

SITA: Protecting Internet Trade Agents from Malicious Hosts

Mark Perry & Qin Zhang

Mark PERRY (Assistant Professor, Department of Computer Science
University of Western Ontario, London, Ontario N6A 5B7
email: mperry@uwo.ca Phone: 519 661 2111 x86644 Fax:519 661-3515
Web: <http://www.csd.uwo.ca/markp/>) (Corresponding author)
Qin ZHANG (Department of Computer Science, University of Western Ontario,
London, Ontario N6A 5B7)

Abstract: The role of agents and their potential in the electronic marketplace has been discussed widely, but the issue of mobile agent vulnerability to attack, particularly from malicious hosts, needs further development. This paper describes our Secure Internet Trade Agents (SITA) framework that allows for multiple ‘window shopping’ agents to retrieve results, whilst providing anonymity for the user, and providing a manageable key structure.

1 The Secure Internet Trade Agent (SITA) Framework

The SITA model we propose is intended to offer better security for trading with Mobile Agents on the internet, whilst at the same time providing a level of anonymity for purchasers and ‘window shoppers’. It relies on a master agent running on a trusted host that dispatches a series of slave agents to carry out the designated tasks.

One advantage offered by mobile agents is support for concurrent job processing. Thus, a task can be separated into several sub-tasks that can be delegated to several ‘slave’ agents, each of which can execute the task in parallel. We use a layered approach to agent initiation, with one superior agent taking control of the task and dispatching child agents, to give improved security. Since the slave agent is sent to one specific shop server only, the control flow of the code is eliminated (i.e., no comparison is done on that server) and agent itinerary modification is avoided. This allows for confidentiality of the slave agent to be achieved by partial encryption of the agent’s components — namely the agent data. Using this mechanism, an agent protects data that must be used at a particular site by encrypting that data with the site’s public-key. In this way, the data is accessible only when the agent reaches the intended execution environment.

We divide the process of inquiring and purchasing in the electronic marketplace into seven stages — ITA (*Internet Trade Agent*) initialization, ITA migration, directory search, product information inquiry, negotiation, evaluation, and purchase and delivery. Fig. 1 shows a simple architecture of an electronic market with secure mobile agents that also makes provision for anonymity.

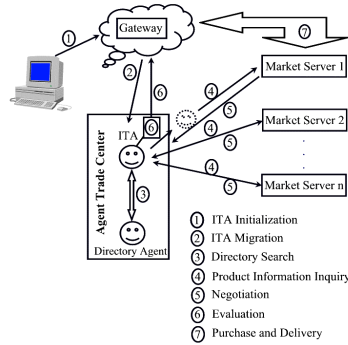


Fig. 1. Secure trade agents in the electronic market

- 1) The user creates an ITA and specifies the name of an item and purchase conditions he/she wants to purchase and delegates this task to an ITA.
- 2) The user sends the instructed agent to an Agent Trade Centre (ATC), which separates the agent from its home address.
- 3) The trade agent queries a directory agent on the ATC and receives a list of destinations it should visit (for example, ‘ask for all addresses of servers that provide airline tickets’).
- 4) The trade agent sits on the ATC as a static agent, dispatches one child mobile agent to each destination server using concurrent parallel scheme.
- 5) The child agent migrates to a market server where it negotiates with the market server and collects offer and reports to the parent agent.
- 6) The trade agent is responsible for evaluation of the collected offers. It can, after it has finished its task, send a message (by email or mobile phone call or pager) back to its user, giving evaluated result. Alternatively, it queries the database of the ATC for a home address and dispatches a result back to the user with the best offer.
- 7) The user reviews the offer found by the ITA. If the offer is reasonable, he/she contacts the best-offer server, does the transaction under the terms of the specific signed offer. Eventually, the user receives the purchased goods

By using this architecture we can make sure that the market servers have no chance of getting any information about the user or about other servers that have serviced such requests. In the case of traditional pseudonyms a trusted third party signs the pseudonyms and thus ensures that in case of need it can identify the user. In our architecture the ATC could do this job, because it is able to identify the user, register the user and ITA together with their home addresses. The ATC can digitally sign the mobile agent and thereby guarantee the trustworthiness of the agents. On the other hand, the agent is ensured and guaranteed by the user who also provides the ATC with her certified personalities (e.g., digital certificates).

For the security of the mobile agents in this system, the ATC can also take the place of the trusted server in one of the trust approaches. The purchasing stage can then be executed over the net between the ATC and the appropriate market server by using secure electronic payment system, such as *Secure Electronic Transaction* (SET) [1].

2 The framework

When we consider the security requirements in terms of protecting agents from malicious hosts, different needs exist during the different stages of the electronic transaction. In each stage of the above-mentioned activities, a sequence of messages is exchanged between two entities, that is, each stage is a communication session between two parties. In particular, we need security and accountability for each of the sessions.

Throughout the following discussion, we denote that K is a secret key in a symmetric cryptograph session, K^+ and K^- are public and private key in an asymmetric cryptograph. $K_A(X)$ is the encryption of a message X using the key K that is generated by principal A . $Sig_A(X)$ denotes the digital signature of principal A on object X . $Cert(A)$ denotes the digital certificate of principal A . N_A represents a nonce (i.e. randomly generated integer) generated by the principal A . $K_A^+(X)$ denotes encrypting the object X with principal A 's public key, while $K_A^-(X)$ is encrypting the object X with principal A 's private key. Finally, $h(X)$ means to apply a hash function to X , create a digest of object X .

2.1 ITA (Internet Trade Agent) Initialization

An electronic transaction starts with a user, say Betty (B). Fig. 2 illustrates an initialization by B of an Internet Trade Agent (I) whose unique identifier (ID_I) is created by a pseudorandom generator and then B starts the process of requisitioning competitive purchase contracts. The agent I obtains its own public key (K_I^+) and private key (K_I^-) and is certified by B – thus we have a certificate hierarchy system with the agent I's certificate $Cert(I)$ at the lowest level. B authenticates the agent I as her representative by providing her certificate and the identity ID_B , denoted as $\{Cert(B), ID_B\}$, and specifies her service request. Agent I may learn from B's previous behavior, guide her and make suggestions. After agent I and B exchange messages interactively, they reach the final shopping requirements (SR). Before agent I migrates, it generates a random secret key K_I , uses K_I encrypts the SR and current time stamp T , denoted as $\{K_I(SR, T)\}$. The secret key was encrypted by the public key of ATC, denoted as $K_A^+(K_I)$. The whole message carried by the ITC would look like this: $\{Cert(B), Cert(I), ID_B, ID_I, Sig_I(SR, T), K_A^+(K_I), K_I(SR, T)\}$

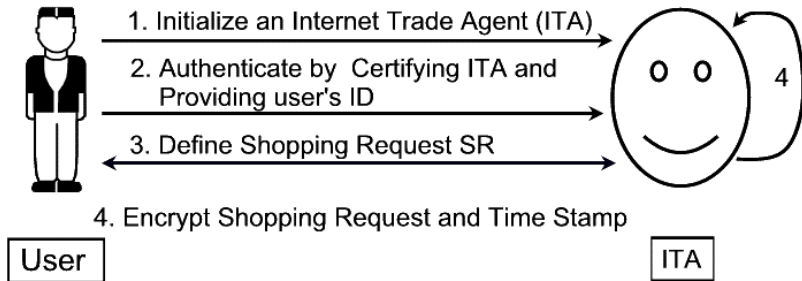


Fig. 2. ITA initialization

2.2 ITA Migration

The instructed ITA migrates to the gateway of the client organization and tries to contact the Agent Trade Centre (ATC). After successful authentication of the ITA's host and the ATC, ITA migrates to ATC carrying the encrypted shopping requirements. Fig. 3 shows the procedures in this stage. At this point, the user can disconnect from its client computer. The trade agent will continue the request, without bothering the buyer again until the delivery stage.

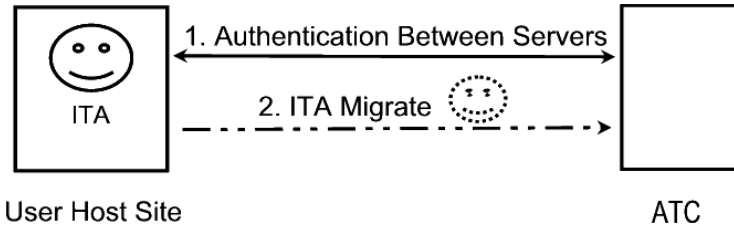


Fig. 3. Migration to Agent Trade Centre

2.3 Directory Search

Fig. 4 illustrates the hidden home address and directory inquiry. When the ITA arrives at the ATC, the ATC first checks if $Cert(B)$ and $Cert(I)$ has been issued by a trusted certification authority. If they are valid and not in the certificate revocation list, the ATC starts to decrypt the encrypted message and check the integrity of the message. The ATC retrieves the secret key by using its private key: $K_A^-(K_A^+(K_I)) \Rightarrow K_I$, then uses this key K_I to retrieve the shopping requirement SR and time stamp T . Also, the ATC checks if the time stamp T is valid. The ATC retrieves the incoming ITA's public key (K_I^+) from its key management file, computes and compares $K_I^+(Sig_I(SR, T))$ and $h(SR, T)$ to see message integrity (if these two values are the same, the

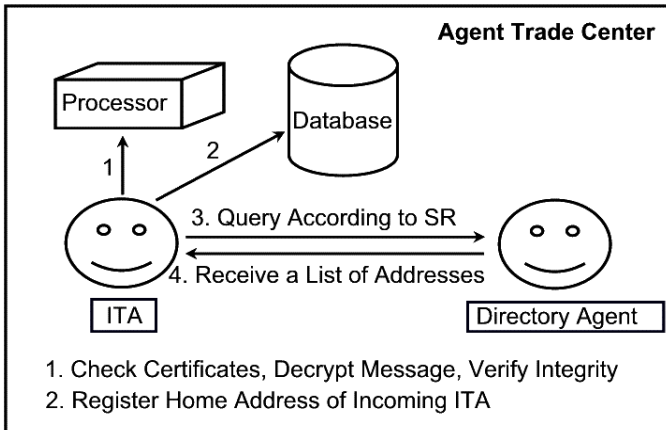


Fig. 4. Directory search

message has not been modified, otherwise the integrity check fails.). The ATC refuses the delegation of the user when integrity check fails, otherwise it registers user B if she hasn't registered before. Only when all answers are certain, the ATC stores the home address of the ITA in a database associated with the identity of user and ITA. The user's information (e.g., identity and address) is removed from the ITA at the same time. According to the decoded SR, the ITA queries the directory agent in the ATC which keeps information about other web sites and acts as an intermediary broker agent that helps an agent to find business web sites that possess certain required information. The directory agent replies a list of n shop server addresses. Because this communication is executed inside the ATC, and the ATC is a tamper-resistant trust server, we do not use any security technique in this query-response stage.

2.4 Product Information Inquiry

In this stage, the ATC signs the SR and a new time stamp T by using its private key: $Sig_A(SR, T)$. Then it generates a random secret key K_A and encrypts the SR and T : $K_A(SR, T)$, encrypts K_A by using the public key of each destination server: $K_{host}^+(K_A)$. Finally, as shown in Fig. 5, the ITA dispatches one child ITA for each destination server using a concurrent parallel scheme — by employing more than one agent simultaneously in the application. Each child agent carries the certificate of ATC ($Cert(A)$), the identity of the ATC (ID_A), the encrypted message and key, and the digital signature: $\{Cert(A), ID_A, Sig_A(SR, T), K_{host}^+(K_A), K_A(SR, T)\}$. The contacting shop servers in the electronic market only know that they are communicating with the ATC, and have no knowledge of the actual user.

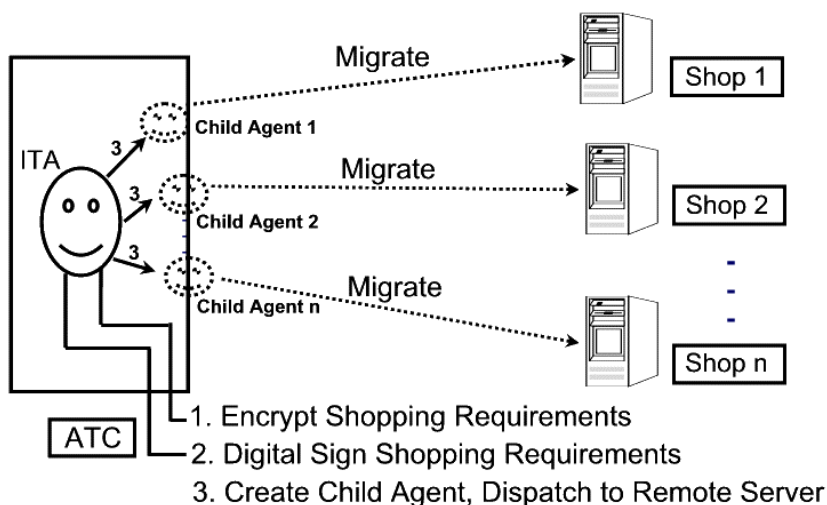


Fig. 5. Product information inquiry

2.5 Negotiation

For the security of the shop server, the server requires adequate authentication proof, such as an authenticated legitimate and traceable signature of the buyer (the ATC in this case), before accepting further interaction with an agent. Otherwise, the transaction will be denied. A server prevents attacks by denying access to any mobile agent that does not have adequate authentication proof. Additionally, the server of the shop checks the code and data of an agent using anti-virus software before it provides the mobile agent with the required execution environment. Fig. 6 depicts the negotiation stage.

So the shop S_i ($i=1,2,\dots, m, m \leq n$) requires authentication proof from the child agent C_i before accepting its execution request. The child agent C_i gives the shop S_i such proof by showing ATC's certificate and ATC's digital signature. While receiving the message $\{Cert(A), ID_A, Sig_A(SR, T), K_{host}^+(K_A), K_A(SR, T)\}$, the electronic shop S_i first checks the validity of $Cert(A)$, then decrypts the secret key by using its own private key if the certificate is valid. After retrieving the secret key K_A , it decrypts $K_A(SR, T)$ and get the shopping requirements SR and the time stamp. Thus it can check the validity of the sender's signature by using the sender's public key: $K_A^+(Sig_A(SR, T)) \stackrel{?}{=} h(SR, ID_A)$. If the verification process succeeds, the shop S_i provides the child agent C_i with execution environment. The child agent asks for the specific goods under the decrypted SR . The result of the communication is a purchase offer signed and encrypted by the shop S_i :

$$\{Cert(S_i), ID_{S_i}, Sig_{S_i}(Offer, T_{S_i}), K_A^+(K_{S_i}), K_{S_i}(Offer, T_{S_i}) \}$$

The child agent receives the encrypted offer (with the valid time limitation T_{S_i}), terminates the negotiation, sends back the encrypted offer to parent ITA, and then disposes itself on shop server side.

When a child agent goes to a server that no longer exists, it is able to return back to the ATC and report the failure so that the directory agent can verify and update its database later. If a child agent arrives at one server, whose address has been changed, the child agent is able to send itself to the new address because of its autonomous capability. (We assume that the changed address server retains its origin identity and key pair.)

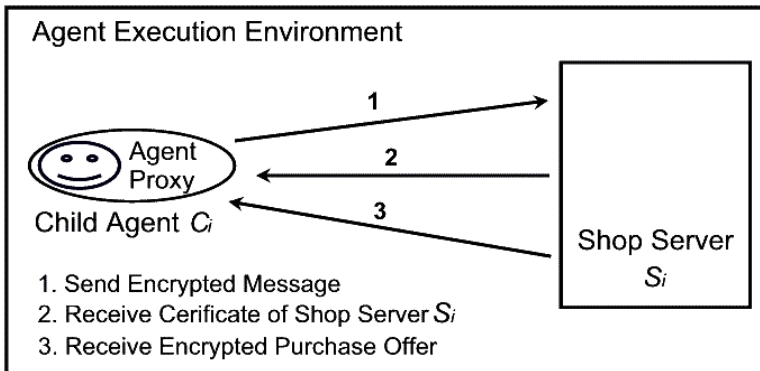


Fig. 6. Negotiation with shop server

2.6 Evaluation

The reply messages, which at this point may be less than n because some child agents may have been killed by malicious shop servers, return to the parent agent site, the ATC. The ITA collects all the information, decrypts and verifies shop's offers. If an offer is not signed or has integrity error, the ITA rejects it and continues the evaluation of those remaining. Eventually, the ITA computes the best offer and queries the ATC database for the user's host address and dispatches it back to the user with the best offer. For security reason, the best offer $bestOffer$ and time stamp T_A will be signed again by ATC and ITA carries back together with the ATC's certificate and encrypted report: $\{Cert(A), ID_A, Sig_A(bestOffer, T_A), K_B^+(K_I), K_I(bestOffer, T_A)\}$ Alternatively, the ITA notifies its user about the accomplished task and the final result by sending a email or by mobile phone call or pager.

2.7 Purchase and Delivery

When Betty connects to the Internet, the ITA reports to Betty the best offer it found. Betty will authenticate the ITA first, then review the offer. If Betty thinks the price is reasonable and is willing to purchase, she contacts the shop server that signed the optimal purchase offer with her acceptance. The shop checks its signature on the offer and verifies the valid time period and if everything is as it should be, it cannot repudiate the terms of the accepted offer. Fig. 7 shows the payment procedure.

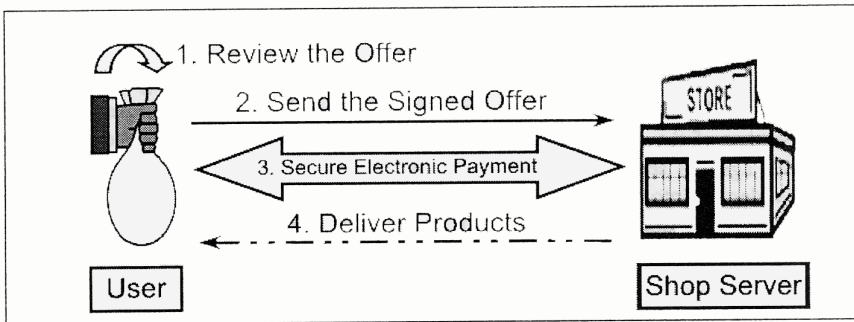


Fig. 7. Payment

Betty uses a secure electronic payment system, such as *Secure Electronic Transaction* (SET), which is accepted by the electronic shop and pays for the requested products. Finally the shop delivers the goods, which may be digital or physical.

3 Security

3.1 Security of ATC against User B and ITA

In SITA, an ITA's certificate $Cert(T)$ is certified by its user B. So it is very important for an ITA to provide its user's valid certificate ($Cert(B)$). All the certificates, the ITA's signature on (SR, T) , and the encrypted (SR, T) authenticate the ITA. An unauthorized ITA or user who intends to enter the ATC is refused if they are not able to show an authorized certificate and correct signature at the same time. Furthermore, the encrypted time stamp makes the replaying attack impossible.

3.2 Security of servers against malicious agents

SITA protects servers against malicious agents mainly by authentication and integrity techniques. In SITA, the shop servers in the electronic market require adequate authentication proof, before accepting further interaction with an agent. A mobile agent must provide the certificate of the ATC ($Cert(A)$), signature on shopping requirements, and time stamp created by the ATC to server if it wants to execute on shop server. The shop server will deny execution of a mobile agent that is not authenticated by the trusted third party, the ATC. When mobile agent returns (or send messages) to the ATC having fulfilled its task, the ATC authenticates it by checking shop server's certificate ($Cert(S_i)$) and their signatures on shopping offer. The user's host has to verify returning ITA on ATC's certificate and ATC's signature on the evaluated best offer. If the authentication succeeds, a server may use anti-viral software checks on the code and data of an agent before it provides the agent with the required execution environment.

All servers will follow an effective access control policy and security policy providing the agent with a limited execution environment and grant access rights only for that environment. If an agent looks suspicious for malicious behavior, the server can suspend the execution of the agent forcing it to migrate back, or even destroy the agent. If the malicious agent is sent by shop servers, the ATC records the hostile behavior with the sender's identity into its revocation list, and will not send a trade agent to that server any longer. If the malicious agent came from the ATC, either the user or the shop server will report to some agent society to verify the reputation of the ATC.

3.3 Security of agents against malicious host

Our SITA framework overcomes most of the malicious host problems. First of all, the proposed ATC is a tamper-resistant trusted third party who provides agent anonymous agent service. The ATC will not engage in any hostile activity against incoming agents. The trust of the ATC is based on using tamper-free trust hardware, administrated by a large commercial institution that has high reputation. It is very unlikely that the ATC turns out to be malicious; if so, not only will the ATC lose its business but also the administrating institution may face legal issues.

In SITA, the ITA uses concurrent parallel mechanism to dispatch one child agent to each shop server in the destination list. Because the ATC is a trusted host, the ITA sitting on ATC as a static agent is resistant to attack by a malicious shop host. More security concerns are related with the protection of the child agents, since they are

mobile. Because a host has to modify an agent in order to give the negotiation result, time stamp, its certificate and digital signature to the agent, a dishonest host may try to alter its state and code or scan the agent for its gathered information. A hostile server may also deny a mobile agent the execution environment, or even kill the agent. If the user sends out only one mobile agent to communicate with several servers in one itinerary, that agent may carry with it sensitive information, which could be used by malicious server for many illegal purposes (inspect other servers' information, try to revise the information, etc.), by the electronic shop server.

However in the SITA model, the ITA sends out a mobile agent to each server respectively and it returns with the only information given by that particular server. The code of the child agent contains the only information given to that host, instead of control flow statement (such as, 'do price comparison') in the code. Thus the code is less likely to be modified. In addition, the child agents do not carry secret keys, offers from other shop servers or other sensitive information (such as a credit card number), since the purchase stage is assigned to the user or the ATC. Therefore, the child agent does not have any information that could tempt the shop server to eavesdrop, intercept or alter. We thus don't need any *detection object* carrying with the mobile agent or *trace* logs on the visited server. We don't even need to encrypt the child agent as a whole, because the only one thing that a child agent needs to keep secret is the user's shopping request information. Therefore, we use encryption mechanism to encode shipping request and only the specified server can decoded it. This saves on computation cost and time when compared with a technique that encrypts the whole agent, whilst at the same time eliminating the problem of revealing sensitive information.

A malicious server may deny service to an authenticated child agent with a valid authentication or even terminate the agent. The parent agent can detect hostile behavior against a particular child agent (it knows the identity of the server that each individual child agent visits). The ITA would report the suspicious behavior to the ATC, later the ATC verifies the suspect host and can add the corresponding shop server to a "revocation list" of servers and cease any future transactions with this server if the malicious activity is verified. Moreover, this model can overcome denial of service of some potential hostile shop servers without restarting the whole process. It is unlikely that every shop server that a child agent visits is malicious and will mount denial of service attack on the incoming agent. In particular, we can perhaps assume that in electronic commerce most of the shop servers are set up for doing business and not for the purpose of attacking other agents or servers. Therefore, if only m out of n child agents return to the ATC, the ITA can continue the evaluation of the valid m offers.

3.4 Anonymous service of ATC

In the information gathering stage, the ATC replaces the home address of incoming ITA, and signs it using ATC's private key. The shop servers in the electronic market only know that they are communicating with an ATC. Once provided the offer, an electronic shop cannot refuse to sell the required products under the terms of the purchase offer it issued, because it has previously signed it. The offer can't be replayed because it has no significance in view of the existence of the time stamp.

During the payment and delivery stage, if the user wants anonymous payment, he can authorize the secure ATC purchasing by providing his/her credit card number (e.g. through Secure Socket Layer [6,9]) to the ATC. Receiving the user's authorization and credit card number, the ATC will contact the designated shop server, use secure electronic payment system, say SET. The electronic products (such as game and software application) can be transferred to the user through the ATC. Other physical products can be arranged to be delivered to the location of institution that is operating the ATC and then be transferred to the user.

4 Conclusion

SITA is different from the "single agent" approach [4-6], because it applies the inherent ability of mobile agent parallel processing. Instead of using one agent to visit multiple hosts (multi-hops) in one trip, one static parent agent spawns several child agents and dispatches one child to each designated host. Looking superficially, it seems computationally heavy because we run $n+1$ agents for information gathering instead of one agent. However, these agents are light-weight threads, containing very little code and data. They can be transferred to remote servers very quickly depending on current bandwidth. SITA simplifies security problems, having similar computation cost as one agent and hosts (including visiting shop servers) taken as a whole, or even less. This is because the agent only needs to encrypt the user's shopping request information, instead of the whole agent. Furthermore, the agent doesn't need a *detection object* [7] or the complicated agent structure as suggested by Wang et al in [8], and the visited shop servers do not need to store a *Login Data Base* nor any other *execution trace* [9].

Furthermore, SITA has improved on the Kotzanikolaou et al. [10] approach in four ways:

1. SITA provides anonymous window-shopping to users;
2. The Kotzanikolaou et al. approach requires that the user keeps on-line connection for the stage of shop server issuing permission-tokens to parent agent. SITA can operate off-line as long as the user creates and dispatches the parent agent to the ATC.
3. SITA removes the need for agent permission tokens. In Kotzanikolaou et al. approach, such token is mainly to provide the authentication of mobile agents to shop servers. In SITA, the certificates and digital signature of the trusted ATC provide such authentication.
4. Kotzanikolaou et al. approach uses sole public-key cryptography algorithm. Although the public-key encryption doesn't have key distribution problem, it has the drawback of having higher processing overhead than the secret key. It is about 100-1000 times slower than secret-key encryption. SITA improves on this by using a hybrid encryption scheme (i.e. encrypt message using secret-key cryptography, and then encrypt the simple secret key using public-key cryptography).

In addition, SITA provides a simplified model for improved security in Internet shopping, with the advantage of lessening loads on hosts through the use of the hybrid encryption, small, efficient agents and the utilization of the ATC.

Reference

1. SET, *Secure Electronic Transaction (SET)*. <http://setco.org>
2. Schneier, B., *Applied Cryptography*. Second ed. 1996: John Wiley & Sons, Inc.
3. SSL, Secure Sockets Layer (SSL), <http://www.netscape.com>.
4. Hohl, F., *Time limited blackbox security: protecting mobile agents from malicious hosts*. Mobile Agent Security, Lecture Notes in Computer Science. Vol. 1419. 1998: Springer-Verlag. 92–113.
5. Yee, B.S., A sanctuary for mobile agents. DARPA Workshop on Foundations for Secure Mobile Code, 1996.
<http://www.cs.nps.navy.mil/research/languages/statements/bsy.ps>
6. Sander, T. and C.F. Tschudin, *Protecting mobile agents against malicious hosts*. Mobile Agent Security, Lecture Notes in Computer Science. Vol. 1419. 1998: Springer-Verlag. 44–60.
7. Meadows, C., *Detecting attacks on mobile agents*. Workshop on Foundations for Secure Mobile Code, Monterey, California, 1997. p.64-65
8. Wang, X.F., X. Yi, and K.Y. Lan, *Secure information gathering agent for Internet Trading*. 11th Australian Joint Conference on Artificial intelligence (AI'98), 1998. Springer-Verlag Lecture Notes in Artificial Intelligence, Vol. 1544, edited by Chengqi Zhang and Dickson Lukose, Springer-Verlag Publishers, 1998. p.183-194.
9. Vigna, G., Protecting mobile agents through tracing. Third Workshop on Mobile Object Systems, June 1997.
<http://www.cs.ucsb.edu/~vigna/listpub.html>
10. Kotzanikolaou, P., G. Katsirelos, and V. Chrissilopoulos, *Mobile agents for secure electronic transactions*. Recent Advances in Signal Processing and Communications, 1999: 363–368.