通信情報保護學會誌
第5卷 第1號, 1995. 3

암호 프로토콜의 정형적 사양 및 분석 기법

Formal Specification & Analysis models for Cryptographic protocol

이 진 석*, 신 기 수*, 이 강 수**

요 약

소프트웨어 개발에서, 사용자 요구 사항을 잘 표현하면 할수록 시스템은 오류가 적고, 사용자가 요구하는 시스템으로 만들어지며, 시스템 검증이 쉬워지는 것 같이, 정형적 사양(formal specification)은 시스템 개발 전반에 영향을 준다. 이 정형적 사양은 암호 프로토콜이 완전(completeness)하고 안전(soundness)한가를 검증하는 데에도 유용하게 사용될 수 있다. 암호 프로토콜을 사양하고 분석하는 방법은 크게 대수적, 논리적, 상태 변환적 접근 방법등과 통신 프로토콜의 검증에 쓰는 패트리넷을 이용한 방법등이 있다. 이들 방법중에는 프로그램으로 구현되어 자동 검증 tool로 개발된 사례도 있다. 본 고에서는 암호 프로토콜을 위한 사양 기법과 그에 따른 분석 기법을 알아보고 그것들을 비교, 분석하였다.

1. 서 론

컴퓨터의 발달과 함께, 전자 결재 시스템, 전자 우편 및 민감한 병원 정보나 개인 정보를 저장하는 데이터베이스등에 정보보호를 위해 "암호학"이 도입되고 있고, 비밀 정보를 얻고자 하는 "공격 기술"도 암호학의 발전과 함께 날로 진보되고 있는 실정이다. 일반적으로, 암호 시스템은 수학적인 측면인 암호 알고리즘과 절차적 규칙인 프로토콜로 구성된다. 따라서, 암호 시스템의 오류에는 알고리즘적 오류와 프로토콜상의 오류가 존재하게 되고, 올바른 암호 시스템을 설계하기 위해서는 알고리즘과 프로토콜의 두가지 측면 모두를 고려해야 한다[1]. 암호 시스템에서 수학적으로 잘 정의된 알고리즘을 사용하더라도 잘못된 암호 프로토콜의 사용으로 치명적인 보안 누출이 생기는 사례는 잘못된 프로토콜이 암호 시스템을 무용지물로 만들 수 있다는 것을 잘 보여준다[1,2,3].

어떤 시스템(소프트웨어, 프로토콜)을 올바르게 만든다는 것은 시스템의 요구 사양(requirements specification)을 만족하는 시스템을 만드는 것으로, 요구 사양을 잘 만드는 것이 제대로된 시스템을 만드는 관건이 된다. 시스템 개발 단계에서 보면, 사용자 요구는 사양되고, 요구 사양에 의해 설계, 구현, 테스팅, 인수과정을 거치면서 시스템이

* 한국전자통신연구소
** 한남대학교 전자계산공학과

개발된다. 따라서, 요구 사양을 명확하고 분명하게 표시하면 할수록, 시스템을 각 단계에서 보다 명확하게 개발할 수 있다. 이를 위해 사양에 정형화(formalism) 기법을 사용하게 된다[4,5].

일반적으로 프로토콜을 기술하는 표기법은 자연어에 근거한 서술식 사양으로 프로토콜의 오류를 발견하기 어려울 뿐만 아니라 오류가 없음을 보이기도 어려웠다. 특히 알고리즘과 결합된 암호 프로토콜의 보안성 누출을 발견하기 위해서는 경험과 직관이 필요하고, 철저한 분석이 요구된다. 암호 프로토콜의 사양 방법에 정형화된 기법의 도입은 확실한 문제 정의를 바탕으로 분석을 용이하게 해 줄 수 있는 방법으로 여러 연구자에 의해 제안되고 있다. 이들 각 기법은 나름대로의 장단점을 갖고 있으며, 각 기법에 대한 문제점 지적과 해결 노력도 시도되고 있다. 본 고에서는 통신 프로토콜과 암호 프로토콜의 차이를 2장에서 보이고, 3장에서는 기존의 암호 프로토콜의 사양과 분석 방법에 대해 알아보고, 4장에서 사양과 분석 방법들을 비교하며, 마지막으로 5장에서는 결론 및 추후 연구 방법을 논한다.

2. 프로토콜 엔지니어링

프로토콜은 D. Longly와 M. Shain의 Data & Computer Security 용어 사전(1989년)에 의하면 "자료 통신에서 두 통신 시스템 간의 입·출력의 통제와 자료 형태에 관한 공식화된 규칙의 집합"으로 정의된다. 정보화 사회로의 생활 방식이 전이되고, 이를 지원하는 컴퓨터 시스템이 분산화됨에 따라 다양한 프로토콜이 개발되었다. 개발된 프로토콜은 원래에 정해진 규격에 따라 정확히 동작해야 하는데 자연어에 근거한 서술식 사양(specification)으로 작성된 프로토콜은 오류(error)를 발견하기 어렵고, 결과를 예측하지 못하거나 원하지 않는 프로토콜 행동들을 야기하게 되었다. 이러한 문제점들을 해결하기 위하여 프로토콜을 체계적으로 개발하는 일련의 방법들이 요구되어 프로토콜 공학이 대두하게 되었다. 프로토콜 공학은 주어진 규격에 적합한 프로토콜을 정의하는 프로토콜 정의, 정의된 프로토콜에 따라 원하는 규격을 정형적인 사양으로 변환하는 프로토콜 사양, 변환된 프로토콜 사양에 따라 실제의 프로토콜을 구현하지 않고 오류를 검출하거나 프로토콜 특성을 분석하는 프로토콜 분석, 실제로 프로토콜을 구현하는 프로토콜 구현과 구현된 프로토콜이 규격에 일치하는 지를 미리 정해진 시험 절차에 따라 시험하여 확인하는 프로토콜 시험으로 구성된다. 분산 시스템의 통신 프로토콜은 여러 개의 층(layer)으로 구성되는 계층 구조를 갖고 있다. 특정 층에서 상위의 층을 상위층, 하위의 층을 하위층, 두 통신이 연결된 시스템간의 같은 층을 peer층이라 한다. 이러한 통신 프로토콜의 사양에는

- 각 층의 목적 및 제공되는 서비스(서비스 사양)
- 각 층의 내부 및 상호 관계 구조
- 엔티티(entity)간의 프로토콜 사양 내용(예: 기능, 메세지, 세부 구현, 성능 사항 등)이 포함된다.

특히, 서비스 사양에는 프로토콜 각 층의 입·출력 내용, 접속·비접속, 송·수신등 기본적인 기능이 포함되며, 단계적 세련화(step-wise(layer-wise) refinement) 기법으로 작성된다.

2.1 프로토콜 사양 방법

프로토콜 공학의 구성 요소중 프로토콜 사양은 프로토콜을 정형적으로 기술하므로, 자연어에 근거한 서술식 사양의 모호함이나 불완전한 점을 보완하여 원하는 동작을 정확하게 표현하는 방법이다. 사양 방법은 프로토콜의 특성을 사건(event) 중심으로 기술하는 트랜지션 모형과 알고리즘 중

심으로 기술하는 프로그래밍 언어 모형 및 이들의 혼합형으로 다음과 같이 구분할 수 있다[6].

- **트랜지션 모형** : 외부의 입력 명령어, 메세지 도착, 내부의 time-out 같은 사건(event)이 발생했을 때 대응되는 처리를 해주고 다음 상태로 전이하는 모형을 의미한다.
  - 상태 기계(state machine) : 유한 개의 상태, 외부 및 내부의 입력, 상태 전이로 구성되며, 현재의 상태에 따라 미리 정해진 입력에 반응한다.
  - 정형 언어(formal language) : 넌터미날 심볼(non-terminal symbol), 터미날 심볼(terminal symbol), 행동 규칙(action rule), 시작 심볼(starting symbol)로 구성된다.
  - SDL : ITU-T에서 통신 시스템 사양을 위해 정의된 언어로 그래픽(graphic) 형태와 구문적(textual) 형태가 있으며, 프로시듀어, 결정, 입력, 출력, 채널, 마크로 등 미리 정해진 템플리트로 구성된다. 통신 시스템 구현 언어인 CHILL 언어로 변환하는 방법과 도구가 개발되어 있다. 최근에는 객체지향 개념을 적용한 SDL이 권고되었다.
  - **패트리넷 모형** : 분산 시스템 모델링에 유용한 패트리넷을 이용한 사양 방법으로, 사양과 분석이 용이한 모형이다.
- **프로그래밍 언어 모형** : 프로토콜이 단순 알고리즘이며, 이 알고리즘은 고급 프로그래밍 언어로 명료하게 기술할 수 있기 때문에 고려된 모형으로 언어의 고급 정도나 추상화 정도에 따라 다르지만 프로토콜을 구현할 때 쉽게 구현할 수 있고, 상태 기계 모형과는 달리 큰 값을 갖는 변수나 파라미터를 쉽게 처리할 수 있는 장점이 있지만, 사양하는 방법에 추상화의 개념이 약하므로 어려움이 있다.
- **혼합형** : 상태 기계 모형과 프로그램의 장점을 합친 것으로 프로토콜의 주요 기능을 상태들로 표현하고, 각 상태마다 변수와 처리 루틴을 추가하므로 입력 파라미터의 값에 따라 수행할 행동(action)을 결정할 수 있게 하는 방법이다.

2.2 프로토콜 분석 방법

2.2.1 분석 대상

프로토콜 공학의 구성 요소중 프로토콜 분석 방법은 프로토콜이 사용자의 규격대로 사양되었는지를 확인(validation)하는 과정이다. 즉, 올바른 프로토콜을 개발하였는 지를 판단하는 과정이다. 프로토콜 정의 및 사양시의 오류를 발견하기 위해 분석하는 항목은 다음과 같다[16].

- 데드록(deadlock)
- 완전성(completeness)
- 종료성(termination)
- 보존성(boundness)
- 성능(performance)

2.2.2 분석 방법

상기 항목들을 검증하는 기법으로 현재 주류를 이루는 방법인 "도달성 분석(reachability analysis)"방법에서는 가능한 상태를 모두 생성하므로 상태 수가 기하급수적으로 증가하여 "상태 폭발 문제"가 발생한다. 이 문제는 프로토콜 사양으로부터 assertion을 생성하는 방법인 수학적 증명(proof) 기법, 부분적인 사양 · 검증 기법(partial verification), 거시적 프로토콜(즉, 큰 granule을 가진 좀 더 추상적인 프로토콜) 기법(abstraction), 분할(partition), 상태들의 축소(reduction), 인공 지능을 이용한 검색 기법(heuristic search) 및 분석의 자동화(분석기 개발)에 의해 다소 해결하고 있다[6].

2.3 암호 프로토콜과 공격

프로토콜은 앞절에서 언급한 것처럼 절차적 규칙을 의미한다. 이 프로토콜중 암호 시스템에 관계된 절차를 암호 프로토콜(cryptographic protocol)이라 할 수 있다. 대개 암호 프로토콜이란 키 분배 프로토콜과 인증 프로토콜을 의미하는데, 이는 키 분배와 인증 방식에 절차적인 사항(프로토콜)이 많이 요구되기 때문이다. 통신 프로토콜은 통신하는 당사자들 간에 오류없는 통신을 위하여 프로토콜을 정의하는 반면, 암호 프로토콜은 절차상 오류 뿐만 아니라, 암호 시스템의 특징인 보안성 누출 여부도 중요한 사항이다. 이 암호 프로토콜과 통신 프로토콜 및 프로그램은 유사한 접근 방식과 개념을 보여주고 있으나 이들을 비교하면 표 2-1과 같다.

표 2-1. 암호 프로토콜, 통신 프로토콜 및 프로그램의 차이점

	암호 프로토콜	통신 프로토콜	프로그램
요구 사양 내용	정확성(안전성 및 인증성)	기능, 성능 및 고장 인성 요구 사양	기능 및 성능 요구 사양
모형화 과정	필요함, 모형화 도구 필요	필요함, 모형화 도구 필요	불필요(실시간, 분산 소프트웨어에서는 필요)
설계 및 구현 방법론	방법론화 안됨	방법론화 부족	구조적 설계, 객체 지향 설계 방법론화
검증과정	피 공격 가능성 검증	테스트 및 증명 기법	테스트 및 증명 기법
테스트 케이스	공격 시나리오	환경 및 운영 시나리오	테스트 케이스
공격개념	공격자에 의한 공격 (공격자에 대한 가정 필요)	하드웨어에 의한 공격 (즉, 메세지 손실, 변형 등 하드웨어 환경에 대한 가정 필요)	바이러스에 의한 공격
에러의 형태	정보 누설, 변조, 파괴	기능, 시간 및 순서 에러	고유, 논리 및 계산 에러
특 성	피 공격성, 비동기성, 병행성, 비결정성, 안전성	비동기적, 병행성, 비결정성	순차성, 반복성, 계산성, 입출력성
지원 도구, 언어	Interrogator, Inast, PA, Prolog, LISP	SDL, CHILL	각종 CASE Tool들

암호 시스템은 알고리즘과 프로토콜로 구성되어 있으며, 원하는 수준의 보안성을 유지할 수 있는 암호 시스템을 설계하기 위해, 안전한 암호 알고리즘도 중요하지만 프로토콜의 설계도 중요하다.

Moore[7]는 시스템 내의 데이터 보안성과 무결성을 유지하기 위해서는 수학적으로 타당한 암호 알고리즘과 프로토콜의 설계와 분석이 필요하다고 주장하였다. 또한, 그는 프로토콜의 잘못으로 인한 보안성 누출에 관해 언급하고, 오류가 있는 프로토콜을 분석함으로써 프로토콜 개발에 대한 올바른 지침을 제공하였다. 그는 키 분배 프로토콜, 디지털 서명, 공증(notarization)등의 비밀(secrecy) 프로토콜들은 기본적으로 공격이 가능하며 오류(failure)가 발생할 수 있음을 보였으며,

완전한 암호 프로토콜은 존재하지 않는다는 것을 보이고 있다. 또한, Moore는 프로토콜의 오류를, (a) 암호 알고리즘 자체의 오류, (b) 암호 알고리즘에 대한 어떤 원리 적용의 실패에 의한 오류, (c) 프로토콜이 제공하는 비밀성의 크기를 과장하는 오류로 분류하였다.

Simmons[1]은 90's Workshop on the Mathematical Concepts(or Principles) of Dependable Systems(독일)에서 Moore가 제시한 프로토콜의 오류에 대해 소개하고 보안 프로토콜의 설계와 분석시 정형적(formal) 사양을 이용해야 하며, 보안 프로토콜의 개발 및 분석의 어려움을 지적하고 있다. 또한, RSA 암호 기법을 바탕으로 한 TMN 프로토콜[3]의 문제점도 지적하였다.

Denning-Sacco[8]는 NS 프로토콜[9]에 대한 공격자의 replay 가능성을 최초로 지적하면서, 임의의 암호 프로토콜은 불안전한 상태에 도달할 가능성이 있다는 점을 반증하였다. 따라서, 암호 프로토콜의 분석 목표(즉, 공격 가능한가?)를 제시해 주고 있다.

3. 사양 및 분석 기법

본 장에서는 암호 프로토콜의 정형적 사양과 분석 방법에 대해 고찰한다. 본 연구에서 고찰한 각 사양 방법들의 분류는 2장에서 논한 통신 프로토콜의 사양 및 분석 방법들의 분류법과 KMM[12]에서의 분류 방법을 적용하였다.

3.1 대수적 접근 방법

3.1.1 Dolev-Yao 기법

공개키 암호에 대한 관심이 집중되고 있었던 시기였던 1980년대 초에 Dolev와 Yao[10]는 두 가지 프로토콜 모형에 대해 보안적 특성을 보여주기 위해 대수적 기법(algebric approach)을 적용하였다. 이들이 사용한 기법은 규칙 기반(rule-based)적 사양모형과 term-rewrite rule을 이용한 정형적이며 오퍼레이션적인 분석 방법이다. Term-rewriting rule은 후속 연구[11,12]들에서도 많이 사용되는 규칙으로서, 일종의 대수의 축소(간략화) 규칙이라 할 수 있다. 이 축소 규칙은 공개 암호 시스템에서 사용하는 알고리즘의 수학적 특성인 $Dec_{key}(Enc_{key}(m)) = m$, $Dec_{s\text{-}key}(Enc_{p\text{-}key}(m)) = Dec_{p\text{-}key}(Enc_{s\text{-}key}(m)) = m$ 관계를 이용해 수식을 간략화하는 것이다.

Dolev-Yao는 공개키 암호 시스템이 수동적인 도청자("passive" eavesdropper)에 효과적이나, 프로토콜을 잘못 설계하면 능동적인 공격자(active saboteur)에게 약점이 있음을 주장하고, cascade 프로토콜과 name-stamp 프로토콜 모형을 제시하여, 이 두 프로토콜이 안전하기 위한 조건을 분석하였다. 이 분석은 첫째로, 사용자가 메세지를 만들기 위해서 각 단계에 적용하는 연산(operation)인 프로토콜 문법(syntax)을 사양하고, 둘째로, 침입자가 평문을 알아내는데 사용할 수 있는 추론 규칙(inference rule)을 사양하는 순서로 진행된다. 그들은 공개키 암호 시스템과 공격자 가정에 대해 언급하고, 심볼과 연산 표기를 정의하고 있으나, 이에 대한 자세한 사항은 본고에서 생략하고, cascade 프로토콜에 대해 사용한 사양과 분석 과정을 보이고자 한다.

- **cascade 프로토콜 일반적 표기**

  사용자 A, B사이에 평문 M을 보내기 위해

  $$A \rightarrow B : (A, E_B(M), B) \quad \text{------} \ ①$$
  $$B \rightarrow A : (B, E_A(M), A) \quad \text{------} \ ②$$

  의 순서로 송수신하는 프로토콜이다. ■

  이 프로토콜의 문제점으로, 능동적인 공격자 Z에 의해 다음과 같은 방법으로 평문 M을 얻을 수 있다.

ⓐ Z가 ①번 단계에서 보내는 중간에서 메세지를 가로채고

ⓑ 가로챈 메세지에서 A를 Z로 바꾸어 $(Z, E_B(M), B)$로 만든 다음 B로 보낸다.

ⓒ 그러면 B에서 $(B, E_Z(M), Z)$로 된 응답이 Z로 되돌아온다.

ⓓ Z는 자신이 갖고 있는 D_Z를 이용하여 평문 M을 얻는다.

● cascade 프로토콜의 사양 정의

〔정의 1〕 two-party cascade protocol T는 유한 스트링의 열로,

$$\widetilde{\alpha}_i \in \{z_1, z_2, z_3\}^* \ (1 \le i \le t)$$

$$\widetilde{\beta}_i \in \{z_1, z_2, z_4\}^* \ (1 \le i \le t \ , \ t' = t \text{ 또는 } t-1)$$

로 정의한다. 사용자 X와 Y에 대해, $\alpha_i(X,Y)$, $\beta_i(X,Y)$를 E_x, E_y, D_x, D_y로 대치시킨 심벌 z_1, z_2, z_3, z_4를 갖는 스트링 $\widetilde{\alpha}_i$, $\widetilde{\beta}_i$라 하면, $\alpha_i(X,Y) \in \{E_X, E_Y, D_X\}^*$와 $\beta_i(X,Y) \in \{E_X, E_Y, D_Y\}^*$이다. ■

〔정의 2〕 T를 $\{\widetilde{\alpha}_i, \widetilde{\beta}_j \mid 1 \le i \le t, \ 1 \le j \le t'\}$에 의해 지정된 two-party cascade 프로토콜이라 하고 X, Y를 사용자라 하면,

$$N_1(X,Y) = \alpha_i(X,Y)$$
$$N_{2j}(X,Y) = \beta_i(X,Y)\ N_{2j-1}(X,Y) \ (1 \le j \le t')$$
$$N_{2i+1}(X,Y) = \alpha_{i+1}(X,Y)\ N_{2i}(X,Y)$$
$$(1 \le i \le t-1)$$

로 정의하며, X가 Y에게 평문 M을 보낼 때 교환되는 메세지는 $N_i(X,Y)M$이다 $(i = 1, 2, \cdots, t+t')$. ■

● cascade 프로토콜의 보안성 개념 정의

E를 모든 E_A의 집합, D를 모든 D_A의 집합이라 하고 X, Y, Z를 사용자명이라 할 때

〔정의 3〕 T를 $\{\widetilde{\alpha}_i, \widetilde{\beta}_j\}$에 의해 지정되는 two-party cascade 프로토콜이라 하면

$$\Sigma_1(Z) = E \cup \{D_Z\},$$
$$\Sigma_2 = \{\alpha_i(A,B) \mid \text{for all } A \ne B \text{ and } i \ge 2\},$$
$$\Sigma_3 = \{\beta_i(A,B) \mid \text{for all } A \ne B \text{ and } i \ge 1\}$$

로 정의하고, 만일, $N_i(X,Y)$내에 $\overbrace{\gamma N_i(X,Y)} = \lambda$인 $\gamma \in (\Sigma_1(Z) \cup \Sigma_2 \cup \Sigma_3)^*$인 γ가 존재하면, T는 "불안전(insecure)"하다. 그렇지 않으면, T는 "안전(secure)"하다. ■

● 보안 프로토콜의 특성

〔정의 4〕 $\pi \in \{E,D\}^*$가 스트링이고 A는 사용자 이름일 때, $D_A \in lt(\pi) \Rightarrow E_A \in lt(\pi)$라면, π가 A에 대해서 "balancing property"를 갖는다고 말한다. 이 balancing property는 secure cascade 프로토콜에 본래부터 가지고 있는 것이다. ■

〔정의 5〕 X, Y를 사용자라 하고, 모든 $i \ge 2$인 경우에 대해 $\alpha_i(X,Y)$가 X에 대해 balancing property를 갖고, 모든 $i \ge 1$인 경우에 대해 $\beta_i(X,Y)$가 Y에 대해 balancing property를 갖는다면, two-party cascade 프로토콜 $T = \{\widetilde{\alpha}_i, \widetilde{\beta}_j\}$는 "balanced cascade 프로토콜"이다. i, $j \ge 1$에 대해 $\alpha_i(X,Y)$, $\beta_j(X,Y)$는 축소된 형태이다. ■

〔Lemma 1〕 Z가 사용자 이름이고 T가 balanced cascade 프로토콜이면, $(\Sigma_1(Z) \cup \Sigma_2 \cup \Sigma_3)^*$내의 모든 스트링 η에 대해, $\bar{\eta}$는 모든 $A \ne Z$에 대해서 balancing property를 갖는다. ■

〔정리 1〕 X, Y는 두 구별되는 사용자이고, two-party cascade 프로토콜 $T = \{\widetilde{\alpha}_i, \widetilde{\beta}_j\}$가 안전할 필요·충분 조건은 $lt(\alpha_i(X,Y)) \cap \{E_X, E_Y\} \ne \phi$이고 T가 balanced이다. cascade 프로토콜 T가 doubly verified 되었다는 것은 어떤 i에 대해, $lt(N_i(X,Y) \subseteq \{E_Y, D_Y, D_X\}$이고, $j \ge 2$에 대해 $lt(N_j(X,Y)) \subseteq \{E_X, D_X, D_Y\}$인 경우를 말한다. ■

〔정리 2〕 모든 doubly verified 프로토콜은 불안전하다. ■

정리 2는 수신자 Y가 암호화된 메세지 M을 복호화할 수 있고, 송신자 X는 X로 돌아온 메세지의 어떤 부분을 복호화함으로써 M을 획득할 수 없다면 안전한 cascade 프로토콜이라는 것을 의미한다. 즉 X가 M을 첫 번째 보내고 난 후 없애 버린다면 M을 재구성할 수 없어야 한다는 의미이다. 이 정리에 의해 Diffie-Hellman 프로토콜[13]은 공개키 인증 획득에 대해 안전하지 않다(즉, $E_B(D_A(M)) \neq E_A(D_B(M))$).

이상과 같이 프로토콜을 무한 갯수의 상태 공간으로 사양화하므로써 현실적이며, 프로토콜 공격자는 잘 정의된 규칙을 가지고 우회 공격을 할 수 있음을 보이고 있다. 특히, cascade 프로토콜은 암호 및 복호 연산자가 대응할 경우에만 안전하다는 점을 알 수 있다. 그러나, 초기 메세지의 비밀성(secrecy)만을 프로토콜 정확성의 기준으로 하여, 정확성 결정을 위한 최적 알고리즘을 찾고 있다는 단점을 갖고 있다. 또한, 메세지 및 객체의 인증성(authentication)문제는 다루지 않고 있으며, 메세지를 보내고 그 메세지를 응답하는 핑퐁형 프로토콜만을 분석할 수 있으므로 일반성이 결여되어 있다.

3.1.2 Longley-Rigby 방법

Longley-Rigby[14]는 키 관리 기법에서 보안 누출(security flaw)을 찾기 위해 규칙 기반적 사양 모형을 사용하여 사양하고 전문가 시스템 기법을 이용해 분석하는 방법을 제시하였다. 분석을 위해, 프로토콜의 보안 오류를 찾는 Prolog 프로그램을 개발하였다. 이들의 방법에서는 분석 시의 자료 구조로서 검색 트리를 이용하여 공격자의 공격 가능성(즉, 정보의 누설)을 검색하고 있다. 검색 트리의 루트는 검색 목표(정보)이고, 노드는 공격자에게 필요한 정보들을 나타낸다. 아래 복호 함수에 대한 검색 트리는 그림 3-1과 같다.

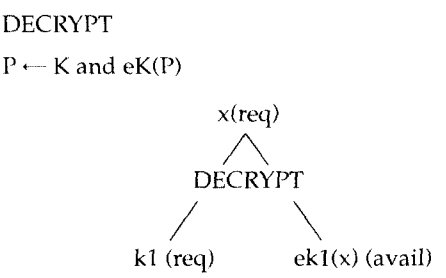

그림 3-1. 복호 함수의 검색 트리

여기서 available 노드는 이미 공격자가 알고 있는 정보를 나타내고, required 노드는 공격자가 알아야 할(모르는) 정보를 나타낸다. 따라서 오른쪽 노드의 정보(k1)를 알면 루트 노드의 x의 평문을 알 수 있게 된다. 따라서 암호 프로토콜의 검색 트리를 만들고 이에 대해 required 노드의 값을 알아낼 수 있는 방법만 안다면, 또는 이전 값으로 구해 낼 수 있다면, 암호 프로토콜에 보안 누출이 있다는 것을 보여 줄 수 있다. 또한 이들은 Jones[15]가 처음 제시한 Tagged Key Management기법에 Draft 개념을 확장 · 제시하였다. 그러나 Longley-Rigby의 패키지는 "what if"에 대한 개발 지원을 제공하지만 다음과 같은 단점이 있다: (a) 자동 검증 시스템이 아니다. (b) 공격자가 쓸 수 있는 모든 가능 데이터를 제공하는 기능이 없다. (c) 완전히 이론적으로 분석하고 찾을 수 있는 방법을 보장할 수 없다.

3.2 논리적 접근 방법

일반적으로 논리적 접근 모형들은 모달 논리(modal logic)를 이용하여 암호 프로토콜에 관련된 각종 정보의 믿음(belief)과 지식(knowledge)을 표현하며, 논리를 전개함으로써 프로토콜을 검증한다. epistemic logic은 지식의 논리,

doxastic logic은 믿음의 논리가 된다. 이 접근 방법의 대표적 모형은 BAN(Burrows-Abadi-Needham) 모형[16]으로 알려져 있고, 이에 대한 많은 보완 연구[17,18,19,20,21] 결과들이 발표되었다. 이 모형에서는 행위(action)와 믿음의 논리를 이용하고 있으며, 서로 인증하려는 각 파트는 정직하며 프로토콜을 충실하게 따른다고 가정하고 있다. 이런 가정으로부터 프로토콜이 행할 수 있는 여러가지 수준(order)의 믿음을 기술하여 공리들을 구하고 이를 이용해 분석하였다. 단점으로서, 프로토콜의 각 파트마다 자신의 사양이 필요하며(대부분의 논리적 접근 방법의 단점이다.), 믿음의 상태(state of belief)만을 프로토콜 정확성의 기준으로 하고 있다. 즉, 지식의 상태는 고려하지 않고 있다. 또한, 비 정형적인 이상화(idealization)과정을 통해 프로토콜을 사양하므로, 분석자는 철저한 프로토콜의 이해가 요구된다. 또한, 암호 기능의 temporal reasoning과 의미론적 표현을 다루지 않았고, 믿음은 에이전트가 정상일 때만 지식이 된다.

3.2.1 BAN 논리

BAN 논리는 "인증 논리"라고도 하는데, 인증 논리는 인증 프로토콜에 관한 정형적 사고를 위해 추론 규칙의 집합에 일치함을 기초로 하는 논리적 계산법이다. 이 논리는 암호 프로토콜 사양 및 분석시에 정당성(correctness), 효율성(efficiency) 및 적용성(adaptability)을 높일 수 있다는 장점을 갖고 있다. 프로토콜이 보안적 목표를 만족하는지를 증명해 줄 수 있고, 만약 설정된 목표에 부합하지 않는다면, 무엇 때문인지를 보여줄 수 있다(정당성). 보안 프로토콜에서 메세지의 암호화 없이도 보안 목표를 성취할 수 있다면, 암호화 과정을 없애므로서 프로토콜을 쉽게 사양할 수 있고, 분석도 쉽게 할 수 있다(효율성). 또한 논리는 프로토콜이 실제적인 상황에 사용될 수 있는지 판단하기 위해 프로토콜의 가정을 밝히는데 도움이 될 수 있다(적용성).

BAN 논리에서, 프로토콜은 프로토콜에 개입하고 있는 각 통신 당사자(principal) *P*의 관점에서부터 분석된다. *P*에 의해 수신된 각 메세지는 *P*에 의해 수신된 이전의 메세지와 *P*에 의해 보낸 메세지에 대한 관계(relation)로 고찰된다. 문제는 보내진 메세지와 수신된 메세지를 기초로 통신 당사자가 "무엇을 믿어야(belief) 하느냐"이다. 통신 당사자가 공식의 진실을 확신할 때 또는 그것이 사실이라는 결론에 도달할 때, 통신 당사자는 "그것을 믿는다(believes it)"라고 하며, believe, see, control 등과 같은 술어를 사용하여 논리를 전개한다. BAN 논리에는 오직 현재와 과거 표기밖에 없고, 메세지에는 타임 스탬프가 없다고 가정한다. 이 BAN 논리를 암호 프로토콜 분석에 적용하기 위해서는 다음과 같은 단계를 거친다.

〔단계 1〕 논리가 알려진 상태(known state)로부터 진행되게 하고, 목표에 실제로 도달될 수 있는지 확인할 수 있게 하기 위해서, 가정과 목표를 심벌 표현인 공식(formulas 또는 statement)으로 표현한다.

〔단계 2〕 프로토콜의 각 단계들을 심벌 표현인 공식으로 변환한다(이상화 메세지로 변환).

〔단계 3〕 가설(postulate)이라 불리는 추론 규칙 집합을 적용한다. 가설은 가정으로부터 출발하여 중간 공식을 통해 인증 목표에 도달된다.

단계 1과 단계 2에서 보여지는 것 처럼, BAN 논리를 적용하기 위해서는 통신 당사자의 모든 메세지와 행위는 공식으로 변환되어야 하는 데, 이 공식은 *"P believes X"*와 같은 표기로 나타내며, *"P*가 *X*를 믿는다"는 의미를 갖는다. BAN 논리에서 사용하는 공식은 여러가지가 있고, 그중 *"P see X"*는 "통신 당사자 *P*가 *X*를 갖는 메세지를 수신했다"는 의미이고, *"P said X"*는 "과거의 어떤 순간

에, X가 있는 메세지를 보냈다는 것을 P는 알게 된다"는 의미이다. 단계 3은 추론 규칙을 적용하는 단계로 BAN 논리의 추론 규칙에는 메세지 의미 규칙(message-meaning rule), 논스 증명 규칙(nonce-verification rule)과 판결 규칙(juris-diction rule)등이 있다. 메세지 의미 규칙에서는 P 자신이 Q와 비밀키 K를 공유한다는 사실을 믿고, K로 암호화된 자료 X가 담긴 메세지를 수신했다면, Q가 언젠가 자료 X를 보냈었다는 것을 P는 확실히 믿을 수 있다. 이 규칙은 다음 두 가지 중요한 가정하에서 유용하다. (a) X는 반드시 키 K가 암호화를 위해 사용되었음을 증명할 수 있는 인식 자료를 갖고 있어야 한다. (b) P는, 메세지가 자신이 이전에 보냈던, K를 아는 믿을 만한 어떤 통신 당사자에 의해 보내졌던 메세지의 재전송이 아님을 말할 수 있어야 한다. 논스 증명 규칙은 Q가 언젠가 X를 보냈었다는 것을 P가 믿는다면, P는 Q가 X를 믿었었다는 것을 알(믿을) 수 있고, X가 새로워(fresh)졌다는 것을 P가 믿는다는 부가적인 주장이 있다면, Q가 현재도 X를 알고(믿고) 있다는 점을 P는 믿어야 한다는 것을 의미한다. 판결 규칙은 통신 당사자가 문장에 대해 판결권을 가짐을 나타낸다. Q가 X의 사실 여부에 대한 판결권을 갖는다는 것을 P가 믿고, Q가 X를 사실이라 믿는다는 것을 P가 믿는다면, 역시 P는 X를 믿어야만 한다. 이는 P가 연관되는 한 Q는 문제에 대해 권위이기 때문이다.

위의 추론 규칙과 다른 가설도 있고, 공개키와 비밀키에 관계된 가설이 BAN에 의해 제시되었지만, 나머지 가설에 대해서는 생략한다. 암호 프로토콜 적용 사례로 Kerberos 프로토콜, RPC 프로토콜[22], NS 프로토콜[9], ITU-U X.509 프로토콜[23]에 대해 BAN 논리를 적용하고 있다. 본고에서는 비밀키 방식의 NS 프로토콜에 대해 BAN 논리 단계 1인, 기본 가정을 표 3-1에, 인증 목표를 표 3-2에 보인다[40]. BAN 논리의 단계 2는 프로토콜의 각 단계를 공식으로 변환하는 것으로, 메세지를 이상화 메세지로 변환하는 것으로 그 예를 표 3-3에 보인다.

표 3-1. Needham-Schroeder 프로토콜의 명시적 가정(assumption)

A believes	B believes	S believes
$A \overset{K_A}{\longleftrightarrow} S$	$B \overset{K_B}{\longleftrightarrow} S$	$A \overset{K_A}{\longleftrightarrow} S,\ B \overset{K_B}{\longleftrightarrow} S$
S *controls* $A \overset{K_{AB}}{\longleftrightarrow} B$	S *controls* $A \overset{K_{AB}}{\longleftrightarrow} B$	$A \overset{K_{AB}}{\longleftrightarrow} B$
S *controls* $fresh(A \overset{K_{AB}}{\longleftrightarrow} B)$	$fresh(N_B)$	$fresh(A \overset{K_{AB}}{\longleftrightarrow} B)$
$fresh(N_A)$		

표 3-2. Needham-Schroeder 프로토콜의 인증 목표

A believes	B believes
$A \overset{K_{AB}}{\longleftrightarrow} S$	$A \overset{K_{AB}}{\longleftrightarrow} S$
B *believe* $A \overset{K_{AB}}{\longleftrightarrow} B$	A *believe* $A \overset{K_{AB}}{\longleftrightarrow} B$

표 3-3. Needham-Schroeder 프로토콜의 이상화 메세지

Message	Idealized Message
1. $A \rightarrow S$: A, B, N_A	
2. $S \rightarrow A$: $\{N_A, B, K_{AB}, \{K_{AB}, A\}K_B\}K_A$	$\{N_A,\ A \overset{K_{AB}}{\longleftrightarrow} B,\ fresh(A \overset{K_{AB}}{\longleftrightarrow} B),\ \{A \overset{K_{AB}}{\longleftrightarrow} B\}K_B\}K_A$
3. $A \rightarrow B$: $\{K_{AB}, A\}K_B$	$\{A \overset{K_{AB}}{\longleftrightarrow} B\}K_B$
4. $B \rightarrow A$: $\{N_B\}K_{AB}$	$\{N_B,\ A \overset{K_{AB}}{\longleftrightarrow} B\}K_{AB}$
5. $A \rightarrow B$: $\{N_B - 1\}K_{AB}$	$\{N_B,\ A \overset{K_{AB}}{\longleftrightarrow} B\}K_{AB}$

BAN 기법의 적용 단계 3은 단계 2에서 변환된 프로토콜 각 단계의 이상화 메세지에 가설(postulate)이라 불리는 추론 규칙 집합을 적용하여, 가설이 가정으로부터 출발하여 중간 공식을 통해 인증 목표(표 3-2)에 도달하는지를 분석하는 것이다. 추론 규칙을 적용하여 인증 목표에 도달하는 과정은 본고에서 생략한다.

BAN 논리는 기본 가정과 인증 목표를 BAN 논리로, 암호 프로토콜을 이상화 메세지로 변환하여 논리를 전개해 나간다. 만약 이상화 메세지가 가설에 의해 인증 목표에 도달할 수 없다면 이 프로토콜에는 불안전한 요소를 포함하고 있다는 의미가 된다. 하지만 BAN 논리중 이상화 메세지 변환 방법은 잘 정의되어 있지 않고 있어 앞으로 연구가 필요한 분야이다. BAN 논리 가설을 적용하는 것은 복잡하고 에러가 있을 수 있으므로, 이를 지원하는 "Jape 에디터"가 개발되어, 증명의 대화식 구성에 도움을 주어 사용자의 명령에 의해 가설이 자동적으로 적용된다[24]. Jape 에디터는 backward reasoning 기능을 제공하며 NRL Protocol Analyzer[2,11,12], Interrogator[2,25] 및 Queens 대학의 사양 모형들[26,27]에 영향을 주었다. Backward reasoning은 주어진 문장(결론)으로 부터 가설로 역 추론하는 기능을 의미하며, foreward reasoning보다 쉬울 경우가 있다.

3.2.2 Syverson 모형

Syverson[17]은 KPL Free logic(Hintikka의 possible world semantic logic)을 이용한 추상적이고 형이상학적인 사양 모형을 제시하였다. 단어 수준의 지식뿐 아니라 명제 지식을 직접 표현할 수 있는 사양 모형으로서, 2 종류의 지식(명제의 진위와 메세지)을 명확히 하고 있으며 NS 프로토콜[9]을 사례로 분석하고 있다. 그러나, 프로토콜의 정당성을 공식화하는 것보다는 논리의 의미론의 중요성만을 강조하고 있다.

3.2.3 Syverson-Meadow 모형

Syverson-Meadow[19]는 템포랄 논리에 기반을 둔 요구(또는, 가정)명세 작성용 언어로서, NRL Protocol Analyzer[2,11,12]의 언어를 위한 요구 사항 사양 언어로 이용하고 있다. ISO 2 패스 메세지 인증 프로토콜을 사례로 하여, NRL Protocol Analyzer를 위한 요구 사항 명세 결과를 보이고 있다. 이 사양 모형은 프로토콜의 통합적 단일 사양을 위한 사양 언어로서 적합하다. 또

한, 분석하려는 프로토콜과는 독립적인 사양이 가능하다. 그밖에 Rangan[20]과 Snekkenes[21]은 BAN 모형[16]의 문제점을 지적하고 있다.

3.3 논리 및 대수적 접근 방법

Beiber[28]는 Syverson[17,18]처럼 프로토콜 분석을 위한 정형적인 오퍼레이션날 사양 모형을 제시하였으며, 추상적이고 형이상학적 수준의 연구이다. 이는 암호 프로토콜의 암호 함수와 악의적인 에이전트, 암호 함수의 epistemic 특성을 사양하고, 암호 프로토콜에 대한 정형 사양을 구축한다. 특히, 분산 시스템의 지식 지향(knowledge-oriented) 사양이라할 수 있고, 시간, 통신, 지식의 논리에 대한 의미론(semantics)을 제공한다. 나쁜 환경 하에서 통신 논리를 제공하는, Hintikka가 제안한 epistemic 논리를 확장한 CKT5 언어(epistemic, temporal, communication modal operator들을 포함한 논리)를 이용해 프로토콜 수행중 에이전트가 갖는 지식의 상태와 무지(ignorance)의 상태를 공식화하고 있다. 그는 프로토콜 검증시, 믿음보다는 지식을 이용하며 객체의 인증성은 논하지 않고, 비밀성과 메세지의 인증성만을 논하고 있다. 암호 함수는 $Key \times Mesg \rightarrow Mesg$, 즉 $(Key, Mesg) \rightarrow Mesg' = Enc(Key, Mesg)$으로 정의하고, 복호 함수는 $(Key, Mesg) \rightarrow Mesg' = Dec(Key, Mesg)$로 정의한다. 이들 함수간의 관계는 $\forall Key, \forall Mesg, Dec(Key, Enc(Key, Mesg)) = Mesg$로 표현하고, 암호 시스템이 이상적(ideal)일 필요·충분 조건은 $Enc(Key, Mesg)$와 $Dec(Key, Mesg)$가 "실제로" 키 Key의 지식이 없이 역변환할 수 없어야 한다고 정의하고 있다. 또한, 지식(knowledge), 무지(ignorance), 학습(learning)과 같은 epistemic 개념을 도입하고 있다.

Snekkenes[29]는 NS 비밀 키 분배 프로토콜[9]과 비슷한 KP 프로토콜을 제시한 후 BAN[16]과 Beiber 사양 모형[28]을 이용하여 비밀키 분배에 적당하다는 것을 보이고 있다. Merritt-Wolper[30]와 Toussaint[31]도 비슷한 형태의 기법을 제시하였다.

3.4 상태 변환적 접근 방법

이 접근 방법에는 대수적인 사양 방법과 논리적인 방법이 있을 수 있으며, 암호 프로토콜 분석에 가장 잘 알려져 있는 사양 방법이다. 또한, 분석 소프트웨어 도구까지 개발된 사양 방법으로서 표 3-5에서 Interrogator, NRL protocol analyzer, Inatest에 대한 사양 방법과 그에 대한 도구를 요약·비교하고 있다. 논리와 상태 변환적인 사양 방법을 혼합한 방법으로 Abadi-Tuttle[32]의 사양 모형도 알려 져 있다.

3.4.1 Interrogator[2,25]

(1) 특 성

Interrogator는 프로토콜에서 주어진 불안전 상태로 도달이 가능한가를 backward이고 깊이 방향 우선 순서로 검색하는 Prolog 프로그램이며, 검색을 실패했을 때는 프로토콜의 검증 기능은 없다. 프로토콜을 communicating state machine으로 보고 있으며, 메세지의 송·수신 사건에 의해서 상태가 전이된다. 모든 공격 행동은 통신 채널 버퍼를 통해 이루어지며, 초보적인 수준의 그래픽 출력을 제공한다.

(2) 구 성

- Knowledge 섹션 : 공격자의 최종 지식 및 공격 규칙을 포함한다.
- Reachability 섹션 : back-tracking을 이용한 공격 경로를 분석한다.
- Protocol 섹션 : 분석할 프로토콜에 대한 사양이며 분석기(Interrogator)의 입력이 된다. 트랜지션(transmit, receive), 초기

상태, 공격자의 지식, 메세지 포맷, 메세지 타입, 키관계, 키 및 목표 상태를 입력으로 한다.

- 시나리오 섹션 : 공격 시나리오 생성, 공격자의 초기 지식 및 제약 조건이 기술된다.
- Algebra 섹션 : 논리식의 간략화에 이용된다.

(3) 분석 프로토콜

NS 프로토콜[9]에 대한 Denning-Sacco 공격[8]과 TMN[3]의 위장 공격을 분석하였다.

(4) 단 점

프로토콜에 관여한 각 파트의 내부에 있는 암호 함수의 작동에 의해서는 상태 전이가 안되며, 오직 송·수신 사건에 의해서만 상태가 전이된다. 즉, 프로토콜의 세부적인 상태 전이를 사양할 수 없는 매우 추상적인 사양 모형이다. 사용자의 직관과 능력에 의존하여 프로토콜 분석시에 발생하는 상태 폭발 문제를 해결하고 있다. 프로토콜의 특성인 병행성(concurrency), 비 동기성(asynchronous) 및 비 결정성(non-determinism)을 사양 및 분석할 수 없다. 프로토콜이 문자적인 형태로 사양되므로 가시성이 결여되어 있다.

3.4.2 NRL Protocol analyzer[2,11,12]

(1) 특 성

Dolev-Yao[10]의 term rewrite rule을 이용하여 Prolog로 구현된 자동화된 분석기이다. 즉, 공격자가 가진 부분적인 지식을 표시하기 위해 term rewrite rule을 이용해 대수적으로 프로토콜을 사양하고 분석하는 프로그램이다. 공격자 관점에서 프로토콜의 사양을 기술하며, 프로토콜의 정확성을 사양 언어 내의 단어의 비 획득성(unobtainability 또는 비 도달성) 여부로서 판단하고 있다. 분석기를 사용할 때, 자유롭게 자동 모드와 반자동 모드의 전환이 가능하다. 원하는 불안전 상태를 기술하고 그 상태에 이르는 모든 중간 과정을 검색함으로써 공격 시나리오를 획득하고 검증할 수 있으며, 분석할 프로토콜을 공격자가 메세지(word, belief, event)를 생산하는 기계로 간주한다. Partial query, language checker 및 static unifier 기법들을 통해 분석시 상태 폭발을 줄이고 분석 속도를 향상시킬 수 있다.

(2) 구 성

- Transition 규칙 : 정상 행동 규칙 및 시스템의 물리적 고장 발생의 규칙을 아래와 같은 형식으로 기술한다.(여기서, +는 1회 이상 반복됨을 의미한다.)

  If: {pre-condition}$^+$,
  then: {post-condition}$^+$,
  EVENT: {event-statement}$^+$
- Operation : 정상적 수행 기능을 기술한다.
- Atom : 프로토콜 내의 단어의 구성 요소들을 기술한다.
- Rewrite rule : 오퍼레이션의 간략화 규칙을 기술한다.

(3) 분석 프로토콜

TMN[3]의 위장 공격과 BM 프로토콜[35]을 분석하였다.

(4) 단 점

분석은 사용자의 직관과 능력에 의존하며, 사양 결과를 읽기가 어렵다(readability 결여). 주어진 상태의 안전성 여부는 보일 수 있지만, 어떤 상태가 불안전한 지는 보일 수 없다. Simmons의 공격을[1] 분석할 수 없으며, 분석자는 관심 있는 불안전 상태와 시퀀스를 분

석하려는 프로토콜 사양 내에 기술해야 하며, 분석할 때마다 사양 부분을 수정해야 한다. 프로토콜의 병행성, 비 동기성 및 비 결정성을 사양하고 분석할 수 없다. 프로토콜이 문자적인 형태로 사양되므로 가시성이 결여되어 있다.

표 3-5. 대수 및 상태 변환적 사양 모형들의 비교

	Meadow[9,12,16,21]	Millen[16,19]	Kemmerer[16,17,18]
사양언어	Prolog	Prolog	Ina Jo
분석기 명칭	NRL Protocol Analyzer	Interrogator	Inatest
분석기 구현	Prolog interpreter	Prolog interpreter	YACC, LISP interpreter
상태검색 방향	Backward(Breadth-first order)	Backward(Depth-first order)	Forward
특성	- Term-rewriting rule[4] - 주어진 불안전 상태의 도달성 증명(어떤 상태가 불안전한지 결정 못함)	- Communicating state machine - 주어진 불안전 상태의 도달성 증명(프로토콜 검증 기능 없음)	- Formal Language - 소프트웨어 테스팅 기법 적용(귀납적 테스팅) - 테스트 사례 = 침입 시나리오, 소프트웨어 = 프로토콜)
분석 프로토콜	TMN[3](가장공격 발견), BM[35]	TMN(가장공격 발견), NS[9]	TMN(Simmon 공격 발견)
단점	- RSA형 암호기법 모형화 못함 - 사양의 가시성 낮음 - 사용자는 높은 수준의 프로토콜 지식과 직관 필요 - 보안 요구사항의 사양용 언어 필요 - 병행성, 비 결정성 표현 못함	- 반 자동적 사용 - RSA형 암호 기법 모형화 못함 - 파트의 내부 상태 변화 표현 못함 - 추상적 수준의 프로토콜 모형화 - 문자적 표현(가시성) - 병행성, 비 결정성 표현 못함	- 수동적 사용 - 사용자의 직관 필요 - 침입자가 모든 침입 행위를 기술 - 병행성, 비 결정성 표현 못함

3.4.3 Inatest[2,33,34]

(1) 특 성

소프트웨어 공학 분야에서 잘 알려진 정형 사양(formal specification) 기법을 암호 프로토콜의 사양과 검증(verification)에 이용한 사양 모형이다. First-order logic인 Ina Jo 언어를 정형 사양 언어로 이용하며, YACC와 LISP를 이용해 문자적 수행(symbolic execution)을 통해 분석하였다. 프로토콜의 각 파트의 상태는 변수 및 상수로 사양하였고, 프로토콜의 규칙들은 상태 전이 규칙으로 사양하며, 암호 알고리즘 및 가정들을 공리로써 사양하였다.

프로토콜이 만족해야 할 특성은 불변(invariant)이 되고 정리(theorem)는 분석할 때 자동으로 생성된다. 분석할 때의 기본 가정으로서, 공격자가 이미 아는 정보, 알아야 할 정보, 공

격자의 행동 및 security에 대한 정의는 반드시 필요하며, 모두 Ina Jo로 기술하고 있다. 귀납법으로 프로토콜을 검증하고 있으며, 분석할 프로토콜을 테스팅 할 프로그램으로 볼 때 공격 시나리오는 테스트 케이스와 같은 개념이다. 모든 상태 공간을 검색하지 않고도 분석이 가능하다.

(2) 구 성

- Type : 각 변수의 형을 선언한다.
- Constant : 각 상수의 형을 선언한다.
- Axiom : 암호 함수들에 관한 공리을 선언한다.
- Variable : 변수을 선언한다.
- Criterion : 각 상태에 대한 만족해야 할 요구 사항을 기술한다.
- Initial : 초기의 요구 사항을 기술한다.
- Transform : 분석될 프로토콜의 규칙을 기술한다.

(3) 분석 프로토콜

TMN 프로토콜[3]에 대한 Simmons 공격[1]을 분석하였다.

(4) 단 점

비밀성(secrecy)만을 프로토콜의 정확성의 기준으로 하였고, 공격자는 모든 공격 행위와 시나리오를 프로토콜과 함께 기술해야만 한다. 프로토콜의 병행성, 비 동기성 및 비 결정성을 사양 및 분석할 수 없으며 프로토콜이 문자적인 형태로 사양되므로 가시성이 결여되어 있다.

3.5 의미론 및 대수적 접근 방법

Woo-Lam[36]은 인증 프로토콜을 의미론 및 대수론을 이용하여 분석하였다. 기존 사양 모형들에서의 프로토콜의 정당성(correctness)에 대한 불확실한 정의를 지적하였고, 정형성과 직관성간에 갭이 존재한다는 점을 지적하였다. 즉, 기존의 정형화 방법들에 대해 견실성(soundness)과 완전성(completeness)을 구분할 수 없음을 들어 비평하고 있다. 이에 대한 대안으로서, Woo-Lam은 의미론적 사양 모형을 이용해 비밀성과 인증성을 동시에 고려한 정당성 개념을 정형적으로 정의하였고, 프로그램과 유사한 프로토콜의 사양을 생성하고 있다.

3.6 패트리넷을 이용한 접근 방법

패트리넷(Petri nets, PN)은 1962년 독일 본 대학의 Carl Adams Petri의 박사 학위 논문에서 처음 제시되었다. C. A. Petri는 컴퓨터 시스템내의 비동기적(asynchronous) 부품들 사이의 통신 이론을 세우기 위해, 시스템 내의 사건들 사이의 인과 관계(causal relationship)를 가시적(그래프 형태)으로 모형화 하였다. 그후 많은 연구자에 의해 패트리넷에 관한 이론 연구와 병행(concurrent) 시스템의 모형화와 분석 연구에 사용되었다[37,38,39]. 또한 패트리넷은 전산학(병행 시스템 모형화 및 분석), 산업 공학(자동 제어, 공장 자동화의 병행성 분석) 및 통신공학(통신 프로토콜의 사양)등에서 응용이 활발하다.

Varadhrajan[40]과 Sassone-Varadhrajan[41]은 패트리넷을 이용하여 컴퓨터 보안의 한 분야인 정보 흐름 모형(information flow modeling, access control modeling)을 사양하였으며, 정보 흐름(또는 접근 제어)을 사양하기 위해 패트리넷을 확장한 Security넷을 제시하였고, 특히, 비밀 등급을 갖는 정보를 토큰으로 사양하였으며, Denning과 Bell-Lapadula 모델에서 제시한 액세스 제어 규칙은 트랜지션의 점화 규칙을 수정함으로써 사양하였다. 이들은 일반적인 정보 흐름적 사양 모형으로서, 암호 프로토콜의 사양과 분석에는 적용하기 어렵다.

Nieh-Tavares[26]와 Stal-Tavares-Meijer[27]

는 캐나다 Queens 대학에서 수행한 암호 프로토콜 분석 기법으로, 유색 패트리넷(Colored Petri Net)[42]을 이용하여 BM[35], MM[43], NS[9], Hwang[44] 및 ISO 프로토콜[45]의 공격 가능성을 분석하였다.

국내에서는, 자신들이 제안하는 키분배 프로토콜에 오류가 없다는 것을 보이기 위해, 패트리넷을 이용한 사례가 있다[46].

3.6.1 Nieh-Tavares사양 모형[26]

(1) 특 성

유색 패트리넷을 이용하여 암호 프로토콜의 정형 사양 모형을 제시하고 분석하여 보안성의 누출을 밝혔다. 이 사양 모형은 프로토콜에 대한 좋은 가시성과 층 추상화(layered abstraction)를 갖는 정형 명세(formal description)를 생성한다. 특히, 이 명세는 데이터 흐름 사이의 제약과 관계를 가시화 한다. 공격자의 공격을 공식화하기 위해 공격자를 사양하여 공격자가 없는 프로토콜 엔티티에 삽입하고 암호 프로토콜에 대한 테스트 케이스를 생성한다. 테스트 케이스를 모두 생성하는 과정과 특별한 보안 기준을 위반하는 상태를 찾기 위해 완전 공격 테스트(exhaustive penetration test)를 하는 과정을 보이고, 이를 이용하여 프로토콜의 보안 취약점을 밝힌다. 이 모형은 공개키 시스템과 비밀키(공통키) 시스템 양쪽 모두에 적용이 가능하다.

(2) 사양 및 분석 방법

사양을 위해 프로토콜 엔티티와 공격자 엔티티로 나누어 처음에는 공격자 엔티티가 없는 프로토콜 엔티티만을 사양하고, 이후 공격자 엔티티가 있도록 사양한다. 프로토콜 사양에서 프로토콜 엔티티에 의해 수행되는 모든 기능은 오류가 없고, 두 프로토콜 엔티티 사이에 메세지는 고장 없는 채널을 통해 전달된다고 가정한다. 사양을 위해 프로토콜 요소와 패트리넷의 요소와의 관계를 표 3-6과 같이 설정한다.

표 3-6. Queens 대학 사양 모형의 사양 기준

프로토콜 요소	패트리넷 요소
Process	트랜지션
Data input/output medium	플레이스
Data item	토큰
Data flow relationship	방향성 호선

위의 관계를 이용하여 프로토콜의 구조적인 형태를 보이는 개념 수준(conceptual level)으로 사양하고, 이를 좀더 구체적인 데이터와 기능까지 보이는 기능 수준(functional level)으로 상세화 한다. 이 사양에 대한 공격 행위는 공격자의 사양으로서 나타내 진다. 공격자는 채널 상의 메세지를 추가, 변경, 삭제하는 행위를 수행하므로, 공격자 엔티티는 통신하는 두 프로토콜 엔티티 사이의 채널 상에 존재하게 되고, 또한 공격자 엔티티는 임의의 프로토콜이 갖는 보안 기준을 공격하는 요소(manipulate, substitute, copy, paste, pass, delete operation)를 반영하여 사양한다. 분석을 위해 프로토콜이 갖는 보안 특성을 분석하여 보안 기준을 마련하고 완전 공격 테스트를 실시하여 보안 기준에 위배되는 지를 조사한다. 즉, 패트리넷의 도달성 트리를 생성하여, 마킹 순차를 따라가면서 불안전(insecure) 마킹이 있는지를 조사해 가는 방법으로 불안전 마킹이 존재하면 누출이 발생됨을 의미한다.

(3) 분석 프로토콜

BM 프로토콜[35]과 MM 프로토콜[43]을 분석하였다.

(4) 단 점

분석 방법인 완전 공격 테스트는 패트리넷의 도달성 트리를 생성하는 것으로 이에 대한 문제는 일반 패트리넷에서와 같이 복잡한 시스템에서는 그 트리 크기가 분석하기에 너무 방대해 진다. 뿐만 아니라, 공격자를 사양하는 방법이 복잡하다.

3.6.2 Stal-Tavares-Meijer 사양 모형[27]

(1) 특 성

유색 패트리넷을 이용하여 암호 프로토콜을 분석하는 방법이다. 패트리넷을 이용하여 암호 프로토콜 엔티티들 사이의 정보 제어와 흐름을 도형적으로 사양하였다. 또한, 분석을 위해, 명확한 공격자가 사양을 만들고 합법적인 프로토콜 엔티티를 공격하는 방법과 상태도를 생성하였다. 불안전 상태(insecure state)를 정의하고, 안전성 누출(security flaw)을 조사하기 위해 후향 상태 분석(backward state analysis)을 사용하였다.

(2) 사양 및 분석 방법

프로토콜을 사양하기 위해 금지 호선을 갖는 유색 패트리넷을 이용하고, 생성된 사양에 공격자 사양을 두 통신자 사이 또는 키 분배 센터 사이에 삽입하여 분석한다. 공격자는 메세지를 가로채어 삭제하고 버전을 변경할 수 있고, 프로토콜에 관한 완전한 지식을 보유하지만, 프로토콜 엔티티가 가진 비밀 키같은 비밀 정보는 직접 접근할 수 없다고 가정하고, 분석을 위해 후향 상태 분석 방법을 사용한다. 이 방법은 Meadow[47]에 의해 제안된 암호 프로토콜 검증과 모델링 방법중 타입 1과 타입 2의 모두에 속한다고 볼 수 있다. 타입 1이란 암호 프로토콜 분석을 위해 특별히 디자인되지 않은 사양 언어와 검증 도구를 이용하여 모델링과 검증하려는 방법을 말하고, 타입 2는 프로토콜 안전성 조사가 완료됨을 보장하지 않는 분석 방법들을 말하지만, 이는 프로토콜에서 이미 알려졌거나, 알려지지 않은 보안 누출을 밝히는 데 사용될 수 있다. 후향 상태 분석 방법은 사양에서 있을 수 있는 불안전성 상태를 조사하여 이 상태에서 초기 상태까지 도달되는 경로가 있는지를 후향으로 찾아가는 방법이다. 만약 그런 경로가 존재하면, 불안전 상태가 일어날 수도 있다는 것을 의미한다.

(3) 분석 프로토콜

안전한 디지털 이동 통신을 위한 Hwangs 프로토콜[44]과 ISO Working draft에 있는 간단한 Protocol[45]에 대해 불안전 상태가 존재하는지를 후향 상태 분석으로 검증한다. 분석 결과, Hwang 프로토콜은 통신 채널에서 메세지의 변형(ID 값의 대치)으로 인가되지 않았거나, 외부의 공격자를 인증해줄 수 있음을 보이고 있다.

(4) 단 점

이 방법에서는 가능한 모든 불안전 상태를 먼저 확인하여야 한다. 이는 미처 확인하지 못한 불안전 상태가 있었다면, 그 불안전 상태는 후향 상태 분석으로는 찾을 수 없다(모든 불안전 상태를 찾을 수 없다는 점을 시인). 이는 후향 상태 분석 자체의 문제이며, 초기 상태로 시작하여 모든 가능한 상태를 찾는 일반적인 방법(forward analysis)이 많은 분석 시간을 요구하지만, 모든 불안전 상태를 찾을 가능성이 높다.

4. 비교 분석

이상과 같이 암호 프로토콜의 사양과 분석을 위한 여러 방법이 연구되고 있고 각 방법들은 제각

기 장단점을 갖고 있으며, 이들 연구에 대한 후속 연구도 활발히 진행되고 있는 실정이다. 이 장에서는 이들 방법들에 대한 장단점을 비교하여 표 4-1에 보인다. 이들 방법들의 공통적인 문제점은 통신 프로토콜에 수학적인 암호 알고리즘이 관계하는 암호 프로토콜의 보안성 누출 여부는 잘 설계되고 구현된 운영 체계를 공격하는 해커들의 기술적, 경험적 접근 방법이 가끔 성공하는 것과 같이 공격자(intruder, attacker)의 모든 방법을 조사해 보기 전에는 알 수 없다는 점이다. 이는 프로그램내의 오류는 모든 경로를 모두 수행해 보기 전에는 에러를 모두 찾을 수 없다는 프로그램 테스팅 이론과 같다[4,5].

대수적 기법인 Dolev-Yao[10]의 사양과 분석 방법은 cascade 프로토콜에 대한 정의와 그에 따른 보안 특성을 정의와 정리로 표현하고 그에 대한 증명을 보이므로 불안전 상태와 안전 상태를 체계있게 보여 준다. 또한 축소 규칙을 이용하여 복잡한 암복호 함수를 간략화한다. Longley-Rigby[14]는 암호 프로토콜을 검색 트리로 변환하고, 트리의 루트에 도달 가능한지를 검색하므로써 보안성 누출여부를 판별한다. 논리적 접근 방법의 BAN 모형[16]은 가정과 인증 목표를 논리로 표현하므로써, 암호 프로토콜이 그 인증 목표에 도달되는 가

표 4-1. 각 사양, 분석 기법들의 비교

	대수적 접근	논리적	논리 대수적	상태 변환적	패트리넷(도형적)
특 징	rule base term-rewrite rule reducing	modal logic belief(trust) knowledge(security)	epistemic 논리 지식 지양 사양 무지, 학습개념 도입	state machine deadlock 발견 backword 분석 공격자 관점에서 프로토콜 사양	정보 흐름 모형 도달성 분석 도형적 모형 backword 분석 가능
분석 프로토콜	cascade[10] name-stamp[10]	Kerberos, RPC[23], NS[9],BM[35]	NS[9], KP[29]	TMN[7]	NS[9], BM[35], MM[43], Hwang[44]
도구 사용언어	prolog	prolog, KPL[18]	CKT5[28]	prolog, Ina Jo YACC, LISP	PN 분석 도구 사용 가능
장 점	규칙을 사용한 수학적 분석 expert system 구현	규칙 기반 공식 전개 방법 사양 언어로 적합	지식 지향 사양 시간, 통신, 지식의 논리에 대한 의미론 제공	분석 도구 있음	가시성, 추상화를 가진 정형적 사양 제공 PN 이론 연구 많음
단 점	인증성 문제 무시 일반성 결여 검색트리 생성 방법	프로토콜의 각 파트마다 자신의 사양이 필요 비 정형적인 이산화 과정	객체의 인증성 결여	검색 실패시 검증 불가 병행성,비동기성, 비결정성 사양 불가 readability 결여	도달성 트리 복잡 불안전 상태 발견 난이 공격자 모델링 필요
연 구 자	Dolev-Yao[10] Longley-Rigby[14]	BAN [16] syverson[17,18,19] Meadow[19] Rangan[20] Snekkenes[21]	Beiber[28] Merritt-Wolper[30] Toussaint[31]	Millen[25] Meadow[11,12,19] Kemmerer[33,34]	Varadhrajan[40,41] Sassone[41] Nieh-Tavares[26] Meijer-Tavares[27]

를 검증하는 방법으로 초기 가정까지 일관된 논리를 사용하고 있으나, 안전(security)보다도 신임(trust)에 중점을 둔다[18]. 그리고 메세지 이상화 방법이 비정형화 되어 있는 문제점을 갖고 있다. 상태 변환적 접근 방법은 분석 도구까지 개발된 방법으로 그들의 비교는 표 3-5에 보인다.

패트리넷을 이용한 방법은 앞절에서 언급한 것처럼 정보 흐름 모형을 위한 것과 암호 프로토콜을 위한 두 가지 방법이 있고, 암호 프로토콜에 관한 모형은 캐나다의 퀸대학에서 주로 연구되었다. 패트리넷 모형은 통신 프로토콜의 사양 방법에 이미 사용되고 있고 도형적인 모형으로 "백문이 불여일견"의 철학을 수용하는 접근 방법이며 기존의 사양 모형들에 대해 패트리넷이 갖는 장점은 아래와 같은 것이 있다.

- 기존 사양 모형들이 공통적으로 지닌 프로토콜 사양에 대한 비 가시성(문자적으로 표현되므로)문제를 해결할 수 있다. 즉, 패트리넷은 도형적인 사양 모형이다.
- Topdown, stepwise-refinement로 사양 및 분석이 가능하다.
- 상태 폭발 문제를 해결하는 방법이 잘 알려져 있다(예: 축소 규칙 등).
- 프로토콜과 같은 병행적, 비 동기적 및 비 결정적인 시스템들에 대한 사양과 분석이 용이하다.
- 패트리넷은 논리나 대수처럼 이론적인 특성들이 잘 알려져 있는 하나의 오토마타이므로, 정형적으로 프로토콜을 검증할 수 있다.
- 패트리넷은 사용자의 해석에 따라서, 논리형, 대수형 및 상태 변환형 사양으로 간주할 수 있다(사실, 패트리넷은 1st order predicate logic을 기본으로 하는 Prolog로 표현 가능하며, 기본의 암호 프로토콜 검증 방법들은 대부분 Prolog로 구현되었다).
- 패트리넷은 사양하고 분석하려는 목표, 대상 및 수준에 따라 자유롭게 수정하여 사용할 수 있는 비 해석적(uninterpreted)인 사양 모형이다.
- 이미 패트리넷을 분석하는 소프트웨어 도구들이 상품화되어 있고, 많은 이론적인 결과와 응용 사례가 발표되어 있다.
- 시스템의 정성적인 특성(예: 도달성, 데드록 등)뿐 아니라 정량적 특성(예: 성능, 신뢰도등)을 분석할 수 있다.
- 기존의 연구들에서는 통신망의 안전성을 가정하고 있다. 그러나, 패트리넷을 이용한 방법에서는 메세지의 보안 문제 뿐아니라, 메세지 분실에 의한 문제를 fault-tolerant 기법을 이용하여 해결하고, 정량적인 성능 및 안전도 등을 분석할 수도 있다.
- 패트리넷은 시스템 내의 자료 흐름과 제어 흐름을 통합적으로 표현할 수 있다.
- 패트리넷으로 사양된 프로토콜은 실행이 가능하다(executable).

5. 결 론

암호 프로토콜에 대한 분석 기법은 소프트웨어 개발에서 있어 사용자가 원하는 소프트웨어의 구현을 위해 요구 사양을 정형화하는 것과 같은 접근 방법으로 시작할 수 있다. 잘 정의되고 정형적으로 표현된 사양은 소프트웨어의 특성과 품질을 잘 나타내 주고 개발의 용이성 뿐만 아니라, 인수를 위한 좋은 자료가 되는 것 처럼, 암호 프로토콜의 특성과 보안성 누출 여부에 대한 검증 자료가 된다. 이를 위해 암호 프로토콜에 대한 연구자들은 암호 프로토콜을 대수적으로 정의하고, 그에 따른 보안성 특성을 살펴봄으로써 안전한 프로토콜이 되기 위한 조건을 규명하기도 하고, 통신자들이 상대방이 보낸 메세지에 대한 믿음과 지식을 이용한 논리 전개등을 이용하여 분석등을 수행하여 왔다. 즉, 프로토콜을 정형적으로 나타낼 수 있는 방법과 그에 맞는 분석 방법을 제시하고 있다.

그러나 많은 분석 기법에서는 공격자 모형까지 사양되어야 하는 어려움이 있다. 이는 분석자는 분석에 앞서 공격 가능 방법(경로)를 알고 있어서, 그에 대한 검출이 가능하도록 사양하여야 한다. 따라서 분석자는 공격자 가정에서 보여지는 것 처럼 키 이외의 모든 정보를 아는, 암호 프로토콜에 대한 전문가 이어야 한다. 정형적 사양은 암호 프로토콜을 이해하기(다른 사람에게 이해시키기) 쉽게 해주므로 분석하는 노력을 줄여 줄 수 있다. 따라서 암호 프로토콜의 분석을 위해, 각 기법들의 문제점을 해결하려는 노력이 필요하며, 실제로 각 기법에 대한 후속 연구도 발표되고 있는 실정이다. 특히 가시성이 높은 패트리넷을 이용한 연구는 아직 초기 단계에 있으나, 앞장에 언급하는 것과 같은 장점으로 인해 좋은 결과가 있으리라 예상된다. 또한 상태 변환적 방법에서와 같이, 분석 도구의 개발은 암호 프로토콜을 연구하는 연구자에게 빠른 암호 프로토콜 분석이 가능하게 하여 보다 완벽한 암호 프로토콜을 제시할 수 있게 해주므로 이에 대한 연구와 도구 개발이 있어야 할 것이다.

참 고 문 헌

[1] G. J. Simmons, "Proof of soundness (Integrity) of cryptographic protocols", Journal of Cryptology, vol. 7, pp. 69-77, 1994.

[2] R. Kemmerer, C. Meadow, J. Millen, "Three systems for cryptographic protocol analysis," Journal of Cryptology, vol. 7, no. 7, pp.79-130, Spring 1994.

[3] M. Tatebayashi, N. Matsuzaki and D. Newman, "Key distribution protocol for digital mobile communication systems," Lecture Notes in Computer Science, vol. 435, pp.324- 333, 1991.

[4] 이 주헌, 실용 소프트웨어 공학론 상권, 법영사, 206-213, 1993.

[5] Fairley, R., "Software Engineering Concepts," McGraw-Hill, 1985.

[6] C. A. Sunshine, "Communication protocol modeling," Artech House, selected papers, 1981.

[7] J. H. Moore, "Protocol failures in cryptosystems," Proceedings of the IEEE, vol. 76, no. 5, pp. 594-602, May 1988.

[8] D. Denning and G. Sacco, "Timestamps in key distribution protocols," Comm. of ACM, vol. 24, no. 8, pp.533-536, Aug. 1981.

[9] R. Needham and M. Shroeder, "Using encryption for authentication in large networks of computers," Communications of ACM, vol. 21, no. 12, pp.993-999, Dec. 1978.

[10] D. Dolev and A. Yao, "On the security of public-key protocols," IEEE tran. on Information Theory, vol. IT-29, no.2, pp. 198-203, March 1983

[11] C. Meadows, "Representing partial knowledge in an algebraic security model," Proc. the Computer Security Foundations Workshop III, pp.23-31, 1990.

[12] C. Meadows, "A system for the specification and analysis of key management protocols," 1991 IEEE Computer Society Symp. on Research in Security and Privacy, pp. 182-195, May 1991.

[13] W. Diffie and M. Hellman, "Multiuser cryptographic techniques," Proc. AFIPS 1976 NCC., Montvale, NJ, AFIPS Press, pp109-112, 1976.

[14] D. Longley and S. Rigby, "An automatic search for security flaws in key management schemes," Computer & Security, vol. 11, no.1, pp. 75-89, 1992.

[15] R. W. Jones, "Some techniques for handling encipherment keys," ICL Tech. J., vol. 3, pp175-188, 1982.

[16] M. Burrows, M. Abadi and R. Needham, "A logic of authentication," ACM Tran. on Computer Systems, vol. 8, no.1, pp.18-36, 1990.

[17] P. Syverson, "Formal semantics for logics of cryptographic protocol," Proc. the Computer Security Foundations Workshop III, pp.32-41, 1990

[18] P. Syverson, "The use of logic in the analysis of cryptographic protocols," Proc. 1991 IEEE Computer Society Symp. on Research in Security and Privacy, pp.156-169, May 1991.

[19] P. Syverson and C. Meadow, "A logical language for specifying cryptographic protocol requirement," 1993 IEEE Computer Society Symp. on Research in Security and Privacy, pp.165-177, May 1993.

[20] P. V. Rangan, "An axiomatic theory of trust in secure communication protocols," Computer & Security, vol. 11, no.2, pp.163-172, 1992.

[21] E. Snekkenes, "Exploring the BAN approach to protocol analysis," Proc. 1991 IEEE Computer Society Symp. on Research in Security and Privacy, pp. 171-181, May 1991.

[22] Satyanarayanan, M., "Integrating security in large distributed system," ACM Trans. Comput. Syst., 7, 3, pp.247-280, Aug. 1989.

[23] CCITT. CCITT draft recommendation X.509. The directory-authentication framework, version 7. CCITT, Gloucester, Nov. 1987.

[24] G. Coulouris, et al., "Distributed systems concepts and design," , Addison-Wesley, pp.477-616, 1994.

[25] J. K. Millen, S. C. Clark and S. B. Feedman, "The Interrogator: protocol security analysis," IEEE Tran. on Software Engineering, vol. SE-13, no.2, pp.274-287, Feb. 1987.

[26] B. B. Nieh and S. E. Tavares, "Modeling and analyzing cryptographic protocols using Petri nets," Proc. of AUSCRYPT'92, pp.275-295, 1993.

[27] D. M. Stal, S. E. Tavares and H. Meijer, "Backward state analysis of cryptographic protocol using colored Petri nets," Proc. of Workshop on Selected Areas in Cryptography, pp.275-295, May 1994.

[28] P. Beiber, "A logic of communication in hostile environment," Proc. the Computer Security Foundations Workshop III, pp.14-21, 1990.

[29] E. Snekkenes, "Roles in cryptographic

protocols," 1992 IEEE Computer Society Symp. on Research in Security and Privacy, pp.105-119, May 1992.

[30] Merritt, M., and P. Wolper, "State of knowledge in cryptographic protocols", unpublished manuscript. 1985.

[31] M. J. Toussaint, "Separating the specification and implementation phases in cryptology," Lecture Notes in Computer Science, vol. 648, Springer-Verlag, pp. 77-101, 1992.

[32] M. Abadi and M. R. Tuttle, "A semantics for a logic of authentication," Proc. of 10th Annual ACM symp. on Distributed Computing, pp.201-206, Aug. 1991.

[33] R. A. Kemmerer, "Analyzing encryption protocol using formal verification techniques," IEEE Journal Selected Areas in Communication, vol.7, pp.448-457, 1989.

[34] R. A. Kemmerer, "Integrating formal methods into the development process," IEEE Software, pp.37-50, Sep. 1990.

[35] J. Burns and C. Mitchell, "A security scheme for resource sharing over a network," Computer & Security, vol. 9, no. 1, pp.67-75, 1990.

[36] T. C. Woo and S. S. Lam, "A semantic model for authentication protocols," 1991 IEEE Computer Society Symp. on Research in Security and Privacy, pp.178-194, May 1993.

[37] J. L. Peterson, "Petri nets theory and the modeling of systems," Prentice-Hall, 1981.

[38] T. Murata, "Petri nets: properties, analysis and application," Proceedings of the IEEE, vol.77, pp.541-580, 1989.

[39] 이강수, "패트리넷에 관한 기술 해설," 정보과학회지, 1권 2호, pp.39-47, 1983.

[40] V. Varadhrajan, "Petri net based modeling of information flow security require- ments," Proc. the Computer Security Foundations Workshop III, pp.51-61, 1990.

[41] V. Sassone and V. Varadhrajan, "A unifying Petri net model of non-interference and non-deducibility information flow security," Hewlett-Packard Corp., Tech. Paper (HPL-91-44), Feb. 1991.

[42] K. Jensen, G. Rozenberg, "High-level Petri nets: theory and application," Springer-Verlag, p.724, 1991.

[43] C. H. Meyer and S. M. Matas, "Cryptography: a new dimension in computer data security," John Wiley & Sons, pp.578-583, 1982.

[44] T. Hwang, "Scheme for secure digital mobile communication based on symmetric key cryptography," Information Processing Letters, vol. 48, pp.35-37, 1993.

[45] ISO/IEC JTC1/SC27/N832, ISO/IEC 11770-2:Information technology -- Security techniques -- Key Management -- Part 2: Key Management Mechanisms Using symmetric techniques. draft

Document, International Organization for standardization, 1993.

[46] C. Meadow, "Applying formal methods to the analysis of a key management protocol," Journal of Computer Security, vol. 1 no. 1, pp.5-35, 1992.

[47] 허용도, 손진곤, "새로운 인증방법과 키 분배 프로토콜," 통신 정보 보호 학회지, 제3권, 제2호, pp.52-63, 1993.6.

□ 著者紹介

이 진 석(정회원)

1986년 대전 산업대 전자계산학과(학사)
1990년 한남대학교 대학원 수학과(전산전공)(석사)
현 한국전자통신연구소 선임연구원

※ 관심분야 : 컴퓨터 보안, 암호 프로토콜, 프로그램 테스팅

신 기 수(정회원)

1975년 서강대학교 전자공학과(학사)
1989년 충북대학교 대학원 전자계산기공학과(석사)
현 한국전자통신연구소 실장

※ 관심분야 : 네트워크 보안, ISDN 관련 분야, 이동통신

이 강 수(정회원)

1981년 홍익대학교 전자계산학과(학사)
1983년 서울대학교 대학원 계산통계학과(석사)
1989년 서울대학교 대학원 계산통계학과(박사)
1992 ~ 1993년 미국 일리노이 대학 객원 교수
1995 ~ 한국전자통신연구소 초빙연구원
현 한남대학교 전자계산공학과 부교수

※ 관심분야 : 실시간 및 병행처리 시스템 모형, 소프트웨어 공학, 패트리넷 응용, 암호 프로토콜 모형, EDI 보안 모형