Annales UMCS Informatica AI 6 (2007) 171-183

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI

http://www.annales.umcs.lublin.pl/

Attack and revison of an electronic auction protocol using OFMC

Bogdan Księżopolski[1], Pascal Lafourcade[2]

[1]*Institute of Computer Science, Maria Curie-Skłodowska University,*
pl. M.Curie-Skłodowskiej 5, 20-031 Lublin, Poland
[2]*Information Security, ETH Zürich, IFW C46.1 Haldeneggsteig 4 CH-8092 Zürich, Switzerland*

Abstract

In the article we show an attack on the cryptographic protocol of electronic auction with extended requirements [1]. The found attack consists of authentication breach and secret retrieval. It is a kind of "man in the middle attack". The intruder impersonates an agent and learns some secret information. We have discovered this flaw using OFMC an automatic tool of cryptographic protocol verification. After a description of this attack, we propose a new version of the e-auction protocol. We also check with OFMC the secrecy for the new protocol and give an informal proof of the other properties that this new e-auction protocol has to guarantee.

1. Introduction

Nowadays, we are witnesses and participants of ubiquitous changes in everyday activities. These changes are connected with the development of technological world. Among the researchers information communications are the most widespread. The high stress is put on the development of well-available, mobile information services called "e-anything", like e-government, e-money, e-banking and e-auction. These mentioned processes are fulfilled mainly in an electronic way, thanks to which one can increase their availability, cutting down the costs according to the traditional way of these services. There are many electronic services in the e-commerce one of them are electronic auctions. For instance, the auction websites, such as eBay, are popular the number of and their users is still growing.

e-auction: The auction schemes can be divided into four groups:

– The English auction [2], for instance eBay, is the most widespread. In this auction the users bid the price for a given object. The prices grow till the end of the auction. Hence the winner is the bidder who proposed the highest price.

- 1st Price Sealed-Bid [1,3], in this case the users independently define the price for given goods. The price defined by the bidder can not be increased and has only one value. The user who will declare the highest price wins goods and he has to pay the announced price.
- The Vickrey auction [4], alternatively called 2nd Price Sealed-Bid, is very similar to the previous scheme. The difference is that the auction is won by the person, who declared the highest sum the given goods but he pays the second highest price proposed.
- The Dutch auction [5] starts with the highest possible price and the bidders decrease the price until one bidder decides to pay the current value. The winning bidder pays the price on which the auction is stopped.

Attack and Verification: After creating the cryptographic protocol the post-designing analysis should be done. It is important because there are no guarantees that a protocol is secure. In literature a taxonomy of cryptographic protocol attacks is proposed. The taxonomies can be based on the informal methods or formal verification. Informal taxonomies are based on the protocol flaws [6] or on replay attacks in terms of message origin and destination [7] or on an intruder's attack objectives and different roles of the parts of protocol [8]. For instance in [8] the authors grouped the attacks into seven categories:

- *Authentication Breach*: The intruder finishes a protocol run in order to impersonate a legitimate principal.
- *Authentication Breach + Secret Retrieval*: The intruder finishes a protocol run in order to impersonate a legitimated agent which accepts the secret and to retrieve the secret.
- *Authentication Breach + Secret Revival*: The intruder finishes a protocol run in order to impersonate a legitimate principal and to receive an old secret. The intruder must impersonate the agent which generates the secret.
- *Authentication Breach + Secret Injection*: The intruder finishes a protocol run in order to impersonate a legitimate principal and to inject a secret of his own. The intruder must impersonate the agent which generates the secret.
- *Message Generation*: The intruder makes a protocol run until some stage such that he obtains a new valid fake message.
- *Secret Retrieval*: The intruder retrieves the secret distributed between two legitimate principals.
- *Session Hijacking*: The intruder takes over a protocol run after two legitimate principals successfully authenticates each other and before the secret is received by the participant which accepts the secret.

The usage and the popularity of the electronic auction is connected with the fulfillment of a proper level of security of information sent between different parties using cryptographic protocols [9]. For instance, the e-auction protocols

have to guarantee confidentiality, privacy, integrity, non-repudiation of agents and anonymity of bidders. These security requirements are solved by the conception of the protocols using some cryptographic primitives. Achieving these properties correctly is not obvious and an automatic verification is useful to avoid some flaws. There are many tools to check formally and automatically the cryptographic protocols [10-17].

In the remaining part of the paper, we present an attack on the electronic auction protocol with extended requirement [1], found by the tool OFMC [17] one of the tools of AVISPA [16]. This flaw is based on an authentication breach and a secret retrieval. We also revise the mentioned protocol of e-auction by improving its security properties.

Plan of the paper: In the following section, we give the properties satisfied by an e-auction protocols and the notations used in the paper. In Section 3 we recall the e-auction protocol [1]. In Section 4, we present the attack on this protocol. Then we give a new version of this protocol in Section 5. Before to concluding in the last section, we explain different security properties satisfied by our new version of the protocol.

2. Properties of e-auction and notations

The e-auction protocols aim at guarantee of the following features:

- *Secrecy of bids*: Nobody, except the bidder himself and the auctioneer, can establish the contents of the sent offers.
- *Integrity of data*: The sent offers and the final results of e-auction cannot be modified.
- *Non-repudiation*: Bidders cannot deny the contents of the offer and the fact that he made it.
- *Authentication of participants*: Only registered persons can announce e-auction and make auction offers.
- *Anonymity of bidders*: The true identity of bidder who won the auction and also the identity of all the bidders are not public.
- *Public verification*: Everyone can check, which offer won e-auction. Participants of e-auction can check if their offers were taken into consideration.

In the remaining part of the paper, we use the following notations to describe the entities which will take part of the e-auction protocol:

- *A*: registered principal who wants to sell the object (Auctioneer).
- *B*: registered principal who wants to win the object (Bidder).
- *C*: principal who wants to register.
- *TTP*: trustworthy third part.
- *WWW*: place for public information.

Notice that C is a generic name representing either an auctioneer A or a bidder B. We also use for the messages the following notations:

- PK_C: public key of the agent C.
- SK_C: private key of the agent C.
- N_A: nonce associated to the agent A.
- T_{N_A} : time stamp associated to the nonce N_A.
- NR_A: registration number associated to the agent A.

3. Description of the old electronic auction protocol

We now describe the 1st Price Sealed-Bid protocol proposed in [1]. The cryptographic protocol with extended requirements consists of four subprotocols: certification, notification of auction, notification of offer and the choice of offer. The first step is the registration of participants taking part in the e-auction using *TTP*. The subprotocol of certification is used by all the participants: the bidders B and the auctioneer A. The next step is the auction notification by registered principal A. In the subprotocol of notification of auction, the *TTP* publishes the notified by A auction properties. In the next subprotocol for offer notification, each registered person can take part in auction by sending his offer to *TTP*. The last subprotocol is executed once the e-auction is closed i.e. after elapsing of time for notification of offers. Then the auctioneer A and the bidders, send to *TTP* their parts of secret needed to read offers. After decoding them, the *TTP* sends it to the auctioneer A, who chooses the vinning offer and sends back information about the vinning offer to *TTP*. Finally, the *TTP* publishes the winning auction number using *WWW*. In the following we describe more precisely each step of the protocol.

3.1. The certification subprotocol

The participation in e-auction has to be preceded by obtaining suitable authorizations, and is described in Figure 1.

1. $C \rightarrow TTP : \left\{ \{ D_C \}_{SK_C} \right\}_{PK_{TTP}}$
2. $C \leftarrow TTP : \left\{ \left\{ \{ NR_C \}_{SK_{TTP}}, T_{NR_C}, (SK_C, PK_C) \right\}_{SK_{TTP}} \right\}_{PK_C}$

Fig. 1. Certification subprotocol

This subprotocol works as follows. A person who applies for certificate, denoted by C, is either an auctioneer or a bidder. He should possess appropriate documents D_C as well as private key SK_C achieved from one of indicated earlier

centers of authorization. After that the documents mentioned above are digitally signed by using SK_C and encrypted by the public key PK_{TTP}. Then C sends them to *TTP*. Hence the *TTP* decrypts documents and then verifies them. After positive verification, the *TTP* generates unique registration number for a given person NR_C. Registration number is valid during definite time, given by the time stamp of registration number T_{NR_C}. The *TTP* generates also the private key SK_C and the public key PK_C, which will be used in the next subprotocol. Validity of these keys ends along with crossing the time given by T_{NR_C}. The *TTP* digitally signs the generated data, encrypts them by the public key PK_C and then sends them to C.

3.2. The auction notification subprotocol

This subprotocol is designed for the agent A which wants to announce the electronic auction. In the protocol only registered principals can take part in the rest of the e-auction protocol. That requirement will be fulfilled when the agent finishes with success the previous subprotocol of certification.

1. $A \rightarrow TTP : \left\{ \left\{ NR_A, T_{NR_A}, AP_A, N_A \right\}_{SK_A} \right\}_{PK_{TTP}}$
2. $A \leftarrow TTP : \left\{ \left\{ SK_{P(A)} \right\}_{SK_{TTP}} \right\}_{PK_A}$
3. $TTP \rightarrow WWW : Nb_{Au}, AP_A, PK_{Au}$

Fig. 2. Auction notification subprotocol

This subprotocol works as follows and is described in Figure 2. In the first step, A sends to *TTP*, digitally signed by SK_A and encrypted by PK_{TTP} the following information: the registration number NR_A, the time stamp T_{NR_A}, the conditions of auction AP_A and his individual number N_A. The main auction agency *TTP* verifies the registration number of A, NR_A and the validity of his time stamp. After positive authorization *TTP* generates the individual number of auction Nb_{Au} and the pair of keys for concrete auction (SK_{Au}, PK_{Au}). The private key of auction SK_{Au} is divided by the use of the threshold scheme of dividing secret [19]. Secret is divided into three parts, designed for A: $SK_{P(A)}$, for *TTP*: $SK_{P(TTP)}$ and for bidders in auction: $SK_{P(B)}$. Each part is necessary to reconstruct the full private key SK_{Au}. The *TTP* sends digitally signed by SK_{TTP} and encrypted by PK_A – the part of secret designed for A, $SK_{P(A)}$. Hence, the *TTP* publishes the number of auction Nb_{Au}, auction properties AP_A and the public key of the concrete auction.

3.3. The auction offer subprotocol

After the auction is notified and published, the interested parties can notify their offers. A bidder who wants take part in the auction should gets earlier the registration number NR_B, the private key SK_B and his offer OF_B. Then the bidder B, generates his individual number Nb_B and marks his offer by time stamp KT_{OF_B}.

1. $B \rightarrow TTP : \left\{ \{OF_B\}_{SK_B} \right\}_{PK_{Au}}, \left\{ \{Nb_B, NR_B, Nb_{Au}, T_{OF_B}\}_{SK_B} \right\}_{PK_{TTP}}$
2. $B \leftarrow TTP : \left\{ \{Confirmation\}_{SK_{TTP}} \right\}_{PK_B}$

Fig. 3. Auction offer subprotocol

This subprotocol works as follows and is described in Figure 3. Firstly, bidders send to *TTP* digitally signed by SK_B and encrypted by PK_{TTP} the following information: Nb_B, Nb_{Au}, NR_B, T_{OF_B}. The offer OF_B is also digitally signed by SK_B and encrypted by the public key of a given auction PK_{Au}. Then these messages are sent to *TTP*. If sent data are correct, then the *TTP* sends the confirmation of the offer notification. Finally, the *Confirmation* is digitally signed by SK_{TTP} and encrypted by the public key PK_B of a given bidder.

3.4. The offer choice subprotocol

The last subprotocol is executed after elapsing the time designed for making offers.

1. $TTP \rightarrow B_i : \left\{ \left\{ SK_{P_{(B_i)}} \right\}_{SK_{TTP}} \right\}_{PK_{B_i}}$
2. $A \rightarrow TTP : \left\{ \left\{ SK_{P_{(A)}} \right\}_{SK_A} \right\}_{PK_{TTP}}$
2. $TTP \leftarrow B_i : \left\{ \left\{ SK_{P_{(B_i)}} \right\}_{SK_{B_i}} \right\}_{PK_{TTP}}$
3. $A \leftarrow TTP : \left\{ \left\{ \left\{ OF_{B_i} \right\}_{SK_{B_i}} \right\}_{SK_{TTP}} \right\}_{PK_A}$
4. $A \rightarrow TTP : \left\{ \left\{ NR_{B(win)}, NR_A, N_{B_i}, N_A, Nb_{Au} \right\}_{SK_A} \right\}_{PK_{TTP}}$
5. $TTP \rightarrow WWW : NR_{B(win)}, N_{B_i}$

Fig. 4. Offer choice subprotocol

This subprotocol, described in Figure 4, works as follows. Knowing the number N of bidders who sent their offers, the *TTP* divides earlier split parts of main secret of auction into N smaller parts $SK_{P(B_i)}$. He uses again the safe threshold scheme dividing the secret into N part which the following profile (2,N), i.e. two persons are sufficient to reconstruct the secret divided into N parts. Created parts $SK_{(B_i)}$ are digitally signed by SK_{TTP}, encrypted by SK_{B_i} and sent to the appropriate bidder B_i. In the next step, the auctioneer A and the bidders B_i send digitally signed and encrypted, their parts of secret to *TTP*. After that *TTP* joins the received parts of the secret into the main secret of the auction SK_{Au}. Having the whole secret of given auction the *TTP* can decrypt all sent offers OF_{B_i} in the previous protocol. After the *TTP* sends to the auctioneer A, which announced the auction, all offers OF_{B_i} digitally signed by the bidders. All offers are earlier decrypted by SK_{TTP} and encrypted with PK_A. After that the auctioneer A has received the offers, he chooses the best offer and sends the result to *TTP* in order to notify the winner. The results include the following information: the registration number $NR_{B(win)}$ of the bidder who has won the auction, the auctioneer registration number NR_A, the individual numbers N_{B_i} of the bidders who sent the offer, the auctioneer individual number N_A and the number of auction Nb_{Au}. These exchanged information is digitally signed by SK_A and encrypted by PK_{TTP}. When that the *TTP* has received this information, he publishes the individual number of the bidder who won the auction $SKNR_{B(win)}$ and the numbers of bidders N_{B_i}.

4. The attack

In this section, we describe the flaw found on the first phase which is a kind of "man in the middle" attack. It is based on the fact that the messages sent by the agent C and the *TTP* in the first phase are not authenticated and on the fact that the answer given by the server does not contain any information about the identity of the agent C.

Notation: The intruder is denoted by I, and $I(C)$ means that the intruder is impersonating the agent C.

$$1.1\ C \rightarrow\ I(TTP) \qquad : \left\{\{D_C\}_{SK_C}\right\}_{PK_{TTP}}$$

$$2.1\qquad I(C) \rightarrow TTP \quad : \left\{\{D_I\}_{SK_I}\right\}_{PK_{TTP}}$$

$$2.2\qquad I(C) \leftarrow TTP \quad : \left\{\{NR_I\}_{SK_{TTP}}, T_{NR_I}, \{SK_I, PK_I\}_{SK_{TTP}}\right\}_{PK_I}$$

$$1.2\ C \leftarrow\ I(C) \qquad : \left\{\{NR_I\}_{SK_{TTP}}, T_{NR_I}, \{SK_I, PK_I\}_{SK_{TTP}}\right\}_{PK_C}$$

Fig. 5. Description of the attack on the first phase in Figure 1

Description of the attack: This attack requires two sessions of the first phase of the protocol and is described in Figure 5. The agent C starts the step 1.1 of a session of this protocol and sends the message $\{\{D(C)\}_{SK_C}\}_{PK_{TTP}}$ on the network. The intruder controls the network, in consequence he blocks the first message sent by C. In parallel the attacker plays the first step of a second session 2.1 with the server *TTP*. The intruder generates a new document D_I and sends it to the *TTP* instead of the message generated by C in the first step of the first session. The server *TTP* generates automaticaly a registration number, a timestamp, a set of keys encrypted with his private and sends all this information ecncrypted with the public key of the intruder in step 2.2. The intruder uncrypts this message with his private key and learns the new sets of keys generated by the server. He is now able to forge the message 1.2 of the first session to convince the agent C that everything is normal.

This attack on the secrecy of the new keys was find by our modeling of this first phase of the old protocol in the OFMC tool. This attack implies that the intruder falsifies the document produced by the agent C and the most important that the intruder can understand all the encrypted information exchanged between the server and the agent C, during the next steps of the protocol.

5. The solution

In this section, we correct the first phase of the previous protocol and we also give a new version of the other phases of the protocol.We have checked all these phases of this new version for the secrecy property with OFMC.

Notice that the new protocol also optimizes the old protocol because we decrease the complexity of different subprotocols. First of all, the changes are connected with the fulfillment of confidentiality of bids. In the old protocol it was gained by dividing the secret key of the given auction SK_P into the three parts. As a result, the offer could be decrypted only then if the parts of the secret key are joined together. In the proposed revision of the protocol the message containing the offer are encrypted by the public key PK_{TTP} and after that the offer OF_B is encrypted by the public key of the auctioneer PK_A. Due to the use the public key of TTP nobody who will intercept the message can decrypt it. Moreover, the TTP can not learn the offer because it is encrypted by the key of the auctioneer. Secondly, we decrease the computation operation in the new version of protocol because we do not use the needless digital signatures.

5.1. First phase

In Figure 6 the new description of the first phase is given. The client sends his identity and a fresh nonce N_C encrypted by the public key PK_{TTP}. The *TTP* answers sending the following message: the nonce of the client N_C, a new and

fresh register number NR_C, a timestamps T_{NR_C} to control the validity of the registration and his identity. That message is encrypted by the public key of the client. Finally the client confirms his registration by sending back to the server his registration number and a form D_C encrypted by the public key PK_{TTP}.

1. $C \rightarrow TTP : \{C, N_C\}_{PK_{TTP}}$
2. $C \leftarrow TTP : \{N_C, NR_C, T_{NR_C}, TTP\}_{PK_C}$
3. $C \rightarrow TTP : \{D_C, NR_C\}_{PK_{TTP}}$

Fig. 6. New certification subprotocol

After this first phase the client has a new registration number NR_C and a time stamp T_{NR_C} according to the document D_C he sends. The third exchange is necessary to correct the previous version of the protocol. It assures the server that the client received the registration number and in consequence, denies that he is talking with somebody who wants to impersonate the client. Notice that the identity of *TTP* in second message is crucial to avoid a kind of man in the middle attack.

5.2. Second phase

This phase of the protocol is composed of 2 parts, described in Figures 7 and 8:

1. The auctioneer proposes his offer and the server publishes it, Figure 7.
2. The bidders make an offer, Figure 8.

1. $A \rightarrow TTP : \{NR_A, AP_A, A, N_{AP_A}\}_{PK_{TTP}}$
2. $A \leftarrow TTP : \{N_{AP_A}, Nb_{Au}, TTP\}_{PK_A}$
3. $A \rightarrow TTP : \{Nb_{Au}, T_{Nb_{Au}}\}_{PK_{TTP}}$
4. $TTP \rightarrow WWW : Nb_{Au}, T_{Au(open)}, T_{Au(close)}, AP_A, PK_A$

Fig. 7. New subprotocol for the e-auction notification by the vendor

Auctioneer stage: In this phase the auctioneer A submits his auction to the server. In the first message the sends his registration number NR_A obtained in the previous phase, the proposal of the auction AP_A, his identity and a new nonce N_{AP_A}. The message is encrypted by the key of *TTP*. The server checks the validity of the received registration number and if it is positive the *TTP*

generates a fresh auction number Nb_{Au}. After that he sends the auction number Nb_{Au}, the nonce N_{AP_A} and his name. The auctioneer confirms the reception of the message by sending the server a new number of auction and a time stamp $T_{Nb_{Au}}$. Then the server publishes on a web site, the open time for a given auction $T_{Au(open)}$ and the close time $T_{Au(close)}$ for that auction. During that time the received auction properties will be taken into account. Except for this information, *TTP* publishes the description of the auction AP_A and the public key PK_A of the auctioneer.

5. $A \rightarrow \quad TTP \quad : \left\{ NR_B, Nb_{Au}, B, N_{OF_b}, \{OF_B\}_{PK_A}, h(OF_B) \right\}_{PK_{TTP}}$

6. $A \leftarrow \quad TTP \quad : \left\{ Nb_{OF_b}, N_{OF_b}, TTP \right\}_{PK_B}$

7. $A \rightarrow \quad TTP \quad : \left\{ Nb_{OF_b}, T_{Nb_{OF_B}} \right\}_{PK_{TTP}}$

Fig. 8. New subprotocol for the notification of the offers

Bidders stage: The next phase, described in Figure 8 based on the collection of all the propositions done by bidders during the time interval allowed for the auction.

The bidder B makes an offer OF_B and generates a new nonce N_{OF_B}. He encrypts with the public key of the server and sends to *TTP*: his registration NR_B, obtained during the first phase, the auction number Nb_{Au}, his name, the new nonce N_{OF_B}, his offer encrypted by the public key of the vendor (avoiding that the server read it), and the hash of his offer (giving the possibility to the server to find the bidder in the last phase). The server answers giving a new registration number Nb_{OF_B}, the nonce received N_{OF_B} and his identity. The bidder confirms to the server by sending a time stamp $T_{Nb_{OF_B}}$ and the number Nb_{OF_B}.

5.3. Last phase

Once the auction is closed, the auctioneer chooses the winner and the server publishes his identity during the subprotocol described in Figure 9. First the auctioneer sends to the *TTP*: the registration number of the auction Nb_{Au}, all the offers OF_{B_i} encrypted by the bidders with the public key PK_A of the auctioneer, his identity and a new nonce N_{TTP}. The auctioneer makes the choice and communicates it to the *TTP* by sending the hash of the winner offer, the auction number, the nonce and his identity. Finally, using the hash of the offers obtained in the previous phase, the *TTP* finds and publishes the number of the winner $N_{OF_{B(win)}}$, all bidder's individual numbers $N_{OF_{B(i)}}$ and the number of the auction

Nb_{Au}. This step assures that all the propositions made by all the bidders were transmitted to the auctioneer and taken into account. Hence everybody can check anonymously who is the winner of the e-auction.

1. $A \leftarrow \quad TTP \quad : \left\{ Nb_{Au}, \left\{ OF_{B_i} \right\}_{PK_A}, TTP, N_{TTP} \right\}_{PK_A}$
2. $A \rightarrow \quad TTP \quad : \left\{ h\left(OF_{B(win)} \right), Nb_{Au}, N_{TTP}, A \right\}_{PK_{TTP}}$
3. $TTP \quad \rightarrow WWW \quad : N_{OF_{B(win)}}, N_{OF_{B_i}}, Nb_{Au}$

Fig. 9. New offer choice subprotocol

5.4. Security analysis

We identify properties that this protocol has to verify and give some explications how the new e-auction protocol satisfies them.

Secrecy: The integrity and confidentiality of transaction must be protected. Except for the information published on the *TTP* website *WWW*, all message transactions are protected by the public encryption system to ensure the integrity and confidentiality of messages. Moreover, we check the secrecy of all data using the formal verification OFMC tool. Moreover by construction of the protocol the OFMC does not have access to the bids because they are directly encrypted by the public key of the auctioneer.

Authentication: Only registered persons can make or announce an e-auction. The certification subprotocol is responsible for main verification of the auction participants. In other subprotocols the *TTP* as the third trustworthy part checks the required documents and verifies that the participants in auction have a valid registration number. This registration number is a fresh number generated by *TTP* during the first phase.

Non repudiation: The winner and the bidders cannot deny the contents of their offers. The auction and the bids are firstly transmitted to the trustworthy third part. The *TTP* stores the received data and information about the identity of the auctioneer or bidder. In this way the *TTP* can prove that he received some information from an agent, exhibiting for instance the hash of the auction done or the proposition submitted by the participant.

Anonymity: The auctioneer can not know the true identity of the bidder. The name of the winning bidder is not public because only the numbers associated to the agents are published. When the bidding time expires, the auctioneer receives the offers sent. Those offers do not include bidders' identity and are stored only by the *TTP*. They are encrypted by the public key PK_A which denies that *TTP* knows the content of the auctions made by the bidders. When the auctioneer chooses the winning offer only the bidder's individual number is published. This

number shows the winning bidder identity and assures the anonymity of the true winner and the bidders.

Public verification: Everybody can check if an offer has to be taken into account in the e-auction. When the e-auction is finished, all bidder's individual numbers taking part in the auction are published. Every participant can check if his number is on the list which is equivalent with the fact that the offer was taken into consideration by the auctioneer.

6. Conclusion

The security of electronic auction is a crucial issue on electronic market. Security requirements are defined by security properties such as secrecy, authentication, anonymity. These features are guaranteed by protocols including cryptographic primitives and other security mechanisms. Designing cryptographic protocol is a complex process and assures that the security properties verification of the protocol is not an easy task. In the article we have presented the authentication breach and secret retrieval attack on the cryptographic protocol with extended requirements [1]. The attack was discovered by the formal verification tool OFMC. We also propose the revision of e-auction protocol which corrects founded attack and optimizes the complexity of old protocol. We have also checked with OFMC the secrecy of the data exchanged for the new protocol. The next step will be to develop some formal methods for verifying automatically all other properties that an e-auction protocol has to assure.

Acknowledgments

This work was partially supported by the DGA under grant number 06 60 019 00 470 75 01, and by the Zurich Information Security Center. It represents the views of the authors.

References

[1] Księżopolski B., Kotulski Z., *Cryptographic protocol for electronic auctions with extended requirements*. Annales UMCS Informatica, 2 (2004) 391.
[2] David E., Azoulay-Schwartz R., Kraus S., *Tan english auction protocol for multi-attribute items*. In Workshop on Agent Mediated Electronic Commerce on Agent-Mediated Electronic Commerce IV, Designing Mechanisms and Systems, Springer-Verlag, 2531 (2002) 52.
[3] Juels A. Szydlo M., *A two-server, sealed – poverties auction protocol*. In roceedings of the 6th Annual Conference he Financial Cryptography (FC), Springer-Verlag, 2357 (2002) 72.
[4] Vickrey W., *Counter speculation, auctions, and competitive sealed tenders*. Journal of Finance, 16(1) (1961) 8.
[5] Wolfstetter E., Auctions: An introduction. Journal of Economic Surveys, (1996) 367.
[6] Gritzalis S., Spinellis D., *Cryptographic protocols over open distributed systems: a taxonomy of flaws and related protocol analysis tools*. In In Proceedings of the 16th International Conference on Computer Safty, Reliability and Secutity, (1997) 123.

[7] Syverson P., A taxonomy of replay attacks. In In Proceedings of Computer Security Foundations Workshop VII, (1994) 187.
[8] Xu C., Kedem G., Gong F., *Categorizing attacks on cryptographic protocols based on intruders*. In In Proceedings of the FMCS'2000 Foundational Methods in Computer Science Conference, (2000).
[9] Księżopolski B., Kotulski Z., *Adaptable security mechanism for dynamic environments*. Computers & Security, (2007) 367, to appear.
[10] Lowe G., *Casper: A compiler for the analysis of security protocols*. In Proc. 10th Computer Security Foundations Workshop (CSFW'97), Rockport, Massachusetts, USA, IEEE Comp. Soc. Press, (1997) 18.
[11] Meadows C., *Language generation and verification in the NRL protocol analyzer*. In Proc. 9th Computer Seurity Foundation Workshop (CSFW'96), Kenmare, Ireland, IEEE Comp. Soc. Press, (1996) 48.
[12] Mitchell J.C, Mitchell M., Stern U., *Automated analysis of cryptographic protocols using murphi*. In IEEE Symposium on Security and Privacy, May (1997).
[13] Blanchet B., *An efficient cryptographic protocol verifier based on prolog rules*. In Proc. 14th Computer Security Foundations Workshop (CSFW'01), Cape Breton, Canada, 2001. IEEE Comp. Soc. Press., (2001) 82.
[14] Bozga L., Lakhnech Y., Perin M., *HERMES: An Automatic Tool for Verification of Secrecy in Security Protocols*. In Computer Aided Verification, (2003).
[15] Corin R., Etalle S., Saptawijaya A., *A logic for constraint-based security protocol analysis*. In IEEE Symposium on Security and Privacy, (2006).
[16] Armando A., Basin D., Boichut Y., Chevalier Y., Compagna L., Cuellar J., Drielsma P.H., Heám P.C., Kouchnarenko O., Mantovani J., Mödersheim S., Oheimb D., Rusinowitch M., Santiago J., Turuani M., Viganò L., Vigneron L., *The avispa tool for the automated validation of internet security protocols and applications*. In Proceedings of CAV'2005, LNCS 3576, Springer-Verlag, (2005) 281.
[17] Cremers C., *Scyther – Semantics and Verification of Security Protocols*. PhD thesis, Eindhoven University of Technology, (2006).
[18] Basin D., Mödersheim S., Viganò L., *Ofmc: A symbolic model checker for security protocols*. International Journal of Information Security, , 4(3) (2005) 181.
[19] Kulesza K., Kotulski Z., *On automatic secret generation and sharing for karin-greene-hellman scheme*. In J.Soldek and L. Drobiazgiewicz, editors, In Proceedings of Artificial Intelligence and Security in Computer Systems, (2003) 281.