AVISPA를 이용한 전자상거래 시스템의 보안 프로토콜 명세 및 검증

정연오○ 김주배 김현석 최진영
고려대학교 컴퓨터학과
{yojeong○, jbkim, hskim, choi}@formal.korea.ac.kr

The Specification and Verification of Security Protocols on E-commerce System Using AVISPA

Yeonoh Jeong○ Joobae Kim Hyunseok Kim Jinyoung Choi
Dep. Of Computer Science & Engineering, Korea University

요 약

최근 인터넷을 이용한 전자상거래(e-commerce)가 확산되면서 많은 편의를 제공하고 있다. 하지만, 인터넷을 통한 거래는 개인정보의 유출이나 주문 및 지불 정보의 유출 등의 취약점들에 대한 보안상의 요구를 만족하면서도 사용자들이 접근하기 쉽고 사용하기 편리해야 한다는 점에서 이중적인 어려움을 가지고 있다. 본 논문에서는 현재 전자상거래에서 사용되고 있는 보안 프로토콜들의 문제점들을 분석하고 이를 보완한 새로운 보안 프로토콜을 제안한다. 그리고 제안된 프로토콜을 정형검증 툴인 AVISPA를 이용하여 명세 및 검증함으로써 안전성을 검증한다.

1. 서 론

최근 인터넷의 규모가 확산 되면서 인터넷을 이용한 물품거래가 매우 활발해지고 있다. 과거의 오프라인상의 거래에서는 소비자와 상인이 서로를 직접대면하고 거래를 하고, 거래하고자 하는 물품도 직접보고 금액을 지불하였다. 하지만 최근 인터넷이나 통신기술 등의 발달로 인해 온라인상에서의 거래가 발달하게 되었다. 이는 시간, 공간의 제약을 거의 받지 않아 사용자에게 매우 편리한 반면, 거래의 참여자 간에 서로를 믿을 수 있는지에 대한 신뢰성 문제와 개인 정보의 유출 위험에 따른 보안상의 문제, 구매한 물건에 대한 정확한 주문과 지불에 관한 안정성 문제 등 여러 가지 요구사항을 가지고 있다. 따라서 사용자들이 사용하기 편리하고 간단하면서도 위의 요구사항들을 만족시키기 위해서는 안정적인 보안 프로토콜의 구성이 필수적이다.

전자상거래(e-commerce)는 개인의 카드 및 계좌정보를 이용한 지불 거래와 주문한 물품의 거래가 인터넷을 통해 이루어지므로 보안상의 문제는 곧 개인 정보의 누출이나 금전적인 피해로 이어질 수 있다. 이러한 문제를 해결하기위하여 강력한 암호화 방법과 암호화 프로그램의 설치 및 믿을 수 있는 제 3의 인증기관의 참여 등으로 보안성을 강화할 수 있지만, 대부분의 이용자들이 보안에 대한 인식이 부족한 일반 사용자들이고, 일반적으로 인터넷을 통한 물품구입은 소액 결제가 대부분이어서 고도의 안전성 보다는 빠르고 사용하기 편리한 방법을 선호하므로, 보안 프로토콜의 요구사항을 만족하면서도 보다 편리하고 빠르게 거래를 할 수 있는 프로토콜의 구성이 요구된다.

현재 전자상거래 시스템에서 사용되고 있는 대표적인 보안 프로토콜은 SSL(Secure Sockets Layer)/TLS(Transport Layer Security)[1,2]와 SET(Secure Electronic Transaction)[3,4,5]이 있다. SSL/TLS는 대부분의 전자상거래 시스템에서 사용하고 있는 프로토콜로써 사용하기 편리하여 널리 사용되고 있지만 보안상의 취약점을 안고 있으며, 지불 시스템을 위하여 개발된 SET 프로토콜은 강력한 보안의 장점에도 불구하고 복잡한 구성으로 인해 전자상거래 시스템의 보안 표준으로 자리메김하지 못하고 있다.

본 논문에서는 앞서 소개한 전자상거래 시스템의 대표적 프로토콜인 SSL/TLS와 SET 프로토콜 대해 알아보고, 이 프로토콜들의 장점과 단점을 분석하여 전자상거래에서 요구하는 조건들을 좀 더 만족 시킬 수 있는 개선된 전자상거래 프로토콜을 제안하고, 정형 기법을 이용하여 이를 명세 및 검증하였다.

정형 기법은 설계된 시스템이나 프로그램이 개발자의 요구사항에 맞게 설계된 것인지를 수학적 이론이나 명세 및 검증 도구를 이용하여 검증하는 방법을 말하며, 크게 시스템이나 프로그램의 동작이나 특성들을 정형적으로 표현하는 정형 명세와 명세 된 시스템이 요구사항을 잘 만족하는지 검증하는 방법인 정형 검증으로 나눌 수 있다.

보안 프로토콜의 정형 검증 방법은 수학적 논리를 바탕으로 한 정리 증명과 정형 검증 툴인 FDR, SPIN, AVISPA(Automated Validation of Internet Security Protocols and Applications)[6]를 이용한 모델체킹 방법이 있는데, 본 논문에서는 인터넷 상의 보안 프로토콜 검증에 적합한 AVISPA를 이용하여 명세 및 분석하고,

검증결과를 통해 프로토콜의 안전성을 확인하고자 한다.
본 논문의 구성은 다음과 같다. 2장에서는 전자상거래 시스템의 보안 요구사항에 대해 알아보고, 3장에서는 현재 사용되고 있는 전자상거래 프로토콜들을 관련연구로 소개하고, 4장에서는 전자상거래 프로토콜들을 분석하기 위한 정형 명세 언어와 툴에 대해 소개하며, 5장에서는 본 논문에서 제안하는 개선된 전자상거래 프로토콜인 EPP (Electronic Payment Protocol) 프로토콜의 명세 및 검증을, 마지막으로 6장에서 결론 및 향후 연구방향을 제시한다.

2. 전자상거래 시스템의 보안 요구사항

인터넷의 발달과 함께 대부분의 업체나 기관들 그리고 개인들이 웹사이트를 이용하여 정보를 공유하고 있다. 이러한 인터넷상에서의 활동은 시간과 공간의 제약을 받지 않는 이점을 토대로 인터넷을 통한 전자상거래의 발전을 이끌었다. 하지만 이러한 발전의 이면에는 보안상의 취약점이 존재하여 전자상거래 상의 피해가 속출하고 있다. 더욱이 웹을 통한 지불 거래는 금전적인 피해 뿐 아니라 개인 정보의 유출로 인한 제 2, 제 3의 피해를 낳을 수 있으므로 보안에 대한 중요성이 더욱 증가하고 있다.
비록 웹 브라우저를 쉽게 구성하고, 편리하게 사용할 수 있도록 설계되었지만, 내부적으로는 굉장히 복잡하게 구성되어있어 사람이 생각하지 못한 취약점들을 많이 내포하고 있다. 또한 전자상거래의 이용자들이 대부분 훈련이 되지 않은 일반사용자들이기 때문에 보안상 위험에 대한 인식과 그에 따른 대책이 많이 미흡하다.
이러한 이유로 전자상거래에서의 보안은 매우 중요한 사항이며 이에 따라 시스템이 갖추어야할 보안 요구사항은 다음과 같다.

가) 데이터의 무결성(integrity)
프로토콜을 통해 전송되는 모든 정보에 대해 공격자에 의해 데이터가 훼손되거나 수정이 되어서는 안된다.

나) 주문 및 지불 정보에 대한 기밀성(confidentiality)
주문 및 지불 정보가 수신해야 할 수신자만 데이터에 접근할 수 있도록 하여 악의적인 공격자로부터의 공격을 방지한다.

다) 고객과 상인 간의 인증(authentication)
고객이 유효한 계좌의 합법적 사용자임을 인증하고, 판매자 역시 안전한 거래를 위해 인증을 받아야만 한다.

라) 참여자들 사이에 공정한 거래(fair exchange)
고객은 지불에 합당한 물품을 받아야하고, 상인 역시 거래에 합당한 지불 승인을 받아야한다.

3. 전자상거래 시스템 보안프로토콜 연구현황

3.1 SSL(Secure Sockets Layer)/TLS(Transport Layer Security)

SSL은 1994년 네스케이프가 처음으로 만든 프로토콜로서 HTTP와 같은 응용 계층 프로토콜들에 대한 암호화를 지원하는 트랜스포트 계층에서의 보안 프로토콜이며 비록 전자상거래 시스템을 위해 설계된 프로토콜은 아니지만, 신용카드 지불(Credit card payments)에 적합하게 사용되고 있는 가장 일반적인 프로토콜이라고 할 수 있다. TLS는 1995년에 개발된 SSL의 최종버전인 SSLv3.0을 기초로 IETF에서 1999년에 RFC2246로 제안된 표준으로 정의된 것으로 SSL과 거의 흡사하다.
SSL/TLS에 대한 많은 연구가 진행되었다. 특히 정형검증 도구인 AVISPA를 이용한 안전성 검증에 관한 연구가 진행되고 있고, 이를 통해 몇 가지 문제점이 존재함을 확인 하였다[7].

3.2 SET (Secure Electronic Transaction)

SET은 SSL/TLS와는 달리 지불 시스템(payement system)을 위해 설계되었다. 프로토콜의 구성은 customer, merchant, payment gateway로 구성이 되며, 이들은 디지털 서명을 이용하여 상호 인증을 수행하며, 이는 CA(Certification Authority)로 알려진 제 3의 인증기관(trusted third party)에 의해 보장이 된다.
또한 SET은 대칭 암호화 방법인 DES와 비대칭 암호화 방법인 RSA를 혼합하여 사용하며, 강력한 암호화 방법으로 안전성을 강화하였다.
SET 역시 전자상거래의 발전과 함께 많은 연구가 진행되었는데, 강력한 암호화와 인증방법을 사용하고 있지만, 역시 AVISPA를 이용한 안전성 검증에서 보안상의 취약점을 발견하였다[7].

본 논문에서는 전자상거래용 프로토콜들을 분석하는데 사용된 정형검증 도구를 이용하여 새로운 전자상거래용 프로토콜을 제안하고자 한다.

4. HLPSL과 AVISPA 소개

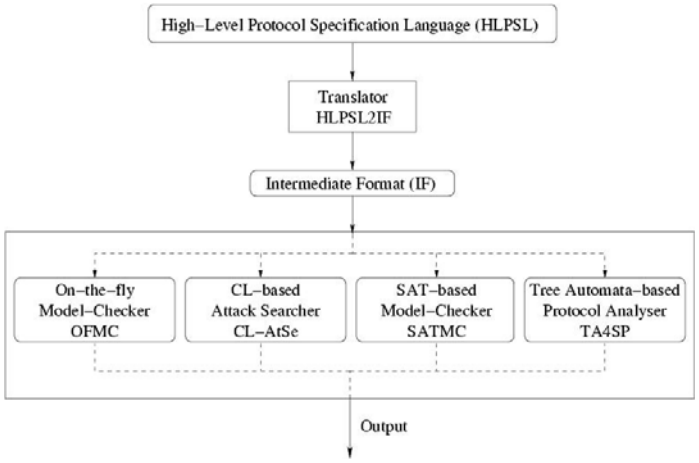
4.1 HLPSL(High Level Protocols Specification Language)

HLPSL[8]은 role을 기반으로 하는 언어로써, 각각의 role들은 서로 독립되어 구성이 되어있고, channel을 통해 의사소통을 한다.
role은 역할에 따라 두 가지로 나누어 구분할 수 있는데, 프로토콜을 구성하는 각각의 개체들을 기술하는 basic role과 basic role들의 시나리오를 기술하기위한 composition role로 구성된다.
basic role은 각 개체들이 가지고 있는 정보를 표기하고, SND와 RCV 명령어를 이용하여 다른 개체들과 서로의 정보를 교환하여 의사소통을 한다.
composition role은 전체 프로토콜의 구성을 나타내는 role로써 각 role들이 가지고 있는 각각의 구조와 공격자(intruder)가 가지고 있는 정보, 프로토콜의 검증 속성을 포함하는 goal들을 표기한다.

4.2 AVISPA(Automated Validation of Internet Security

Protocols and Applications)

AVISPA[6]는 인터넷 보안 프로토콜과 응용프로그램을 자동으로 검증해주는 도구로써, HLPSL이라는 정형 명세 언어로 명세 된 프로토콜을 입력으로 받아서 translator를 통해 IF 파일로 변환이 되고, IF파일은 다시 4가지 back-end들의 입력으로 들어가게 된다. 이 4가지 back-end들은 각각 다른 특성을 가진 model checking tool로써 On the fly Model-Checker(OFMC)[9], CL-based Attack Searcher(CL-ALse), SAT-based Model-Checker(SATMC), Tree Automata-based Protocol Analyser(TA4ST) 로 구성이 되어있으며, 입력으로 들어온 명세 된 보안 프로토콜에 대해 4가지의 검증 도구에 대해 각각의 검증 결과를 얻을 수 있다.



[그림 1] AVISPA 툴의 구조

4.3 SPAN

SPAN[10]은 AVISPA의 web graphical interface의 local version이다. 범용 적으로 사용되는 컴파일러의 형태를 차용하여 시각적으로 이해하기 쉽게 설계되었고, 아직 툴이 완벽하지는 않지만, 단점을 보완한 새로운 버전이 계속 나오고 있다.

5. EPP (전자 지불 프로토콜: Electronic Payment Protocol) 제안

5.1 EPP 프로토콜의 설계의 목표

EPP 프로토콜 설계의 목표는 2장에서 설명한 전자상거래에서 만족해야할 기본적인 요구사항 4가지를 만족시키고, 현재 적용되어 사용되고 있는 SET 프로토콜에 웹 보안기능을 추가한 새로운 프로토콜을 설계하고자 한다.

5.2 EPP 프로토콜의 모델링 및 분석

5.2.1 EPP 프로토콜의 모델링

프로토콜에 대한 전체 구성은 아래의 [그림 2]와 같다.

표기법	
표기	내용
A.B	두 메시지 A와 B의 접합(concatenation)
h(M)	메시지 M을 hash함수를 이용해 hashing
{M}_A	Key A로 메시지 M을 암호화
inv(A)	Key A의 역(inverse)이 되는 키
Enck(A)	A의 공개키로 암호화
A -> B : M	A에서 B로 메시지 M을 보냄

암호화 기법

- Sign_A(Msg) = Msg.{h(Msg)}_inv(SignK(A))
- Encrypt_A(Msg1, K, Msg2) = {Msg2}_K.{Msg1.K}_EncK(A)
- Encrypt_B(Msg, K) = {Msg}_K.{K}_EncK(B)
- DualSign_A(M1,M2) = Sign_A(h(M1), h(M2))

프로토콜의 구성

```
% Temporary customer certificate
1. C -> M : Encrypt_M(C, K1, Nc)
% Temporary certificate response
2. M -> C : {M.Sign_M(Nc.Nm.Sid) }_K1
% Purchase request
3. OI = OrderDesc.PurchAmt.Sid
   PI = CardInf.PurchAmt.Sid.M
   C -> M : {Nm.OI.DualSign_C(OI.PI)}_inv(PubK_C).
            Encrypt_P(DualSign_C(OI.PI).PI, K1)
% Purchase Authorization Request
4. AuthReq = h(PI).OI.DualSig_C(PI.OI)
   M -> P : Encrypt_P(Sign_M(AuthReq), K2).
            Encrypt_P(DualSign_C(OI, PI).PI, K1)
% Purchase Authorization Response
5. P -> M : Encrypt_M(Sign_P(Response.Sid.PurchAmt), K3)
% Purchase Response
6. M -> C : Encrypt_C(Sign_P(Response.Sid.PurchAmt), K1)
```

[그림 2] EPP 프로토콜 구성

EPP 프로토콜은 모두 6단계로 구성되며, 각 단계에서 포함하고 있는 메시지들에 대한 설명은 다음과 같다.

1단계: 임시 인증 요청(Temporary certificate request)
고객과 상인간의 임시적 상호 인증을 위한 메시지로서 고객은 자신의 ID값인 C와 자신만이 만들 수 있는 nonce값인 Nc, 그리고 상인과의 메시지 전달에 사용할 대칭키인 K1을 상인만이 볼 수 있도록 상인의 공개키로 암호화 하여 상인에게 보낸다.

2단계: 임시 인증 응답(Temporary certificate response)
고객의 인증 요청에 응답하기 위하여 상인 역시 자신만이 만들 수 있는 nonce 값인 Nm과 거래를 하는 동

안 사용할 수 있는 ID값인 Sid 값을 포함시켜 자신의 개인키로 서명하여 자신이 보낸 메시지임을 증명하고 고객과 공유한 대칭키로 암호화하여 보낸다.

3단계: 구매 요청(Purchase request)
거래를 위한 상호 인증을 마친 고객은 상인에게 주문 정보와 지불 정보를 보낸다. 이때 주문정보(OI)와 지불 정보(PI)를 이중 서명(DualSign_C) 방식을 이용하여 고객과 지불 게이트웨이 사이에 불필요한 정보 유출을 막는다.

4단계: 지불 인증 요청(Payment certificate request)
고객에게 정보를 받은 상인은 고객의 주문 정보를 확인하고, 자신의 개인키로 서명하한 지불 인증(AuthReq) 정보를 지불 게이트와의 메시지교환에 사용할 대칭키(K2)와 함께 지불게이트웨이의 공개키로 암호화하여 보낸다.

5단계: 지불 인증 승인(Payment certificate response)
상인으로부터 받은 지불에 대한 정보를 확인하고 고객에 대한 정보를 확인하여 올바른 사용자임을 확인 한 후 지불 인증 요청에 대한 응답메시지(Response)를 보내어 지불에 대한 승인을 한다.

6단계: 구매 승인(Purchase response)
지불 게이트웨이로부터 지불 승인을 받은 상인은 지불 승인에 따른 구매 승인 메시지(Response)를 고객에게 보냄으로서 거래를 완료한다.

5.2.2 EPP 프로토콜 분석

다음은 제안한 프로토콜을 HLPSL로 명세한 부분 중에서 고객(cardholder)에 대한 role의 명세부분이다.

```
role cardholder(C, M, P: agent,
                CardInf : text,
                PurchAmt : nat,
                OrderDesc : text,
                PubK_C,PubK_M,PubK_P : public_key
                ) played_by C def=
local State : nat,
      Nc, Nm, Sid, Response : text,
      OI, PI, DualSig : message,
      K1 : symmetric_key,
      SND, RCV : channel (dy)
init State := 0
transition
1. State = 0 ∧ RCV(start)
   =|>
   State' := 1 ∧ K1' := new()
                ∧ Nc' := new()
                ∧ SND({Nc'}_K1'.{C.K1'}_PubK_M)
2. State = 1 ∧ RCV({M.Nc.Nm'.Sid'.{h(Nc.Nm'.Sid')}
                _inv(PubK_M)}_K1)
   =|>
   State' := 2 ∧ OI' := OrderDesc.PurchAmt.Sid'
                ∧ PI' := CardInf.PurchAmt.Sid'.M
                ∧ DualSig' := h(OI').h(PI').
                  {h(h(OI').h(PI'))}_inv(PubK_C)
                ∧ SND({{Nm'.OI'.DualSig'}_inv(PubK_C).
                  {DualSig'.PI'}_K1.{K1}_PubK_P}_K1)
                ∧ witness(C,M,c_nonce,Nc.Nm')
                ∧ witness(C,M,cm_deal,Sid'.OI')
                ∧ secret(OrderDesc,order_1,{C,M})
                ∧ secret(PurchAmt,order_2,{C,M,P})
                ∧ secret(CardInf,payment,{C,P})
3. State = 2 ∧ RCV({Response'.Sid.PurchAmt.
                {h(Response'.Sid.PurchAmt)}_inv(PubK_P)}_K1)
   =|>
   State' := 3 ∧ request(C,M,m_nonce,Nc.Nm)
                ∧ request(C,M,mc_deal,OI.h(PI))
end role
```

[그림 3] EPP 프로토콜의 HLPSL 명세 코드

[그림 3]의 명세를 통해 최초거래를 위해 고객과 상인의 상호 인증부분(1.State의 SND부분과 2.State의 RCV 부분)과 상인과 지불 게이트웨이 사이의 불필요한 정보의 누출을 방지하기 위한 이중 서명 부분(2.State 중간의 SND부분)과 인증 요청 및 이에 대한 승인 메시지를 각 프로토콜 참여자들의 role에 명세하였고, 프로토콜의 검증 속성인 보안(secrecy)과 인증(authentication)을 검증하기 위하여 secret()과 witness(), request()의 함수를 사용하였다. 위에 명세 된 코드에서의 각 함수의 의미는 다음과 같다.

secret(OrderDesc,order_1,{C,M}) : 고객(C)과 상인(M)사이에 주문 정보(OrderDesc)에 대한 보안요구 사항이 만족이 되는지를 검증한다. order는 goal 부분에 명세하기위한 ID값이다.

witness(C,M,c_nonce,Nc.Nm') : Nc.Nm'의 값에 대해서 M에 의한 C의 약한 인증(weak authentication)을 의미하며, c_nonce는 goal에 명세하기 위한 ID값이다.

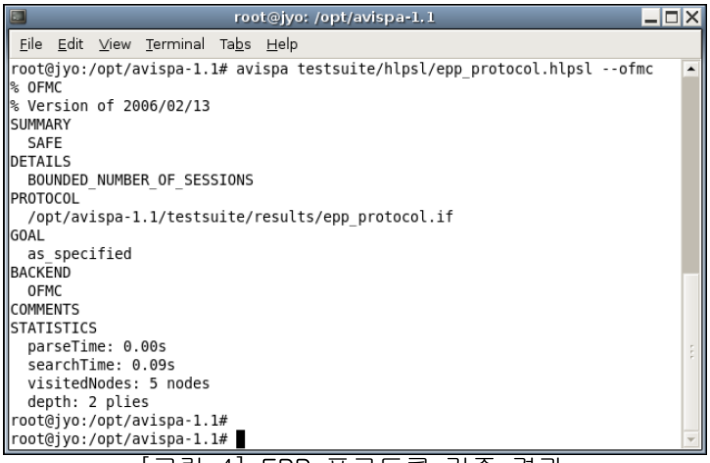
request(C,M,m_nonce,Nc.Nm) : Nc.Nm의 값에 대해서 M에 의한 C의 강한 인증(strong authentication)을 의미 하며, c_nonce는 goal에 명세하기 위한 ID값이다.

request()는 witness()와 함께 인증을 위한 함수로 사용된다.

위와 같은 검증 속성을 통해 제안한 프로토콜의 보안 요구 사항과 인증문제의 안전성을 검증하였다.

5.3 EPP 프로토콜 검증 결과

[그림 4]는 AVISPA를 이용하여 HLPSL로 명세한 EPP 프로토콜을 검증한 결과이다.



[그림 4] EPP 프로토콜 검증 결과

위의 검증결과는 AVISPA의 4가지 back-ends 중에서 OFMC에 대한 검증결과로서, [그림 4]에서 볼 수 있듯이 SUMMARY 부분이 SAFE 함을 통해 안전한 프로토콜임을 확인할 수 있고, 이로써 GOAL부분을 통해 명세 된 EPP 프로토콜의 검증 속성인 보안(secrecy)과 인증(authentication)에 대한 요구사항을 만족함을 확인할 수 있다.

6. 결론 및 향후 연구

인터넷을 이용한 물품 거래는 사용자가 공간과 시간의 제약이 거의 존재하지 않는 장점으로 인해 계속 확산되고 있다. 하지만 인터넷을 통하여 물건을 구입하고 돈을 지불한다는 점에서 전자상거래에서의 보안 프로토콜의 역할은 매우 크다고 할 수 있다. 이러한 보안 요구 사항을 만족시키기 위해 현재 많은 연구가 진행되고 있지만, 다양해진 공격 형태로 인해 계속적인 연구가 요구된다.

본 논문에서는 현재 사용되고 있는 전자상거래 보안프로토콜들을 분석하고 이들의 장단점을 분석 및 보완하여 기존의 전자상거래 프로토콜들을 개선한 프로토콜을 제안하였다. 그리고 제안된 보안 프로토콜의 안전성을 검증하기 위하여 정형 명세언어인 HLPSL을 이용하여 명세하고, 이를 정형 검증 툴인 AVISPA를 통해 요구된 검증 속성에 대한 안정성을 검증하였다.

향후 연구로서 제안된 프로토콜을 개선하여 모바일 환경에 적합한 M-Commerce 프로토콜을 연구하고자 한다.

참고 문헌

[1] E. Rescorla, "SSL and TLS : Designing and Building Secure Systems", Addison-Wesley, 2001.
[2] William Stallings, "Network Security Essentials : applications and standards", Prentice hall, 2000.
[3] Mark Stamp, "Information Security : Principles and Practice", Wiley, 2005.
[4] G. Bella, F. Massacci, L.C. Paulson, "An Overview of the verification of SET", International Journal of Information Security, p.17-28, 2005.
[5] G. Bella, F. Massacci, L.C. Paulson, "The Verification of an Industrial Payment Protocol : The SET Purchase Phase", ACM Conference on Computer and Communications Security, 12-20, 2001.
[6] AVISPA. "AVISPA v1.1 User Manual", Available at http://www.avispa-project.org, 2006.
[7] AVISPA. "The AVISPA Library of Protocols", http://www.avispa-project.org/library/index.html.
[8] AVISPA. "HLPSL Tutorial : A Beginner's Guide to Modelling and Analysing Internet Security Protocols", Available at http://www.avispa-project.org, 2006.
[9] D. Basin, S. M¨odersheim, L. Vigano`, "OFMC : A symbolic model checker for security protocols", International Journal of Information Security, 2004.
[10] AVISPA. "SPAN : A Security Protocol Animator for AVISPA Version 1.1 User Manual", Available at http://www.avispa-project.org, 2007.