情報保護學會論文誌
第13券 第5號, 2003. 10

MIP 환경에서의 Diameter Security Association 정보 분실시의 재설정 기법

유희종*, 김말희**, 김현곤***

Efficient Re-Establishment Mechanism of Diameter Security Association lost in MIP Application

Hui-jong Yu*, Mal-hee Kim**, Hyun-gon Kim***

요 약

Diameter 프로토콜에서는 송수신되는 메시지가 아무런 문제없이 중간 경유 노드에서 변경, 삭제, 추가될 수 있다. Diameter CMS Security Application은 공개키 기반 구조를 이용하여 Diameter 응용에서 종단간 보안 기능을 제공함으로서 이러한 문제점을 해결하였다. Diameter CMS Security Application은 두 종단 노드 사이에 DSA(Diameter Security Association)을 설정하여 이후의 통신의 안전성을 보장하도록 하고 있다. 그러나 시스템 장애 등의 이유로 한 노드가 DSA 정보를 분실하였을 경우, 이 Application을 사용한 두 노드 사이의 안전한 통신에는 문제가 발생하게 된다. MIP(Mobile IP)와 같은 응용에서는 이러한 문제로 사용자 등록이 불가능하여 서비스가 이루어지지 못할 수도 있다. 따라서 본 논문에서는 이러한 경우 DSA를 재설정 할 수 있는 방법을 제시하고 이 해결책을 실제 구현한 결과를 보인다.

ABSTRACT

AAA(Authentication, Authorization, Accounting) protocol is a framework that propose functions of AAA on multiple networks and platforms. AAA protocol is extending from previous RADIUS protocol to Diameter protocol. There are some Diameter applications for variety purpose. Diameter CMS Application makes Diameter messages more secure by using PKI. Diameter CMS Application establish DSA(Diameter Security Association) for end to end security. However the Application has some problems to establish DSA(Diameter Security Association), which can make Diameter system unstable. If one system lose DSA information for some system error - for example, reboot -, the secure communication between two nodes may not be possible. At the application such as MIP, even user registration can't be done. In this paper, we propose a mechanism for DSA re-establishment, and also show the result of the implementation.

keyword : *AAA, Diameter, MIP*

1. 서 론

랩탑, 노트북, 이동전화와 같은 장비들이 사무실 밖에서 이동 중에도 사용할 수 있도록 소형화되었다. 그러나 크기와 이동중의 사용 뿐 아니라 안전성도 제공되어야 한다. 또한 사용자의 수가 급격히 늘어나면서 발생하는 문제점들을 해결해야 한다. 불법적으로 서비스를 사용하는 것을 방지해야 하고, 가입자의

* 한국전자통신연구원 AAA정보보호연구팀 유희종(anny5@etri.re.kr)
** 한국전자통신연구원 AAA정보보호연구팀 김말희(mariekim@etri.re.kr)
*** 한국전자통신연구원 AAA정보보호연구팀 김현곤(hyungonkim@etri.re.kr)

권한 레벨을 부여하고 검증해야 하며, 과금 및 자원 계획을 수립하기 위해 네트웍 사용에 대한 측정이 필요하다.

AAA(Authentication, Authorization, Accounting) 프로토콜은 다중 네트워크와 플랫폼 상에서 인증(Authentication), 권한검증(Authorization), 과금(Accounting) 등의 기능들을 조정하는 프레임웍으로서 위의 여러 문제점들을 해결할 수 있는 방법이다. AAA 프로토콜의 기능은 다음과 같다.

• 인증(Authentication)
망 접근을 허용하기 전 사용자의 신원을 검증하는 것이다. AAA 서버는 사용자가 제공한 인증 데이터와 자신의 데이베이스 안의 사용자 관련 데이터를 비교하여 인증서가 일치하면 망에 대한 접근을 허락한다. 만일 일치하지 않으면 인증실패로 인해 망 자원 사용을 허용하지 않는다.

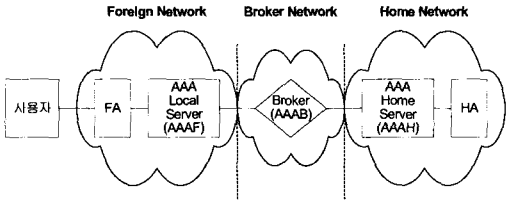
• 권한검증(Authorization)
망 사용이 허락된 사용자에 대해 어떤 권한과 서비스를 허용할 것인지를 정하는 것이다. 여기에는 IP 주소, 제공될 응용 및 프로토콜을 결정하기 위한 필터 등이 포함된다. 인증과 권한검증은 AAA 동작 환경에서 일반적으로 함께 수행된다.

• 과금(Accounting)
사용자의 자원 사용에 관한 정보를 모으는 방법을 제공한다. 그리고 이 정보는 사용요금, 회계 그리고 용량 증설에 사용된다.

일반적으로 PPP(Point-to-Point Protocol)나 터미널 서버 액세스와 같은 서비스를 위한 AAA 프로토콜로 RADIUS(Remote Authentication Dial-In User Service) 프로토콜이 사용되어 왔다. 그러나, 급격하게 증가하는 네트웍 환경 하에서 RADIUS는 Scalability와 Security, 프로토콜의 기술적인 한계 등에서 AAA 서비스를 위한 프로토콜로 부적합한 것으로 밝혀지고 있다.

Diameter는 이러한 환경에서 RADIUS를 대체할 수 있는 AAA 서비스 프로토콜로서 RADIUS가 가지는 많은 단점을 개선하여 이동인터넷 환경에서 AAA 서비스에 적합한 프로토콜로 현재 규격 정의 작업이 진행 중이다.

이러한 Diameter 프로토콜은 기본 프로토콜 이외에 여러 Application들을 통하여 다양한 서비스를 제공 할 수 있으며 다음의 Application들이 지원된다.



[그림 1] MIP를 위한 AAA 망 참조 모델

- Diameter Mobile IP Application[2]
- Diameter CMS(Cryptographic Message Syntax) Security application[1]
- Diameter NASREQ(Network Access Server REQiurement) Application[4]
- Diameter Base Protocol[3]

위 응용 중 본 논문에서는 MIP 응용(Diameter Mobile IP Application)을 위한 Diameter 프로토콜에 기밀성 및 무결성,부인 봉쇄 등의 보안 기능을 제공하는 Diameter CMS Security application의 실제 구현 결과를 제시하며 구현시 발생하는 문제점에 대하여 분석하고 해결책을 제시한 결과를 보인다.

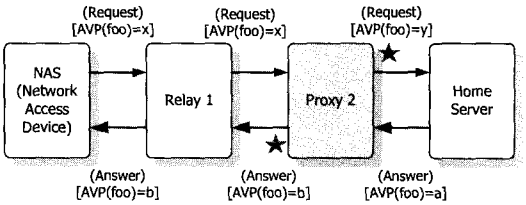
MIP를 위한 AAA 망 참조 모델은 [그림 1]과 같다. 이에 관한 자세한 설명은 4.1절을 참조한다.

II. Diameter CMS Security Application

Diameter 메시지는 AVP(Attribute-Value-Pair)라는 단위로 구성되어 전송되며 Diameter 기본 프로토콜은 이러한 Diameter 메시지가 지나는 모든 노드들에서 기본적으로 IPsec이나 TLS 보안 프로토콜을 제공하도록 한다. AVP는 일반적인 통신 프로토콜의 TLV (Type-Length-Value) 개념과 비슷한 것으로 AVP 종류와 길이 및 값으로 구성되어 있다.

TLS와 IPsec은 경유 노드에서의 보안을 제공한다. 그러나 [그림 2]는 악의적인 목적을 가진 메시지의 경유 노드(proxy)에서 AVP들이 변조됨을 보인다. 즉, 각 경유 노드 사이에서의 무결성, 기밀성만이 보장될 뿐 송수신 종단 노드의 안전성은 보장되지 않는다. 따라서 Diameter 기본 프로토콜은 CMS 응용을 이용하여 이 문제를 해결하였다.

본 절에서는 이러한 Diameter CMS 응용에 대하여 설명한다. Diameter CMS 응용은 비대칭 암호 방식을



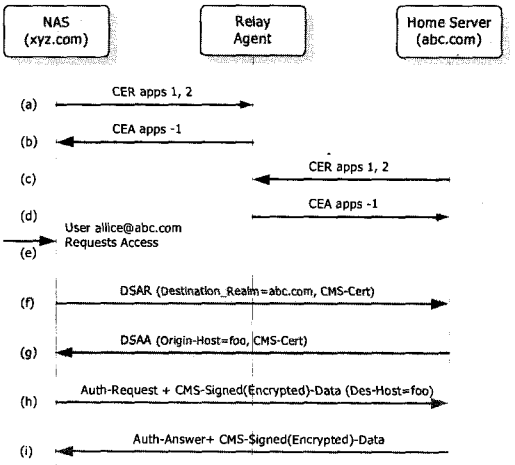
〔그림 2〕 AVP를 변경하는 Diameter 노드

사용하여 Diameter 종단 노드간에 Diameter 메시지들의 무결성, 기밀성, 인증 기능을 제공하도록 하는 응용이다. CMS 응용은 크게 DSA(Diameter Security Association) 설정 과정, 암호화/복호화 및 서명/검증의 두 과정으로 나눌 수 있다.

2.1 DSA 설정 과정

[그림 3]에 DSA Diameter 노드간 중계 에이전트가 존재하는 환경에서 DSA 설정 과정을 포함하여 Diameter CMS 메시지 흐름 예를 보였다. NAS가 xyz.com 도메인내 자신의 로컬 중계 에이전트를 통해 abc.com에 있는 서버와 통신하는 예이다. 이 예에서는 NASREQ 환경의 경우를 가정하였다. MIP 환경이라면 NAS는 FA가 될 수 있다. 여기서 초기에 Capabilities Exchange(CE)가 이루어진 이후, NAS는 alice@abc.com으로부터 액세스 요청을 받는다. 이 때 application-s pecific 인증 요청에 의해 DSA 설정이 시작된다.

(a) NAS는 자신의 중계 에이전트에게 CER(CE Request) 메시지를 송신하고, 응용 1(NASREQ)과 응용 2(CMS Security)를 지원한다는 것을 알린다.
(b) 중계 에이전트가 CEA(CE Answer) 메시지를 NAS에게 전송하고 모든 Diameter 응용들을 지원한다는 것을 알린다.
(c) abc.com의 홈 서버는 CER 메시지를 중계 에이전트에게 송신하고, 응용 1과 응용 2를 지원한다는 것을 알린다.
(d) 중계 에이전트는 CEA 메시지를 abc.com의 홈 서버에게 송신하고, 모든 Diameter 응용들을 지원한다는 것을 알린다.
(e) NAS는 사용자에게 접근 요청을 받는다.
(f) NAS는 Destination-Realm AVP에는 abc.com로 지정하여 DSAR(Diameter Security Association Request) 메시지를 전송하며, CMS-Cert AVP에 자신의 인증서를 포함시키고 필요한 경우 AVP들을 서명하여 CMS-Signed-Data AVP를 포함시킨다.



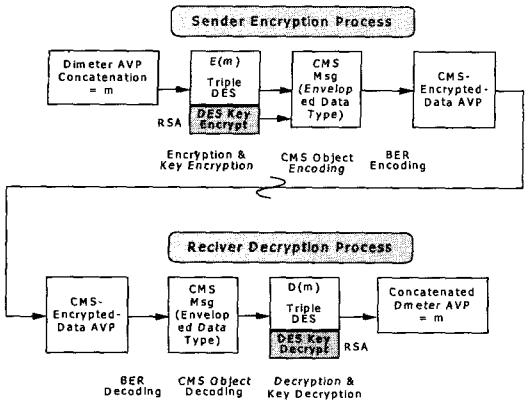
〔그림 3〕 Diameter CMS 메시지 흐름

(g) abc.com의 홈 서버는 DSAR 메시지를 처리하고 DSAA(Diameter Security Association Answer)메시지로 응답한다. DSAA 또한 CMS-Cert AVP에 자신의 인증서를 포함시키고 필요한 경우 AVP들을 서명하여 CMS-Signed-Data AVP를 포함시킨다.
(h) NAS는 DSAA내의 Destination-Host AVP를 Origin-Host AVP의 값으로 지정하고, 인증 요청을 한다. 이 경우 설정된 DSA를 사용하여 CMS-Signed-Data AVP, CMS-Encryped-Data AVP가 포함될 수 있다.
(i) 홈 서버는 성공적으로 사용자를 인증하면 응답을 리턴한다. 이 경우 설정된 DSA를 사용하여 CMS-Signed-Data AVP, CMS-Encryped-Data AVP가 포함될 수 있다.

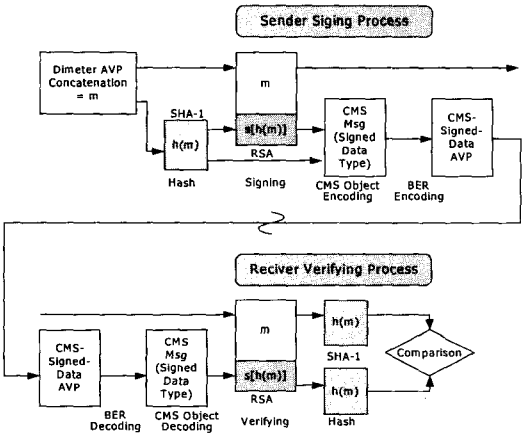
위 과정 중 서명과 암호화의 자세한 과정은 다음 절에 설명된다.

2.2 암호화/복호화 및 서명/검증 과정

[그림 4]에 나타낸 암호화 및 복호화는 Diameter 메시지의 기밀성을 제공한다. 로컬 정책 및 Diameter 규격에서 암호화를 추천하고 있는 AVP의 예를 들면, MIP의 동적 세션 키들을 들 수 있다. 암호화 절차는 다음과 같다. 암호화할 AVP들을 Concatenation시키고, 원문을 Triple DES로 암호화하고 DES Key를 RSA로 암호화 한다. 암호화된 값을 [5]에 기술된 CMS의 EnvelopedData Type에 포함시키고 BER 엔코딩하여 CMS-Encrypted-Data AVP를 최종적으로 생성한다. 이



〔그림 4〕 Diameter AVP들의 암호화 및 복호화



〔그림 5〕 Diameter AVP들의 서명 및 검증

AVP는 다른 AVP들과 동일하게 Diameter 메시지에 패킹되어 수신측에 전달된다. 수신 노드는 복호화를 수행한다. CMS-Encrypted-Data AVP에서 원래의 EnvelopedData Type을 추출한다. 그리고 Triple DES로 복호화하여 원문을 추출하고, RSA로 복호화하여 DES 키를 추출한다. DES 키는 실제 Concatenated된 AVP의 복호화에 사용된다. 어떤 AVP들은 암호화와 서명이 동시에 이루어질 수 있다. 이 경우, 암호화를 먼저 수행하고, 그 결과 값을 포함한 CMS-Encrypted-Data AVP가 하나의 AVP로서 서명된다.

[그림 5]에 나타낸 디지털 서명 및 검증은 인증, 무결성, 부인 봉쇄 기능을 제공한다. 두 사업자간 양방향 승인이 필요한 과금 정보와 같은 중요한 데이터들이 서명되며, 서명이 필요한 AVP는 Diameter 헤더에 'P' 비트가 셋팅된다. 서명 절차는 다음과 같다. 송신 노드는 서명할 AVP들을 Concatenation시키고, 원문을 해쉬함수 SHA-1으로 압축(Digest)한다. 그리고 이 결과 값을 RSA로 서명한다. 압축된 값과 서명문을 [1]에 기술된 CMS의 SignedData Type에 포함시키고, BER 엔코딩하여 CMS-Signed-Data AVP를 최종적으로 생성한다. 이 AVP는 다른 AVP들과 동일하게 Diameter 메시지에 패킹되어 수신측에 전달된다.

서명문의 수신자인 수신 노드는 검증을 수행한다. CMS-Signed-Data AVP에서 원래의 SignedData Type을 추출한다. 그리고 RSA 알고리즘을 통해 검증한다. 검증을 통해 나온 값을 SHA-1으로 해쉬하고, 다시 원래의 메시지를 해쉬하여 두 값을 비교한다. 비교한 결과가 같으면 검증은 성공한 것이므로 이후 Diameter 메시지 처리를 계속 수행한다. 비교 결과가 다르면 해당 Diameter 메시지는 폐기한다.

2.3 알고리즘

Diameter CMS 보안 응용에서는 아래의 알고리즘을 사용한다. 추후 Triple DES는 AES로 대치될 것이다.

- Asymmetric : RSA
- Hashing : SHA-1
- Signature : RSAwithSHA-1
- Symmetric Encryption : Triple DES

Ⅲ. MIP 기반 AAA 시스템의 구현

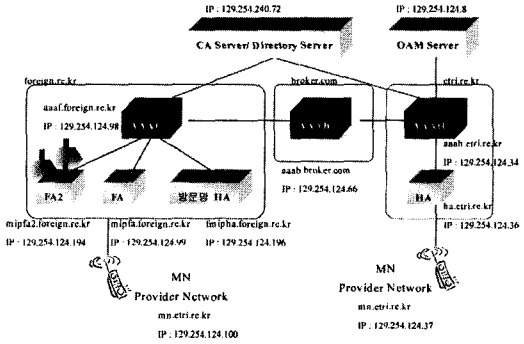
본 장에서는 망 참조 모델의 일반적인 서브 시스템 구조와 Diameter CMS Security 응용을 실제 구현한 내부 블록 구조를 제시한다.

[그림 6]은 MIP 기반 AAA 시스템의 실제 구현 환경이다.

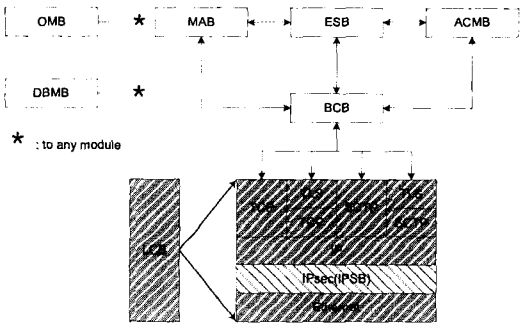
각 시스템의 구현 환경은 다음과 같다.

- AAA 서버 및 클라이언트 플랫폼(AAAH, AAAF, AAAB) : 자이온 리눅스 시스템 Focus 510
- AAA 서버 및 클라이언트 운영체계(HA, FA) : Red Hot Linux 6.2(Kernel v. 2.2.x)
- 단말 플랫폼(MN) : 리눅스 탑재 타이닉스 박스(이동 단말 개발용 Embedded Linux 장치)
- 기타 : 블록간 통신은 IPC 메시지 큐를 사용

[그림 7]은 실제 구현한 Diameter 서브 시스템의 일반적인 구조이다.



〔그림 6〕 MIP 환경 AAA 시스템의 실제 구현 환경



〔그림 7〕 AAA 서브 시스템의 논리적 구조

BCB(Base Control Block) 블록은 Diameter 기본 프로토콜에 해당되는 블록으로 전송 계층을 연결할 때 능력 정보 교환 기능을 제공하며, 연결 설정이 끝난 후에 Command 처리 및 전달, 오류 신호 보고, peer connection 관리, 라우팅 서비스, 세션 관리의 기능을 제공한다.[3] 또한 ACMB(Accounting Management Block)는 기본 프로토콜 문서에 포함되어 있으나 과금 기능을 수행하는 개별 블록으로 구현하였다.

ESB(End-to-End Security Block) 블록은 PKI를 이용하여 Diameter 프로토콜 메시지에 기밀성, 인증, 무결성, 부인봉쇄 등의 보안 기능을 제공하는 블록으로 Diameter CMS Security 응용의 실제 구현 블록이다. DBMB(Database Management Block) 블록은 사용자 인증 및 과금을 위한 가입자 정보와 과금 정보, 초기화 파라미터, 세션 정보, 라우팅 정보, Mobile IP 세션 키 정보 등을 저장한 저장소를 관리하는 기능을 담당한다. IPSB(IPsec Block) 블록은 Diameter Client와 Diameter Server사이의 안전한 AVP전송을 위한 IP Security를 제공한다.

LCB(Low-layer Communication Block) 블록은 전송 프로토콜인 SCTP나 TCP를 이용하여 연결을 설정하는 기능 및 설정된 연결을 통하여 Diameter AVP 메시지를 송신 및 수신하는 기능을 제공한다

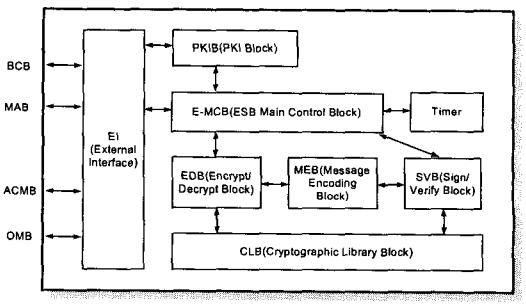
MAB(Mobile IPv4 Application Block) 블록은 Mobile IP 응용에서 이동 단말의 인증 기능을 제공한다.[2]

본 구현에서 OMB(O&M Block) 블록은 ACMB, BCB, MAB, ESB의 블록의 운용에 필요한 초기화 정보, 가입자 프로파일, 과금 정보, 보안 정책, Mobile IP 세션 키 등의 정보관리를 수행 하는 정보관리 기능을 제공한다.

3.1 Diameter CMS Security 응용의 구현 구조

본 절에서는 Diameter CMS Security 응용을 실제 구현한 내부 블록 구조를 제시하며 이는 [그림 6]에서 ESB 블록에 해당된다. 그 구조는 [그림 8]과 같다.

E-MCB(ESB-Main Control Block) 블록은 ESB 블록을 제어하고 관리하는 기능을 한다. PKIB(PKI Block) 블록은 공개키 기반 구조와 관련된 기능을 수행하는 블록이다. EI(External Interface) 블록은 외부 블록과의 인터페이스를 담당한다. Timer는 DSA(Diameter Security Association)의 유효 시간인 DSA-TTL(DSA-Time To Live), 인증서의 유효 기간 등을 측정하는 기능을 담당한다. 또한 DSA 설정 요청 메시지를 송신한 후 일정 시간 동안 응답 메시지가 수신되지 않을 경우 다른 블록들에게 알리는 경우에도 사용된다. EDB (Encrypt/Decrypt Block) 블록은 AVP들의 암호화/복호화시 E-MCB 블록에 의해 호출되는 블록이다. SVC (Sign/Verify Block) 블록은 AVP들의 서명/검증시 E-MCB 블록에 의해 호출되는 블록이다. MEB(Message Encoding Block) 블록은 암호화/복호화, 서명/검증된 AVP들의 BER/DER 인코딩을 담당한다.[5] CLB(Cryptographic Library Block) 블록은 ESB 블록에서 사용되는 기반 암호 라이브러리 함수이다.



〔그림 8〕 ESB 블록 내부 구조

3.2 구현 내용

현재 Diameter CMS Security 응용이 Draft 상태이므로 다양한 문제점들이 존재한다. 따라서 본 절에서는 이러한 문제점들을 해결하여 실제 구현한 ESB 블록에 대하여 기술한다.

3.2.1 블록 초기화

ESB 블록은 초기 구동될 때에 OMB에게 초기화 데이터를 받아온다. 이 초기화 데이터에는 DSA-TTL, 인증서 경로 정보, 암호화/서명이 요구되는 데이터 등이 포함되어 있다. 초기화 데이터를 수신한 ESB는 초기화 데이터를 검증하고 로컬에 저장된 인증서를 읽어 초기화를 수행한다.

Diameter 프로토콜에서 CMS 응용은 다른 실제 응용 서비스를 안전하게 하기 위한 장치이다. 따라서 다른 응용 서비스 모듈에 최소한의 영향을 주는 동시에 안전함을 보장하는 것이 효율적인 Diameter 프로토콜을 위한 CMS 응용의 설계의 목적이다. 따라서 본 구현에서는 CMS에서 제공하는 안전성을 효율적으로 보장하는 동시에 다른 블록들이 CMS 응용에서 사용되는 정보들을 최소한으로 관리하도록 구현하였다. 또한 ESB 블록은 다른 응용 서비스를 안전하게 하기 위한 보조적인 블록으로 본 구현에서는 다른 응용에 최소한의 영향을 주도록 하기 위해 라이브러리 형태와 같이 능동적인 동작보다는 다른 응용 블록의 영향 하에 수동적으로 동작하도록 구현하였다.

3.2.2 DSA 설정

Diameter 노드가 Diameter CMS Security 응용을 사용하기 위해서는 먼저 DSA를 설정해야 한다. DSA는 PKI를 이용하여 두 노드 사이에 비밀 정보를 공유하기 위해 설정하는 보안 연관이다.

DSAR(Diameter Security Association Request) 메시지는 DSA 설정 요청 메시지로 송신자의 인증서, DSA-TTL, OCSP 지원 요청 등의 정보가 포함된다. DSAA(Diameter Security Association Answer) 메시지는 응답 메시지로 DSAA 송신자의 인증서 등의 정보를 포함한다.

Diameter CMS Security 응용에는 DSAR, DSAA 메시지에 포함되는 인증서 필드에 해당 노드의 동일 도메인 내의 노드들의 인증서를 모두 포함하도록 되어있다. 인증서 하나의 크기를 대략 700k 정도로 가정할 때 이는 2개 이상이면 1메가가 훨씬 넘는 DSA 메시지가 송수신되게 된다. Diameter 메시지는 실제 서비스를 위한 신호 메시지이다. 신호 메시지는 그 크기가 크면 실제 프로토콜에 부하를 주게 된다. 따라서 Draft에서는 동일 도메인 내의 모든 노드들이 동일한 인증서를 사용하도록 권고하고 있다. 그러나 DSA는 종단 노드 사이의 보안 설정이기 때문에 사실상 DSA 메시지에 동일 도메인의 모든 노드의 인증서가 포함될 필요가 없다. 즉, DSA정보는 DSA를 설정한 두 노드만 가지고 있기 때문에 다른 노드의 인증서를 가지고 있다 하더라도 다른 노드와 통신하기 위해서는 새로운 DSA를 맺어야 한다. 따라서 본 구현에서는 DSA 설정 메시지를 송신할 경우 인증서 개수를 1개로 한정하고 해당 노드의 인증서만 포함시킴으로써 DSA 설정 메시지의 크기를 최소화하였다. 그러나 다른 Diameter 시스템과의 호환을 위해 다수의 인증서 수신은 가능하도록 구현하였다.

3.2.3 메시지 암/복호화, 서명/검증

ESB 블록은 초기 구동시 OMB 블록으로부터 암호화, 서명될 AVP들을 수신하여 그 정보를 유지한다. AVP는 Diameter 메시지를 구성하는 단위이다. DSA가 설정된 후 다른 응용 블록들(MAB, BCB, ACMB, NASB)은 송신할 Diameter 메시지를 ESB에게 보내 암호화/서명 요청, 복호화/검증 요청을 한다. 이 과정에서 타 블록들은 어떤 AVP에 ESB가 적용되는지 상관하지 않는다. ESB는 이 메시지를 처리하여 역시 Diameter 메시지 단위로 각 응용 블록에게 넘겨준다.

Diameter CMS Security 응용은 PKI를 이용하기 때문에 공개키 암호 방식을 지원한다. 그러나 공개키 암호 방식은 비밀키 암호 방식에 비해서 속도가 느리기 때문에 키 재사용 기법을 선택 사항으로 권고한다. 키 재사용 기법은 공개키 암호 방식을 이용하여 키가 공유되면 이후의 통신은 비밀키 암호 방식을 이용하여 암/복호화하는 방법이다.[6] 본 구현에서는 키 재사용 기법을 지원하였다. Diameter CMS Security 응용에서는 키 재사용 기법을 지원하지 않는 노드가 키 재사용 기법이 지원된 메시지를 수신할 경우 암호학적 오류로 정하고 있다. 그러나 이 경우 실제의 암호학적 오류와 구분이 되지 않는 문제가 발생한다. 따라서 본 구현에서는 키 재사용 기법을 지원하여 메시지를 보냈을 경우 응답 메시지에 암호학적 오류가 발생했다면 이후의 메시지에는 키 재사용 기법을 적용하지 않고 송신하도록 구현하였다.

본 구현은 Diameter CMS Security 응용 Draft를 따

라 2.3절에서 기술된 알고리즘을 기반으로 Diameter 시스템을 구현하였다.

Ⅳ. MIP 환경에서의 효율적인 DSA 재설정

4.1 MIP 사용자 등록 과정

MIP는 Mobile 노드가 홈망에서 타망으로의 이동 시 사용자의 서비스를 끊임없이 받는 것이 가능하도록 하는 응용 서비스이다.

MIP 환경에서 AAA를 이용한 사용자 등록 과정은 [그림 9]과 같다. MN(Mobile Node)이 Foreign Network으로 이동했을 때 하부 연결 설정 과정과 DSA 설정 과정 이후가 실제적인 MIP 등록 과정이다. MIP 등록 메시지는 설정된 DSA를 이용하여 인증과 기밀성을 제공받는다.

사용자가 등록 요청을 하면 FA(Foreign Agnet)는 AAAF(Foreign 도메인의 AAA 서버)에게 등록 요청 메시지(AMR:AA Mobile Node Request)를 전송한다. AAAF는 사용자의 홈 도메인과 DSA가 설정되었는지 검사한다. DSA가 설정되어 있지 않다면 2.1절에 따라 DSA를 설정하고 AMR을 수신한 AAAH는 HA(Home Agent)와 HAR(Home Agent MIP Request), HAA(Home Agent MIP Answer) 메시지를 주고받아 사용자의 정보를 확인한다. 사용자의 등록 요청이 승인되면 AAAH (Home 도메인의 AAA 서버)는 FA에게 AMA(AMR: AA Mobile Node Answer)를 전송한다. 이 메시지가 FA까지 전송되고 결과가 성공이면 사용자 등록 과정이 완료된다. 자세한 과정은 [2]에 나타나 있다.

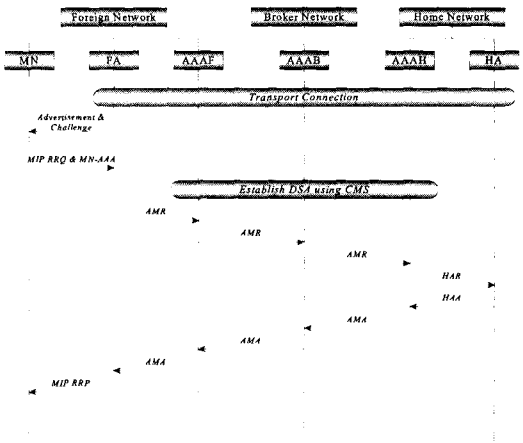
사용자가 위의 과정을 통하여 등록을 완료하면 서비스를 제공받을 수 있다. 이 사용자와 같은 망의 사용자가 추후에 등록 요청을 하였을 경우에는 연결 설정과 DSA 설정 과정이 다시 요구되지 않는다. 기존의 설정을 이용한 등록 과정만이 필요하다.

4.2 DSA 정보 유실시의 사용자 등록

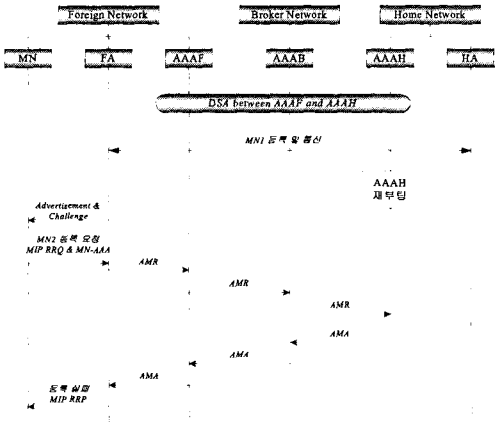
Diameter CMS 응용에서는 키 정보 등 DSA 정보가 시스템 재부팅 등 어떤 이유로 인하여 유실되었을 경우에 대하여 기술하고 있다. 이 경우 에러를 반환하거나 DSA를 다시 맺을 수도 있다고 기술하고 있다. 그러나 확실한 해결책을 제시하지 않고 미해결 문제로 남겨두고 있다. Diameter 노드들이 DSA를 설정하여 통신하던 중 한 노드의 시스템에 문제가 발생하여 DSA 정보를 잃게 된다면 시스템 이중화 등의 물리적인 해결책 없이 DSA를 사용한 두 노드 사이의 통신은 불가능하다. 본 경우에서는 AAAH 혹은 AAAF가 재부팅 되었을 경우 DSA 정보의 유실 상황을 구현하였다.

4.2.1 AAAH의 재부팅

[그림 10]과 같이 MN1이 DSA를 사용한 등록을 완료하고 서비스를 받은 이후 AAAH가 재부팅되어 DSA 정보를 유실하였을 경우 MN2의 등록 과정은 실패하게 된다. ESB 블록은 다른 블록에 최소한의 영향을 주기 위해서 수동적으로 동작하기 때문에 DSA 설정이나 암복호화 동작시 다른 블록의 요구에 응답하는 형태로 구현되어 있다. 따라서 DSA 설정 또한 타 블록의 필요에 의한 요청이 요구된다. AAAH가 리부팅 되었으므로 등록 요청 메시지는 등록 실패를 포함한



〔그림 9〕 MIP 등록 과정

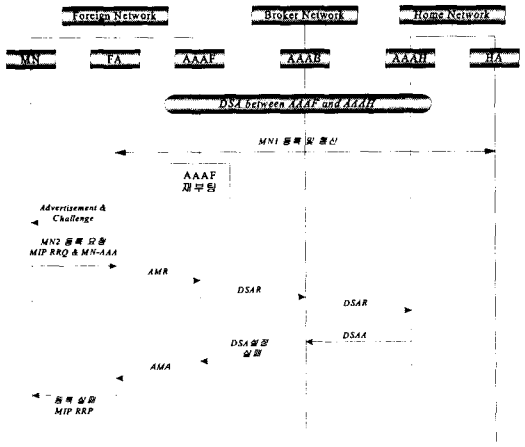


〔그림 10〕 AAAH에서의 DSA 정보 유실

응답 메시지를 생성시키며 이 메시지의 Origin-State-ID-AVP는 1 증가되게 된다. Origin-State-ID-AVP는 노드의 재부팅시 1 증가되는 값으로 BCB 블록에서 생성, 분석되는 AVP이다. 따라서 본 구현에서 기존의 값보다 1 증가된 값을 수신한 AAAF의 BCB 블록은 ESB 블록에게 DSA의 재설정을 요청한다. 이 요청에 따라 AAAF의 ESB 블록은 AAAH와의 DSA를 재설정 하도록 구현하였다. 또한 DOS 공격의 방지를 위하여 Origin-State-ID-AVP는 항상 서명되도록 한다.

4.2.2 AAAF의 재부팅

[그림 11]과 같이 MN1이 DSA를 사용한 등록을 완료하고 서비스를 받은 이후 AAAF가 재부팅되어 DSA 정보를 유실하였을 경우 MN2의 등록 과정은 실패하게 된다. 이는 이미 AAAF와의 DSA 정보를 가진 AAAH가 다시 DSA 요청이 왔을 경우 본 구현에서는 에러로 처리하도록 하였기 때문이다. 이것은 동일 메시지를 여러번 수신하였을 경우 첫 번째 메시지만이 유효하고 나머지는 중복으로 처리해서 버려야 하는 Diameter Draft를 따라 처리했기 때문으로 본 구현에서는 DSA 설정 요청을 받은 AAAH가 DSAR 메시지의 Origin-State-ID-AVP를 확인하여 다시 DSA를 맺도록 하였다. 이 경우에도 DOS 공격의 방지를 위하여 Origin-State-ID-AVP는 항상 서명되도록 한다. 따라서 Draft를 따르면서 DSA를 다시 설정할 수 있도록 구현하였다.



〔그림 11〕 AAAF에서의 DSA 정보 유실

V. 구현 결과

다음 [표 1]의 결과는 DSA 설정에 요구된 시간을 측정한 것이다. 전체적인 DSA 메시지 처리 시간과 AAAF와 AAAH 사이의 DSA 설정에 걸리는 시간을 10회 측정하여 평균을 계산하였다.

〔표 1〕 DSA 설정 시간

횟수	DSA 메시지 처리 시간(초)	AAAF-AAAH간 DSA 설정시간(초)
1	0.0864	2.7567
2	0.5889	4.9305
3	0.3967	6.4668
4	0.1585	5.5766
5	0.1688	4.5364
6	0.1569	5.1389
7	0.1979	5.5225
8	0.2525	5.9639
9	0.1687	4.3694
10	0.0923	5.1316
평균처리시간	0.2268	5.0394

〔표 2〕 MIP 메시지 처리 시간

횟수	AAAH	AAAB	AAAF	AAA 전체시스템
1	1.10	0.00	0.18	1.28
2	1.20	0.00	0.18	1.38
3	1.10	0.00	0.26	1.36
4	1.22	0.00	0.22	1.44
5	1.18	0.00	0.16	1.34
6	1.24	0.04	0.20	1.48
7	1.26	0.02	0.16	1.44
8	1.36	0.00	0.18	1.54
9	1.22	0.00	0.18	1.40
10	1.10	0.00	0.22	1.32
평균처리시간	1.1980	0.0060	0.1940	1.3980

[표 2]의 결과는 [표 1]의 결과로 설정된 DSA를 사용하여 MIP 사용자의 등록 시간을 나타낸 것이다. AAAH, AAAB, AAAF에서 CMS 응용을 적용한 MIP 등록 메시지를 처리하는 시간을 측정하고 전체적으로 걸린 시간을 계산하였다. 각 MIP 메시지는 CMS 응용을 사용하여 암호화, 복호화, 서명, 검증이 적용되었다.

VI. 결 론

Diameter 프로토콜은 다양한 응용에서 사용자의

인증, 권한 검증, 과금등의 서비스를 지원할 수 있다. Diameter 프로토콜은 IPsec, TLS등을 사용하여 노드 간 보안을 제공한다. 그러나 경유 노드에서의 메시지 변조가 가능하기 때문에 안전성에 문제가 발생한다. 따라서 Diameter CMS Security 응용은 공개키 기반 구조를 이용하여 DSA를 설정함으로써 Diameter 응용에서 종단간 보안 기능을 제공하였다. 본 논문에서는 실제 Diameter CMS Security 응용을 구현하고 구현시 발생하는 문제들을 수정하였다. 또한 시스템 장애 등의 이유로 한 노드가 DSA 정보를 분실하였을 경우, 두 노드 사이의 통신에는 심각한 문제가 발생하게 되는 문제점도 존재한다. MIP와 같은 응용에서는 이러한 문제로 사용자 등록이 불가능하여 서비스가 이루어지지 못할 수도 있다. 본 논문에서는 이러한 경우의 DSA를 재설정하는 방법 또한 제시하였고 이 해결책을 실제 구현한 결과와 성능을 보였다. 또한 다음은 실제 MIP 사용자의 등록 시간에 관한 테스트 결과이다. 이 테스트는 CMS를 사용했을 경우와 적용하지 않았을 경우로 나누어 시행되었으며 MN이 등록 요청을 한 후 최종적으로 등록 승인 메시지를 수신한 시간까지의 결과이다. 결과에서 알 수 있듯이 CMS를 사용하여 안전하게 통신한 경우의 결과가 그렇지 않은 경우에 비하여 0.56초의 차이가 남을 알 수 있다.

〔표 3〕 MIP 사용자 등록 시간

횟 수	CMS 적용시의 MIP 등록시간(초)	CMS 미적용시의 MIP 등록시간(초)
1	3.68	3.00
2	3.60	2.98
3	3.96	3.38
4	3.70	3.30
5	3.58	3.02
6	3.58	3.16
7	3.88	3.18
8	3.56	3.18
9	3.72	3.14
10	3.86	3.26
평균처리시간	3.71	3.16

이와 같은 결과를 바탕으로 CMS를 적용한 실제 Diameter 프로토콜의 상용화를 위해서 본 논문의 결과가 유용하게 사용되리라 사료된다.

참 고 문 헌

[1] Diameter CMS Security Application(draft-ietf-aaa-diameter-cms-sec-04), Internet-Draft, march 2002.

[2] Diameter Mobile IPv4 Application(draft-ietf-aaa-diameter-mobileip-11), Internet-Draft, June 2002.

[3] Diameter Base Protocol(draft-ietf-aaa-diameter-12), Internet-Draft, July 2002.

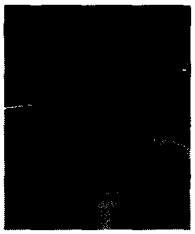
[4] Diameter NASREQ Application(draft-ietf-aaa-diameter-nasreq-09.txt), Internet-Draft, March 2002.

[5] Cryptographic Message Syntax, RFC 2630, June 1999.

[6] Farrell, Turner, "Reuse of CMS Content Encryption Keys", RFC 3185, October 2001.

〈著者紹介〉

유 희 종 (Hui-jong Yu) 정회원
1999년 2월 : 성균관대학교 정보공학과 졸업
2001년 2월 : 성균관대학교 전기전자 및 컴퓨터공학과 석사 졸업
2001년 1월~현재 : 한국전자통신연구원 연구원
<관심분야> 암호이론, 네트워크 보안, AAA



김 말 희 (Mal-hee Kim) 정회원
1996년 2월 : 서강대학교 전자계산학과 졸업
1998년 2월 : 서강대학교 전자계산학과 석사 졸업
1998년 1월~2000년 10월 : 삼성전자 통신연구원 연구원
200년 11월~현재 : 한국전자통신연구원 연구원
<관심분야> 네트워크 보안, AAA

김 현 곤 (Hyun-gon Kim) 정회원
1992년 : 금오공과대학교 전자공학과 학사
1994년 : 금오공과대학교 전자공학과 석사
2003년 : 충남대학교 전자공학과 박사
1994~현재 : 한국전자통신연구원 정보보호연구본부 AAA정보보호연구팀장
<관심분야> IP 기반의 이동통신 네트워크 및 정보보호, 무선 인터넷 정보보호