밍

SCTP의 멀티스트리밍 특성은 하나의 세션을 통해 다양한 종류의 응용데이터를 보낼 수 있도록 한다. 세션 초기화 단계에서 송신자는 자신이 전송할 스트림의 개수를 수신자에게 통보하며, 전송 단계에서 각 스트림 별로 독립적인 순서화(ordering) 기능이 제공된다. 데이터 복구 및 재전송 과정 또한 스트림 ID 별로 수행되어, 기존의 TCP에서 문제시 되었던 HOL 블로킹 문제를 해결한다.

지금까지 기술한 멀티스트리밍 및 멀티호밍 특성이외에도, SCTP는 TCP와는 다른 세션 초기화, 데이터 전송 및 종료 절차를 사용한다.

3) 세션 초기화

초기화 과정에서 SCTP는 TCP의 three-way handshake와는 달리, four-way handshake 절차를 사용한다. 이러한 4단계 초기화 절차는 외부의 DoS 공격을 방어하는 데에 적합하다. 또한 SCTP는 verification tag와 COOKIE 값을 사용하여, 소위 TCP SYN 공격(blind attack)을 방지한다.

4) 데이터 전송

SCTP 데이터 전송 단계에서, 각 데이터 chunk 에게 TSN(Transmission Sequence Numbers) 및 SSN(Stream Sequence Numbers) 값이 부여된다. TSN은 세션에 대한 흐름제어 및 오류복구를 위해 사용되며, SSN은 수신단 SCTP에서 스트림별 순서화를 위해 사용된다.

5) 세션 종료

SCTP 세션 종료(shutdown)의 절차는 TCP의 “half-open” 문제를 해결한다. TCP에서는 4단계 절차를 통해 각 SCTP 개체별로 데이터 전송을 종료했으나, SCTP에서는 3단계 절차를 사용하여 두 개체의 데이터 전송 종료가 통합적으로 이루어지도록 하였다.

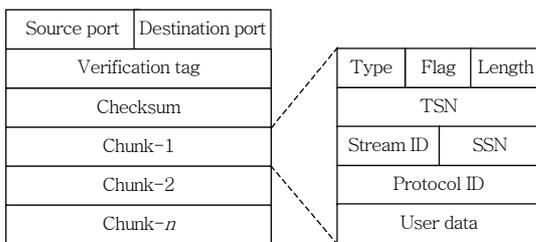
다. SCTP 패킷 구조

(그림 2)는 SCTP 패킷 구조를 보여준다.

SCTP 패킷의 헤더에는 송신자 포트번호(16bits), 수신자 포트번호(16bits), verification tag(32bits) 및 전체 패킷에 대한 Checksum(32bits) 정보가 포함된다. Verification tag 값은 association 별로 할당되며, 세션 식별자로 사용된다.

하나의 SCTP 패킷은 여러 개의 데이터 및 제어 chunks를 포함할 수 있으며, 데이터 chunk의 경우 Type, Length 정보와 함께 해당 데이터 chunk에 대한 TSN, SSN 번호를 포함하여 추후 오류제어 및 흐름제어 등에 사용된다.

SCTP는 DATA chunk와 함께 여러 개의 제어



(그림 2) SCTP 패킷 구조

<표 1> SCTP 제어 Chunks

제어 Chunk	내용
INIT	SCTP association의 초기화를 위해 initiator에 의해 전송됨
INIT-ACK	INIT chunk에 대한 응답 메시지
COOKIE-ECHO	SCTP initiator는 INIT-ACK를 받은 후 세션설정을 마무리 하기 위해 COOKIE-ECHO를 전송
COOKIE-ACK	COOKIE-ECHO chunk에 대한 응답 메시지
SACK (Selective ACK)	데이터 수신상태 정보를 나타내기 위해 사용되며, 수신 데이터 chunk에 대한 TSN, SSN 정보 포함
HEARTBEAT	SCTP 경로관리를 위해 사용되며, SCTP는 자신의 가용 IP 주소에 대해 HEARTBEAT chunk 전송
HEARTBEAT-ACK	HEARTBEAT chunk에 대한 응답 메시지
ABORT	상대 SCTP endpoint에게 비정상적인 세션종료를 나타냄
ERROR	SCTP 운용상의 오류를 나타냄
SHUTDOWN	정상적인 세션 종료 메시지
SHUTDOWN-ACK	SHUTDOWN에 대한 응답메시지
SHUTDOWN-COMplete	최종 세션 종료를 나타냄

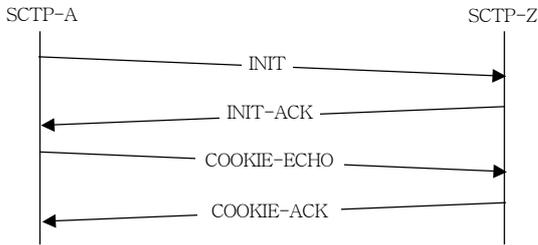
chunks를 사용하며, 추후 SCTP 확장작업과 함께 제어 chunk의 수는 증가할 수 있다. SCTP 기본 규격(RFC 2960)에 정의된 주요 제어 chunk의 종류를 정리하면 <표 1>과 같다.

2. 프로토콜 동작방식

가. SCTP 세션 초기화

SCTP 세션 초기화는 4단계 절차로 구성된다. 먼저 SCTP-A는 INIT 메시지(INIT chunk를 포함하는)를 SCTP-Z에게 전송하여 세션 초기화를 시작한다. INIT 메시지는 A에서 사용할 IP 주소목록을 포함한다. SCTP-Z는 INIT에 응답하여 INIT-ACK 메시지를 전송하며, 자신이 사용할 IP 주소 목록을 포함시킨다. INIT 메시지 교환은 세션 초기화의 첫 번째 절차이지만, 이 단계의 각 endpoint에서는 SCTP PCB(Protocol Control Block) 상태정보를 생성시키지 않는다(그림 3) 참조.

이 후에 COOKIE-ECHO 및 ACK 메시지를 교



(그림 3) Sctp 초기화

환함으로써 세션 초기화가 마무리되며, 이 단계의 각 endpoint에서 PCB를 생성시킨다. COOKIE의 사용은 세션에 대한 사용자 인증 기능을 제공하며, 또한 정당한 COOKIE에 대해 PCB를 생성시킴으로써 TCP-SYN(blind attack) 문제를 해결한다.

(그림 4)는 Sctp 초기화 단계에서의 각 종단에서의 상태천이도(state transition diagram)를 보여준다. 각 단계별로 Sctp 종단에서는 INIT, COOKIE 타이머를 사용하며, COOKIE-ECHO 및 COOKIE-ACK chunks는 데이터 chunks와 함께 전송될 수 있다.

나. 데이터 전송

Sctp는 오류 제어를 위해서 Selective ACK (SACK) 방식을 사용한다. A에서 전송된 데이터 chunks에 대해, 수신자 Z는 수신된 데이터 chunks

의 목록을 SACK chunk를 통해 A에게 알려준다. SACK 전송규칙은 A에서의 재전송 타이머 및 흐름 제어 등을 고려하여 제정되었다.

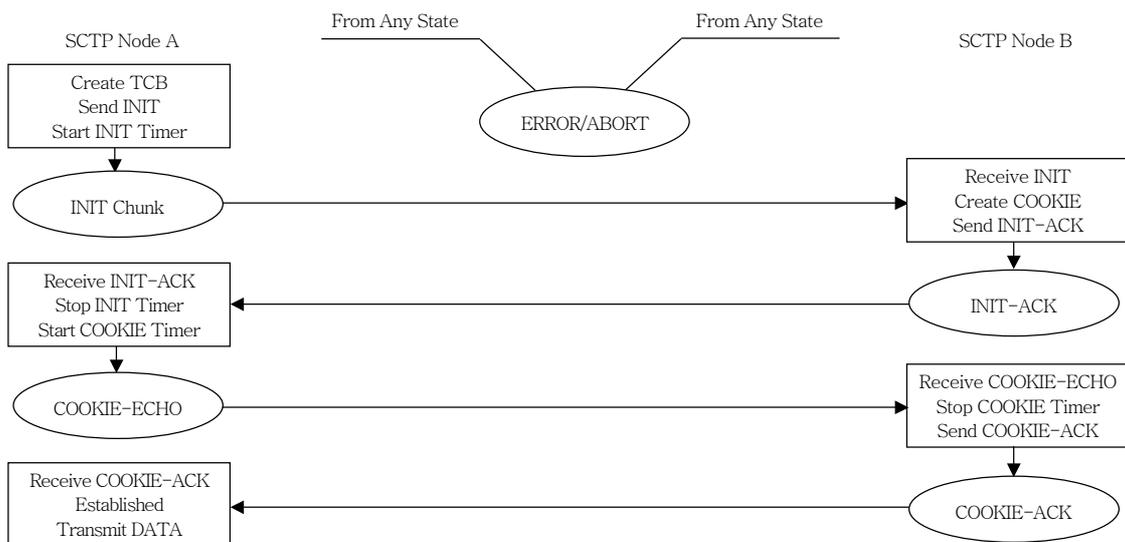
(그림 5)에서 보여지듯이, 적어도 2개의 데이터 Sctp 패킷(여러 개의 데이터 chunks 포함 가능)마다 하나의 SACK chunk를 보내야 하며 (흐름제어를 위해), SACK chunk는 데이터 chunks와 함께 전송될 수 있다.

(그림 6)에서 보여지듯이, 데이터 전송단계에서는 각 Sctp 종단은 SACK와 함께 경로 (목적지 IP 주소) 관리를 위해 HEARTBEAT 메시지를 전송한다. 이를 통해 상대방의 어떤 IP 주소들이 가용한지를 주기적으로 파악한다.

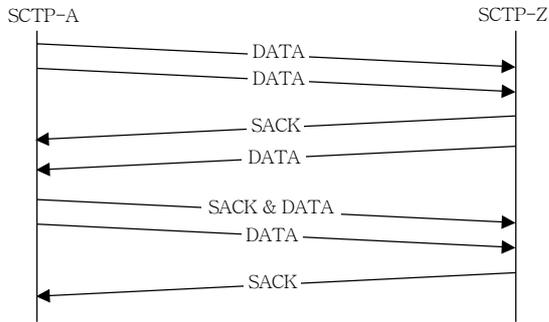
다. Sctp 세션 종료

Sctp 세션종료 절차는 TCP의 4단계와는 달리, 3단계로 구성된다. 이를 통해 TCP의 “half-open closing”을 해결함으로써 프로토콜 상태관리를 최적화 하였다.

먼저 (그림 7)에서 보는 바와 같이, A는 데이터 전송 종료 후에 SHUTDOWN 메시지를 전송한다. Z는 자신의 데이터 전송을 마무리하고 SHUTDOWN-ACK를 A에 전달한다. 마지막으로 A는 SHUT-



(그림 4) Sctp 초기화 단계 상태천이도



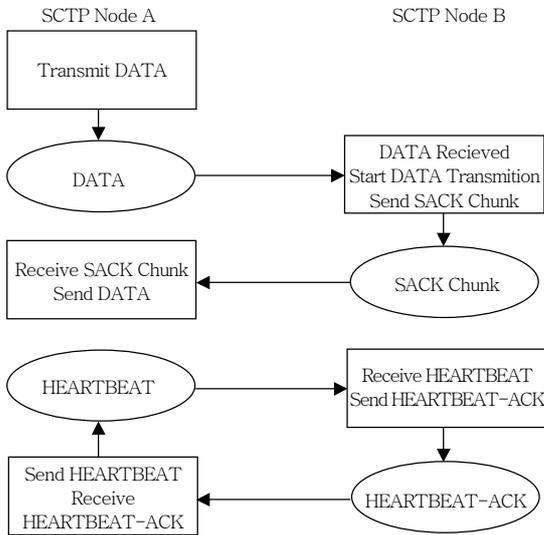
(그림 5) Sctp 데이터 전송

DOWN-COMPLETE를 전송하여 최종적으로 세션 종료를 마무리한다.

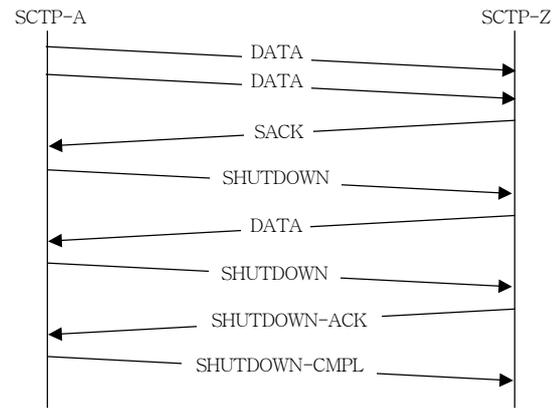
(그림 8)은 세션 종료단계에서의 프로토콜 상태천이도를 보여준다. 노드 A는 SHUTDOWN 타이머를 가동하여 B로부터 SHUTDOWN-ACK가 오지 않는 경우 SHUTDOWN 메시지를 재전송할 수 있다.

3. Sctp 개발 현황

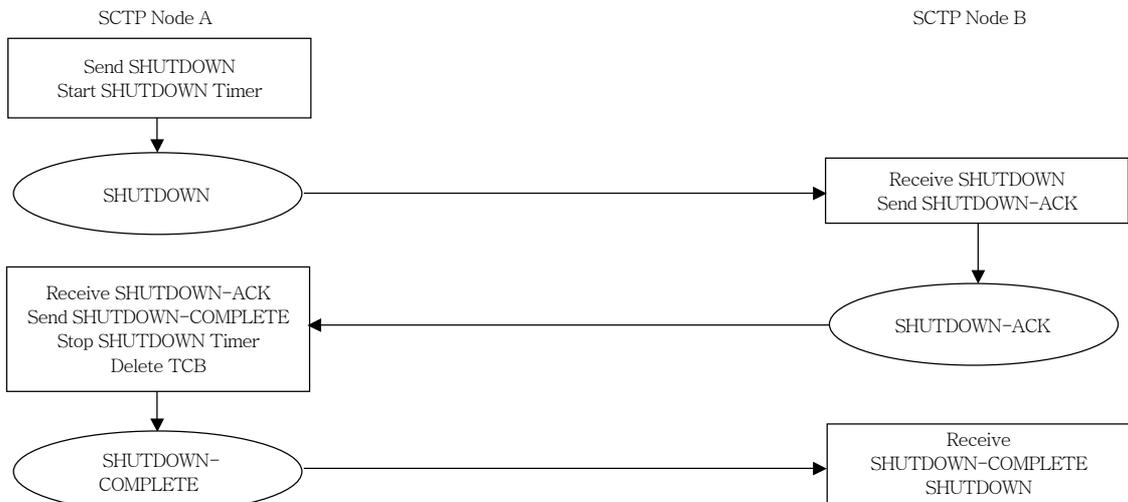
Sctp 표준개발 작업은 IETF SIGTRAN WG에서 시작되어 현재 기본 규격 제정은 마무리된 상태이다. RFC 2960은 Sctp 기본 규격문서이며, 보조



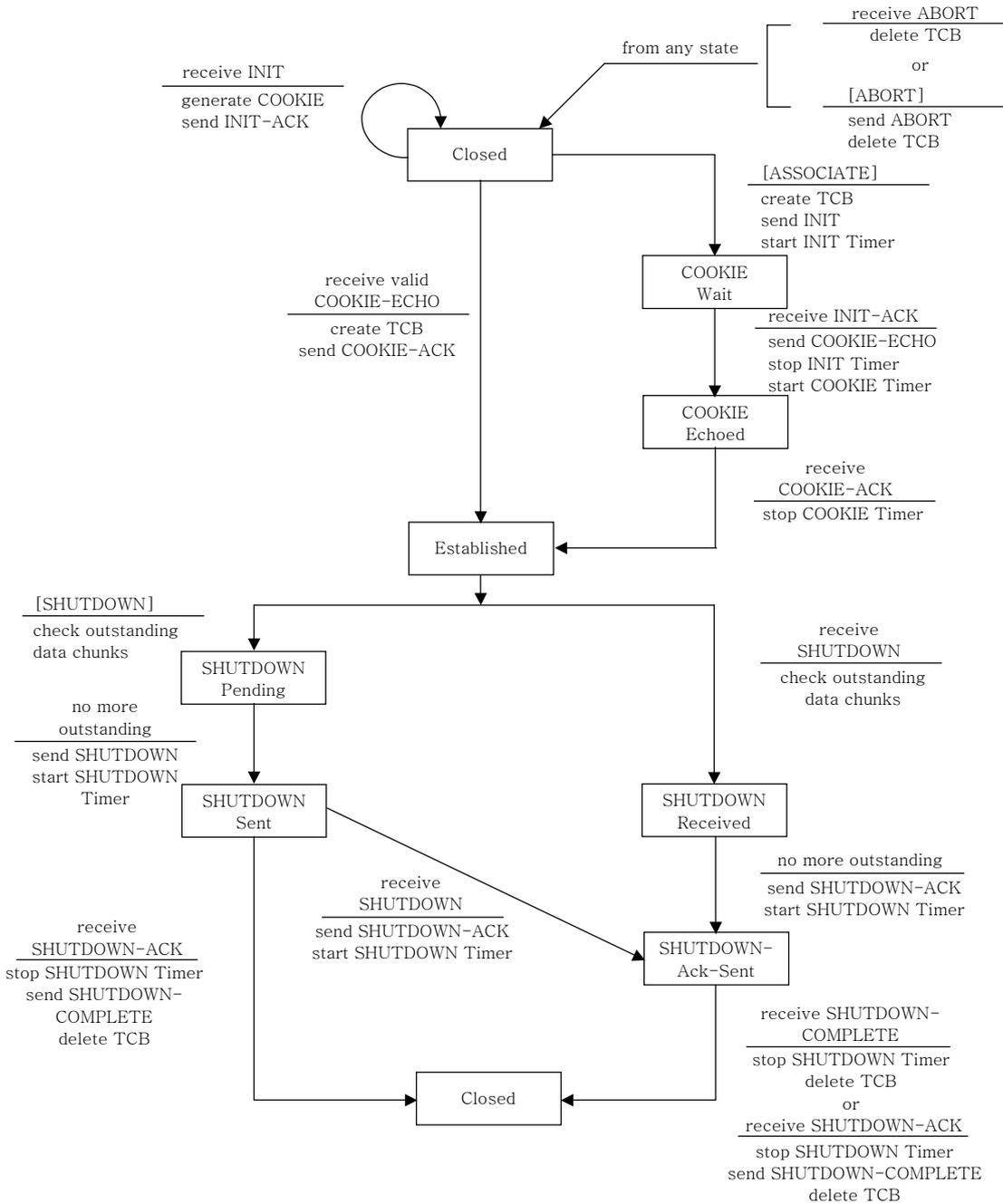
(그림 6) Sctp 데이터 전송단계 상태천이도



(그림 7) Sctp 세션 종료



(그림 8) Sctp 세션 종료단계 상태천이도



(그림 9) SCTP 프로토콜 상태천이도

문서로서 RFC 3257, 3286 문서가 작성되었다. 기본 규격 제정 후에 SCTP 표준화 작업은 IETF TRANSPORT AREA WG으로 이관되었으며, 현재는 VoIP 시그널링용 프로토콜이 아닌 TCP 차기버

전의 범용 수송계층 프로토콜로서 표준규격 개발작업이 진행되고 있다.

이후에 추가 규격확장작업이 계속 진행되고 있으며, SCTP 관련 수송계층 보안문서가 RFC 3436으

로 2002년 12월에 제정되었다. 현재 WG에서 개발 중에 있는 문서로는 SCTP 구현 가이드[6], SCTP를 위한 소켓 API 확장[7], SCTP MIB(Management Information Base)[8] 등이 있다. 최근에는 이동통신에서 핸드오버를 지원하기 위해 세션 도중에도 새로운 IP 주소를 추가, 삭제, 변경할 수 있는 기능이 추가적으로 개발되고 있다[13].

SCTP 구현 및 개발 작업은 미국, 독일 등 전세계적으로 진행중에 있다[9]-[12]. 현재 BSD 및 리눅스 플랫폼 환경에서 개발이 진행중에 있으며, 관련 ns-2 시뮬레이션 코드도 개발되었다. 구현작업 또한 커널 환경 및 user space에서 모두 개발이 진행중이다.

(그림 9)는 SCTP 구현을 위한 전체적인 동작 상태천이도이다.

III. SCTP 확장

SCTP 프로토콜은 범용(general-purpose) 수송 계층 프로토콜로서 사용되기 위해, 향후 추가 기능 확장작업이 계속될 것으로 전망된다. 현재 진행되고 있는 주요 기능확장작업으로는 “부분 신뢰성(partial reliable) 제공 기능” [13] 및 “동적인 IP 주소 구성” [14] 등이 있다.

1. PR-SCTP

PR-SCTP(Partial Reliable-SCTP)는 실시간 응용을 위해 제정되고 있으며[13], SCTP 데이터 전송 도중에 “time-critical” 스트림에 대해서는 오류제어를 중단하고 곧바로 응용계층에 전달될 수 있는 기능을 정의한다.

PR-SCTP 기능 사용을 위해서는 SCTP 양방간에 세션 초기화 단계에서부터 PR-SCTP 기능을 지원할 것인지 여부를 협상해야 한다. 일단 데이터 전송이 시작된 후에, 송신자는 필요에 따라 특정 TSN 번호 이전의 데이터에 대해서는 신뢰성 제공 기능을 중지하도록 상대방 종단에 통지할 수 있다. 수신자는 PR-SCTP 신호를 받았을 경우, 자신의 버퍼에

있는 해당 데이터 chunks들을 곧바로 상위 응용에 전달한다.

이러한 확장 기능은 주로 시간적 제약을 갖는 응용 데이터 전송에 사용될 것으로 전망된다.

2. Mobile SCTP

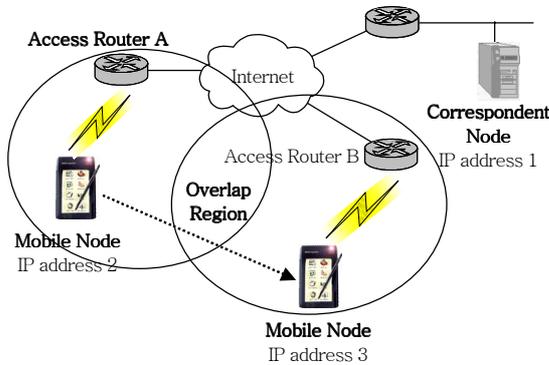
또 하나의 주요 SCTP 기능 확장으로는 ADDIP 확장을 들 수 있다[14]. SCTP 기본 규격에서는 세션에 사용되는 IP 주소를 세션초기화 단계에서만 지정하도록 되어 있었으나, 본 확장 규격에서는 세션 도중에도 신규 IP 주소를 세션에 등록하거나 혹은 삭제하는 기능을 제공한다. 또한 세션 도중에 primary IP 주소를 변경하는 기능도 포함한다.

이러한 SCTP 세션에 대한 IP 주소의 재구성이 필요할 경우, 해당 SCTP는 관련 주소 정보를 AS-CONF(Address Configuration Change) 제어 chunk에 실어 상대방에게 전송하며, 상대방은 ACSONF-ACK chunk로 응답할 수 있다.

위 기능은 특히 이동단말이 세션 도중에 다른 IP 망으로 이전하게 되는 경우에, seamless 핸드오버 기능 지원을 위해 필수적으로 요구되는 사항이다. 특히 기존의 Mobile IP[15]-[18] 경우, 핸드오버 기능이 취약하였으나 SCTP를 통해 핸드오버 기능이 상당히 개선될 것으로 기대된다[19]. 한편 완전한 IP 이동성 지원을 위해서는 핸드오버와 함께, IP 단말에 대한 위치관리 기능이 제공되어야 하며, 이를 위해 Mobile IP의 위치등록 기능이 함께 사용될 수 있을 것으로 전망된다[20]. 본 절에서는 AS-CONF 기능을 활용한 seamless 핸드오버 절차에 대해 기술한다.

(그림 10)에 보여지듯이, 이동 단말이 SCTP를 사용하며 세션 도중에 IP 지역을 바꾸는 경우 핸드오버 절차는 다음과 같다[19].

- (1) 먼저 그림에서처럼, 세션 초기화 단계에서 이동단말(Mobile Node: MN)은 주소 2를, 상대방(Correspondent Node: CN)은 주소 1을 사용하여 SCTP 세션을 설정하였다고 가정한다.



(그림 10) SCTP 핸드오버

- (2) MN이 다른 IP 영역으로 이동하는 경우, 중첩 지역(overlapping region)에서 신규 주소 3을 하위 네트워크 계층으로부터 받게 되면, 이를 ASCONF chunk를 통해 CN에게 통지한다. 이를 통해 MN은 dual-homing 상태가 되며, CN으로부터의 데이터를 주소 2 뿐만 아니라 주소 3을 통해서도 받을 수 있게 된다.
- (3) MN은 무선계층의 신호세기에 따라 primary 주소를 주소 3으로 변경할 수 있다.
- (4) MN이 중첩지역을 벗어나는 경우 ASCONF를 통해 기존 주소 2를 SCTP 세션에서 삭제할 수 있다.
- (5) 이러한 절차가 SCTP 세션 도중에 IP 지역을 바꿀 때마다 되풀이 된다.

SCTP 기반의 핸드오버는 기존의 MIP(Mobile IP)와는 달리 종단간 수송계층에서 이루어진다는 특징이 있으며, 따라서 터널링 등의 네트워크 라우터의 도움이 없이도 적용 가능하다. SCTP 핸드오버에 대한 유일한 요구사항은 MN 및 CN에 SCTP가 사용되어야 한다는 점이다. 이러한 Mobile SCTP 기능은 차세대 all-IP 기반 이동통신망의 IP 이동성 관리 기법으로 적용될 수 있을 것으로 전망된다.

IV. 결론 및 향후 전망

지금까지 본 고에서는 SCTP 프로토콜의 기본

특징 및 주요 확장기능에 대하여 살펴보았다. SCTP는 TCP 이후의 차세대 수송계층 프로토콜로서 지속적인 표준확장 및 보급이 이어질 것으로 전망된다. 시간이 지남에 따라 SCTP 보급이 확대되면, 기존에 TCP를 통해 제공되던 응용들도 SCTP를 통해 보다 효율적으로 제공될 수 있을 것으로 전망된다.

특히, 실시간 멀티미디어 전송 및 고도의 신뢰성이 요구되는 응용에 대해서는 SCTP의 적용이 선호된다. 또한, 차세대 이동통신망에서의 IP 이동성 제공 측면에서도 SCTP의 사용이 긍정적으로 검토될 수 있을 것이다.

참고 문헌

- [1] R. Stewart and Xie Qiaobing, "Stream Control Transmission Protocol," Addison-Wesley, 2001.
- [2] R. Stewart et al., "Stream Control Transmission Protocol," IETF RFC 2960, Oct. 2000.
- [3] L. Ong and J. Yoakum, "An Introduction to the Stream Control Transmission Protocol(SCTP)," IETF RFC 3286, May 2002.
- [4] L. Coene et al., "Stream Control Transmission Protocol Applicability Statement," IETF RFC 3257, April 2002.
- [5] A. Jungmaier et al., "Transport Layer Security over SCTP," IETF RFC 3436, Dec. 2002.
- [6] R. Stewart et al., "SCTP Implementers Guide," IETF Internet Draft, draft-ietf-tsvwg-sctpimp-guide-sctp-07.txt, Oct. 2002.
- [7] R. Stewart et al., "Sockets API Extensions for SCTP," IETF Internet Draft, draft-ietf-tsvwg-sctpsocket-05.txt, Oct. 2002.
- [8] J. Pastor and M. Belinchon "SCTP Management Information Base," IETF Internet Draft, draft-ietf-sigtran-sctp-mib-08.txt, Nov. 2002.
- [9] SCTP implementations by BSD, <http://sourceforge.net/projects/lksctp>
- [10] SCTP implementations by Linux, <http://trivus.sourceforge.net/>
- [11] SCTP ns-2 simulations codes, <http://pel.cis.udel.edu/#downloads>
- [12] SCTP tutorial, <http://www.iec.org/online/tutorials/sctp/>

- [13] R. Stewart et al., "SCTP Partial Reliability Extension," IETF Internet Draft, draft-stewart-tsvwg-prsctp-02.txt, Dec. 2002.
- [14] R. Stewart et al., "SCTP Dynamic Address Reconfiguration," IETF Internet Draft, draft-ietf-tsvwg-addip-sctp-05.txt, May 2002.
- [15] C. Perkins(ed.), "IP Mobility Support for IPv4," IETF RFC 3344, Aug. 2002.
- [16] D. Johnson et al., "Mobility Support in IPv6," IETF Internet Draft, draft-ietf-mobileip-ipv6-20.txt, Jan. 2003.
- [17] K.L. Malki et al., "Low Latency Handoffs in Mobile IPv4," IETF Internet Draft, draft-ietf-mobileip-lowlatency-handoffs-v4-04.txt, June 2002.
- [18] R. Koodli et al., "Fast Handovers for Mobile IPv6," IETF Internet Draft, draft-ietf-mobileip-fastmipv6-05.txt, Sep. 2002.
- [19] S.J. Koh et al., "Use of SCTP for Seamless Handover," IETF Internet Draft, draft-sjkoh-mobile-sctp-handover-00.txt, Feb. 2003.
- [20] S.J. Koh et al., "SCTP with Mobile IP for IP Mobility Support," IETF Internet Draft, draft-sjkoh-mobile-sctp-mobileip-00.txt, Feb. 2003.