World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:5, No:12, 2011

Mechanized Proof of Resistance of Denial of Service Attacks in voting protocol with ProVerif

Bo Meng, Wei Wang

Abstract—Resistance of denial of service attacks is a key security requirement in voting protocols. Acquisti protocol plays an important role in development of internet voting protocols and claims its security without strong physical assumptions. In this study firstly Acquisti protocol is modeled in extended applied pi calculus, and then resistance of denial of service attacks is proved with ProVerif. The result is that it is not resistance of denial of service attacks because two denial of service attacks are found. Finally we give the method against the denial of service attacks

Keywords—applied pi calculus, protocol state, symbolic model, availability

I. Introduction

THE securities of internet voting protocols are the key requirements of electronic government and electronic commerce. People have paid serious attentions on receipt-freeness and coercion-resistance. Many internet voting protocols claimed on their securities [1-5]. Besides the previous security properties, owning to the big damage of denial of service attacks in security protocols [6], in order to protect the security of voting system, internet voting protocol should also have resistance of denial of service attacks.

The formal method is a powerful tool used to analyze the resistance of denial of service attacks. To our knowledge there are mainly three formal models: Yu-Gligor model [7] based on user agreement; Meadows's cost-based model [6] based on fail-stop protocol; Meng-Huang model [8] based on protocol state. Among the above three formal models, Meng-Huang model is the only one which support the mechanized tool ProVerif.

Acquisti protocol [1] plays an important role in development of internet voting protocols and claims its security without strong physical assumptions. Until now resistance of denial of service attacks in Acquistic protocol has not been analyzed. So here we use ProVerif to verify resistance of denial of service attacks in Acquisti protocol based on Meng-Huang model.

The main contributions of this paper are summarized as follows:

- Apply the mechanized formal model proposed by Meng and Huang for mechanized proof of Acquisti protocol and its resistance of denial of service attacks. Hence the extended applied pi calculus is used to model Acquisti protocol, and then according to the formal definition of resistance of denial of service attacks, Acquisti protocol is analyzed with ProVerif.
- The result we obtain is that Acquisti protocol is not resistance of denial of service attacks. Two denial of service attacks are found by us. At the same time we give the method against the denial of service attacks.

Bo Meng is with school of computer, South-Center University for Nationalities, Wuhan 430074, China e-mail:(mengscuec@gmail.com).
Wei Wang is now with school of computer, South-Center University for Nationalities, Wuhan 430074, China e-mail: (wangwei9852003@yahoo.com.cn).

II. Related Work

Yu and Gligor [7] propose may be the first formal model on resistance of denial of service attacks based on temporal logic. They use user agreement to describe resistance of denial of service attacks. But their formal framework does not support the automated tools. Following this line Bacic and Kuchta [9] argue that the core problem of resistance of denial of service attacks is resource allocation. They introduce the notion of a resource allocation monitor. Millen [10] extended Yu-Gligor model by representing the passage of time explicitly. He also proposes a resource allocation model for resistance of denial of service attacks.

Meadows [6] makes a great contribution to development of the formal model on resistance denial of service attacks. He introduces a formal framework based on the costs spending on computation by the principles in security protocols. He argues that his formal framework can be supported by modification of NRL protocol analyzer and points out that it is not resistance of denial of service attacks. But we argue that Meadows's formal model may be not practical because the costs of generating a bogus message are small than costs of checking, so each protocols is not resistance of denial of service attacks. Based on Meadows's cost-based model Ramachandran [11] analyzes JFK protocol and point that it is resistance of denial of service attacks with the conditions bogus messages are handled in an appropriate way. Smith et al. [12] also analyze JFK protocol with Meadows's cost-based model. They point that because both of the Diffie-Hellman exponentials can be reused the coordinated attackers can launch the denial of services attacks. Tritilanunt et al.[13] firstly point out that the cost analysis has only taken into account honest runs of the protocol in Meadows's cost-based model. At the same time they also think that Meadows used only a coarse measure of computational cost. In practice it can be quite difficult to classify and compare operations in such a coarse measure. So they use the colored Petri nets to model the denial of services attacks based on cost-based and time-based model and analyzed the HIP protocol.

Far from the idea of Yu-Gligor and Meadows, Meng and Huang [8] present the first automatic method of resistance of denial of service attacks based on the extended applied pi calculus. They extended applied pi calculus from the attacker contexts and process expression, and then from the view of protocol state, an automatic method of resistance of denial of service attacks is introduced. At the same time they analyze two protocols: JFK protocol and IEEE 802.11 4 handshake protocol and find that JFK protocol is and IEEE 802.11 four-handshake protocol is not. Huang and Meng [14] also use the model to analyze Meng voting protocol.

Besides the previous models, Cuppens and Saurel [15] formalize availability policy by the four predicates expression of right. Followed Cuppens and Saurel model, Gabillon and Gallon [16] model availability as where the distribution of rights varies with the time.

Owning Meng-Huang model is the only one which supports the mechanized tool ProVerif, here we use it to analyze resistance of denial of service attacks in Acquisti protocol.

III. REVIEW OF MENG-HUANG MODEL

Here we only review the definition of resistance of denial of service attacks and method of automated proof of resistance of denial of service attacks.

A. *Definition: resistance of denial of service attacks*

P is an annotated Alice-and-bob specification in protocol, B is resistance of denial of service attacks if and only if set of association ω between any message and in set :

1) ω is null set $\emptyset$;
2) Any data items in ω are authenticated.

Where $Recv(B)$ is set where data items are in operations that are ordered in casually precedes in $act_j(B)[M_j, O_1^j, \cdots, O_k^j]$, $i, j \in [1, n], i < j$.

B. *Method of Automated Proof of Resistance of Denial of Service Attacks*

$$Alice,Bob(\rightarrow \cup \equiv)^* \ 0 \qquad Alice,Bob(\rightarrow \cup \equiv)^* \ !P$$
$$Alice,Bob(\rightarrow \cup \equiv)^* \ \nu n.P \qquad Alice,Bob(\rightarrow \cup \equiv)^* \ P \mid P'$$
$$Alice,Bob(\rightarrow \cup \equiv)^* \ c(x).P \qquad Alice,Bob(\rightarrow \cup \equiv)^* \ \bar{c}\langle N \rangle.P$$
$$Alice,Bob(\rightarrow \cup \equiv)^* \ if \ M = N \ then \ P \ else \ Q$$
$$Alice,Bob(\rightarrow \cup \equiv)^* \ if \ M = N \ then \ P \ else \ C\left[\bar{c}\langle S \rangle\right].Q \quad c \notin \tilde{n}$$

Fig. 1. Processes

Applying the extended applied pi calculus, the protocol can be modeled. We assume that the protocol exchanges messages between principles $Alice$ and Bob in a run. Principles Bob receives n messages $M_i, i \in [1, n]$. Principles Bob sends n messages $M_i', i \in [1, n]$. Protocol process $pp \equiv \nu\tilde{n}.(!Alice|!Bob)$ is a closed process and consists of parallel composition of any initiator processes $Alice$ and responder processes Bob. According to the extended applied pi calculus process $Alice$ and Bob can be reduced into one process in Fig.1.

In order to use ProVerif to automatic proof of resistance of denial of service attacks of Bob, the any messages $M_i, i \in [1, n]$ is modeled with the extended applied pi calculus. If the adversary can get the secret $secret$ on the public channel c,

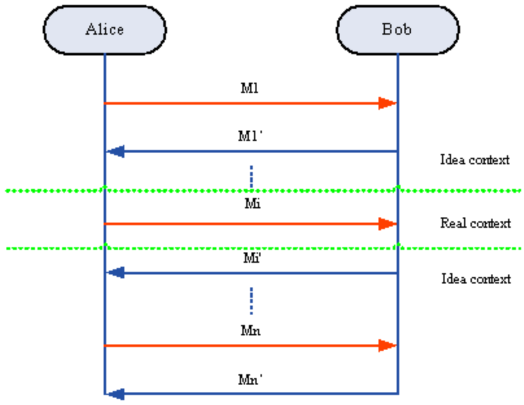

Fig. 2. The formal model of messages $M_i \quad i \in [1, n]$

then the adversary can launch a denial of service attacks by attacks of message M_i.

The method is used to model the messages $M_i, i \in [1, n]$ in Fig.2. The message M_i is exchanged and processed in real context. Real context is insecure environments where the adversary is in Dolev-Yao model. The adversary in real context can overhear, intercept, and synthesize any message and is only limited by the constraints of the cryptographic methods used. The messages $M_1, M_1', \cdots, M_{i-1}, M_{i-1}', M_i, M_{i+1}, M_{i+1}', \cdots, M_n, M_n'$ are exchanged and processed in idea context. Ideal context is secure environments. Protocol process PP is $pp \equiv \nu\tilde{n}.(!Alice|!Bob)$,$c$ is public channel.$c_j, j \in [2, n] \cap j \neq i$ are private channels used to receive messages $M_j, j \in [2, n] \cap j \neq i.Alice_i(\rightarrow \cup \equiv)^* C[\bar{c} < c_i >]\bar{c}_i < M_i > .Alice_i + 1$ $c_i \notin \tilde{n}$,$Bob_i(\rightarrow \cup \equiv)^* C[c(x)]x(m_i).Bob_i + 1$ $c \notin \tilde{n}$, $Alice_j(\rightarrow \cup \equiv)^* C[\bar{c} < c_j >]\bar{c}_j < M_j > .Alice_j + 1, c_j \in \tilde{n}, j \in [1, n] \cap j \neq i$, $Bob_j(\rightarrow \cup \equiv)^* C[\bar{c} < c_j >]\bar{c}_j < m_j > .Bob_j + 1, c_j \in \tilde{n}, j \in [1, n] \cap j \neq i$. If the adversary can get the secret message $Secret$ on the public channel c, then the adversary can launch a denial of service attacks by attacks of message M_i .

IV. ACQUISTI PROTOCOL

Acquisti protocol promises that it can implement securities without strong physical assumptions. It assumes that the private key is private and that an attacker cannot control every possible communication between the voter and an authority. In Acquisti protocol there are five entities: registration authority, issue authority, bulletin board, voters, tallying authority. Registration authority is responsible for authenticating the voters. Issue authority takes charge of issuing the related key and credentials. Voters register for voting, get their credentials and post a vote. Tallying authority is responsible for tallying ballots.

A. *Preparation phase*

Every issue authority $A_i(i = 1 \cdots l)$ creates l random numbers l as $c_{i,j}$, representing shares of credentials, for each eligible voter $voter_j(l = 1 \cdots l)$. For each $c_{i,j}$, A_i performs two operations: first, it encrypts $c_{i,j}$ using PK^c and appropriate secret randomization, signs the resulting ciphertext with SK_i^c, and publishes it on bulletin board on a row publicly reserved for the shares of credential of voter v_j : $(E^c(c_{i,j}))SK_{A_i}$. SK_{A_i}: represents the signature of authority A_i Second, A_i also encrypts $c_{i,j}$ using PK^v and appropriate secret randomization, without signing it, but attaching to it a designated verifier proof DVP_{v_j} of equality of plaintexts $E^c(c_{i,j})$ and $E^v(c_{i,j})$. The proof is designated to be verifiable only by $voter_j$. A_i encrypts this second message with $voter_j's$ public key and sends it $voter_j$:$E^{v_j}(E^v(c_{i,j}), dvp_{v_j})$ .E^{voter_j} represents RSA encryption under $voter_j's$ public key.

B. Voting phase

For each encrypted share of credential she receives,$voter_j$ verifies the designated verifier proof of equality between $E^v(C_{i,j})$ and the corresponding $E^c(C_{i,j})$ that has been signed and published in her reserved area of bulletin board. Upon successful verification, she multiplies together the shares $E^v(C_{i,j})$ Voter chooses the ballot shares $E^v(b_1^i), \cdots, E^v(b_1^s)$, generates $E^v(C_J)E^V(B_j^t) = E^B(\sum_{i=1}^{s} c_{i,j} + \sum_{i=1}^{s} b_{i,j}^t) \equiv E^V(C_J + B_J^t)$ and sends $E^s(E^v(C_J + B_J^t))$ to bulletin board.

C. Tallying phase

After the voting time expires, all ballots on bulletin board posted by allegedly eligible voters are mixed by the tallying authorities. The shares of credentials posted by the registration authorities are also combined and then mixed. Tallying authorities thus obtain two lists: a list of encrypted, mixed credentials the registration authorities themselves had originally posted on the bulletin board; and a set of encrypted, mixed sums of credentials and ballots, posted on the bulletin board by the voters. Using threshold protocols for the corresponding sets of private keys, the tallying authorities decrypt the elements in each list and then compare them through a search algorithm and publish the tallying result on bulletin board.

V. MODELING ACQUISTI PROTOCOL WITH EXTENDED APPLIED PI CALCULUS

A. Function and equational theory

We use the extended applied pi calculus to model Acquisti protocol. We model cryptography in a Dolev-Yao model as being perfect. The functions and equational theory are described in reference [17].

B. Processes

The complete formal model of Acquisti protocol in extended applied pi calculus is given in Figures below. Figures from 3 to 6 reports the basic process include main process, voter process, corrupted voter process, registration authority process, issuer authority process and tallying authority process in Acquisti protocol. The issuer authority process and tallying authority process here are described in reference [17].

The main process in Fig.3 sets up private channels $chVR, chRI_i, chRI_2, chRI_2$and specifies how the processes

```
Acquistiprotocol@ new C;new V;new S; new keyV;new keyI_1;
        new keyI_2;new chVR;new chRI_1;new chRI_2;
        out(pub,(PK(C),PK(V),PK(S),PK(keyV),PK(keyI_1),PK(keyI_2)));
        (!voter|!corrupted voter|!tallying authority|
        !issuer authority_1|!issuer authority_2|!registration authority)
```

Fig. 3. Main process

are combined in parallel.chVR is the private channel between voter and registration authority.$chRI_1$ and $chRI_2$ are the private channel between registration authority and issuer authority. At the same time the main process generates the key parameters for credentials,V for vote,S for non-homomorphic cryptosystem,$keyV$ for voter, and $keyI$ for issuer authority.

Voter process is modeled in extended applied pi calculus in Fig.4. Each voter get the shares ciphertext $KVenccered_1$ and $KVenccred_2$ from registration authority, then decrypt and get the credentials $venccered_1$,$venccred_2$ and the designated verifier proof $NZDVP_1$ and $NZDVP_2$. After that the voter verify $NZDVP_1$ and $NZDVP_2$ and the equivalence between the encrypted share $Public(NZDVP_1), decsign(Public_2(NZDVP_1))$ and the one $Public(NZDVP_2), decsign(Public_2(NZDVP_2))$.The voter also gets the encrypted shares $vencballot_i^t$ of the ballot. If the verification is true then he multiplies $cred = \prod_{i=1,2} venccred_i$ and $vote = \prod_{i=1,2} vencballot_i^t$ else output$Sectet$ by the public channel pub .finally the resulting ciphertext $TpTKenc(result, PK(s), r)$ is sent to the bulletin board.

Corrupted voters process is modeled in Fig.5. The corrupted voter will register and get his secret credentials shares $kVenccred_1$ and $kVenccred_2$ from registration authority, then decrypt and get the credentials $venccred_1$, $venccred_2$ and the designated verifier proof $NZDVP_1$ and $NZDVP_2$,after that ,he outputs $venccred_1$ and $venccred_2$ on a public channel, so that the attacker can impersonate them.

The registration authority process is modeled in Fig.6. The registration authority generates the voters id, then get $cred_1$ and $cred_2$. After that the registration authority creates designated verifier proof $NZDVP_1$ and $NZDVP_2$.

VI. MECHANIZED PROOF OF ACQUISTI PROTOCOL WITH PROVERIF

We use the extended pi calculus in Meng-Huang model as the input of ProVerif. In order to prove resistance of denial of services attacks in Acquisti protocol, the extended applied pi calculus is needed to be translated into the syntax of ProVerif and generated the ProVerif inputs. The input code is in Fig.7. The result of resistance of denial of services attacks in Acquisti protocol is in Fig.8. Owning to that the adversary can get the secret message on the public channel , Acquisti protocol is not resistance of denial of services attacks. In Acquisti protocol there are two resistance of denial of services attacks by us.

1) In preparation phase issuer authority publishes public keys PK^c, PK^v, PK^s on bulletin board without protecting security of these public keys by public channels. Thus the adversary can intercept public keys

```
Voter ≜
    in(chVR,id);in(pub,pkv);in(pub,pkc);in(chVR,kencNZDVP_1);in(chVR,kencNZDVP_2);
    let  NZDVP_1 = PKdec[(kencNZDVP_1,r_1,r_2),SK(keyV)]  in
    let  NZDVP_2 = PKdec[(kencNZDVP_2,r_1',r_2'),SK(keyV)]  in
    if   CheckNZDVPp(DVPsign(Public_3(NZDVP_1),SK(keyV)),VK(keyV),Public_3(NZDVP_1))   then
    if   CheckNZDVPp(DVPsign(Public_3(NZDVP_2),SK(keyV)),VK(keyV),Public_3(NZDVP_2))   then
    if   checkciphertext(Public_1(NZDVP_1),decsign(Public_2(NZDVP_1)),(pkv,r_1),(pkc,r_2)) = true  then
    ( if  checkciphertext(Public_1(NZDVP_2),decsign(Public_2(NZDVP_2)),(pkv,r_1'),(pkc,r_2')) = true  then
          ( let  cred = ∏_{i=1,2} Public_1(NZDVP_i)   in
                let  vote = ∏_{i=1,2} vencballot_i^t   in
                let   result = cred × vote   in              else   out(pub,Secret)
                new    r;
                out(pub,TpPKenc(result,PK(S),r));
          )
    )
    else   out(pub,Secret)
```

Fig. 4. Voter process

```
Corrupted voter ≜
    in(chVR,id);in(pub,pkv);in(pub,pkc); in(chVR,kencNZDVP_1);in(chVR,kencNZDVP_2);
    let   NZDVP_1 = PKdec[(kencNZDVP_1,r_1,r_2),SK(keyV)]  in
    let   NZDVP_2 = PKdec[(kencNZDVP_2,r_1,r_2),SK(keyV)]  in
    if   CheckNZDVPp(DVPsign(Public_3(NZDVP_1),SK(keyV)),VK(keyV),Public_3(NZDVP_1))  then
    if   CheckNZDVPp(DVPsign(Public_3(NZDVP_2),SK(keyV)),VK(keyV),Public_3(NZDVP_2))  then
    out(pub,(Public_1(NZDVP_1),Public_1(NZDVP_2)));
```

Fig. 5. corrupted voter process

```
Registration authority ≜
    new id;out(pub,id);out(chVR,id);new cred;
    out(chRI_1,(id,cred));out(chRI_2,(id,cred));
    in(chRI_1,(id,cred_1));in(chRI_2,(id,cred_2));
    new r_1;new r_2;
    out(chVR,PKenc((NZDVP_1,r_1,r_2),PK(keyV)));
    out(chVR,PKenc((NZDVP_2,r_1,r_2),PK(keyV)));
    NZDVP_i = ZK_{6,4}( cred_i,r_1,r_2,C,V,DVPsign(m,SK(keyV));pPKenc(cred_i,PK(V),r_1),
                        sign(pPKenc(cred_i,PK(C),r_2),SK_i(C)),m,VK(keyV); )
```

Fig. 6. Registration authority process

PK^c, PK^v, PK^s and modify it, then send it to bulletin board. In voting phrase voter v_j verifies the designated verifier proof of equality between $E^v(c_{i,j})$ and the corresponding $E^c(c_{i,j})$ that has been signed and published in her reserved area of bulletin board. $E^c(c_{i,j})$ has been publish on bulletin board with digital signature with authority. Owning the adversary has modified the public keys PK^c, PK^v, PK^s, hence the verification is not success, thus voter v_j can not vote. Hence attacker can make a resistance of denial of services attacks. In order to protect Acquisti protocol against the denial of service attack we can use the digital certificate to distribute these public keys: PK^c, PK^v, PK^s.

2) In preparation phase if it is not at the same time that the issuer authority publishs $(E^c(c_{i,j}))_{SK_{A_i}}$ on bulletin board and sends $E^{v_j}(E^V(c_{i,j}, P_{v_j}))$ to voter v_j, then for voter v_j there also is a resistance of denial of services attacks. The adversary can intercept $(E^c(c_{i,j}))_{SK_{A_i}}$ or $E^{v_j}(E^V(c_{i,j}, P_{v_j}))$ and modify it, then send it to BB and voter v_j, respectively. Voter v_j verifies $E^{v_j}(E^V(c_{i,j}, P_{v_j}))$ in voting phrase, the verification will fail, thus voter v_j can not vote. Hence

```
fun pPKenc/3.
fun pPKdec/2.
fun PKenc/2.
fun PKdec/2.
fun check/2.
fun sign/2.
fun decsign/2.
fun verifysign/2.
fun TPKsubdec/3.
fun TpPKsubdec/3.
fun TPKdec/2.
fun TPKenc/2.
fun TpPKenc/3.
fun TpPKdec/2.
fun SK/1.
fun VK/1.
fun PK/1.
fun checkciphertext/4.
fun add/2.
fun multi/2.
fun equals/2.
fun selfBlinding/2.
fun projection1/1.
fun projection2/1.
fun zk/2.
fun zkver/1.
fun public1/1.
data true/0.
equation pPKdec(pPKenc(x,PK(y),z),SK(y))=x.
equation PKdec(PKenc(x,PK(y)),SK(y))=x.
equation decsign(sign(x,SK(y)),PK(y))=x.
equation verifysign(sign(x,SK(y)),x)=true.
equation equals(x,x)=true.
equation check(sign(x,y),VK(y))=x.
equation add(projection1(x),projection2(x))=x.
equation add(projection2(x),projection1(x))=x.
equation TPKdec(TPKenc(x,PK(y)),SK(y))=x.
equation multi(TpPKenc(a,PK(y),r),
TpPKenc(b,PK(y),z))=TpPKenc(add(a,b),PK(y),r).
equation checkciphertext(TpPKenc(x,y,r1),TpPKenc(x,z,r2),(y,r1),(z,r2))=true.
equation checkciphertext(TpPKenc(x,PK(y),r1),TpPKenc(x,PK(z),r2),y,z)=true.
equation public1(zk(x,y))=y.
equation zkver(zk((cred1,cred2,fake),(TpPKenc(cred1,PK(V),r1),
TpPKenc(cred2,PK(V),r2),PK(V),r1,r2,x,y)))=true.
equation zkver(zk((fake,cred2,sign(m,voter)),
(x,TpPKenc(cred2,PK(V),r2),PK(V),r,r2,m,VK(voter))))=true.
equation zkver(zk((cred1,fake,sign(m,voter)),
(TpPKenc(cred1,PK(V),r1),x,PK(V),r1,r,m,VK(voter))))=true.
free pubp,pub.
private free pubv,pubt,chvote.
free va,vb.
free n1,n2.
private free secret.
query attacker:secret.
let votechooser =
          out(chvote,va) | out(chvote,vb).
let voter=
        new nonce;
        new nonce1;
        out(chVR,(n1,nonce));
        in(chVR,(=n2,=nonce,cenccred1,cenccred2,r1,r2,ct));
        let zkp=PKdec(ct,SK(voter)) in
        if zkver(zkp)=true then
        (
        let (enccred1,enccred2,PKV,r3,r4,m,vk)=public1(zkp) in
        in(chBBV,(pkc,pkv));
        if checkciphertext(enccred1,cenccred1,(pkv,r3),(pkc,r1))=true then
(
       if checkciphertext(enccred2,cenccred2,(pkv,r4),(pkc,r2))=true then
       (
          let cred=multi(enccred1,enccred2) in
          in(chvote,vote);
          new r5;new r6;
 let encvote=multi(TpPKenc(projection1(vote),PK(V),r5),
          TpPKenc(projection2(vote),PK(V),r6)) in
          let ballot=multi(cred,encvote) in
          let res=PKenc(ballot,PK(S)) in
          out(pubv,res)
          )
          else out(pub,secret)
          )
          else out(pub,secret)
          )
          else out(pub,secret).
let corruptedvoter=
        new nonce;
        out(chVR,(n1,nonce));
        in(chBBV,(pkc,pkv));
        in(chVR,(=n2,=nonce,cenccred1,cenccred2,r1,r2,ct1));
        out(pub,ct1).

let tallying_authority=
        new nonce;
        out(chRT,(n1,nonce));
        in(chRT,(=n2,=nonce,enccred1,enccred2));
        in(pubv,res);
        let cenccred=multi(enccred1,enccred2) in
        let result=PKdec(res,SK(S)) in
        new r1;new r2;
        let cencvotea=multi(TpPKenc(projection1(va),PK(C),r1),
TpPKenc(projection2(va),PK(C),r2)) in
        let cencvoteb=multi(TpPKenc(projection1(vb),PK(C),r1),
TpPKenc(projection2(vb),PK(C),r2)) in
        let test1=multi(cenccred,cencvotea) in
        let test2=multi(cenccred,cencvoteb) in
        if true=checkciphertext(test1,result,C,V) then out(pubt,va) else
         if true=checkciphertext(test2,result,C,V) then out(pubt,vb).
let BB=
        in(pubp,pkc);
        in(pubp,pkv);
        in(pubp,pks);
        !(out(chBBV,(pkc,pkv))).
 let registration_authority=
        in(chVR,(=n1,nonceV));
        in(chRT,(=n1,nonceT));
        new nonce;
        out(chIIR,(n1,nonce));
        in(chIIR,(=n2,=nonce,id));
        new cred;
        let cred1=projection1(cred) in
        let cred2=projection2(cred) in
        new r1;new r2;new m;new r3;new r4;
        out(chRT,(n2,nonceT,TpPKenc(cred1,PK(C),r1),
TpPKenc(cred2,PK(C),r2))); out(chVR,(n2,nonceV,TpPKenc(cred1,PK(C),r1),
TpPKenc(cred2,PK(C),r2),r1,r2,PKenc(zk((cred1,cred2,sign(m,voter)),
(TpPKenc(cred1,PK(V),r3),
TpPKenc(cred2,PK(V),r4),PK(V),r3,r4,m,VK(voter))),PK(voter)))).
  let issuer_authority=
        in(chIIR,(=n1,nonceR));
        new id;
        out(chIIR,(n2,nonceR,id));
        out(pub,id).
process new C;new V;new S;
         new voter;
         new chVII;
         new chIIR;
         new chRT;
         new chBBV;
         new chVR;
         out(pubp,PK(C));
         out(pubp,PK(V));
         out(pubp,PK(S));
         out(pub,PK(voter));
         ((!voter)|(!corruptedvoter)|(!tallying_authority)|
(!registration_authority)|(!issuer_authority)|(!votechooser)|(BB))
```

Fig. 7. The input code for Acquisti Protocol

the adversary constructs a resistance of denial of services attack. In order to protect Acquisti protocol against the denial of service attack we can make the operation on publishing $(E^c(c_{i,j}))_{SK_{A_i}}$ on bulletin board and on sending $E^{v_j}(E^V(c_{i,j}, P_{v_j}))$ to voter v_j as an atomic action.

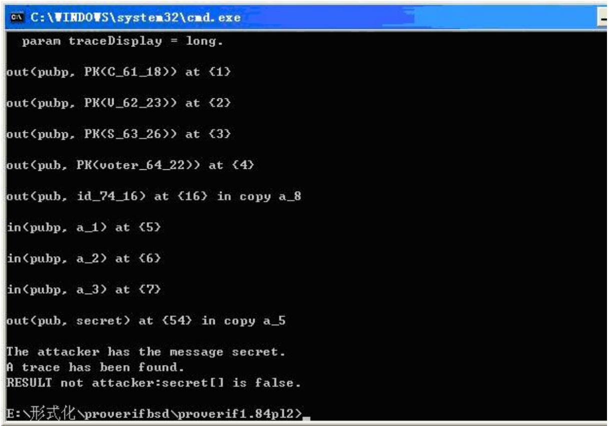

Fig. 8. The result of resistance of denial of services attacks in Acquisti protocol

VII. CONCLUSION

Internet voting protocol play an important role in remote voting system. Acquisti protocol is one of the most important remote internets voting protocol that claims to satisfy formal definitions of key properties without strong physical constrains. Owning to the huge damage and hard to prevention of denial of service attacks in security protocol, the secure remote internet voting protocol should also have resistance of denial of service attacks. To our best knowledge until now resistance of denial of service attacks in Acquistic protocol is not analyzed. Recently owning to the contribution of Meng and Huang, Acquisti protocol can be proved with mechanized proof tool ProVerif. In this paper we apply the mechanized formal model proposed by Meng and Huang for mechanized proof of resistance of denial of service attacks. The result we obtain is that Acquisti protocol has not resistance of denial of service attacks. Two denial of service attacks are found by us. At the same time we give the method against the denial of service attacks. To our best knowledge, we are conducting the first mechanized proof of resistance of denial of service attacks in Acquisti protocol for an unbounded number of honest and corrupted voters. As future work, it would be interesting to formalize the security properties of remote internet voting protocols in the computational model with mechanized tool CryptoVerif.

ACKNOWLEDGMENT

This study was supported in part by Natural Science Foundation of The state Ethnic Affairs Commission of PRC under the grants No: 10ZN09.

REFERENCES

[1] A.Acquisti. Receipt-Free Homomorphic Elections and Write-in Voter Verified Ballots. *Technical Report 2004/105*. International Association for Cryptologic Research, May 2, 2004, and Carnegie Mellon Institute for Software Research International, CMU-ISRI-04-116, 2004.

[2] M.R.Clarkson, S. Chong, and A.C. Myers, "Civitas: Toward a secure voting system," *In Proceeding of the 2008 IEEE Symposium on Security and Privacy,* Oakland, California, USA, 2008, p.354-368.

[3] B.Meng,"A critical review of receipt-freeness and coercion-resistance". *Information Technology Journal, vol.8*, vol.8, no. 7, pp. 934-964, 2009.

[4] B.Meng, "A secure non-interactive deniable authentication protocol with strong deniability based on discrete logarithm problem and its application on internet voting protocol," *Information Technology Journal*, vol.8, no.3, pp. 302-309, 2009.

[5] B.Meng, Z. Li ,and J. Qin, " A receipt-free coercion-resistant remote internet voting protocol without physical assumptions through deniable encryption and trapdoor commitment scheme," *Journal of software,* vol.5, no.9, pp. 942-949, 2010.

[6] C.Meadows , "A cost-based framework for analysis of denial of service networks," *Journal of Computer Security,* vol.9, no.1/2, pp. 143-164, 2001.

[7] C.F.Yu ,and V.D. Gligor, 1990. "A formal specification and verification method for the prevention of denial of service," *Journalc on communictions*, to be published.

[8] B.Meng ,and W.Huang, "Automated Proof of Resistance of Denial of Service Attacks with Theorem Prover," *Journalc on communictions*, to be published.

[9] E.Bacic, and M. Kuchta, "Considerations in the preparation of a set of availability criteria," *In Proceedings of 3rd Annual Canadian Computer Security Symposium,* Ottawa, Canada, 1991,p. 283-292.

[10] J.K.Millen, "A Resource Allocation Model for Denial of Service Protection,"*Journal of Computer Security,* vol.2, no.2-3, pp. 89-106, 1993.

[11] V.Ramachandran, Analyzing DoS-resistance of protocols using a cost-based framework, *Technical Report DCS/TR-1239*, Harlow, Yale University, USA, 2002.

[12] J.Smith, J.M. Gonzalez-Nieto, and C. Boyd, "Modelling denial of service attacks on JFK with Meadows's cost-based framework," *In Proceedings of the 2006 Australasian workshops on Grid computing and e-research,* Australia, 2006, p.125-134.

[13] S.Tritilanunt, C. Boyd, E. Foo, and J.M.G. Nieto, "Cost-based and time-based analysis of DoS-resistance in HIP," *In Proceedings of the thirtieth Australasian conference on Computer science,* 2007, p.191-200.

[14] W.Huang ,and B.Meng, "Automated Proof of Resistance of Denial of Service Attacks in Remote Internet Voting Protocol with Extended Applied Pi Calculus,"*Information Technology Journal*, vol.10, no.8, pp. 1468-1483, 2011.

[15] F.Cuppens ,and C. Saurel, "Towards a formalization of availability and denial of service,". *In Proceedings of In Information Systems Technology Panel Symposium on Protecting Nato Information Systems in the 21st Century*, Washington, 1999.

[16] A.Gabillon, and L. Gallon, "An Availability Model for Avionic Data Buses," *In Proceedings of. Workshop on Issues in Security and Petri Nets.* University of Eindhoven, Netherlands, 2003.

[17] B.Meng,W.Huang,and D.J.Wang "Automatic Verification of Remote Internet Voting Protocol in Symbolic Model," *Journal of Networks*, Vol 6, No. 9 pp. 1262-1271, 2011.

Bo Meng was born in 1974 in China. He received his M.S. degree in computer science and technology, Ph.D. degree in traffic information engineering and control from Wuhan University of Technology, at Wuhan, China, in 2000, 2003, respectively. From 2004 to 2006, he works in Wuhan University, China as Postdoctoral researcher in information security. Currently he is an Associate Professor in school of computer, South-Center University for Nationalities, China. He has authored/coauthored over 50 papers in International/National journals and conferences. His current research interests include electronic commerce, Internet voting, and protocol security.