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. 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.

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 φ;
2) Any data items in ω are authenticated.

Where Receive(B) is set where data items are in operations that are ordered in casually precedes in actj(B)[Mj, O1, ..., Oi, i, j ∈ [1, n], i < j.

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

![Image](https://example.com/image)

Fig. 2. The formal model of messages Mi, i ∈ [1, n]

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 ∈ [1, n]. Principles Bob sends n messages M_i, i ∈ [1, n]. Protocol process PP ≡ vni.(|Alice||Bob),c is public channel c, j ∈ [2, n] and j ≠ i are private channels used to receive messages M_i, j ∈ [2, n] and j ≠ i. Alice_c_(i→∪≡^*C[ε < c_j >]ε_j < M_i > .Alice_c_i + 1 c_i的食物 n,Bob_i_(→∪≡^*C[c(x)x|m_i,)].Bob_i + 1 c ≠ n, Alice_j_(→∪≡^*C[ε < c_j >]ε_j < M_j > .Alice_j + 1, c_j ∈ n, j ∈ [1, n] and j ≠ i, .Bob_j_(→∪≡^*C[ε < c_j >]ε_j < M_j > .Bob_j + 1, c_j ∈ n, j ∈ [1, n] and j ≠ 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.

The method is used to model the messages M_i, i ∈ [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_2, ..., M_n, Mi, M_{i+1}, ..., Mn, M_i are exchanged and processed in idea context. Ideal context is secure environments. Protocol process PP is pp ≡ vni.(|Alice||Bob),c is public channel c, j ∈ [2, n] and j ≠ i are private channels used to receive messages M_i, j ∈ [2, n] and j ≠ i. Alice_c_(i→∪≡^*C[ε < c_j >]ε_j < M_i > .Alice_c_i + 1 c_i的食物 n, Bob_i_(→∪≡^*C[c(x)x|m_i,)].Bob_i + 1 c ≠ n, Alice_j_(→∪≡^*C[ε < c_j >]ε_j < M_j > .Alice_j + 1, c_j ∈ n, j ∈ [1, n] and j ≠ i, .Bob_j_(→∪≡^*C[ε < c_j >]ε_j < M_j > .Bob_j + 1, c_j ∈ n, j ∈ [1, n] and j ≠ 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, ..., 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 $c_{i,j}$, $A$ performs two operations: first, it encrypts $c_{i,j}$ using $PK^v$ and appropriate secret randomization, signs the resulting ciphertext with $SK^e$, and publishes it on bulletin board on a row publicly reserved for the shares of credential of voter $v_j$ : $(E^e(c_{i,j}))SK_A$, $SK_A$: represents the signature of authority $A$. Second, $A$ also encrypts $c_{i,j}$ using $PK^v$ and appropriate secret randomization, without signing it, but attaching to it a designated verifier proof $DV_{P_2}$ of equality of plaintexts $E^e(c_{i,j})$ and $E^v(c_{i,j})$. The proof is designed to be verifiable only by $voter_j$. $A$ encrypts this second message with $voter_j$’s public key and sends it $voter_j: E^v(\sum_{i=1}^{l} c_{i,j})$, $dep_{v_j}$. $E^v(\sum_{i=1}^{l} c_{i,j})$ 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 $E^e(C_{i,j})$ and the corresponding $E^v(C_{i,j})$ that has been signed and published in her reserved area of bulletin board. Upon successful verification, she multiplies together the shares $E^e(C_{i,j})$ Voter chooses the ballot shares $E^e(b_i^j), \cdots, E^e(b_l^j)$, generates $E^e(C_j) E^e(B_j^l) = E^e(\sum_{i=1}^{l} c_{i,j} + \sum_{j=1}^{l} b_i^j) \equiv E^v(C_j + B_j^l)$ and sends $E^v(\sum_{i=1}^{l} c_{i,j})$ 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 $chVR, chRI_1, chRI_2, chRI_2$ and specifies how the processes 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 voter,$S$ for non-homomorphic cryptosystem. key$V$ for voter, and key$I$ for issuer authority.

Voter process is modeled in extended applied pi calculus in Fig.4. Each voter get the shares ciphertext $KVenccred_1$ and $KVenccred_2$ from registration authority, then decrypt and get the credentials $venccred_1, venccred_2$ and the designated verifier proof $NZDVPI$ and $NZDVPI_2$. After that the voter verify $NZDVPI$ and $NZDVPI_2$ and the equivalence between the encrypted share Public(NZDVPI), desgn(Public2(NZDVPI_2)) and the one Public(NZDVPI_2), desgn(Public2(NZDVPI_2)). The voter also get the encrypted shares venccballot$1$ of the ballot. If the verification is true then he multiplies $cred = \prod_{i=1}^{2} venccred_i$ and $vote = \prod_{i=1}^{2} venccballot_i$. The voter another outputs Secret by the public channel pub finally the resulting ciphertext $TipTKenc(result, PK(s), r)$ 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 $NZDVPI$ and $NZDVPI_2$ 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 the designated verifier proof $NZDVPI$ and $NZDVPI_2$.

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. Acquisti protocol there are two resistance of denial of services attacks by us.

1) In preparation phase issuer authority publishes public keys $PK^e, PK^v, PK^s$ on bulletin board without protecting security of these public keys by public channels. Thus the adversary can intercept public keys.
In voting phrase, the voter votes to protect Acquisition protocol against the denial of service attacks. The adversary can intercept denial of services attacks. In order to make a resistance of denial of services attacks, the voter must verify its identity, then for voter \( v_j \), then for voter \( v_j \) there also is a resistance of denial of services attacks. The adversary can intercept denial of services attacks. In order to protect Acquisition protocol against the denial of service attack we can use the digital certificate to distribute these public keys: \( PK^e, PK^v, PK^s \).

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

\[
\text{Voter} v_j
\]

\[
in χVR(id);in (pub,pk), in χVR, kencNZDVp
\]

\[
\text{let} \ NZDVp = PKdec\left(\text{kencNZDVp}_{,r}, SK_{keyV}\right)
\]

\[
\text{let} \ NZDVp = PKdec\left(\text{kencNZDVp}_{,r}, SK_{keyV}\right)
\]

\[
\text{if} \ CheckNZDVp\left(DVPSign\left(Public\left(\text{NZDVp}\right), SK_{keyV}\right), VK_{keyV}, Public\left(\text{NZDVp}\right)\right)
\]

\[
\text{if} \ CheckNZDVp\left(DVPSign\left(Public\left(\text{NZDVp}\right), SK_{keyV}\right), VK_{keyV}, Public\left(\text{NZDVp}\right)\right)
\]

\[
\text{if} \ checkciphertext\left(Public\left(\text{NZDVp}\right), \text{decSig}\left(Public\left(\text{NZDVp}\right)\right)\right) = \text{true} \text{ then}
\]

\[
\text{let cred} = \prod_{i=1}^{n} Public\left(\text{NZDVp}\right)
\]

\[
\text{let vote} = \prod_{i=1}^{n} \text{votecallot}_{i}
\]

\[
\text{let result} = \text{cred} \cdot \text{vote} \text{ in new } r;
\]

\[
\text{out} \left(\text{pub}, TP\text{PKenc}(\text{result}, PK\left(S_{i}, r\right))\right)
\]

\[
\text{else out} \left(\text{pub}, \text{Secret}\right)
\]

\[
\text{Corrupted voter} v_j
\]

\[
in χVR(id); in (pub,pk), in χVR, kencNZDVp
\]

\[
\text{let} \ NZDVp = PKdec\left(\text{kencNZDVp}_{,r}, SK_{keyV}\right)
\]

\[
\text{let} \ NZDVp = PKdec\left(\text{kencNZDVp}_{,r}, SK_{keyV}\right)
\]

\[
\text{if} \ CheckNZDVp\left(DVPSign\left(Public\left(\text{NZDVp}\right), SK_{keyV}\right), VK_{keyV}, Public\left(\text{NZDVp}\right)\right)
\]

\[
\text{if} \ CheckNZDVp\left(DVPSign\left(Public\left(\text{NZDVp}\right), SK_{keyV}\right), VK_{keyV}, Public\left(\text{NZDVp}\right)\right)
\]

\[
\text{if} \ checkciphertext\left(Public\left(\text{NZDVp}\right), \text{decSig}\left(Public\left(\text{NZDVp}\right)\right)\right) = \text{true} \text{ then}
\]

\[
\text{let cred} = \prod_{i=1}^{n} Public\left(\text{NZDVp}\right)
\]

\[
\text{let vote} = \prod_{i=1}^{n} \text{votecallot}_{i}
\]

\[
\text{let result} = \text{cred} \cdot \text{vote} \text{ in new } r;
\]

\[
\text{out} \left(\text{pub}, Public\left(\text{NZDVp}\right), Public\left(\text{NZDVp}\right)\right)
\]

\[
\text{Registration authority}
\]

\[
\text{new } id; \text{out} \left(\text{pub}, id\right); \text{out} χVR(id); \text{new cred};
\]

\[
\text{out} χRI(id, cred); \text{out} χRI(id, cred);
\]

\[
\text{in} χRI(id, cred); \text{in} χRI(id, cred);
\]

\[
\text{new } r, \text{new } r;
\]

\[
\text{out} \left(\text{pub}, \text{PKenc}(NZDVp_{,r}, SK\left(KV\right))\right);
\]

\[
\text{out} \left(\text{pub}, \text{PKenc}(NZDVp_{,r}, PK\left(KV\right))\right);
\]

\[
\text{NZDVp} = ZK_{v_j}(cred_{,r}, C, V, DVPSign(m, SK\left(KV\right)); PKenc cred_{,PK\left(KV\right)});
\]

\[
\text{sign}_{PKenc cred}\left(PK\left(C\right)_{,r}, SK\left(C\right), m, VK\left(KV\right)\right);
\]
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 $(E^p(v_{i,j}))_{SKA}$ on bulletin board and on sending $E^p(E^q(c_{i,j}, P_{v_j}))$ 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 internet 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 Acquisti 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


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.