Chinese Journal of Electronics
Vol.22, No.1, Jan. 2013

Formal Analysis of Information Card Federated Identity-Management Protocol*

WANG Juan[1,2], HU Hongxin[3], ZHAO Bo[1,2], YAN Fei[1,2], ZHANG Huanguo[1,2] and WU Qianhong[1,2]

(1.*Computer School, Wuhan University, Wuhan 430072, China*)

(2.*Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education, Wuhan 430072, China*)

(3.*SEFCOM, Arizona State University, AZ 85287, USA*)

Abstract — Information Card (InfoCard) is a user-centric identity management metasystem. It has been accepted as a standard of OASIS Identity Metasystem Interoperability Technical Committee. However, there is currently a lack of security analysis to InfoCard protocol, especially, with formal methods. In this paper, we accommodate such a requirement by analyzing security properties of InfoCard protocol adopting a formal protocol analysis tool. Our analysis result discovers that current InfoCard protocol is vulnerable against the session replay attack. Furthermore, we reveal the importance of two optional elements in InfoCard metasystem, token scope and proof key, and found that InfoCard protocol will be susceptible to man-in-the-middle attack and token replay attack if these two optional elements lack.

Key words — Information card, User-centric, Identity, Automated validation of Internet security protocols and applications (AVISPA).

I. Introduction

In open and collaborative Internet, managing the necessary digital identities is a complex hurdle. Traditionally, Service providers collect and manage user identity related information which has led to proven attacks like identity theft and privacy violation. Recently, user-centric identity management approaches have emerged. User-centric identity management allows users to control their own digital identities. Users can select their credentials when responding to an authenticator or attribute requester, giving users more rights and responsibility over their identity information. Information Card (InfoCard)[1−3] is a user centric federated identity management metasystem for reducing the reliance on passwords and improving the privacy of personal information.

InfoCard protocol is as the foundation of an InfoCard system implementation and its security should be guaranteed. However, the design of security protocols is highly error-prone. Security flaws have been found in many protocols[4]. For InfoCard protocol, security issues have not yet been explored in depth, especially using formal methods. In this paper, we focus on the formal analysis of InfoCard protocol. We first provide an informal description about the detailed communication proceeds and message format of InfoCard protocol and thoroughly analyze the implementation mechanism of security goals. Then, we translate the informal description of InfoCard protocol and its security goals to a formal specification based on High level protocol specification language (HLPSL). Finally, we verify the protocol using an automatic formal analysis tool, Automated validation of Internet security protocols and applications (AVISPA)[5,6].

Our contribution is three-fold. First, we model and analyze the authentication and secrecy goals of InfoCard protocol using a formal method. Second, we uncover a session replay attack. Finally, we reveal the importance of two main optional elements, token scope and proof key, in an InfoCard metasystem. We found that InfoCard protocol is susceptible to man-in-the-middle attack and token replay attack if these two optional elements lack in the implementation of InfoCard protocol. To the best of our knowledge, this is the first time that above attacks to Infocard protocol are identified.

The remainder of this paper is organized as follows. In Section II, we describe the detailed communication proceeds and message format of InfoCard protocol and analyze the implementation mechanism of security goals. In Section III, we analyze and verify InfoCard protocol by a formal analysis tool, AVISPA, and discuss the importance of two main optional elements, token scope and proof key. Section IV discusses the related work. Finally, Section V provides conclusions.

II. Infocard Protocol

In accordance with the specification of OASIS Identity Metasystem Interoperability standard, the InfoCard protocol involves five components: User (U), Identity selector (IS), Client application (A), Relying party (RP) and Identity provider (IP).

The interaction process of the InfoCard system is depicted in Fig.1.

(1) A user accesses a relying party (a website) which allows

*Manuscript Received Oct. 2011; Accepted Apr. 2012. This work is supported by the National Natural Science Foundation of China (No.61003268, No.61173138) the Fundamental Research Funds for the Central Universities (No.211274629).

the user to log in the relying party with InfoCard.

(2) The relying party returns the requested page and sends the relying party's policy.

(3) The client application calls the identity selector and forwards the relying party's policy.

(4) In accordance with the relying party's policy, the identity selector searches in the user's information cards collection and finds the information cards which satisfy the requested claims in the relying party's policy. The user then selects one of these cards to log on the relying party.

(5) The identity selector sends a token request to the corresponding identity provider who issued the selected card.

(6) The identity provider authenticates the user by the provided authentication credential. If the user has passed the authentication, the identity provider issues a security token to the identity selector.

(7) The identity selector sends the service request message and the security token to the relying party.

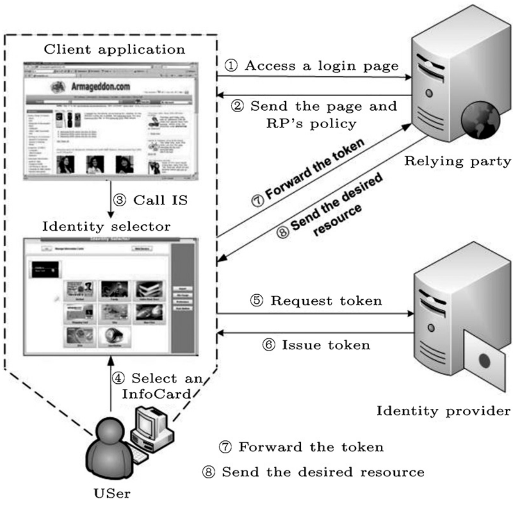

Fig. 1. Framework of InfoCard

(8) The relying party successfully authenticates the user and sends desired service or resource to the client.

Although the InfoCard system includes five components and eight operating steps, but in fact it is a typical three party authentication protocol. We can combine the user, identity selector and client application as "Client", and neglect the steps used to retrieve the policy from the point view of security protocol analysis, just considering the steps using the mutual authentication process. Moreover, it must be explained that we just consider the managed card protocol in this paper because the selfissued card protocol does not have a separate identity provider and can be regarded as a special case of the managed card protocol.

On the basis of above considerations, we can give the main authentication procedure of managed card protocol. The main authentication procedure and message format are according to Microsoft's implementation of InforCard protocol[7,8] and the OASIS identity metasystem specification[1].

The elements of n_1 to n_9, n_{ce} and n_{se} are random numbers. k_1, k_2 and k_t are symmetric keys. PK_{IP} represents the public key of IP and PK_{IP}^{-1} represents the private key of IP. Ck_{sig} is the signature key of client. Ck_{enc} is the encryption key and $C_{kendorse}$ is the endorsing signature key. RPk_{sig} is the signature key of RP and RPk_{enc} is the encryption key. M_{rst} indicates request message and M_{resp} indicates response message. M_{tok} indicates the issued token by identity provider. SAML is the type of issued token. M_{enctok} represents the encyphered token. $cardId$ is the unique identity of a card. $types_{RP}$ stands for the types of the required claims. $claims_U$ represents claims with the attribute values for the required $types_{RP}$. M_{mac} is the message signature. M_{proof} is the endorsing signing message proving possession of proof key.

The detailed communication proceeds and message format of managed card protocol are described as following:

(1) C->IP: $\{\{k_1\}_{PK_{IP}}, n_1, n_2, \{\{M_{user}, M_{rst}\}_{Ck_{sig}}, M_{user}, M_{rst}\}_{Ck_{enc}}\}$

$M_{user} = (U, pwd)$,

$M_{rst} = (cardId, types_{RP}, RP, n_{ce})$

$Ck_{sig} = \mathrm{PSHA1}(k_1, n_1)$,

$Ck_{enc} = \mathrm{PSHA1}(k_1, n_2)$

(2) IP->C: $\{n_3, n_4, \{\{M_{rstr}\}_{IPk_{sig}}, M_{rstr}\}_{IPk_{enc}}\}$

$M_{rstr} = (\{k_t\}_{PKRP}, \{SAML(M_{tok}, \{M_{tok}\}^{-1}_{PK_{IP}})\}_{k_t}, n_{se})$

$M_{tok} = (IP, \{PSHA1(n_{ce}, n_{se})\}_{PK_{RP}}, claims_U, RP, ppid)$

$IPk_{sig} = PSHA1(k_1, n_3)$,

$IPk_{enc} = PSHA1(k_1, n_4)$

(3) C->RP: $\{\{k_2\}_{PK_{RP}}, n_5, n_6, n_7, M_{enctok}, \{M_{mac}, M_{proof}, M_{req}\}_{Ck_{enc2}}\}$

$M_{enctok} == (\{k_t\}_{PK_{RP}}, \{SAML(M_{tok}, \{M_{tok}\}^{-1}_{PK_{IP}})\}_{k_t})$,

$M_{mac} = \{M_{req}\}_{Ck_{sig2}}$,

$M_{proof} = \{M_{mac}\}_{Ck_{endorse}}$,

$Ck_{enc2} = PSHA1(k_2, n_6)$,

$Ck_{sig2} = PSHA1(k_2, n_5)$,

$k_{proof} = PSHA1(n_{ce}, n_{se})$,

$Ck_{endorse} = PSHA1(k_{proof}, n_7)$

(4) RP->C: $\{n_8, n_9, \{\{M_{resp}\}_{RPk_{sig}}, M_{resp}\}_{RPk_{enc}}\}$

$RPk_{enc} = PSHA1(k_2, n_9)$,

$RPk_{sig} = PSHA1(k_2, n_8)$

III. Formal Verification of Infocard Protocol

In this paper, we use the formal tool-Automated validation of Internet security protocols and applications (AVISPA)[5,6] to analyze InfoCard protocol.

1. Modeling InfoCard protocol and security goals

AVISPA is a formal tool for building and analyzing security protocols. AVISPA integrates four different formal backends: On-the-fly model-checker (OFMC), CL-based Attack searcher (CL-AtSe), SAT-based model-checker (SATMC) and Tree-automata-based protocol analyzer (TA4SP). These backends implement different automated reasoning techniques for formally analyzing protocol specification. To our knowledge, no other tool exhibits the same level of scope and robustness while being with the same performance and scalability.

AVISPA uses a special language called High level protocol specification language (HLPSL)[9,10] to model the analyzed protocol. A HLPSL specification is composed of three parts: a list of definitions of basic roles (the principals of protocol), protocol session, Intruder capabilities and a list of declarations of goals.

(1) Modeling InfoCard protocol in HLPSL

Specification of protocol basic roles. The specification of InfoCard protocol contains three basic roles: client, Identity provider (IP), and Relying party (RP). Every role owns a list of global variables, a list of local variables and a lot of transitions. Global variables represent the public and share initial knowledge. Messages are exchanged via channels, Sending channel (SND) and Receiving channel (RCV). dy, the channel of type, stands for a channel of Dolev-Yao[11] which is under complete control of intruder. The following is the HLPSL specification of client.

```
  role client (C, IP, RP        : agent,
               Kip, Krp         : public_key,
               Muser, Cardid    : text,
               SND, RCV         : channel (dy),
               PSHA1            : hash_func)
 played_by C def=
   local State : nat,
   N1, N2, N3, N4, N5, N6, N7, N8, N9, Nce, Nse : text,
   K1, Kt, K2 : symmetric_key,
   Ipksig, Ipkenc, Cksig, Ckenc, Cksig1, Ckenc1,
Rpksig, Rpkenc :
   hash(symmetric_key.text),
   Kproof : text,
   Ckendorse : hash(text. text),
   Mresp, Mreq: message,
   SAMLtoken:agent.{text}_public_key.text.agent.text.{agent. {text}_public_key.text.agent.text}_inv(public_key)
```

The transition section of basic roles describes the procedure of message reception and emission (behavior of principals). Every transition has the form "precondition =>action", which represents that when precondition is satisfied, action will be executed. For instance, the state transition of the client is as following:

```
 transition
   0. State = 0 ∧ RCV(start) = | >
   State':=2 ∧ N1':=new() ∧ N2':=new()∧ K1':= new()
   ∧ Cksig':=PSHA1(K1'.N1')∧ Ckenc' :=PSHA1
   (K1'.N2')∧ SND({K1'}_Kip. N1'.N2'. {{Muser.Cardid. typesrp.RP.Nce'}_Cksig'. Muser. Cardid. typesrp. RP. Nce'}_Ckenc')
   2.State=2∧RCV(N3'.N4'.{{{Kt'}_Krp.{SAMLtoken'}_ Kt'.Nse'}_Ipksig'.{Kt'}_Krp.{SAMLtoken'}_
   Kt'.Nse'} _Ipkenc') ∧ Ipksig' = PSHA1(K1.N3')∧
   Ipkenc' = PSHA1(K1.N4') = | >
   State':=4∧K2' := new()∧N5' := new()∧N6' := new()
   ∧N7' := new() ∧Cksig1' := PSHA1(K2'.N5')/\ Ckenc1' := PSHA1 (K2'.N6') ∧Kproof' := PSHA1(Nce.
   Nse)∧ Ckendorse' := PSHA1 (Kproof'.N7') ∧SND
   ({K2'}_Krp. N5'.N6'.N7'.{Kt'}_Krp. {SAMLtoken'}
   _Kt'.{Mreq.{Mreq}_Cksig1'.{{Mreq}_Cksig1'}_Ckendorse'} _Ckenc1')
   4.State=4 ∧ RCV(N8'.N9'.{{Mresp} _Rpksig'. Mresp}_Rpkenc')∧
   Rpksig' = PSHA1(K2.N8') ∧ Rpkenc' = PSHA1
   (K2.N9') = | >
   State':= 6
 end role
```

Protocol session. A session is used for describing how to combine roles which is to run roles in parallel. The run of a basic role is a process. In the session of InfoCard protocol, the processes of three basic roles (client, IP, RP) run parallel which is described as following:

```
 Role session(C, IP, RP: agent, Kip, Krp: public_key,
Muser, Cardid: text, PSHA1: hash_func) def=
     local SC, RC, SIP, RIP, SRP, RRP: channel (dy)
     composition
       client(C, IP, RP, Kip, Krp, Muser, Cardid, SC, RC,
PSHA1)
     ∧ ip(C, IP, RP, Kip, Krp, Muser, Cardid, SIP, RIP,
PSHA1)
     ∧ rp(C, IP, RP, Kip, Krp, SRP, RRP, PSHA1)
 end role
```

Intruder capabilities. Intruder capabilities are defined in the environment role. This role comprises global constant and a composition of session, where the intruder may play some roles as legal users. Composition session includes two parallel protocol running in which the intruder is participating. The definition of environment role is as follows:

```
role environment() def=
  const
     client, ip, rp                             : agent,
     krp, kip, ki                               : public_key,
     typesrp, muser, cardid, claimsu, ppid      : text,
     samltoken                                  :protocol_id,
     psha1                                      :hash_func

    intruder_knowledge = {client, ip, rp, krp, kip, ki, inv(ki),
psha1}
    composition
    session(client, ip, rp, kip, krp, muser, cardid, psha1)
  ∧ session(client, ip, rp, kip, krp, muser, cardid, psha1)
end role
```

(2) Modeling security goals in HLPSL

Authentication and secrecy are two basic security goals of InfoCard protocol. We model them as following:

```
  request (IP, C, muser, Muser) witness(C, IP, muser,
Muser)
  request (IP, C, cardid, Cardid) witness(C, IP, cardid, Cardid)
  request(RP,C,samltoken,SAMLtoken')
  witness(C,RP,samltoken,SAMLtoken')
  secret (SAMLtoken, samltoken, {IP, C})
```

request$(IP, C, muser, Muser)$ means that IP receives the message$(Muser)$ from the *client*(C). *witness*$(C, IP, muser, Muser)$ means that the client sends the message $(Muser)$ to IP. The paired predicates (*request and witness*) describe strong authentication[4] properties in HLPSL.

$secret(SAMLtoken, samltoken, \{IP, C\})$ represents the main secrecy goal that SAML type token should be confidential between the IP and the client. An attacker can not know the token.

2. Security verification and verification results

Given a HLPSL specification, the next step is to translate it into a lower level specification. This is automatically done by the translator hlpsl2if, generating a specification in an intermediate format (IF). In the end, IF specification is verified by the underlying automated reasoning module. During the verification process of InfoCard protocol, we mainly used the OFMC (On-the-fly model-checker)[12], which is the most important and powerful reasoning module of AVISPA.

We have found three types of attacks: session replay attack, man-in-the-middle attack and token replay attack. The man-in-the-middle attack and token replay attack are found based on the assumption that two optional elements, token scope and proof key, lack in InfoCard protocol implementation.

(1) Session replay attack

This attack is found by authentication goal. An attacker can intercept the old message from a client to IP and replay it to IP in a new session. The attack is showed as follows.

i -> (client, 3): start

(client,3)-> i:

$\{K1(1)\}\_kip.N1(1).N2(1).\{\{muser.cardid.typesrp.rp.Nce(1)\}\_(psha1(K1(1).N1(1))).muser.cardid.typesrp.rp.Nce(1)\}\_\ (psha1(K1(1).N2(1)))$

i->(ip,3):

$\{K1(1)\}\_kip.N1(1).N2(1).\{\{muser.cardid.typesrp.rp.Nce(1)\}\_(psha1\ (K1(1).N1(1))).muser.cardid.typesrp.rp.Nce(1)\}\_(psha1(K1(1).N2(1)))$

(ip,3)-> i:

$N3(2).N4(2).\{\{\{Kt(2)\}\_krp.\{ip.\{psha1(Nce\ (1).Nse(2))\}\_krp.claimsu.rp.ppid.\{ip.\{psha1(Nce(1).Nse(2))\}\_krp.claimsu.rp.ppid\}\_inv(kip)\}\_Kt(2).Nse(2)\}\_(psha1(K1(1).N3(2))).\{Kt\quad (2)\}\_krp.\{ip.\{psha1(Nce(1).Nse(2))\}\_krp.claimsu.rp.ppid.\{ip.\{psha1(Nce(1).Nse(2))\}\_krp.claimsu.rp.ppid\}\_inv(kip)\}\_Kt\ (2).Nse(2)\}\_(psha1(K1(1).N4(2)))$

i->(ip,7):

$\{K1(1)\}\_kip.N1(1).N2(1).\{\{muser.cardid.typesrp.rp.Nce(1)\}\_(psha1(K1(1).N1(1))).muser.cardid.typesrp.rp.Nce(1)\}\_(psha1(K1(1).N2(1)))$

(ip,7)-> i:

$N3(3).N4(3).\{\{\{Kt(3)\}\_krp.\{ip.\{psha1(Nce(1).Nse(3))\}\_krp.claimsu.rp.ppid.\{ip.\{psha1(Nce(1).Nse(3))\}\_krp.claimsu.rp.ppid\}\_inv(kip)\}\_Kt(3).Nse(3)\}\_(psha1(K1(1).N3(3))).\{Kt(3)\}\_krp.\{ip.\{psha1(Nce(1).Nse(3))\}\_krp.claimsu.rp.ppid.\{ip.\{psha1(Nce(1).Nse(3))\}\_krp.claimsu.rp.ppid\}\_inv(kip)\}\_Kt(3).Nse(3)\}\_(psha1(K1(1).N4(3)))$

where i represents an intruder. "(client, 3) -> i, i-> (ip, 3) and (ip, 3) -> i" is a round execution of protocol. The intruder is a man in middle, who can intercept the all of message from client to identity provider. "i -> (ip, 7) and (ip, 7) -> i" is another round execution, in which the intruder replay the old message intercepted from the former round.

Although the intruder can not decrypt the token issued by IP in the above attack, the attack has breached the authentication property. It also induces DOS attack if the intruder replays continuously the old authentication request message to IP.

(2) Man-in-the-middle attack

In the specification of OASIS identity metasystem interoperability, two optional elements, token scope and proof key, are extremely critical. We model and analyze the revised InfoCard protocol where these two optional elements lack. We found that InfoCard protocol is susceptible to man-in-the-middle attack if the element of token scope lacks in the implementation of InfoCard protocol. We also found the attack of replaying the token without the optional element of proof key.

Token Scope element which represents the identity of RP in the request token message is optional. An Identity Selector, by default, should not convey information about the relying party where an issued token will be used (i.e., target scope) when requesting security tokens. This helps safeguard user privacy. However, if token scope lacks in request-token message, our analysis can find a man-in-the-middle attack that allows a dishonest replying party (an intruder) to impersonate a user at another replying party. The attack is shown as Fig.2.

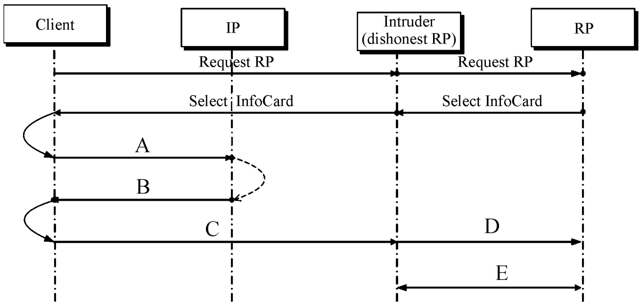

Fig. 2. Man-in-the-middle attack. A—$\{\{k_1\}_{PKIP},\ n_1,\ n_2,\ \{\{M_{user}, M_{rst}\}_{Cksig},\ M_{user}, M_{rst}\}_{Ckenc}\}$; B—$\{n_3,\ n_4,\ \{\{M_{rstr}\}_{IPksig},\ M_{rstr}\}_{IPkenc}\}$; C—$\{\{k_2\}_{PKi},\ n_5,\ n_6,\ n_7,\ M_{enctok},\ \{M_{mac},\ M_{proof},\ M_{req}\}_{Ckenc2}\}$; D—$\{\{k_2\}_{PKRP}, n_5, n_6, n_7, M_{enctok},\ \{M_{mac},\ M_{proof},\ M_{req}\}_{Ckenc2}\}$; E—$\{n_8,\ n_9,\ \{\{M_{resp}\}_{RPksig}, M_{resp}\}_{RPkenc}\}$

In Fig.2, Pki represents the public key of the intruder. M_tok does not contain the identifier of RP. Hence when the intruder receives the token, he can forward directly the signed token and forge other items of the message to another RP. As a result, intruder impersonates a legal client at another RP to get the resource of RP. If token scope is included in issued token, the dishonest RP can not impersonate a legal user at another RP, because the dishonest RP can not tamper the identifier of RP contained in the signed token by private key of IP. Therefore, the dishonest RP can not abuse the token issued by IP at another RP.

(3) Token replay attack

Proof key is a subject confirmation key. In identity metasystem interoperability version 1.0, proof key is an optional element, but it is very important. Proof key is a credential between client and IP. Meanwhile it is included in the encyphered token. In the third step of InfoCard protocol, a client uses the proof key to produce the endorse key which is used to sign the MAC of request message. Thus, RP authenticates client based on the token and the proof key. Although an attacker can get the signed token, he can not tamper the signed token. It means that he can not tamper the proof key included in signed token. Therefore the attacker can not impersonate a legal user without the proof key. Consequently, proof keys provide stronger authentication guarantee.

If the proof key lack in InfoCard protocol, an intruder who have stolen the token by some attack methods, such as DNS dynamic pharming attack[18], can impersonate legal user and replay the signed token to legal RP. If we add the token to the knowledge of intruder and delete proof key of exchanged messages in the formal model of InfoCard protocol, our analysis can find an attack to authentication goal as follows where an intruder, represented as i, can pass the authentication to RP by the stolen token.

$i >$(rp,6):

$\{K2(1)\}\_Krp.N5(1).N6(1).N7(1).\{Kt(1)\}\_Krp.\{IP.claimsu.RP.ppid.\{IP.claimsu.RP.ppid\}\_inv(Kip)\}\_Kt(1).\{Mreq.\{Mreq\}\_psha1(K2(1).N5(1))\}\_psha1(K2(1).N6(1))$

(rp,6)$>i$:

$N8(2).N9(2).\{\{Mresp\}\_psha1(K2(1).N8(2)).Mresp\}\_psha1(K2(1).N9(2))$

IV. Related Work

Gajek *et al.*[13] analyzed the attacks to InfoCard based on weak security policy enforcements in commodity browsers. Hoang *et al.*[14] discussed the problem InfoCard secure Roaming across different terminals. Alrodhan and Mitchell[15] analyzed its privacy property of card space and suggested a zero-knowledge cryptographic technique to improve privacy protection. Ahn *et al.*[16] proposed a user preference expression language that is crucial to manage user's privacy in federated identity management. Ahn *et al.*[17] provided category-based privacy preference approach to enhance the privacy of user-centric identity management systems.

The work most closely related to ours is Ref.[18]. In this paper, Bhargavan *et al.* formally verified a reference implementation of InfoCard with ProVerif, a formal tool based on pi calculus. They found a man-in-the-middle attack where attacks get the unencyphered signature and replace it with own signature. Their analysis is based on the assumption configuration that signed messages were not encyphered. Without the assumption, their method can not find any attack. However in most cases, signed messages should be encrypted when protocols are designed, because attacks can get the unencyphered signature and replace it with own signature. In our analysis, the configuration of InfoCard protocol is encryption after signature. Therefore, the new attacks found in the paper are different from Ref.[18].

V. Conclusion

In this paper, we have successfully analyzed the authentication and secrecy properties of Infocard protocol using a formal analysis tool, AVISPA. Our analysis showed that InfoCard protocol is easily subjected to a session replay attack because this protocol is lack of freshness numbers. In addition, we analyze the importance of token scope and proof key, the two main optional elements in the specification of OASIS Identity Metasystem Interoperability. Our analysis indicated that a man-in-the-middle attack where a dishonest RP can impersonate a legal user with acquired token to pass authentication of another RP, and token replay attack where a intruder can replay the stolen security token, are possible if these two optional elements lack in InfoCard protocol. We have implemented an InfoCard identity selector using Java on android platforms. Meanwhile, we are implementing an entire InfoCard system. In the future, we would like to verify the privacy properties for federated identity-management protocols based on our implementation using formal methods.

References

[1] Identity Metasystem Interoperability Version 1.0. OASIS Standard. *http://docs. oasis-open. org/imi/identity/v1.0/ identity. html.*

[2] Higgins Open Source Identity Framework, *http: //www. eclipse. org/ higgins/.*

[3] Information Card Foundation, *http://informationcard. net/.*

[4] G. Lowe, "Breaking and fixing the Needham-Schroeder public-key protocol using FDR", *Software-Concept and Tools*, Vol.17, pp.93–102, 1996.

[5] AVISPA. The AVISPA User Manual. *http://avispa-project. org/publications.*

[6] A. Armando *et al.*, "The AVISPA tool for the automated validation of Internet security protocols and applications", *Proc. of the 17th International Conference on Computer Aided Verification* (*CAV'05*), Scotland, UK, Springer-Verlag. 2005.

[7] A. Nanda, "A technical reference for the information card profile V1.0", *Technical Report*, Microsoft Corporation, 2006.

[8] K. Bhargavan, C. Fournet, A.D. Gordon and N. Swamy, "Verified implementations of the information card federated identity-management protocol", *Proc. of ASIACCS '08*, Akihabara Convention Hall, Tokyo, ACM, pp.123–135, 2008.

[9] Y. Chevalier, L. Compagna *et al.*, "A high level protocol specification language for industrial security sensitive protocols", *Proc. of SAPS'04*, Austrian Computer Society, pp.1–13, 2004.

[10] Yannick Chevalier and Laurent Vigneron, "Rule-based programs describing Internet security protocols", *Electronic Notes in Theoretical Computer Science*, Vol.124, No.1, pp.113–132, 2005.

[11] Dolev, A. Yao, "On the security of public-key protocols", *IEEE Transactions on Information Theory*, Vol.29, No.2, pp.198–208, 1983.

[12] D. Basin, S. Mödersheim, L. Vigan'o., "OFMC: A symbolic model-checker for security protocols", *International Journal of Information Security*, No.4, pp.181–208, 2005.

[13] Sebastian Gajek, Jörg Schwenk and Xuan Chen, "On the insecurity of microsoft's identity metasystem cardspace", *Technical Report*, Ruhr University, Bochum, Germany, 2008.

[14] L.N. Hoang, P. Laitinen and N. Asokan, "Secure roaming with identity metasystems", *Proc. of IDtrust '08*, New York, USA, ACM, pp.36–47, 2008.

[15] Waleed A. Alrodhan and Chris J. Mitchell, "Improving the security of cardspace", *EURASIP Journal on Information Security*, Vol.2009, Article ID 167216, pp.1–8, 2009.

[16] GailJoon Ahn, John Lam, "Managing privacy preferences for federated identity management", *Proc. of DIM'05*, Fairfax, Virginia, USA, pp.28–36, 2005.

[17] Gail-Joon Ahn, Moonam Ko, Mohamed Shehab, "Privacy-enhanced user-centric identity management", *Proc. of IEEE International Conference on Communication*, Dresden, Germany, pp.1–, 2009.

[18] C. Karlof, J. Tygar, D. Wagner, and U. Shankar, "Dynamic pharming attacks and locked same-origin policies for web

browsers", *Proc. of the 14th ACM Conference on Computer and Communications Security* (*CCS*), Virginia, USA, pp.58–71, 2007.

WANG Juan was born in China, in 1976. She is an associate professor of Computer School of Wuhan University and Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education. She received the B.S., M.S. and Ph.D. degrees from Computer School of Wuhan University. From 2010 to 2011, as a visiting scholar, she did some research on information security in the Laboratory of Security Engineering for Future Computing at Arizona State University, USA. Her current research interests include identity authentication, security protocol, access control, trusted computing and cloud security. (Email: jwang@whu.edu.cn)

HU Hongxin was born in China, in 1974. He is currently a Ph.D. student in Computer Science and Engineering, Ira. A. Fulton School of Engineering at Arizona State University, USA. His current research interests include security and privacy in healthcare systems and social networks, network and system security, security in distributed, cloud and mobile computing, access control models and mechanisms.

ZHAO Bo was born in China, in 1972. He is a professor of Computer School of Wuhan University and Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education. His current research interests include trusted computing and cloud security.

YAN Fei was born in China, in 1980. He is an associate professor of Computer School of Wuhan University and Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education. His current research interests include trusted computing, access control and cloud security.

ZHANG Huanguo is a professor of Computer School of Wuhan University and Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education. His current research interests include trusted computing and cloud security.

WU Qianhong is a professor of Computer School of Wuhan University and Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education. His current research interests include cryptography and information security.