An Agile Protocol for E-Commerce

Yong Wang [1, 2], Qianxing Xiong [2]

[1] Management School, Wuhan University of Science and Technology, Wuhan 430081, China
[2] School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430063, China
Witsbank@sohu.com

ABSTRACT

With merchant information increasing rapidly, customers have to spend a lot of time and energies to search for goods they need, as a result, they maybe find nothing. If merchant can respond customer's requirements agilely when the event defined by customer appears, blindness of customer can be avoided, merchant wins business opportunities too. Protocol is the technology fundament for e-commerce. An idea for agile mechanism is introduced, this paper designs an e-commerce protocol with agility, and it discusses its security. This protocol improves e-commerce efficiency and solves activity problems.

Keywords: e-commerce, protocol, agility, E-P executing model

1. INTRODUCTION

Higher efficiency and lower cost of e-commerce makes more and more companies take part in this business mode. Without e-commerce, enterprise can not participate in global challenge on business, it not only changes enterprise mode of product, operation and management, but also influences economic operating of society entirely[1]. However, development of e-commerce is influenced by informative fundament, security payment, credit system, delivery link and law environment. When customers roam in vast Internet sea to search goods they need, they have to spend a lot of time and energy, as a result, they maybe find nothing. Much overflow information makes e-commerce become lower efficiency and not meet rapidly enterprise application. How to reverse the positive situation to make e-commerce serve enterprise agilely?

Protocol is support technology to guarantee e-commerce security, transaction and application. Many protocols can be concluded three varieties; they are security protocol, transaction protocol and special application protocol. Summarizing these protocols, each does as he likes. Netbill protocol only is used for digital goods sales. SET has more than 3000 lines grammar definition, 28 steps transaction, each step has 6 times RSA. Digicash pays by means of digital cash, because of its uniqueness, it has the same anonymous and non-traceable features as digital cash. The core of Digicash is blind signature[2]. TLS (Transport Layer Security), IPSEC and PTTP, variation of SSL, are transmitting control protocols over TCP/IP[3]. Either these protocols are too easy and only can meet some special application, or they are too complicate and lead to lower efficiency and a large expense. They are considered insufficiently in customer requirement, perfection and intelligence, such as lacking credit, negotiation, active service and multi-medium payment.

Protocol is not designed perfectly so that customers undergo much suffering of overflow information. This paper researches on aspect of business agility; it introduces agile mechanism and makes protocol have agile property.

2. AGILE E-COMMERCE SYSTEM

2.1 Conception on Agility

Let's see an instance for e-commerce, when a customer accesses a merchant web site to buy something, being out of stock, it can not meets customer this time, when the customer comes here next time, it is maybe short yet, but between the two times, perhaps the commodities are in there and sold out again. In this case, it is possible that customer loses the chance not to obtain goods he needs in the end, as a result, customer waste a lot of time, merchant loses business chance too. If merchant can inform customer actively in replenishing stock, this problem can be solved better. This case demands e-commerce system have an ability to serve customer actively, the system is agile. Agility means it executes transaction automatically to finish business activity in coming definition event.

2.2 E-P Executing Model

Following gives E-P executing model for agility mechanism.

Event Definition: Events causing agile service consist of different kind of events.

$$E = \{p_t(o) \in (E_t, E_e, E_c, E_u) | o \in D_T \wedge p \in p(o)\}$$

Where E stands for event set relating to transaction t, $p_t(o)$ stands for the event which transaction t executes operating p on object o, D_t is data set of t, $p(o)$ is operating set of o, E_t, E_e, E_c, E_u respectively stands for time event, external event, complex event and user-defined event.

Transaction Definition：Transaction set deposits a series trigged transaction, which corresponded with different event.

$T=\{p(o)|\exists o \exists p(o) \in (T_s, T_c, T_m, \ldots)\}$

Where T_s, T_c and T_m respectively represent system transaction, customer transaction and merchant transaction.

Trigger Definition：Trigger is a dual series of condition and trigged transaction.

$R=\{Cond, T\}$

Where *Cond* and *T* respectively stand for condition and trigged transaction.

Agility Mechanism：Agility mechanism is a mate of event and process.

$A=\{<e, p>|e \in E, p \in P，P=<CD,R,TMG>\}$

This is *E-P* model. *CD* is event detector, process *P* is a triple set, and *TMG* is transaction management program.

How to create an agile e-commerce system? Using *E-A* model on active database by means of document [4], when merchant system can not meet the customer's requirement, it allows customer define events and monitor it in real time, once the defined event appears, condition detector sends message to trigger management program, it evaluates the condition and requests *TMG* creating a transaction and executes it, so that it can realize the agile service function of merchant system. Figure 1 is the e-commerce system with agile ability.

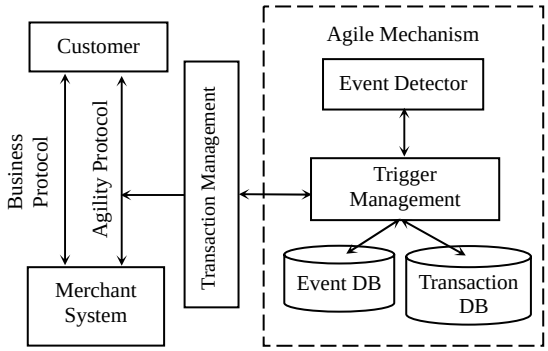

Figure 1 E-commerce system with agile ability

3. PROTOCOL DESIGN

3.1 Symbol

E：Entity. $E \in \{C, M, T, B\}$，*C*, *M*, *T* and *B* respectively stand for customer, merchant, TTP(Trusted Third Part) and payment gateway.

A|*B*：It is entity object. $A|B \in \{C, M, P\}$， *C*, *M* and *P* respectively stand for customer, merchant and payment gateway.

Req：It is goods information that is searched.

A?B:(*X*)：It is that entity *A* ask entity *B* about information *X*.

X+*Y*：It is link of message *X* and message *Y*.

X, *Y*：It is series of message *X* and message *Y*.

A→*B*:(*X*)：It is that entity *A* send message *X* to entity *B*.

Prg.Opt[(Obj)]：It is that procedure *Prg* executes operation *Opt*. *Obj* is operation object. $Prg \in \{CD, R, TMG\}$.

$D_{ASK}(X)$：It is signature to message *X* with secret key *ASK* of entity *A*.

$E_{BPK}(X)$：It is encryption to message *X* with public key *BPK* of entity *B*.

H(*X*)：It is abstract of message *X*, *H* is strongly one-sided Hash function.

OI：It is order information. *OI*=(*GName*, *GType*, *GNumber*, *dod*), *GName*, *GType*, *GNumber* and *dod* respectively stand for goods name, specification, amount and date of delivery.

GDB：It is merchant database.

CertE：It is certificate of entity *E*.

PI：It is payment information. *PI*=<*CAN*, *PW*>

CAN：It is customer account.

PW：It is password of customer account.

MAN：It is merchant account.

receipt：It is merchant receipt.

CID：It is customer identity.

3.2 PROTOCOL DESCRIPTION

It is provided *C*, *M*, *T* and *P* have had verified certificate. *C*, *M* and *B* have registered from *T*, *M* has conserved the copy of certificate of *T*, *T* has conserved the copy of certificate of *B*. *T* acts as certificate, notarization, payment, arbitrate and time postmark in e-commerce protocol[5], *T* plays the role of identity verifying， payment and evidence-protected in this protocol. This protocol consists of two sub-protocol, they are marked as Agility and Business. Generally, it executes Business protocol, Agility protocol is executed when merchant do not meet customer requirements.

Business：

B1：$C \rightarrow M$:($E_{MPK}(D_{CSK}(OI, Cert_C), H(OI))$)，*C* sends order information by way of signature to *M*.

B2：*M* decrypts the message with secret key *MSK*, and decrypts *OI* and $Cert_C$ with secret key *CSK* sent by *C*, and verifies truth of order information and identity of *C*.

B3：$M \rightarrow C$:($E_{CPK}(D_{MSK}(Cert_M, Cert_T, PI)$，*M* submits signature certificates of *M* and *T* and request payment information to *C*, certificate of *T* is preserved by *M* in advance.

B4：*C* decrypts the message with secret key *CSK*, and verifies certificates of *M* and *T* with secret key *MSK* sent by M.

B5：$C \rightarrow M$:(*OP*, *CM*, *CT*), $OP=H(OI)+H(PI)$, $CM=E_{MPK}(D_{CSK}(OI, H(PI))$, $CT=E_{TPK}(D_{CSK}(PI, H(OI))$, *C* sends payment information by way of dual signature to

M.

B6：*M* decrypts *CM* with secret key *MSK*, and decrypts it again with secret key *CSK* sent by *C*, *OI* and *H*(*PI*) are obtained, then $OI' = H(OI)$, $OP' = OI' + H(PI)$.

B7：It creates *H*(*OP*) and *H*(*OP′*) to verify their consistency, and compares them whether they are equal, if they are equal, it manifests *OI* is the order information related to *PI*, if not, it demands *C* send message again or cancel it.

B8：$M \rightarrow T$:(*OP, CT, KI*), $KI=E_{TPK}(D_{MSK}(MAN, Cert_M, Cert_C, dod))$, *M* sends request payment information by way of dual signature to *T*.

B9：*T* decrypts *CT* and *KI* with secret key *TSK*, and decrypts it again with secret key *MSK* sent by *M*, *PI*, *H*(*OI*) and certificates of *M* and *C*, then $PI' = H(PI)$，$OP' = H(OI) + PI'$.

B10：It creates *H*(*OP*) and *H*(*OP′*) to verify their consistency, and compares them whether they are equal, if they are equal, it manifests *PI* is the payment information related to *OI*, if not, it demands *M* send message again or cancel payment.

B11：Prepayment. *T* transfers *pm* to *MAN* from *CAN*, and sends successful message *prepaid* to *M*. $T \rightarrow M$: E_{MPK}(*prepaid*).

B12：*M* decrypts it with secret key *MSK*, then sends payment time *dop* to *C*, $M \rightarrow C$: $E_{CPK}(dop)$, $dop < dod$.

B13：*C* decrypts it with secret key *CSK*, then $C \rightarrow M$: E_{MPK}(*agreement*), *C* agrees to the payment time.

B14：*M* decrypts it with secret key *MSK* and gets message *agreement*, then merchant delivers goods to customer, and sends receipt to *T*, $M \rightarrow T$:$E_{TPK}(D_{MSK}(receipt, H(Cert_M))$.

B15：Customer receives goods and verifies it is no problem, and sends message to *C*, $C \rightarrow T$:$E_{TPK}(D_{CSK}(H(Cert_C))$.

B16：

(1) *T* decrypts them and gets message *receipt* and $H(Cert_C)$, and verifies identification of *M* and *C*, then sends payment information to payment gateway *B*, $T \rightarrow B$:$E_{BPK}(D_{TSK}(PI, pm, MAN, Cert_T, H(Cert_T))$, $Cert_B$ is conserved in *T* when merchant registers .

(2) *T* has only received message *receipt* from *M*, and verified identity of *M* is legitimate, or received unreal identity message from *C*, when payment time *dop* comes, *T* send payment request to *B*.

(3) *T* has only received message $H(Cert_C)$ from *C*, and verified identity of *C* is legitimate, *T* send payment request to *B*.

(4) T has not received any message within *dod*, it cancels payment automatically and informs *M* and *C*.

B17：*B* decrypts message from *T* with its secret key *BSK*, and verifies signature identity of *T* with secret key sent by *T*, *B* formally transfers *pm* to *MAN* from *CAN*.

B18：*B* returns successful message *success* of payment，$B \rightarrow T$: E_{TPK} (*success*).

B19：*T* decrypts *success*, $T \rightarrow C$: E_{CPK}(*finish*)，$T \rightarrow M$: E_{MPK}(*finish*)，*T* sends finished message to *C* and *M*.

Agility:

A1：*C*?*M*:(*Req*). *C* inquires *M* for *Req* .

A2：If *M* has *Req*, then $C \rightarrow M$:($E_{MPK}(D_{CSK}(CID, H(CID), OI))$, OI=(*GName*, *GType*, *GNumber*, *dod*), *GName*, *GType*, *GNumber* and *dod* respectively stand for goods name, specification, amount and delivery of date. *C* sends message with signature and encryption to *M*.

A3：*M* decrypts message with its secret key *MSK*, then verifies identity of *C* with the secret key *CSK* sent by *C* according to $H'(CID) = H(CID)$.

A4：*Define* E_u(Have *GName* in *GDB*), *M* defines event E_u.

A5：*CD.Start*，Event detector *CD* starts.

A6：*CD.Scan*(*GDB*), *CD* scans *GDB*.

A7：$CD \rightarrow R$:($E_{MPK}(E_u$ *arises*), When defined event appears, *CD* sends the message to trigger management program *R*.

A8：*Evaluate Cond*(*GType*, *GNumber*, *dod*), Trigger management program *R* evaluates condition.

A9：*R.Activate*(*TMG*), *R* activates *TMG* when condition is given.

A10：*TMG.Create*(*t*), *TMG* creates relative transaction *t* and executes it.

A11：$M \rightarrow C$:($E_{CPK}(OI)$), *M* sends message *OI* to C.

A12：*C* decrypts message *OI* with its secret key *CSK* and obtains the order information. The protocol is end till now.

3.3 ATOMICITY AND SECURITY

Protocol should meet atomicity of money, goods and affirmable message[6]. Known from B11 and B17, decrease of customer fund is increase of merchant fund, so it conforms to atomicity of money. Prepayment at step B11 meets both atomicity of money and preventing deceit deed of merchant and customer. Known from B15 and B16, customer begins to pay while he gets goods. Known from B14, B16 and B17, merchant deliveries goods while he receives payment money, so deed which customer gets goods without payment or merchant receives payment without delivery is not happened. This way of payment in getting goods conforms to atomicity of goods. Known from B14, B15 and B16, TTP asks payment gateway for payment and preserves transaction evidence when TTP has received receipt of merchant and identity message consent to pay by customer, this realizes atomicity of message who send it.

Truth to identity of transaction entity is important guarantee of protocol security. Certificate is the legal mark of identity, entities apply certificate authority for certificate. It needs to identify the truth of identity through delivering certificate in transaction entities. There is six times identification among four entities, *C* and *M*, *C* and *T* verify each other, and certificates of *T* and *B* are respectively conserved in *M* and *T* in advance, so *T* only needs to verify identity of *M*, *B* only needs to

verify identity of *T*. Process to identification consists of B1, B3, B8 and B16. Figure 2 shows the process.

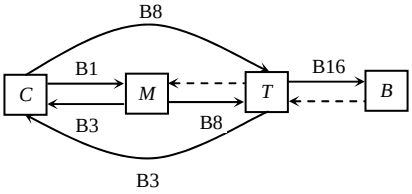

Figure 2 Identification process

Before formal payment, *T* needs to verify identity consistency of *M* and *C*. Certificates of *M* and *C* are sent to *T* at step B8, $H(Cert_M)$ and $H(Cert_C)$ are created, *T* compares them to these sent through B14 and B15, if they are equal, it shows identity of *M* and *C* are consistent and true.

Deceit means *M* gets *C* money by cheating without delivering goods or *C* gets goods without payment. To prevent deceit is one of security problems for e-commerce protocol. Obviously, deceit deed may happen in period of Business protocol. *M* may send a false receipt to *T* without delivering goods when protocol executes step B14, *C* does not send its certificate or send a false certificate to T when protocol executes B15. Considering success or failure of communication, there may have nine kinds of situations when protocol executes.

(1) *M* and *C* are both honest, they send message to *T* successfully.

(2) *M* and *C* are both honest, *M* sends message to *T* successfully, *and C* is failure.

(3) *M* is honest, *C* is deceitful, they send message to *T* successfully.

(4) *M* is honest, *C* is deceitful, *M* sends message to *T* successfully, *C* does not send message to *T*.

(5) *M* and *C* are both honest, *M* fails to send message to *T*, *and C* is successful.

(6) *M* and *C* are both honest, *M* and *C* fail to send message to *T*.

(7) *M* is honest, *C* is deceitful, *M* fails to send message to *T*, C does not send message to *T*.

(8) *M* is honest, *C* is deceitful, *M* and *C* fail to send message to *T*.

(9) *M* is deceitful and sends message to *T* successfully.

Step B16 can solve above-mentioned problems. For situation (1), step B16(1) solves it, step B16(2) solves situation (2), (3) and (4), step B16(3) solves situation (5), step B16(4) solves situation (6), (7) and (8).

TTP is an entity of prepayment, it preserves certificates of *M* and *C* and transaction information as evidence at the same time. Responsibility in protocol executing is traceable, it resists denying. For situation (9), *M* deceives successfully, that is to say, *M* sends a false receipt to *T*, *T* pays only when signature certificate of *M*. If *C* does not get goods when *dod* expires, it can find out *M* is to blame.

4. CONCLUDING REMARKS

E-commerce protocol with agility can handle customer active request in real time, it makes merchant e-commerce system respond to a customer request agilely, and changes positive e-commerce into active e-commerce, and wins more opportunities for merchant.

Security to e-commerce includes three layers, they are physical layer, data layer and business layer. This protocol bases on business application layer. Signature guarantees security of business information. MD5 is a kind of common Hash to signature. Found by Wang Xiaoyun, MD5 exists “conflict”, two files can create the same finger mark, this discovery makes law effect of current signature challenged[7]. Universal used signature algorithm in e-commerce is worth to research.

Agent technology becomes more and more mature, it has wide prospect in e-commerce. Protocol introduced here is only tentative program, it does not introduce the content of intelligence search, negotiation, transaction agent and so on, and these valuable tasks are worth to study for us.

ACKNOWLEDGEMENT

This paper is supported by my tutor , Prof. Xiong Qianxing.

REFERENCES

[1] Chunxiao L., Hui A., *E-commerce—from idea to action*, Beijing: Tsinghua University Press, 2001.

[2] Aslam, Taimur. Dr. Dobb, “Protocols for E-Commerce”, *Software Tools for the Professional Programmer*, Vol. 23, No.12, pp52-57, 1998.

[3] Renee Gotcher, “Tools and Protocols for E-Commerce”, *Information Security Technical Report*, Vol. 3, No. 2, pp34-40, 1998.

[4] Yunsheng L., *Advanced Database Technology*, Beijing: National Defence Industry Press, 2001, 3.

[5] Sihan Q., “TTP Roles in Electronic Commerce Protocols”, *Journal of Software*, Vol. 14, No. 11, pp1936-1943, 2003.

[6] Longxiang Z., “An Overview of Protocol Research in Electronic Commerce”, *Journal of Software*, Vol. 12, No. 7, pp1015-1031, 2001.

[7] Cryptograph of finger mark is found out, Signature is absolute security or not, http://www.sict.ac.cn/news/xwcx.asp?corp_code=00000001960, 2004.