Diameter Mobile-IP 응용을 지원하는 Diameter 선불 시스템의 설계 및 구현

정회원 유 상 근*, 김 현 곤**

Design and implementation of prepaid service for Mobile-IP in Diameter

Sangkeun Yoo*, Hyungon Kim** *Regular Members*

요 약

본 논문은 Diameter 기반 AAA 시스템에서 Mobile-IP 응용과 연동하는 선불 서비스의 설계 및 구현에 대하여 기술하였다. 선불 서비스 기능은 Diameter의 Credit-Control 응용에 기반 하였으며, Credit-Control 응용은 기존의 RADIUS 과금 방식이나 Diameter 베이스 프로토콜의 과금 방식이 지원하지 않는 선불 과금 서비스를 제공하기 위한 응용 기능이다. 실시간 선불 서비스 기능은 실시간 서비스 비용 결정과 서비스를 제공하기 위한 사용자의 계정 잔액 확인 절차 등과 같은 추가의 기능들을 요구한다. 본 논문에서는 위와 같은 추가의 기능들을 지원할 수 있도록, Mobile-IP 응용 서비스인 Diameter Mobile-IP 응용과 연동하는 Diameter Credit-Control 응용의 설계와 구현에 대하여 기술한다.

Key Words : AAA, Diameter, Prepaid-Service, Credit-Control, Mobile-IP

ABSTRACT

This paper presents the design and implementation of credit-control application to provide prepaid service for Diameter Mobile-IP application in Diameter-based AAA system. Diameter credit-control application is designed to support prepaid accounting service, which is not supported in RADIUS and Diameter accounting. Real-time credit-control requires that an application must be able to rate service information in real-time. In addition, it is necessary to check that the end user's account provides coverage for the requested service, prior to initiation of that service. In this paper, we design and implementation Diameter credit-control to provide prepaid service for Diameter Mobile-IP application.

Ⅰ. 서 론

선불 서비스(Prepaid Service)란 사용자가 서비스 제공자에게 미리 일정금액의 돈을 지불한 후, 서비스 사용 시 미리 지불한 금액에서 서비스 비용을 실시간으로 차감하는 과금 방식이다. 이러한, 선불 서비스는 GSM (Global System for Mobile communication) 네트워크에서 매우 성공적인 지불방식으로 여겨지고 있으며, 가입자와 시장이 계속 성장하고 있는 추세이다[1]. 또한, 선불 방식의 과금 서비스는 현재 유선 환경뿐 아니라 무선 환경에서도 그 중요성이 증대되고 있는 실정이다.

차세대 무선 네트워크 환경에서는 Diameter 베이스 프로토콜(Diameter Base Protocol)[2]에서 정의한 기본적인 과금 기능 이외의 추가적인 기능들을 요구한다. 예컨대, 3GPP의 과금 방식 (3GPP Charging and

* 한국전자통신연구원 정보보호연구단 AAA정보보호연구팀(lobbi@etri.re.kr),
**한국전자통신연구원 정보보호연구단 AAA정보보호연구팀(hyungon@etri.re.kr)
논문번호 : KICS2004-09-193, 접수일자 : 2004년 9월 13일

Billing)에서는, 응용 서비스는 실시간으로 서비스의 비용을 결정하는 기능을 제공해야 한다[3]. 또한, 서비스를 제공하기 전 사용자의 계정이 사용자가 요구한 서비스의 비용을 지불할 수 있는지에 대한 확인 등의 절차가 필요하다. 만일 사용자의 계정이 비어 있거나 요구한 서비스 비용에 부족한 경우 서비스는 제공되지 않는다.

그러나, 현재 널리 사용되고 있는 과금 프로토콜인 RADIUS 과금[4] 방식이나 Diameter 베이스 프로토콜의 과금 방식은, 서비스 사용 시 과금 정보를 수집한 후 추후 정산하는 후불제 과금 방식이라는 점에서 실시간 선불 서비스 방식에 적합하지 않다. 그리고 Mobile-IP 응용[5]과 EAP 응용[6]과 같은Diameter의 응용들은 각 응용에 맞는 특정 서비스 권한검증(Authorization) 기능을 제공하지만, 선불 서비스를 위한 권한 검증 기능은 제공하지 않는다.

따라서 Diameter 환경에서 이러한 선불 서비스를 위한 별도의 응용인 Diameter Credit-Control(이후 CC라 지칭)[6] 응용이 필요하다. CC 응용이 제공하는 권한 검증 기능은 일반적이어야 하며, 선불 서비스를 사용하고자 하는 모든 응용에 적용될 수 있어야 한다. 본 논문은 이러한 Diameter CC 응용 기능을 제공하기 위해 Diameter 기반의 AAA (Authentication, Authorization and Accounting) 시스템에서 Diameter Mobile-IP 응용에 선불 서비스를 제공할 수 있는 CC 응용을 설계했으며, 이에 따른 구현에 대해 기술하였다.

본 논문의 구성은 다음과 같다. 2장에서 Diameter 기반의 AAA시스템에서 CC의 기능 및 구조에 대해 설명하며, Mobile-IP 응용과 결합된 CC 권한 검증 절차에 대해 제안한다. 3장에서는 본 논문에서 구현한 Diameter AAA 시스템의 구조와 구현에 대하여 설명하고 4장에서 결론을 맺는다.

Ⅱ. Diameter Credit-Control 기능 및 구조

Diameter 기반 AAA 시스템에서 선불 서비스를 제공하기 위해서는, AAA 기반구조에 새로운 Diameter 노드인 Diameter CC 클라이언트와 서버가 필요하다.

그림 1은 일반적인 Diameter CC의 구조를 나타낸다. 그림에서 서비스 노드(서비스를 제공하는 네트워크 노드, Service Element)는 사용자에게 서비스를 제공하는 장비로 Mobile-IP 서비스인 경우 FA(Foreign Agent) 혹은 AR(Access Router)로 간주한다. CC 클라이언트 기능은 서비스 노드에 포함될 수 있고(그림 1), AAA 서버에 포함될 수 있다. 본 논문에서는 서비스 노드에 CC 클라이언트 기능이 포함된 것으로 간주한다.

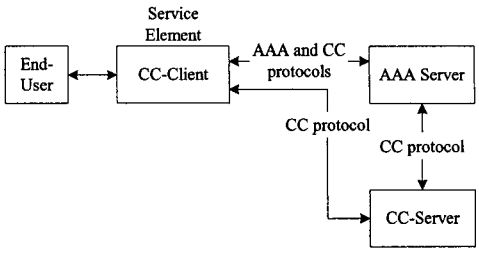

그림 1. 일반적 CC 구조

사용자가 서비스 노드에 접속하여 인증 및 권한 검증 요청을 하면, 서비스 노드는 AAA 서버인 Diameter 서버와 AAA 기반구조를 이용하여 응용의 인증/권한검증을 수행하고, 동시에 CC 권한검증 기능을 수행한다. 응용의 인증/권한검증 절차와 CC 권한검증 절차는 3장에서 설명한다.

그림에서 AAA 서버와 CC 서버는 따로 떨어진 객체로 표현됐지만, 실제 AAA 시스템에서 두 서버는 물리적으로 하나의 서버에 동시에 존재할 수도 있고 그림과 같이 두 개의 서버에 분리되어 존재할 수도 있다. 본 논문에서는 하나의 서버에 Diameter 서버와 CC 서버가 존재하는 것으로 설계하였다.

CC 클라이언트가 탑재된 서비스 노드와 AAA 서버간 프로토콜은 Diameter 프로토콜과 CC 프로토콜이 사용되며, 서비스 노드와 CC 서버간 프로토콜은 CC 프로토콜이 사용된다. CC 프로토콜은 다음의 두 가지 Diameter 메시지를 사용하며, 메시지가 포함한 AVP의 설명은 생략하기로 한다.

표 1. CC 메시지

Command-Name	Abbrev.	Command-Code
CC-Request	CCR	272
CC-Answer	CCA	272

- Credit-Control-Request (CCR)
  CC 클라이언트가 CC 서버에게 크레디트 권한검증 요청 시 사용하며, 중간의 갱신(Update) 요청, 서비스 해제(Terminate) 요청 시에도 사용한다.
  - CCR-INITIAL : 최초 권한검증 시 사용

- CCR-UPDATE : 중간의 갱신 요청에 사용
- CCR-TERMINATION : 서비스 해제 시 사용

• Credit-Control-Answer (CCA)
CCR에 대한 응답으로 CC 서버가 CC 클라이언트에게 전송한다.

1. CC의 동작 절차

Mobile-IP 사용자가 서비스 노드에 접근하여 서비스를 요구하는 경우, 서비스 노드는 사용자에 대한 Mobile-IP 인증/권한검증과 CC 권한검증이 성공하기 전까지 서비스를 제공하지 않는다.

사용자가 서비스를 요청하면, CC 클라이언트는 CCR-INITIAL을 생성하여 CC 서버에게 CC 인증 요청을 한다. CC 서버는 사용자를 인증한 후, CCA-INITIAL을 CC 클라이언트에게 전달한다. 인증 과정에서 CC 서버는 사용자의 계정이 서비스를 제공할 수 있는지 확인한다. 필요 시 서비스의 비용 결정 절차 등을 수행한다. CCA-INITIAL에는 서비스 노드가 사용자에게 서비스를 제공해 줄 수 있는 서비스 사용량이나 시간이 포함되어 있다.

CCR-UPDATE는 최초 CC 권한검증이 성공한 후, CC 서버가 CCA-INITIAL에 설정한 서비스 허용 시간이 만료되거나 허용한 서비스 사용량을 모두 소모한 경우 사용자의 크레디트 정보를 갱신하기 위해 사용한다. 이때, CC 서버는 사용자 계정에서 서비스 금액을 차감하고, 계정이 충분한 경우 다시 사용자에게 서비스 사용량이나 시간을 할당한다. (CCA-UPDATE)

서버가 할당한 서비스 사용량이 사용자의 계정에서 마지막으로 허용된 양이거나, 사용자가 서비스를 종료한 경우 CC 클라이언트는 CC 서비스를 해제하기 위해 CCR-TERMINATION을 전송한다. CC 서버는 사용자의 계정으로부터 사용자가 사용한 서비스 사용량을 가감한 후 서비스를 종료한다. (CCA-TERMINATION)

CCR-EVENT는 일회성 이벤트로 세션을 유지할 필요가 없는 가격질의(Price Inquiry), 잔액확인(Check Balance), 직불요청(Direct Debiting), 환불요청(Refund) 등에 사용한다.

2. Mobile-IP 인증과 결합된 CC 권한검증 동작 절차

Mobile-IP 응용과 결합된 CC 권한검증방식은 다음의 두 가지 방식이 있다.

• Mobile-IP의 인증/권한검증이 완료된 후, CC 권한검증 수행
• Mobile-IP의 인증/권한검증 단계에서 CC 권한검증 수행

본 논문에서는, 전송 메시지의 개수를 줄일 수 있는 두 번째 방식을 이용하여 CC 권한검증을 수행한다. Mobile-IP의 인증/권한검증 단계에서 CC 권한검증을 수행하는 경우, 서비스 노드는 Mobile-IP의 등록요청 메시지에 CC 관련 AVP를 첨가한다. 첨가되는 AVP는 서비스 사업자 또는 서비스 별로 다를 수 있으며, 본 논문에서는 다음의 AVP는 공통적으로 첨가하는 것을 제안한다.

• Credit-Control AVP (AVP Code = 426)
Credit-Control AVP는 서비스 노드가 선불 서비스를 지원하는 경우 반드시 삽입되어야 하는 AVP이다. Mobile-IP 초기 인증/권한검증 시 AAAH (AAA Home : 홈망의 AAA 서버)는 사용자에 대한 선불 서비스 사용자 여부를 판단한다. 사용자가 선불 서비스 사용자인 경우 AAAH는 Credit-Control AVP의 존재 여부와 값을 확인한다. Credit-Control AVP의 존재 여부는 서비스 노드가 Diameter Credit-Control 응용을 지원할 수 있다는 것을 의미한다. 이 AVP의 값이 0(Credit-Authorization)으로 설정되어 있으면 최초의 인증/권한검증 요청이므로 AAAH는 CC 서버로 CC 권한검증 요청을 보낸다. 이 AVP의 값이 1(Re-Authorization)로 설정되어 있으면 Mobile-IP의 재인증이므로 AAAH는 CC 서버로 CC 권한검증요청을 보낼 필요는 없다. 만일 Credit-Control AVP가 존재하지 않으면, 서비스 노드가 Diameter Credit-Control을 지원하지 않는 경우이므로 Mobile-IP 인증/권한검증은 실패처리된다.
• CC-Request-Number AVP (AVP Code = 415)
CC-Request-Number AVP는 진행되고 있는 CC 세션에서 CCR를 구분하고, CCR과 CCA의 쌍을 맞추기 위해 반드시 필요한 AVP이다.
• CC-Request-Type AVP (AVP Code = 416)
CC-Request-Type AVP는 CCR의 종류를 명시하는 것으로 앞 절에서 설명한 INITIAL, UPDATE, TERMINATION, EVENT로 설정할 수 있다. Mobile-IP의 초기 인증/권한검증 메시지에 삽입되는 CC-Request-Type AVP는INITIAL로 설정한다.
• Requested-Service-Unit AVP (AVP Code = 437)

Requested-Service-Unit AVP는 CC 클라이언트가 요구하는 서비스의 양으로 단위는 금액이나 패킷, 시간 등으로 설정할 수 있으며 CC 서버에서 인증이 성공하는 경우 클라이언트가 요구한 서비스 양을 할당한다.

이 외에 사업자 별로, 서비스 비용 결정 등에 필요한 Service-Parameter-Info AVP (AVP Code = 440)나 Service-Identifier AVP (AVP Code = 439), Rating-Group AVP (AVP Code = 432) 등을 첨가할 수 있다.

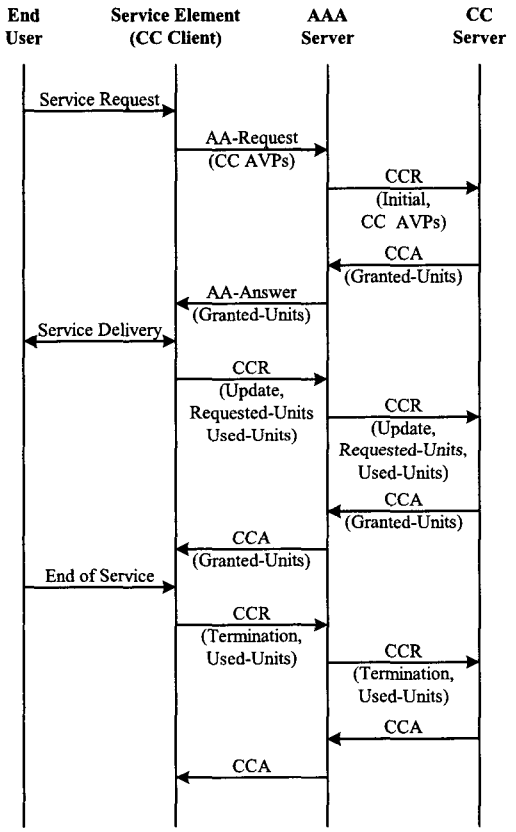

그림 2. Mobile-IP 응용의 인증/권한검증과 결합된 CC 권한검증 절차

그림 2는 AAA 서버와 CC 서버가 분리되어 있는 경우, Mobile-IP 인증/권한검증과 결합된 CC 권한검증 절차로 Mobile-IP의 인증/권한검증 단계에서 CC의 권한검증이 수행되는 방식을 나타낸 것이다. 그림에서 AA-Request/Answer 메시지는 Diameter Mobile-IP의 메시지로 Diameter Mobile-IPv4인 경우 AMR (AA-Mobile-Node Request)과 AMA (AA-Mobile-Node Answer) 메시지와 같다. 본 논문은 Mobile-IPv4와 Mobile-IPv6를 구분하지 않으므로 AA-Request/Answer 메시지로 표시하였다.

그림 2에서 서비스 노드와 AAA 서버 사이에 AAAF (AAA Foreign : 방문망의 AAA 서버)와 AAAB (AAA Broker : 중개망의 AAA 서버)가 존재할 수 있으나, CC과 관련하여 특별한 동작이 없으므로 그림에서 생략하였다.

1. 사용자가 서비스를 요구하면, 서비스 노드의 CC 클라이언트는 AA-Request 메시지에 앞에서 언급한 CC AVP들을 첨가하여 AAA 서버에게 전송한다.
2. AAA 서버는 사용자가 선불 서비스 사용자인지를 판단한 후 선불 서비스 사용자인 경우, AA-Request 메시지로부터 CC AVP들을 추출하여 CCR-INITIAL을 생성한다. AAA 서버는 생성한 CCR-INITIAL 메시지를 CC 서버에게 전송한다.
3. CC 서버는 권한검증을 수행한 후, CCA-INITIAL 메시지를 생성하여 AAA 서버에게 전송한다. 인증이 성공한 경우, CCA-INITIAL 메시지에는 CC-Request-Number AVP, CC-Request-Type AVP와 서비스 허용량을 표시하는 Granted-Service-Unit AVP (AVP Code = 431)와 서비스 만료 시간을 표시하는 Validity-Time AVP (AVP Code = 448) 등이 포함되며, 허용된 서비스 양이 사용자 계정의 마지막 허용량인 경우 Final-Unit-Indication AVP (AVP Code = 430)가 포함된다. 만일, 사용자가 선불 서비스 사용자가 아닌 경우, CCA-INITIAL의 Result-Code를 성공으로 설정한 후 Granted-Service-Unit AVP는 0으로 설정한다.
4. AAA 서버는 CC 서버로부터 CCA-INITIAL을 수신하면 CC-Request-Number AVP, CC-Request-Type AVP, Granted-Service-Unit AVP, Final-Unit-Indication AVP와 Validity-Time AVP 등을 추출하여 AA-Answer 메시지 내에 추가한 후서비스 노드로 전송한다.
5. 서비스 노드는 인증/권한검증이 성공한 경우 사용자에게 서비스를 제공한다. 만일, AA-Answer의 Result-Code가 성공이고 Granted-Service-Unit AVP가 0으로 설정되어 있으면 사용자는 선불 서비스 사용자가 아니므로 일반적인 Mobile-IP 응용의 서비스절차 및 과금절차를 따른다.

6. 서비스 노드에서 서비스 허용량을 모두 소비하거나 서비스 만료 시간이 초과하였을 경우에는 CCR-UPDATE 메시지를 전송한다.
   CCR-UPDATE 메시지에는 사용자가 사용한 서비스양(Used-Units)과 새로 요구하는 서비스양(Requested-Units)을 포함하며, CC 서버에서 CCR-UPDATE를 성공적으로 처리한 경우 새로 요구한 서비스양이 허용된다.
7. 사용자가 서비스를 종료하거나, 허용된 서비스양이 마지막인 경우 서비스 노드는 CCR-TERMINATION을 AAA 서버에게 전송한 후 CC 세션은 종료된다.

Ⅲ. Diameter Credit-Control의 구현

본 논문에서는 하나의 물리적 서버에 AAA 서버와 CC 서버를 구현하였다. AAA 서버는 Diameter Base 프로토콜 엔진(BPB: Base Protocol Block)과 Diameter Mobile-IP 응용(MAB: Mobile-IP Application Block) 및 Diameter CC 응용(CCB: Credit-Control Application Block)이 탑재되었다. 그림 3은 본 논문에서 제안하는 서비스 노드 및 AAA 서버의 구조를 표시한다.

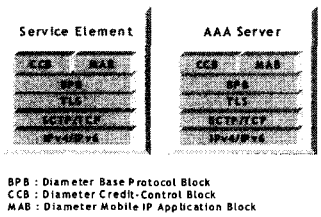

그림 3. 서비스 노드 및 AAA 서버 구조

하나의 서버에 AAA 서버와 CC 서버를 구현하는 경우, 그림 2에서 AAA 서버와 CC 서버간의 인터페이스는 내부 인터페이스로 변경된다. 내부 인터페이스는 추후 설명한다. Diameter Base 프로토콜 엔진 및 Mobile-IP 응용과 CC 서버는 독립된 프로세스로 동작하며 각 프로세스간 통신은 메시지 큐 방식을 사용하였다. 그림 2 의 서비스 노드에서 AA-Request 메시지에 CC AVP를 첨가하는 과정과 인증결과를 처리하는 과정은 그림 4와 같다.

1. 서비스 노드의 MAB는 초기의 인증/권한검증 요청 메시지인 AA-Request를 생성한 후, CCB에게 CC AVP의 첨가를 요청한다. 인증 방식이 EAP와 같이 멀티-라운드를 통한 인증을 이용하는 경우 이후

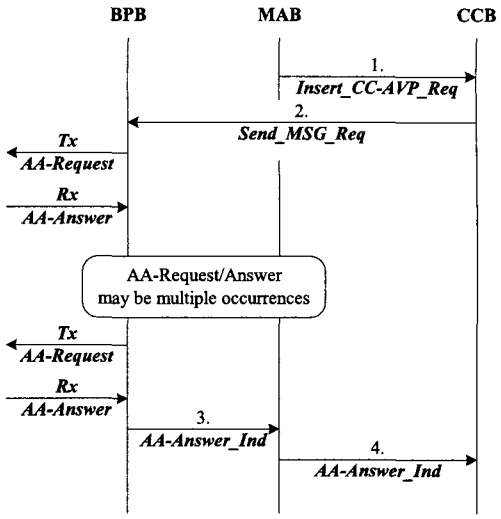

그림 4. 서비스 노드에서 초기 CC 권한검증 처리절차

의 인증 메시지에는 CC AVP를 첨가하지 않는다.
2. CCB는 AA-Request 메시지에 CC 권한검증에 필요한 AVP(Ⅱ.2절에서 설명)를 첨가한 후 BPB에게 메시지 전송요청을 한다. BPB는 AAA 서버로 CC AVP를 포함한 AA-Request 메시지를 전송한다.
3. AAA 서버로부터 AA-Answer 메시지를 받은 BPB는 MAB에게 해당 메시지를 전달한다.
   MAB는 인증의 최종 결과를 포함한 AA-Answer 메시지를 받기 전에는 CCB에게 인증 결과를 전달

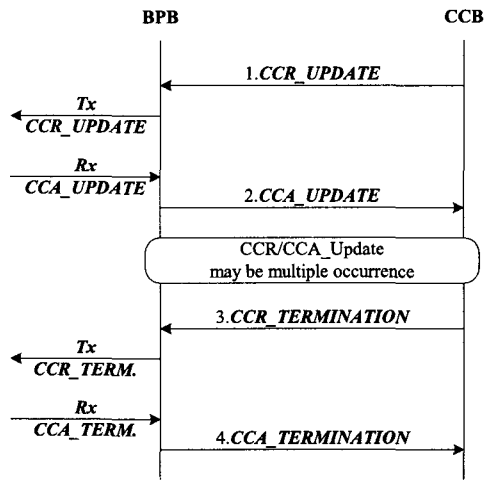

그림 5. 서비스 노드에서 CCR/CCA-UPDATE 및 CCR/CCA-TERMINATION 처리절차

하지 않는다.

4. MAB는 인증의 최종 결과를 포함한 AA-Answer 메시지를 CCB에게 전달하여 CC 권한검증의 결과를 알린다. 인증 결과가 성공인 경우, 서비스 노드는 사용자에게 서비스를 제공하며 실패인 경우에는 서비스를 제공하지 않는다.

그림 5는 CC 서비스가 개시된 이후, 서비스 노드에서 CCR/CCA-UPDATE 및 CCR/CCA-TERM_INATION 메시지가 처리되는 절차를 나타낸다. CCR-UPDATE/TERMINATION 메시지를 전송하는 방법은 Ⅱ.1절에서 설명하였다. 선불 서비스가 개시된 후 CCR-UPDATE/TERMINATION은 MAB를 거치지 않고 BPB-CCB간 직접 처리된다.

그림 6은 AAA 서버에서 CCR/CCA-INITIAL을 처리하는 과정을 보여준다.

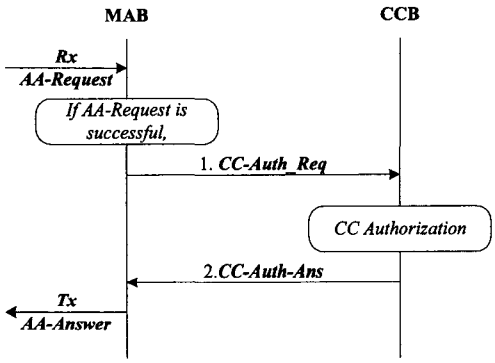

그림 6. AAA 서버에서 CCR/CCA-INITIAL 처리절차

1. AAA서버의 MAB는 Mobile-IP 인증/권한검증이 성공적으로 완료되면 CCB에게 CC 권한검증 요청을 한다. 만일 Mobile-IP 인증/권한검증이 실패하면 CCB에게 CC 권한검증 요청을 하지 않는다. CCB에게 권한검증을 요청하는 메시지에는 MAB가 생성한 AA-Answer 메시지와 AA-Request에 포함되어 있던 CC AVP가 포함된다.
2. CCB는 CC AVP를 이용하여 선불 서비스 권한검증을 수행한다. 권한검증이 성공한 경우 AA-Answer 메시지에 CC AVP를 첨가하여 MAB에게 전달하며, AA-Answer의 Result-Code AVP는 변경하지 않는다. 만일 선불 서비스 권한검증이 실패한 경우 AA-Answer 메시지의 Result-Code AVP를 적당한 값으로 설정한 후 MAB에게 전달한다. MAB는 AA-Answer를 CCB로부터 수신한 후 Result-Code AVP를 검증한다. 성공인 경우 BPB를 통해 서비스 노드로 AA-Answer 메시지를 전송하고, 실패인 경우 MAB는 실패처리를 한 후 BPB를 통해 서비스 노드로 전송한다.

그림 7은 AAA 서버에서 CCR/CCA-UPDATE 및 CCR/CCA-TERMINATION 메시지가 처리되는 절차를 나타낸다. 선불 서비스가 시작된 후 서비스 노드와 마찬가지로 CCR/CCA-UPDATE 및 TERMINATION 메시지의 처리는 MAB를 거치지 않고 BPB-CCB간 직접 처리된다.

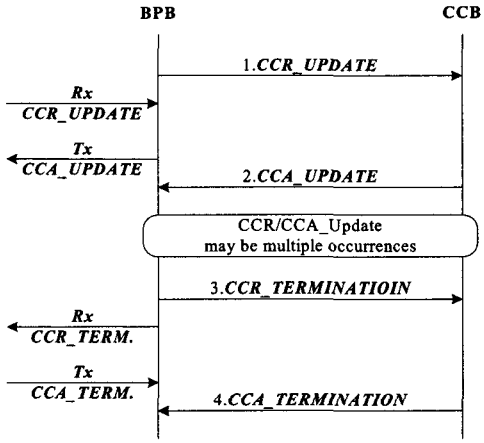

그림 7. AAA 서버에서 CCR/CCA-UPDATE 및 CCR/CCA-TERMINATION 처리절차

표 2는 AAA 서버에서 MAB-CCB간 인터페이스를 위한 메시지 구조체이다.

Evnt_MabTx_CcbRx_CCInit_Req_t는 CCB 에게 CC 권한검증을 요청하는 메시지 구조체 타입이며, Evnt_CcbTx_MabRx_CCInit_Ans_t는 이에 대한 응답의 구조체 타입이다.

표 2. MAB-CCB간 프로세스간 통신을 위한 메시지 구조 - AAA 서버

Evnt_MabTx_CcbRx_CCInit_Req_t
typedef struct { /* AA-Answer message */ u_c8 msgBuf[DIAM_MAX_MSG_LEN]; /* CC AVPs from AA-Request message */ T_CC_AVP credit_control_avp; T_CC_Req_Num_AVP cc_req_num_avp; T_CC_Req_Type_AVP cc_req_type_avp; T_Reqted_Svc_Unit_AVP reqted_svc_unit_avp; }Evnt_MabTx_CcbRx_CCInit_Req_t

```
Evnt_CcbTx_MabRx_CCInit_Ans_t: AAA server
typedef struct
{
  /* AA-Answer with CC-AVPs message */
  u_c8 msgBuf[DIAM_MAX_MSG_LEN];
  u_i32 status_code
}Evnt_CcbTx_MabRx_CCInit_Ans_t
```

표 3은 서비스 노드의 MAB-CCB간 인터페이스를 위한 메시지 구조체이다.

Evnt_MabTx_CcbRx_insertCC_Ind_t는 AA-Request 메시지 내에 CC AVP 첨가를 요청하기 위한 메시지 구조이다. 또한, Evnt_MabTx_CcbRx_AA_Result_Ind_t는 Mobile-IP 인증/권한검증 및 CC의 권한검증 결과를 알려주기 위한 메시지 구조체이다.

표 3. MAB-CCB간 프로세스간 통신을 위한 메시지 구조 서비스 노드

```
Evnt_MabTx_CcbRx_InsertCC_Ind_t
typedef struct
{
  /* AA-Request message */
  u_c8 msgBuf[DIAM_MAX_MSG_LEN];
} Evnt_MabTx_CcbRx_InsertCC_Ind_t;
```

```
Evnt_MabTx_CcbRx_AAResult_Ind_t
typedef struct
{
  /* AA-Answer with CC AVPs message */
  u_c8 msgBuf[DIAM_MAX_MSG_LEN];
} Evnt_MabTx_CcbRx_AAResult_Ind_t;
```

표 4는 BPB-CCB간 AA-Request 및 Diameter CC 메시지를 전송하기 위한 메시지 구조체이다.

표 4. BPB-CCB간 프로세스간 통신을 위한 메시지 구조 AAA 서버 및 서비스 노드

```
Evnt_CcbTx_BpbRx_TxMsg_Req_t
typedef struct
{
  /* AA-Request message */
  u_c8 msgBuf[DIAM_MAX_MSG_LEN];
} Evnt_CcbTx_BpbRx_TxMsg_Req_t
```

본 논문에서는 각 프로세스간 통신을 메시지 큐 방식을 사용하였다. 전체 시스템을 구현하여 동작시키는 경우, 각 블록이 동작하는 플랫폼 시스템에서 정의한 메시지 큐의 크기는 전체 시스템의 성능을 좌우한다. 성능시험을 하는 경우 서비스 노드에 시뮬레이터를 이용하여 다수의 서비스 요청을 동시에 전송하면, 플랫폼 시스템에서 정의된 메시지 큐의 크기는 전체 시스템에서 동시에 수용하여 처리할 수 있는 요청 메시지의 개수를 좌우한다.

만일 시스템에서 정의한 메시지 큐의 크기가 작아, 블록이 자신의 큐에 들어온 요청을 모두 처리하기 전에 블록의 메시지 큐에 메시지가 쌓여 메시지 큐가 모두 차버리면 전체 시스템이 중단되는 일이 발생한다. 예를 들어, EAP-TLS나 EAP-TTLS와 같이 TLS에 기반을 둔 인증방식을 채택하는 경우에 요청 메시지의 크기가 일반적인 경우보다 크기 때문에 메시지 큐의 크기는 전체 시스템의 성능에 크게 영향을 미치게 된다.

따라서 본 논문에서는 메시지 큐의 크기를 최대 크기를 정의한 후 사용하였으며, 동시에 블록별로 다수의 메시지 큐를 사용하는 방법을 이용하여 동시 수용할 수 있는 메시지 수를 최대한으로 설정하여 전체 시스템의 처리성능을 향상 시키는 작업을 수행하였다.

Ⅳ. 결 론

본 논문은 Diameter에 기반을 둔 선불 서비스의 기능 및 구조를 정의하고 이에 따른 구현에 대해 기술하였다. 선불 서비스는 이미 GSM 네트워크에서 매우 성공적인 지불방식으로 받아들여지고 있으며, 가입자와 시장이 계속 성장하고 있는 추세이다.

따라서 차세대에 등장할 네트워크 환경에 적용 가능한 선불 서비스의 설계 및 구현은 매우 유용한 일이라 할 수 있다. 또한 Diameter 프로토콜은 유. 무선 네트워크에서 모두 적용 가능한 AAA 프로토콜이며 차세대에 등장할 많은 네트워크 서비스에서 표준 AAA 프로토콜로 채택되고 있는 상황이다.

본 논문은 현재 Diameter에서 정의하고 있는 Mobile-IP 응용에 적용 가능한 선불 서비스 기능을 구현했지만, EAP 응용이나 NASREQ 응용과도 연동 가능한 구조이므로 Diameter를 적용하는 모든 분야에서 기존의 Diameter 과금 프로토콜이 지원하지 않은 선불 서비스 기능을 제공하는 장점이 있다.

참 고 문 헌

[1] Harri Hakala, Leena Mattila, Juha-Pekka Koskinen, Marco Stura, John Loughney, "Diameter CC Application", IETF work in progress

[2] P. Calhoun, J. Loughney, J. Arkko, E. Guttman, G. Zorn. "Diameter Base Protocol", RFC 3588, September 2003.

[3] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects, Service aspects; Charging and Billing, (release 5), 3GPP TS 22.115 v. 5.2.1, 2002-03

[4] C. Rigney. "Radius Accounting", RFC 2866, June 2000

[5] P. Calhoun, T. Johansson, C. Perkins "Diameter Mobile IP Application", IETF work in progress.

[6] P. Eronen, T. Hiller, G. Zorn. "Diameter Extensible Authentication Protocol (EAP) Application", IETF work in progress.

[7] P. Calhoun, G. Zorn, D. Spence, D. Mitton. "Diameter NASREQ Application", IETF work in progress.

유 상 근(Sangkeun Yoo) 정회원

1997년 2월 : 충남대학교 컴퓨터공학과 졸업

1999년 2월 : 충남대학교 컴퓨터공학과 석사

1999년~2000년 : 씨그마테크(주) 근무

2001년~현재 : 한국전자통신연구원 정보보호연구단 근무

<관심분야> 정보보호시스템, RFID 프라이버시 보호, 전자화폐 및 전자지불 시스템

김 현 곤(Hyungon Kim) 정회원

1992년 금오공과대학교 전자공학과 졸업

1994년 금오공과대학교 전자공학과 석사

2003년 충남대학교 전자공학과 박사

1994년~현재 : 한국전자통신연구원 정보보호연구단 AAA정보보호연구팀장

<관심분야> IP 기반의 이동통신 정보보호, RFID/USN 정보보호