情報保護學會論文誌
第11券 第6號, 2001. 12

W-CDMA 방식 IMT-2000 시스템에서의 인증에 관한 연구

김 건 우*, 정 배 은*, 장 구 영*, 류 희 수*

A study on the authentication mechanism of W-CDMA IMT-2000 system

Keonwoo Kim*, Bae Eun Jung*, Ku-Young Chang*, Heuisu Ryu*

요 약

W-CDMA 방식 IMT-2000 시스템에 대한 인증 메커니즘은 3GPP TSG SA WG3에 의해서 개발되었고, 이에 따라 우리는 인증 메커니즘과 알고리즘을 GUI 환경에서 시뮬레이션 하였다. 본 논문에서는, 3GPP의 인증 절차를 분석하고, GUI 환경에서의 시뮬레이션에 관해서 설명한다. 또한 메커니즘의 타당성을 검증하고, 전송되는 보안 파라미터(security parameter) 중 스펙에서 명확하게 언급되지 않은 부분에 관해서 논의한다.

ABSTRACT

Authentication mechanism for W-CMDA IMT-2000 system is developed by 3GPP TSG SA WG3. We simulated the mechanism and algorithms. In this paper, we overview 3GPP authentication procedures and present results of our simulation. We validate the mechanism and parameters transmitted during authentication procedures and we also discuss parameters which are unclear in specification.

keyword : *3GPP, 인증, AKA, MILENAGE, 보안 파라미터*

Ⅰ. 서 론

GSM이나 IS-95 CDMA 시스템과 같은 2세대 이동통신 시스템과 비교하여, 진보된 3세대 방식인 IMT-2000 시스템은 고속의 멀티미디어 서비스 제공 및 글로벌 로밍을 특징으로 한다.

이러한 이동통신 환경의 변화는 정보보호에 대한 대책을 절실히 요구하고 있고, 또한 정보보호 기술도 새로운 환경 변화에 맞추어 발전해야 한다. 이에 부합하여 IMT-2000의 발전을 주도하고 있는 지역 그룹인 3GPP(3^{rd} Generation Partnership Project)와 3GPP2(3^{rd} Generation Partnership Project2)에서는 각각 자신들의 기술에 맞는 표준화 작업을 진행중이다. 특히, 비동기 방식 표준을 제정하고 있는 3GPP는 ETSI, ARIB, TTA, T1, TTC로 구성되어 있고 여러 작업 그룹에서 활발한 활동을 보이고 있는데, 보안 아키텍쳐, 인증 메커니즘, 암호 알고리즘 등과 같은 정보보호와 관련해서는 TSG SA WG3에서 담당하고 있다.

3GPP에서의 정보보호는 2세대 방식보다 더 강력하고 안전한 메커니즘을 요구하는데, 가입자와 네트워크간의 상호인증, 무선구간에서의 데이터 보호를 위해 KASUMI를 이용한 암호화 함수인 f8과 무결성 제공을 위한 f9 함수가 그것이다.[16,17,19,20]

* 한국전자통신연구원 정보보호연구본부(wootopian, bejung, jang1090, hsryu@etri.re.kr)

특히, RNC(Radio Network Controller)와 ME (Mobile Equipment) 사이에 교환되는 시그널링 메시지에 대한 무결성 서비스는 위장 기지국 공격 (false base attack)에 대항하기 위해서는 필수적인데, 이는 RNC와 MS(Mobile station)간의 지역 인증(local authentication) 과정의 일부로 볼 수 있다. 이에 따라, 3GPP TSG SA는 인증을 위해서 ETSI SAGE에 의해서 개발된 MILENAGE를 사용할 것을 권고하고 있으며, 암호화와 무결성, 그리고 MILENAGE의 안전성에 관해서는 이미 여러 분야에서 검증이 되었다.[1~4,18]

현재 IMT-2000 개발 상황을 고려할 때 시스템이 완벽하게 구축되지 않은 상태에서 전체적인 관점으로 인증과 암호모드, 그리고 보안 파라미터에 대한 검증을 하드웨어적으로 확인하기란 사실상 불가능하다. 하지만, 3GPP의 정보보호 메커니즘이 올바르고 타당한 것인지를 확인하고 거기에 사용된 보안 파라미터가 합리적이고 효율적인가를 검증하는 것은 매우 중요하다. 본 논문에서는 이에 대한 확인 방법으로 AKA(Authentication and Key Agreement) 절차, 신원(identity) 확인 절차, IMSI(International Mobile Subscriber Identity)와 임시로 사용되는 인증 데이터의 분배 절차, 보안 모드 협상 절차, 그리고 TMSI 재할당 절차등 가입자 인증 절차에 관한 메커니즘을 GUI 환경에서 시뮬레이션하고 이것에 대한 검증 결과에 관해서 논한다. 인증 메커니즘의 핵심이라 할 수 있는 AKA가 수행되기 위해서는 알고리즘 $f1, f1^*, f2, f3, f4, f5, f5^*$가 USIM(Universal Subscriber Identity Module)[12]과 AuC(Authentication Center)내에 장착되어야 하고, 또한 AuC는 난수 생성 알고리즘인 f0를 포함하고 있어야 한다.

본 논문의 Ⅱ장에서는 3GPP 인증 메커니즘[13~15]에 관해서 설명하고, Ⅲ장에서 시뮬레이션을 구성하는 각각의 시스템, 즉, USIM, ME, VLR(Visited Location Register), HLR(Home Location Register), AuC에서 분석, 구현되어야 하는 대상과 절차에 관해서 서술한다. 그리고, 이 결과를 바탕으로 AuC에서 인증 벡터를 생성하는 방식에 관해서 논한다. Ⅳ장에서 AuC에서 생성하는 AV(Authentication Vector)의 개수, node identity와 LAI(Location Area Identity)와의 관계와 같은 스펙에서 불분명한 부분에 관한 우리의 의견을 제시한다. 그리고 Ⅴ장에서 시뮬레이션 내용 및 결과에 관하여 간략하게 언급한다.

Ⅱ. 3GPP 인증 메커니즘

2.1 인증 개요

3GPP에서의 가입자 인증은 USIM과 AuC가 같은 비밀키를 소유하고 있음을 증명함으로써 이루어지는데, VLR은 서비스를 제공하기 전에 가입자를 인증하기 위해 HLR/AuC에 의해서 생성된 AV를 사용한다. 인증이 성공하면 VLR과 MS는 보안 모드에서 사용하는 CK(Ciphering Key)와 IK(Integrity Key)를 공유한다. 그리고, VLR은 KSI(Key Set Identifier)의 값을 바탕으로 AKA 절차의 생략 유무를 결정한다. KSI의 값이 '111' 이라면 인증과정을 생략하고 새로운 CK, IK를 생성할 필요가 없이 MS에 저장된 CK와 IK를 사용하도록 한다. 한편, 무선 인터페이스에서 RNC는 암복화에 중요한 역할을 하지만 인증 메커니즘에서는 VLR과 MS 사이의 전송되는 메시지를 전달하는 역할만 수행한다.

〔그림 1〕은 3GPP 보안 구조[5,13]를 나타내고 있는데, $f1, f1^*, f2, f3, f4, f5, f5^*$는 AKA를 위해서 사용되는 암호학적 함수이고 이들의 입출력으로 사용되는 값은 〔표 1〕과 같다.

그리고, $f1 \sim f5^*$ 함수의 출력을 생성하기 위해서 실질적으로 AKA에 사용되는 MILENAGE 알고리즘의 구조를 〔그림 2〕에 도식하였다.[19,20]

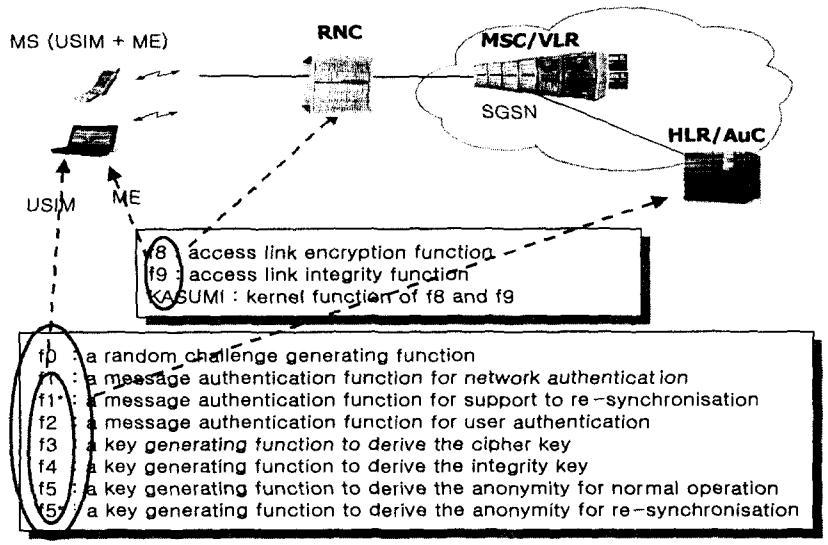
OP_C는 아래 수식과 같이 가입자의 영구 비밀키인 K를 가지고 사업자에 의존하는 값인 OP를 암호화함으로써 생성된다.

$$OP_C = OP \oplus E_K[OP]$$

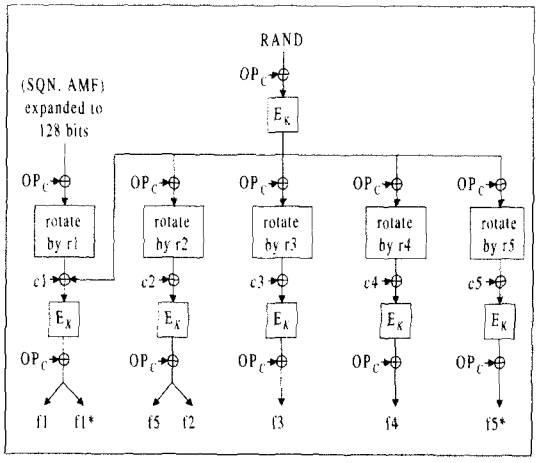
여기에서 $r1, r2, r3, r4, r5$와 $c1, c2, c3, c4, c5$는 고정된 상수이며, MILENAGE의 핵심 함수 E_K에는 Rijndael 알고리즘이 사용된다.

〔표 1〕 f1~f5*의 입력값과 출력값

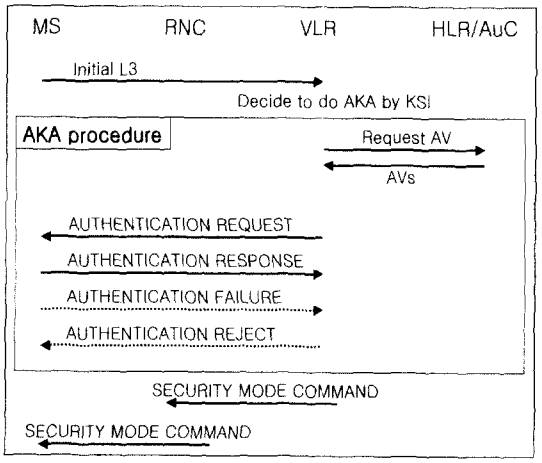
함수	입력값	출력값
f1	K, SQN, RAND, AMF	MAC(XMAC)
f1*	K, SQN, RAND, AMF	MAC*(XMAC*)
f2	K, RAND	RES(XRES)
f3	K, RAND	CK
f4	K, RAND	IK
f5	K, RAND	AK
f5*	K, RAND	AK*



〔그림 1〕 3GPP security 구조와 알고리즘



〔그림 2〕 MILENAGE 구조



〔그림 3〕 3GPP 인증 메커니즘

앞에서 언급한 알고리즘과 구조를 바탕으로 3GPP 인증 메커니즘에 관하여 전송되는 메시지를 〔그림 3〕과 같이 나타낼 수 있다.

1) 먼저 MS는 Initial L3 메시지를 MM(Mobility Management) 연결(connection) 과정에서 VLR로 전송한다.
2) 실제로 AKA 과정이 일어나기 전에 MS와 RNC 사이에는 RRC(Radio Resource Control) 연결 설정이 일어나는 데, 이때 MS는 사용할 암호 알고리즘과 무결성 알고리즘 등을 RNC에 전송한다.
3) Initial L3 메시지에는 사용자 ID, KSI, LAI를 포함하고 있어서 VLR은 KSI 값을 가지고 AKA의 수행여부를 판단한다. 이것과 관련해서 이전 가입자 방문망(VLRo)과 새로운 가입자 방문망(VLRn) 사이에는 IDENTIFICATION 확인 절차가 일어날 수도 있다.
4) AKA가 수행된다면 VLR은 HLR/AuC에게 AV 생성을 요구한다.
5) 인증이 성공하면, SECURITY MODE 협상 단계로 들어간다.

2.2 인증과 Security 관련 절차

이 절에서는 각각의 인터페이스에서의 인증 절차에 관해서 좀더 구체적으로 살펴본다.[13,20]

■ **VLR과 HLR/AuC 구간에서의 AKA**

만약, VLR이 사용자를 인증하는데 필요한 인증 벡터가 없다면 VLR은 HLR/AuC에 새로운 인증 벡터를 요구하게 된다. 이 경우에 VLR은 node identity, node type과 IMSI를 포함하는 메시지를 보내고, HLR/AuC은 이에 응답해서 인증벡터를 생성한 다음 VLR에 전송한다.

■ **MS와 VLR 구간에서의 AKA**

이 구간에서의 AKA의 목적은 MS와 VLR이 상호 인증을 하고, 새로운 CK, IK를 설정하는데 있다. 먼저, VLR이 RAND(Random Number), AUTN(Authentication Token), KSI를 포함하는 AUTHENTICATION REQUEST 메시지를 MS에 보내면 MS는 AUTN을 구성하는 요소중 하나인 MAC(Message Authentication Code)과 자신이 계산할 수 있는 XMAC(Expected Message Authentication Code)을 비교한다. 만약 두 개의 결과가 다르다면 MAC 실패에 대한 이유와 함께 AUTHENTICATION FAILURE 메시지를 VLR에 전송하고, 같다면 MS는 USIM에 저장되어 있는 SQN_{MS}와 AUTN의 또 다른 한 요소인 SQN과 비교한다. 그래서, SQN이 올바른 범위 내에 있는지를 판단해서, 단말기의 네트워크에 대한 인증 성공 응답 메시지인 AUTHENTICATION RESPONSE 메시지를 전송하거나, SQN 범위의 실패에 따른 AUTHENTICATION FAILURE 메시지를 VLR에 전송한다. 만약 SQN이 올바른 범위 내에 있지 않으면, MS는 AUTS를 계산하고, 그것을AUTHENTICATION FAILURE와 함께 보낸다. 한편, MS가 네트워크를 인증하면 응답으로 RES를 계산하는데 이는 AUTHENTICATION RESPONSE 메시지와 함께 VLR로 전송이 되어서 VLR이 가지고 있는 XRES와 같은지를 비교할 수 있게 한다. 이 비교로 네트워크는 단말기에 대한 인증을 성공해서 상호인증 절차를 완료하거나, 인증 실패에 따른 AUTHENTICATION REJECT 메시지를 MS에 전송한다.

■ **이전 방문망으로부터의 IMSI와 인증 데이터의 분배**

이 절차는 동일한 SN(Serving Network) 내에서 이전 방문망인 VLRo에서 새로운 방문망인 VLRn으로 인증 데이터를 제공할 때 발생한다. VLRn은 TMSI와 LAI를 포함하는 Initial L3 메시지를 MS로부터 받자마자 VLRo로 TMSI를 확인하기 위한 메시지를 전송한다. 그러면, VLRo는 TMSI와 관련된 정보를 자신의 데이터베이스에서 찾아서 남아있는 CK, IK 등을 VLRn에게 전송하거나, 혹은 TMSI 확인 실패 메시지를 전송한다.

■ **Identification 절차**

바로 위에서 언급한 TMSI를 사용한 Identity 확인 절차가 실패할 경우, VLRn은 IMSI를 이용하여 MS와 Identification 확인을 하게된다.

■ **보안 모드 setup 절차**

AKA가 성공적으로 완료되거나, KSI 확인으로 인한 AKA가 생략되면 보안 모드 협상 절차를 하게 된다. 이 과정에서 사용할 암호 알고리즘을 선택하는 등 MS와 RNC 사이의 많은 협상 과정을 하게 되지만, 본 논문의 주제와는 약간 거리가 있으므로 상세히 다루지는 않는다. 하지만, 시스널링 메지지에 대한 무결성 체크를 하는 것은 필수적이다.

■ **TMSI 재할당 절차**

MS가 새로운 망으로 들어오면 해당 망의 TMSI를 MS에게 분배하는 것을 TMSI 재할당 절차라고 한다. 물론 이전에 AKA가 성공한 다음이어야 하고 VLRn은 새로운 TMSI를 포함한 TMSI REALLOCATION COMMAND 메시지를 MS에게 전송하고 MS는 TMSI REALLOCATION COMPLETE 메시지로 응답한다. 이 과정에서 ME와 RNC는 전송되는 메시지에 대해서 MAC 값을 덧붙이고, 보안 모드에서 협상된 암호 알고리즘으로 데이터를 암호해서 서로 상대방에게 전송하면, 데이터를 받은 쪽은 메시지를 복호해서 MAC을 검증하게 된다. 이렇게 해서 할당받은 TMSI는 또다시 새로운 방문망으로 접속할 때 IMSI 대신 사용하게 된다.

Ⅲ. 인증 메커니즘의 구현

이 장에서는 3GPP 인증 메커니즘의 소프트웨어 시뮬레이션을 위해서 필요한 파라미터, 절차 및 함수들을 각 시스템별로 나누어 설명한다. GUI 환경에서 USIM, ME, RNC, VLR1, VLR2, VLR3, AuC, 그리고 Registration등 총 8개의 윈도우를 만들었으며 이들간에는 UDP 소켓을 통해서 서로 메시지를 교환한다. 구현 목적이 메커니즘을 분석, 검증하는 것이므로 전송되는 모든 메시지는 각각 메

시지에 대한 설명과 메시지를 구성하는 데이터 포맷, 그리고 실질적인 데이터 내용이 전송된다.[7,8,13]

3.1 AuC

물리적으로 AuC는 HLR과 같은 시스템으로 구성될지도 모른다. 하지만, 시뮬레이션에서는 각 시스템의 동작 분석에 중점을 두고 있으므로 AuC를 HLR과 분리하였다. 인증 메커니즘에서의 AuC의 역할은 가입자에 대하여 인증 벡터를 생성하거나 인증과 관련된 데이터를 관리하는 것인데, HLR이 인증벡터를 요구하거나 재동기 과정을 요구할 때 동작한다. 이를 위해 AuC가 인증 절차를 수행할수 있도록 $f1, f1^*, f2, f3, f4, f5, f5^*$ 뿐만 아니라 난수 생성 함수인 $f0$와 SQN 생성 알고리즘까지 모두 구현하였다. 이밖에, AuC에서 분석, 구현되어야 하는 파라미터와 메커니즘은 다음과 같다.

■ **AMF**

이 파라미터는 망 운영자나 사업자의 인증 관리 정책에 관한 정보를 가지고 있다. 예를 들어, Threshold 값, 유용한 SQN의 범위 등의 정보는 2 바이트 AMF에 기록되며, 관리 정책이란 매우 빈번히 바뀌는 것이 아니기 때문에 AMF는 자주 변경되는 값은 아니라고 보여진다.

■ **MILENAGE**

핵심함수로 128 비트 블록 암호인 Rijndael[19]과 운영자 키인 128 비트 OP를 사용하는 MILENAGE 알고리즘은 Pentium II 750 MHz, 256MB Memory, Visual C++6.0 컴파일러를 사용했을 때 [표 2]와 같은 결과를 얻을 수 있다. MILENAGE는 한 번의 키 스케줄과 7번의 Rijndael 암호화가 반복과정으로 구성되어 있고, 구현에서는 암호화를 위해서 8개의 look-up 테이블을 사용하였다.

그리고, ATMEL AT90SC 스마트카드 개발 툴킷을 사용하여 8 bit 마이크로프로세서로 구현한 결과, 필요한 메모리 크기를 [표 3]에 나타내었는데 이는 표준에서 요구하는 것보다 훨씬 우수한 결과이다.

[표 2] MILENAGE 속도 측정

Rijndael 암호화	256~320 Mbps
AV 생성	34.15 Mbps

[표 3] 8 bit microprocessor 구현결과

ROM(Byte)		RAM(Byte)	
스펙	구현결과	스펙	구현결과
8000	4977	300	240

■ **OP**

운영자는 그들만의 관리 정책에 의해 OP를 결정해야 하는데, 이 값은 모든 가입자에게 동일해도 상관없다. 만약, 가입자마다 하나의 OP를 할당하고자 한다면, AuC에는 모든 가입자에 대한 OP 정보를 저장해야 되고 상당량의 메모리를 필요로 한다. 하지만, OP가 그대로 USIM에 저장되는 것이 아니라 가입자 고유의 영구키로 암호된 OPc가 USIM에 저장되기 때문에 굳이 가입자 마다 다른 OP를 사용할 필요가 없다고 분석된다.[12]

■ **SQN 생성**

AuC는 가입자에 대한 SQN_{HE}를 저장하고 이로부터 SQN을 생성한다. SQN을 생성하기 위해서 time-based, not time-based, partly time-based의 세 가지 방법이 사용되는데,[13] 본 시뮬레이션에서는 not time-based 방법을 사용하였다. 이 방법을 사용한 이유는 실제 IMT-2000 하드웨어 시스템과 위성시각을 맞출 수 없고 또 소프트웨어 구현의 효율성 때문이다. 그러나, 이렇게 생성된 SQN은 $f5$를 이용해서 보호되어야 한다. 한편, MILENAGE의 결과로 $f2$와 $f5$는 동시에 계산이 되기 때문에 이에 대한 오버헤드는 거의 없다.

■ **AV 개수**

한번에 생성해야 하는 인증벡터의 개수는 운영자의 정책에 따라 다른데 가입자의 수, AKA 발생 빈도, 그리고 AuC의 계산 능력 등을 고려해야 한다.

■ **$f0$**

RAND 생성 함수인 $f0$는 하나의 AV가 생성될 때마다 $f0$가 한번 사용되거나, $f1, f1^*, f2, f3, f4, f5, f5^*$가 사용되기 전에 미리 여러 개의 RAND를 생성하는 방법이 있는데, 시뮬레이션에서는 후자의 방법을 이용한다.

위의 암호학적 파라미터를 바탕으로 AuC에서 구현되어야 하는 scheme을 [그림 4]와 [그림 5]에서와 같이 pseudo code로 나타내었다.

```
Input: IMSI, node type, node identity
Actions:
  Step 1 Retrieve K, SQN_HE
  Step 2 Decide AMF
  Step 3 OPc = OP⊕E_K(OP)
  Step 4 Obtain n RANDs from f0
  Step 5 for i=0 to i=n-1 by +1
    Step 5.1 Compute SQN[i]
    Step 5.2 SQN_HE ←SQN[i]
    Step 5.3 Compute f1-f5
    Step 5.4 AUTN[i] = (SQN[i]⊕AK[i])||AMF[i]||MAC[i]
    Step 5.5 AV[i] = RAND[i]||XRES[i]||CK[i]||IK[i]||
                     AUTN[i]
  Step 6 Store SQN_HE
  Step 7 Return IMSI, node type, node identity, AVs
```

〔그림 4〕 AV 생성 scheme

3.2 VLR

〔그림 6〕은 인증과 관련해서 VLR에서 수행되어야 하는 절차를 도식한 것이다. 그리고, 이를 위해서 구현되어야 하는 함수를 VLR을 기준으로 하여 각 구간별로 나누어 설명한다.[8,13]

```
Input: IMSI, node type, node identity, RAND
       AUTS(=SQN_MS⊕AK* || MAC*)
Actions:
  Step 1 AMF* = 0
  Step 2 Retrieve Key, SQN_HE
  Step 3 Compute AK*
  Step 4 SQN_MS = (SQN_MS⊕AK*) ⊕AK*
  Step 5 Verify SQN_HE
    Step 5.1 If correct, then
      Step 5.1.1 Go to Step 6
    Step 5.2 Else
      Step 5.2.1 Compute XMAC*
      Step 5.2.2 If XMAC* == MAC*, then
        Step 5.2.2.1 SQN_HE ← SQN_MS.
        Step 5.2.2.2 Go to Step 6
      Step 5.2.3 Else
        Step 5.2.3.1 Go to Step 6
  Step 6 Generate AVs
  Step 7 Return IMSI, node type, node identity, AVs
```

〔그림 5〕 re-synchronization scheme

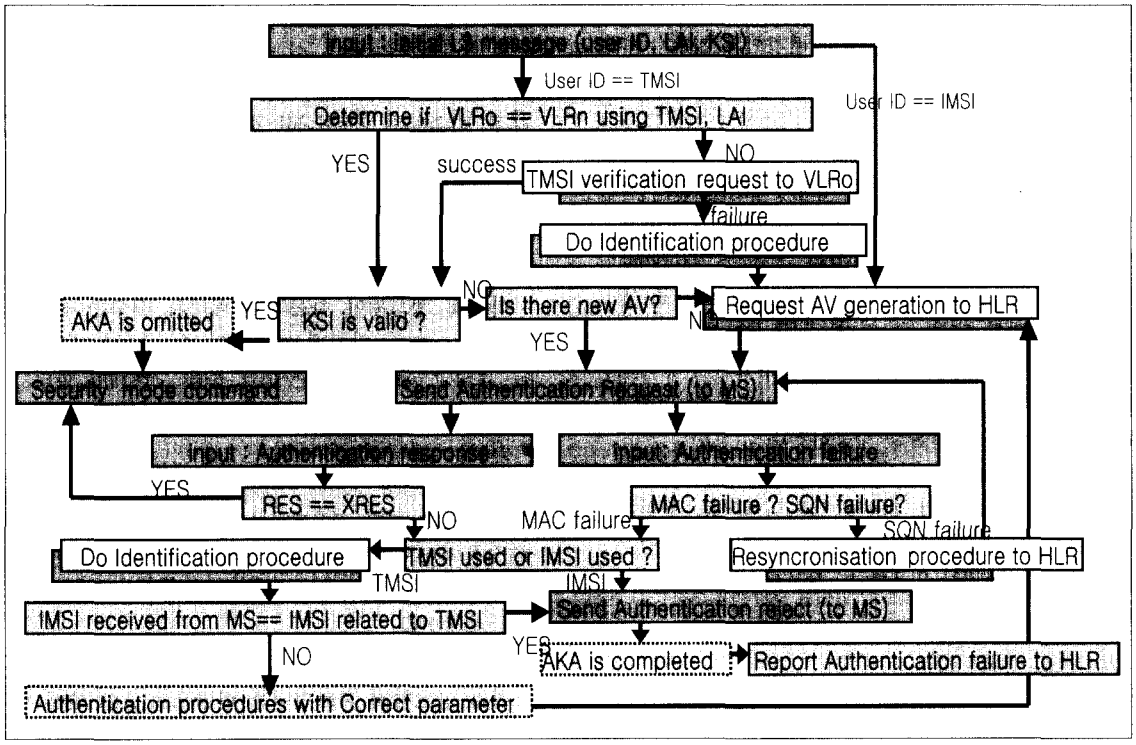
3.2.1 MS나 RNC로부터 메시지를 받았을 때 VLR이 수행하는 함수와 기능

- ***get_initial_L3_message()***

이 함수는 VLR이 MS로부터 Initial L3 메시지를 받았을 때 수행된다.

- ***get_identity_response()***

이것은 MS로부터 IMSI를 포함하는 IDENTITY



〔그림 6〕 VLR에서 수행되는 인증 메커니즘

```
Input: KSI, User ID, LAI (if user ID is TMSI)
Actions:
  Step 1 If user ID denotes IMSI, then
    Step 1.1 Find HLR from IMSI
    Step 1.2 Request AVs to HLR
  Step 2 Else if user ID is TMSI
    Step 2.1 KSI_MS = KSI
    Step 2.2 Find VLRo by LAI
    Step 2.3 If VLRo ==VLRn, then
      Step 2.3.1 Request temporary authentication data
                 and IMSI to VLRo
    Step 2.4 Else
      Step 2.4.1 Find IMSI and temporary data from data
                 base
      Step 2.4.2 If there are data related TMSI, then
        Step 2.4.2.1 IMSI_TMSI = IMSI in data
        Step 2.4.2.2 KSI_TMSI = KSI in data
        Step 2.4.2.3 If KSI_TMSI == KSI_MS and KSI_MS !=
                     '111', then
          Step 2.4.2.3.1 Send SECURITY MODE COMMAND to
                         MS
        Step 2.4.2.4 Else
          Step 2.4.2.4.1 If there are AVs to use, then
                         Send AUTHENTICATION REQUEST to
                         MS
          Step 2.4.2.4.2 Else
            Send a message requesting AVs to HLR/AuC
  Step 3 End
```

〔그림 7〕 *Get_Initial_L3_message*() 함수

RESPONSE 메시지를 받았을 때 처리되는 함수이다. VLR은 TMSI에 해당하는 IMSI를 가지고 있는데 이 IMSI를 $IMSI_{TMSI}$로 표기한다. 만일, VLR이 $IMSI_{TMSI}$를 가지고 있지 않다면 $IMSI_{TMSI}$는 에러 값으로 세팅된다. 이것에 관해서는 뒤에서 다시 설명한다.

```
Input: IMSI contained in IDENTITY RESPONSE, IMSI_TMSI
Actions:
  Step 1 IMSI_MS = IMSI
  Step 2 If IMSI_TMSI == IMSI_MS, then
    Step 2.1 Send AUTHENTICAION REJECT to MS
  Step 3 Else
    Step 3.1 Find HLR from IMSI_MS
    Step 2.2 Request AVs to HLR/AuC
  Step 4 End
```

〔그림 8〕 *get_identity_response*() 함수

■ ***get_authentication_response*()**

이 함수는 AUTHENTICATION REQUEST 메시지에 대한 MS의 응답으로 AUTHENTICATION RESPONSE 메시지를 수신했을 때 수행되는데, 이 때 VLR은 AV의 구성요소인 CK와 IK를 저장한다.

■ ***get_authentication_failure*()**

이 함수는 MS로부터 인증 실패를 나타내는 AUTHENTICATION FAILURE 메시지를 수신했을 때 호출되는데, 인증 실패의 원인이 SQN에

```
Input: RES in AUTHENTICATION RESPONSE, XRES in AV
Actions:
  Step 1 If RES == XRES, then
    Step 1.1 Send SECURITY MODE COMMAND to RNC
  Step 2 Else
    Step 2.1 If there is IMSI received from MS, then
      Step 2.1.1 Send AUTHENTICATION REJECT to MS
    Step 2.2 Else
      Step 2.2.1 Send IDENTITY REQUEST to MS
  Step 3 End
```

〔그림 9〕 *get_authentication response*() 함수

있다면 재동기에 사용될 AUTS 값이 이 메시지에 포함된다. 만약, MS로부터 받은 IMSI가 있다면 $OBTAIN_IMSI_{MS}$를 'TRUE'로 설정하고 그렇지 않다면 'FALSE'로 설정한다.

```
Input: AUTHETNICATION FAILURE, IMSI_MS
Actions:
  Step 1 CAUSE = Failure cause in AUTHENTICATION
  Step 2 If CAUSE == MAC Failure, then
    Step 2.1 If OBTAIN_IMSI_MS == TRUE
      Step 2.1.1 Send AUTHENTICATION REJECT to MS
    Step 2.2. Else
      Step 2.2.1 Send IDENTITY REQUEST to MS
  Step 3 Else if CAUSE == SQN Failure, then
    Step 3.1 Request to resynchronize to HLR/AuC with
             AUTS
    Step 3.2 Delete remaining AVs
  Step 4 End
```

〔그림 10〕 *get_authentication_failure*() 함수

3.2.2 다른 VLR로부터 메시지를 받았을 때 VLR이 수행하는 함수와 기능

■ ***get_verify_TMSI_from_VLRn*()**

이 함수는 VLR이 TMSI를 확인해 달라는 메시지를 받았을 때 수행된다.

```
Input: TMSI
Actions:
  Step 1 Search IMSI, AVs, CK, IK, KSI related to TMSI;
  Step 2 If exist, then
    Step 2.1 Send IMSI and temorary authentication data
             to newVLR
  Step 3 Else
    Step 3.1 Report failure to VLRn
  Step 4 End
```

〔그림 11〕 *get_verify_TMSI_from_VLRn*() 함수

■ ***get_verify_TMSI_response*()**

이 함수는 IMSI와 임시로 사용되는 인증 데이터를 분배하는 과정에서 VLRo로부터 응답 메시지를 받을 때 수행되는 함수이다. 이때 분배 과정이 일어나기 전에 VLRn은 KSI_{MS}를 Initial L3 메시지에

속해있는 KSI로 세팅하는데, 앞에서 설명한 그림 7에서 잘 보여주고 있다.

```
Input: message received from VLRo, KSI_MS
Actions:
  Step 1 If received message denotes a failure, then
    Step 1.1 Set IMSI_TMSI = value denoting error
    Step 1.2 Send IDENTIFICATION REQUEST to MS
  Step 2. Else
    Step 2.1 IMSI_TMSI = IMSI
    Step 2.2 KSI_TMSI = KSI
    Step 2.3 If KSI_TMSI == KSI_MS and KSI_MS !='111',then
      Step 2.3.1 Send SECURITY MODE COMMAND to MS
    Step 2.4 Else
      Step 2.4.1 If ther is a fresh AV, then
        Step 2.4.1.1 Send AUTHENTICATION REQUEST
      Step 2.4.2 Else
        Step 2.4.2.1 Find HLR by IMSI_TMSI
        Step 2.4.2.2 Request AVs to HLR/AuC
  Step 3 End
```

〔그림 12〕 *get_verify_TMSI_response*() 함수

그림 12 step 1.1과 step 2.1에서의 $IMSI_{TMSI}$는 get_identity_response()와 같은 다른 함수에서 사용되는 변수이다.

■ ***get_AV_response_from_HLR*()**

이 함수는 HLR로부터 IMSI와 AV들을 송신했을 때 수행이 되는데, 그중 하나의 인증벡터는 MS에 전송이 되며 IMSI와 남아있는 AV는 저장이 된다.

■ ***get_AV_request_from_VLR*()**

이 함수는 HLR이 VLRn으로부터 인증벡터을 요구하는 메시지를 받았을 때, IMSI, node type, Identity 정보를 바탕으로 AuC에게 인증벡터를 요구하는 역할을 한다.

■ ***get_resync_request_from_VLR*()**

HLR은 VLRn으로부터 재동기를 요구하는 메시지를 받았을 때 이 절차를 수행한다. HLR이 직접 재동기에 응답하는 것이 아니라 IMSI, RAND, AUTS, node type, 그리고 Identity를 AuC에 전송함으로써 HLR은 재동기를 요구한다.

3.2.3 AuC로부터 메시지를 받았을 때 HLR이 수행하는 함수와 기능

■ ***get_AV_from_AuC*()**

이 함수는 IMSI, node type, node identity, 그리고 AV들을 받았을 때 수행된다.

■ ***get_resync_from_AuC*()**

이 함수의 입력 메시지는 IMSI, node type, node identity, AV들로 get_AV_from_AuC()와 같지만, SQN 실패로 인한 재동기 과정의 결과이다. VLR이 MS에 AUTHENTICATION REQUEST 메시지를 보낼 때 KSI를 결정하고, AUTHENTICATION RESPONSE 메시지를 받을 때 KSI_{IMSI}를 저장하는 것으로 분석된다.

```
Input: IMSI, node type, node identity, AVs
Actions:
  Step 1 Retreive node identity
  Step 2 If node identiy is HLR itself, then
    Step 2.1 Send AUTHENTICATION REQUEST to MS
    Step 2.2 Store remaining AV
  Step 3 Else
    Step 3.1 Send data to VLRn
  Step 4 End
```

〔그림 13〕 get_AV_from_AuC() 함수

3.3 ME

인증 메커니즘과 관련해서 ME가 가장 먼저 하는 일은 RRC CONNECTION COMPLETE 메시지에서 security capability와 START 값을 RNC에 전송하는 것이다. MM 연결 설정이 되면 ME는 Initial L3 메시지를 VLR에 전송하는데,[7,8] user ID로 TMSI가 사용된다면 LAI가 Initial L3 메시지에 추가된다. 〔case 1〕과 〔case 2〕는 ME가 VLR로부터 어떤 메시지를 수신했을 때 동작하는 경우를 설명한 것이다.

[case1] SECURITY MODE COMMAND 수신:

이 메시지를 VLR로부터 받으면 AKA 과정이 생략되거나 인증 절차가 완료된다.

[case2] AUTHENTICATION REQUEST 수신:

이 메시지를 VLR로부터 받으면 ME는 USIM이 MILENAGE 알고리즘을 사용해서 인증 메커니즘을 수행할 수 있도록 하기 위해 RAND와 AUTN을 USIM에게 전송한다. 이것에 대한 응답으로 USIM이 인증 성공유무의 결과를 ME에 전송하면 ME는 AUTHENTICATION RESPONSE 혹은 AUTHENTICATION FAILURE 메시지를 VLR에게 송신한다.

■ ***get_from_VLR*()**

이 함수는 VLR로부터 메시지를 수신했을 때 ME가 동작하는 함수이다.

```
Input: message sent from VLR
Actions:
  Step 1 Receive message
    Step 1.1 If message == IDENTIFICATION REQUEST, then
             send IDENTIFICATION RESPONSE containing
             IMSI
    Step 1.2 If message == AUTHENTICATION REJECT, then
             'Abort'
    Step 1.3 If message == AUTHENTICATION REQUEST, then
             Do case 2)
```

〔그림 14〕 get_from_VLR() 함수

3.4 USIM

USIM의 동작환경은 메모리 공간이나 소비 전력의 한계 등 아주 제한되어 있으므로, 알고리즘의 속도나 연산의 효율성을 잘 고려해서 구현하여야 한다.[12]

USIM은 ME와 profile exchange를 할때 지원 가능한 capability와 보안 파라미터, 그리고 ID를 전송한다. 그리고 나서 AUTHENTICATION COMMAND 메시지를 수신하면 AUTN을 체크하고 MLLENAGE 알고리즘을 이용하여 $MAC = XMAC$ 인지를 비교한다. 두 파라미터의 값이 다르다면 ME에게 인증실패 메시지를 보내고 인증절차를 종료한다. 반대로 두 파라미터의 값이 같다면 USIM은 AuC에서 생성한 SQN의 범위를 검증하는데 올바른 범위에 있다면 네트워크에 대한 인증성공을 의미하는 RES 메시지를 전송하고, 올바른 범위에 있진 않다면 재동기 절차를 위한 AUTS를 계산해서 VLR로 전송한다.

USIM에서 사용되는 여러 보안 파라미터 중에 사업자마다 비밀정보로 취급되는 *OP*를 보호하고 알고리즘의 효율을 높이기 위해 *OP* 대신에 *OPc*를 저장하는 것이 합리적이다.

Ⅳ. 불분명한 보안 파라미터에 대한 제안

이 절에서는 인증 절차에 관련된 파라미터 중에 3GPP 스펙에서 명확하게 기술되어 있지 않은 부분에 관하여 논의한다.[13,14] 우리가 제안한 파라미터를 사용하여 시뮬레이션 한 결과, 우리의 제안이 합리적임을 알 수 있었다.

4.1 IMSI, LAI, Node Identity

■ **HLR Identifier**

현재 3GPP 스펙에서는 가입자마다 가지고 있는 고유의 identity인 IMSI의 크기를 15 digit 이하로 정하고 있다. IMSI의 첫 번째 필드는 HLR Identifier로 구성되는데, VLR이 사용자의 HLR을 알 필요가 있을 때 이 값을 참조하게 된다.

■ **LAI**

LAI는 VLR의 Identifier 이며 3GPP 스펙에서는 5 octets 으로 규정하고 있다. 이 값은 TMSI를 할당할 때 함께 전송되며, VLR은 2.2절의 "이전 방문망으로부터의 IMSI와 인증 데이터의 분배" 절차를 수행하기 위해 이 값을 참조한다.

■ **Node Identity**

VLR이 HLR로 인증 벡터 생성 요구 메시지를 보낼 때 함께 보내는 값으로 AuC가 생성한 인증벡터들을 돌려 받기 위해 필요한 파라미터이다. 따라서, Node Identity 역시 VLR을 가리키는 요소이다.

하나의 망이 또다른 하나의 망을 인식하기 위해서는 망을 식별할 수 있는 어떤 식별자가 필요한데, 위에서 기술한 3개의 파라미터는 모두 같은 역할을 하는 망 식별자로 볼 수 있고 시뮬레이션 결과 모두 같은 길이로 사용하는 것이 합리적이라 판단된다. 만약, IMSI가 15 digits 이고 HLR identifier가 5 octets의 길이를 가진다면, 가입자수는 5 digits를 넘지 못한다. 여기서 5 digit이라 함은 IMSI의 MSIN (Mobile Subscriber Identity Number)의 전체 9~10 digit 중에서 앞부분의 4~5 digit를 제외한 나머지 5 digit의 길이이다. 이 수는 MSIN과 HLR의 관계 및 HLR의 수용능력을 고려할 때 충분한 길이라고 할 수 있다.

한편, VLR은 HLR에게 인증벡터 생성을 요구할 때 HLR이 node type과 node identity를 알 수 있게 하여서, HLR은 VLR을 구분하여 식별할 수 있다. 그래서, node identity와 LAI를 동일한 값으로 사용한 것이 더 유리하다.

4.2 올바른 SQN의 범위

USIM은 인증과정에서 SQN이 정상적인 범위에 속하는 가를 검증하여 재동기화 과정의 실시 여부를 판단한다. SQN 생성 원리에 의하면 나중에 생성된 SQN 값이 이전에 생성된 SQN 값보다 커지도록 되어 있다. 따라서, USIM에 저장되어 있는 SQN은 AKA 과정에서 AuC로부터 받은 SQN 보다 작

은 값이어야 한다. 만약 AuC로부터 들어온 SQN이 USIM에 저장되어 있는 SQN보다 작은 경우 인증이 성공적으로 완료되지 못하고 동기 실패에 의한 인증 실패가 발생한다. 이것은 빈번한 재동기 과정을 야기시켜서 무선 채널상의 패킷을 증가시킨다. 예를 들어 VLR이 MS로 AUTHENTICATION REQUEST 메시지를 보낼 때, VLR은 생성된 SQN의 순서와 무관하게 AV를 전송할 수도 있는데 이것은 재동기를 빈번히 발생시키는 원인이 될 수도 있다. 따라서 보안을 위협하지 않으면서도 재동기화 과정의 빈도수를 적게 하도록 사업자의 정책기준이나 전송방법 등을 고려하여 SQN의 범위를 결정하는 것이 중요하다. 이에 대한 해결 방안으로, not time-based SQN 생성 알고리즘을 사용해서 시간 개념과 무관하게 일정한 증가률로 SQN을 생성한다고 하자. 만약 생성된 SQN 값이 USIM에 저장된 SQN 값보다 작다고 하더라도, 재동기 발생을 최대한 억제하도록 하기 위해 $SQN_{AuC} - SQN_{MS} < \Delta$인 어느정도의 차이 만큼은 올바른 SQN 범위로 인정하므로, Δ의 값을 한번에 생성된 인증벡터의 개수보다 작게 하다면 전송 순서와는 무관하게 한계치를 넘지 않는다. 그러므로, 인증벡터 수보다 작은 Δ를 사용하는 것은 재동기화 발생을 줄일 수도 있고 사업자가 수용할 수 있는 합리적인 대안이라고 생각한다. 이때 공격자는 전송되는 SQN의 패턴을 찾아서 일정한 어떤 관계를 알아낼 수 있으므로, SQN의 노출을 막기 위해서는 $SQN \oplus AK$ 형태로 암호화하여 SQN이 전송되어야 한다.

V. 시뮬레이션 설계기준 및 결과

비동기 방식 IMT-2000 시스템에서 보안 관점의 동작을 시뮬레이션하기 위해서, 물리적으로 존재하는 MS, RNC, VLR, AuC 등을 소프트웨어에서는 각각 윈도우로 구성하였다. 그리고, 이들간의 통신은 UDP 소켓을 이용했고, 기본적인 프레임 구조는 Dialog Based로 했다. 시뮬레이션을 위한 각각의 개체는 다음과 같은 8개의 윈도우이다.

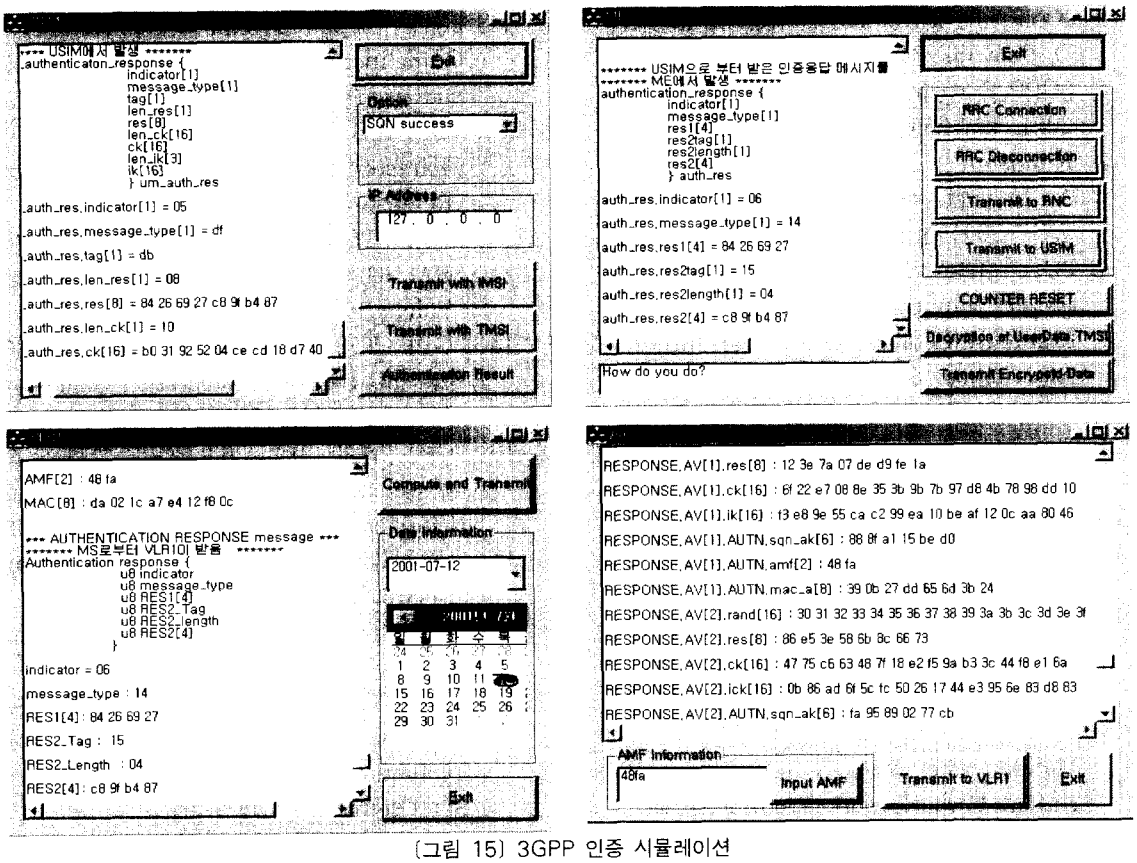
- Registration window
- USIM window
- ME window
- RNC window
- VLR1, VLR2, VLR3 windows
- AuC window

윈도우는 각각 지정된 IP와 port를 가지고 상대방과 통신한다. 단말기(MS)는 기능상 USIM과 ME로 나누었고 RNC는 물리적으로는 여러 개 있어야 하지만 소프트웨어로 시뮬레이션 하는데는 하나의 RNC에서 여러 VLR로 접속할수 있다고 가정하면 되므로 RNC window를 하나만 구현하였다. VLR은 사용자가 처음 접속하는 VLR1을 HLR이라고 가정했고 VLR2, VLR3는 같은 네트웍간의 위치이동, 다른 네트워크간의 위치이동을 위해서 3개의 윈도우를 사용했다. 그리고, AuC window는 인증센터 역할을, Registration window는 처음 단말기를 구입하거나 사용자 키를 분배받을 때 한번만 사용하는 역할을 한다. 통신 개체(각각의 window)간 데이터가 송수신될 때는 데이터의 전송 시 에러 유무를 알고, 전송되는 데이터가 어떤 데이터인지를 쉽게 알 수 있게 하기 위해 송수신 양단에 메시지가 기록되도록 window를 구성했다. 그리고, 송수신되는 메시지 각 개체에서 정보보호 알고리즘과 메커니즘에 관련된 모든 파라미터는 모두 메시지 설명과 함께 정확한 값이 로그 파일로 기록되기 때문에 나중에라도 동작상태를 쉽게 파악 할 수 있다.

3GPP AKA 및 보안 모드를 위한 8개의 윈도우는 송수신되는 정보에 3GPP 표준안에 따른 메시지 포맷을 수용하며, 보안 모드가 협정된 이후에는 암호화된 데이터를 포함하는 PDU를 교환한다.

이 프로그램은 서두에서도 언급했지만 3GPP의 보안 관점에 초점을 맞추고 설계되었다. 그래서, 단말기나 RNC, VLR, AuC 등의 내부 구조나 다른 시스템과의 연동은 고려하지 않았다. 예를 들어서, RNC내부에서 무결성 데이터를 첨가하거나 메시지를 암호를 할 때, RRC 컨트롤[9]을 받아서 어느 계층에서 어떤 베어러을 사용하는가 하는 것은 이 시뮬레이션에서 중요하지 않다. 단지 보안 메커니즘에서 이용되는 파라미터나 물리적인 부분은 시뮬레이션에서 임의로 정해서 사용할 수 있다. 하지만, 3GPP TS WG3 보안 스펙에서 요구하는 형식은 그대로 다 수용하였다.[6~8,10,11]

[그림 15]는 구현한 결과중 일부분인 USIM, ME, VLR1, AuC에 관한 윈도우이다. 수많은 절차들 중에서 하나를 선택해서 처리하는 것은 단지 각 윈도우의 버튼을 클릭하는 것으로서 완료되고 로그창과 로그 파일로 기록이 남게된다. 예를 들어, VLR1 윈도우의 "Compute and Transmit" 버튼을 클릭하면 AuC에 AV를 요구하는 메시지, ME에 인증



〔그림 15〕 3GPP 인증 시뮬레이션

을 요구하거나 보안 모드 협상, 그리고 TMSI 재할당을 요구하는 메시지, 또는 다른 VLR에 TMSI 확인을 요구하거나 응답하는 메시지 등을 전송한다. 즉, 각 윈도우의 버튼은 수신된 패킷을 분석해서 상황에 맞는 알고리즘이나 함수를 자동으로 수행하게 된다. 그리고, 다음에 수행해야 할 절차를 판단한 다음 새로운 보안 파라메테와 적합한 메시지를 계산해서 다음 목적지에 송신하게 된다.

Ⅳ. 결 론

비동기 방식 IMT-2000의 인증 및 암호에 관해서 직접 상용 시스템에서 분석, 연구하기란 현 상황에서는 거의 불가능하다. 하지만, 정보보호 관점에서 이들 메커니즘을 분석, 구현해서 문제점을 파악하고 해결책을 제시하는 것은 매우 중요한 일이라 할 수 있다.

본 논문에서는 3GPP 인증 메커니즘에 관해서 각 시스템에서 수행되어야 하는 절차와 알고리즘 그리고 각각의 함수들에 관해서 분석하였다. 그리고, 3GPP 표준문서에 나와있지만 명확하지 않은 보안 파라미터 등 정보보호 메커니즘을 확인하고 문제가 될 수 있는 부분을 발견할 수 있었고 이에 대한 대안을 제시하였다. 그래서, 다른 시스템적인 문제는 제외하고 보안에 관련한 모든 일들을 소프트웨어로 구현하여 시뮬레이션하였다. 시뮬레이션을 통하여 인증 메커니즘의 합리성과 프로토콜의 효율성을 검증할 수 있었다.

향후, 분석된 결과와 시뮬레이션 결과를 바탕으로 상용 IMT-2000 시스템과 연동해 볼 필요가 있다고 생각된다.

참 고 문 헌

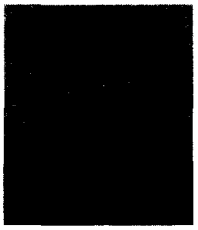
〔1〕 Kang, J., Yi, O., Hong, D. and H. Cho, Pseudorandomness of MISTY-type transformations and the block cipher KASUMI , ACISP 2001.

[2] Kang, J., Shin, S., Hong, D. and Yi, O. Provable Security of KASUMI and 3GPP Encryption Mode f8, ASIACRYPT 2001, pp. 255~271, Dec., 2001.
[3] Knudsen, L. R., and Mitchell, C. J. An analysis of the 3GPP-MAC scheme, WCC 2001, pp. 319~328, Jan. 2001.
[4] 3GPP TSG SA WG3 Security-S3#15, Report on the Evaluation of 3GPP Standard Confidentiality and Integrity Algorithms, Sep., 2000.
[5] 3GPP TS 23.002 v3.4.0 3GPP TSG SSA: Network Architecture(Relesase 1999) 2000. 12.
[6] 3GPP TS 23.003 v3.7.0 3GPP TSG CN: Numbering, addressing and identification (Release 1999) 2000. 12.
[7] 3GPP TS 24.007 v3.6.0 3GPP TSG CN: Mobile radio interface signaling layer 3: General aspects(Release 1999) 2000. 12.
[8] 3GPP TS 24.008 v3.6.0 3GPP TSG CN: Mobile radio interface layer 3 specification: Core Network Protocols-stage 3. 2000. 12.
[9] 3GPP TS 25.331 v3.5.0: 3GPP TSG RAN: RRC protocol Specification(Release 1999) 2000. 12.
[10] 3GPP TS 25.413 v3.4.0 3GPP TSG RAN: UTRAN lu Interface RANAP Signaling (Release 1999) 2000. 12.
[11] 3GPP TS 25.921v3.2.0 3GPP TSG RAN: Guidelines and Principles for protocols and error handling(Release 1999) 2000. 12.
[12] 3GPP TS 31.102 v3.4.0 3GPP TSG T: Characteristics of the USIM Application (Release 1999) 2000. 12.
[13] 3GPP TS 33.102 v3.7.0 3GPP TSG SSA: 3G Security: Security Architecture (Release 1999) 2000. 12.
[14] 3GPP TS 33.103 v3.4.0 3GPP TSG SSA: 3G Security: Integration Guidelines (Release 1999) 2000. 10.
[15] 3GPP TS 33.105 v3.6.0 3GPP TSG SSA: 3G Security: Cryptographic Algorithm Requirements(Release 1999) 2000. 12.
[16] 3GPP TS 35.201: F8 and F9 Algorithms Specification.
[17] 3GPP TS 35.202: Kasumi Algorithm Specification.
[18] 3GPP TR 35.909 v4.0.0 3GPP TSG SSA: 3G Security: Report on the Design and Evaluation of the MILENAGE Algorithm Set (Release 4) 2000. 04.
[19] ETSI/SAGE. Specification of the MILENAGE Algorithm Set. Document 1: Algorithm Specification
[20] ETSI/SAGE Task Force dor 3GPP Authentication Function Algorithms. General Report on the Design, Specification and Evaluation of the MILENAGE Algorithm Set.

〈著者紹介〉

김 건 우 (Keonwoo Kim) 정회원
1999년 2월 : 경북대학교 전자공학과(학사)
2001년 2월 : 경북대학교 전자공학과(석사)
2000년 12월~현재 : 한국전자통신연구원 연구원
〈관심분야〉 이동통신, 암호학, 네트웍 보안

정 배 은 (Bae Eun Jung) 정회원
1993년 2월 : 서울대학교 수학교육과(학사)
1995년 2월 : 서울대학교 수학과(석사)
2000년 2월 : 서울대학교 수학과(박사)
2000년 4월~현재 : 한국전자통신연구원 선임연구원
〈관심분야〉 암호이론, 이동통신 정보보호, 가환대수



장 구 영 (Ku-Young Chang) 정회원
1995년 2월 : 고려대학교 수학과(학사)
1997년 2월 : 고려대학교 수학과(석사)
2000년 8월 : 고려대학교 수학과(박사)
2000년 12월~현재 : 한국전자통신연구원 선임연구원
〈관심분야〉 정보보호 이론, 이동통신 보안, 대수학

류 희 수 (Heuisu Ryu) 정회원
1990년 2월 : 고려대학교 수학과(학사)
1992년 2월 : 고려대학교 수학과(석사)
1999년 5월 : Johns Hopkins University 수학과(박사)
2000년 7월~현재 : 한국전자통신연구원 선임연구원
〈관심분야〉 정보보호, 타원곡선 암호, 이동통신 보안