Selling Telemetry Data Over the Internet
Using SET

Jeffrey R. Kalibjian
CounterSign Software, Inc.

ABSTRACT

Over the past two years the design and implementation of secure Internet based data sharing tools which could enable geographically remote contractor teams to access flight and test telemetry data securely over the Internet were presented [1] [2]. Key technologies facilitating these capabilities were the Hypertext Transfer (HTTP) protocol, the Secure Sockets Layer (SSL) protocol, and the Secure Multipurpose Internet Mail Extension (S/MIME) specification. This year we discuss utilizing the Secure Electronic Transaction (SET) specification in tandem with HTTP, SSL, and S/MIME to deploy a system for securely selling telemetry data over the Internet.

KEYWORDS

electronic commerce, secure data sharing, S/MIME, MIME, SSL, HTTP, SET

INTRODUCTION

The Secure Electronic Transaction (SET) protocol [3], jointly developed by MasterCard and Visa, enables credit card transactions to be securely carried out over the Internet. The protocol provides for confidentiality of cardholder information (e.g. credit card number, etc.), ensures payment integrity, and authenticates both merchants and cardholders. After reviewing SET and the SET paradigm of cardholder, merchant, acquirer interactions, an architecture will be discussed incorporating SSL, S/MIME, SET, and HTTP technologies in an electronic commerce system that can make telemetry data available for purchase over the Internet.

SECURITY USING SSL

Thousands of items each day are purchased on the Internet using credit cards. Typically these transactions are secured using SSL [4]. SSL is a point to point transport level protocol which utilizes dual asymmetric key cryptography and symmetric key
cryptography (see Appendix 1) to both encrypt and authenticate messages. The message content can be any type of data including text, numeric or binary. Since the data that can be secured is arbitrary, systems employing SSL technology are under strict export control.

SET

The SET protocol was designed to only deal with credit card transactions. It's sole purpose is to secure credit card numbers, and purchase information, thereby facilitating a smooth, yet secure transaction path between cardholder (customer), merchant, and acquirer (An Acquirer is the financial institution that provides payment card authorization and payments to the merchant). While SET employs the same type of cryptographic technologies used in SSL, it is not export restricted because of the very limited types of data which may be encrypted.

THE ADVANTAGES OF SET

There are two primary advantages to using SET. First, the protocol can prevent a merchant from gaining access to the customer's credit card number in the Internet environment, the cardholder typically knows very little about the merchant. To prevent the merchant from obtaining the customer's credit card number, the protocol can encrypt the customer's card number so that it can only be decrypted by the acquirer. The merchant will pass on the encrypted credit card number to the acquirer during the authorization process. The merchant does not need to know the customer's credit card number; only that the acquirer is prepared to release funds on behalf of the cardholder for the amount of the purchase.

The other major advantage of SET is that the protocol already mirrors the standard business data flows in place between card holders, merchants and acquiring institutions used in the "non cyber" world. In this model, a person obtains a credit card through a financial institution. Next a person uses the card at a store. The card is swiped through a Veriphone (TM) terminal which is connected to the merchant's acquirer. The acquirer gives the merchant an authorization number indicating that the acquirer will pay the merchant the amount of the purchase, when requested. Finally, the merchant submits the bill to the acquirer in order to get paid. The amount paid to the merchant by the acquirer appears on the customer's credit card bill (the bill, originating from the cardholder’s financial institution) in the next billing cycle. The implication is that no "out of band" processing is required to complete a cardholder/merchant SET transaction (e.g. merchant taking a credit card number delivered to a web site, manually entering it in a veriphone terminal, waiting for the response, etc.).
In the cyber SET model the cardholder, merchant and acquirer entities interact over the Internet using special transactions that comprise the SET protocol. The SET transaction types for each entity are shown in Figure 1,

<table>
<thead>
<tr>
<th>Cardholder</th>
<th>Merchant</th>
<th>Acquirer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purchase request</td>
<td>Authorization request</td>
<td>Authorization response</td>
</tr>
<tr>
<td>Inquiry</td>
<td>Purchase response</td>
<td>Capture response</td>
</tr>
<tr>
<td></td>
<td>Capture request</td>
<td>Authorization reverse resp.</td>
</tr>
<tr>
<td></td>
<td>Authorization reverse request</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Capture reverse request</td>
<td>Capture reverse response</td>
</tr>
<tr>
<td></td>
<td>Credit request</td>
<td>Credit response</td>
</tr>
<tr>
<td></td>
<td>Credit reverse request</td>
<td>Credit reverse response</td>
</tr>
</tbody>
</table>

Figure 1. The SET transactions.

Typical SET interaction is as follows. A customer uses a web browser to view products at a merchant's site. The customer selects the item(s) to be purchased; then utilizes his SET "wallet" software to make a purchase request to the merchant. This request is delivered to the merchant. The merchant takes the purchase request, validates it, formulates an authorization request (passing on among other things the encrypted credit card number of the card holder it can’t read) and delivers the authorization request to the acquirer. The acquirer processes the request and interfaces to legacy financial systems to initiate the authorization process. The result of the authorization is sent back to the merchant in the authorization response message. After receiving the authorization response, the merchant can formulate a purchase response to the card holder. After actually shipping the product to the customer, the merchant can initiate a capture request to have the funds he previously authorized deposited into his bank account. Again, the acquirer will process the capture request message, and actually carry out funds transfer action by interfacing to legacy financial systems. It should be noted that all transactions are encrypted and digitally signed to ensure privacy between entities. The legacy interactions may not as they take place on non public networks.

Other transactions are also possible between the merchant and acquirer (e.g. authorization reverse, if say, items purchased cannot be delivered, etc.) They will not be described here; however, the transaction names shown in Figure 1 clearly conveys the action carried out.
**HOW IT WORKS**

Typically SET messages are delivered to entities by making TCP/IP socket connections to client/server computers connected to the Internet. Entities can be assured messages are coming from legitimate entities by reviewing the X509 certificates (see Appendix 1) enclosed in the SET transaction. In the SET paradigm, merchants and acquirer's must utilize X509 certificates signed by a SET Certificate Authority. While the cardholder doesn't have to have a certificate, he is encouraged to get one to enhance the level of security that may be used for the transaction he wishes to initiate.

**AN INTERNET BASED TELEMETRY PROCUREMENT SYSTEM**

The system will typically have six elements; telemetry decoding software, telemetry post-processing software, data transport and conversion software, SSL based web server or S/MIME [5] server, SET merchant software, and firewall software. The secure SSL server or S/MIME server will be used as the point of presence on the Internet for the company selling the telemetry data. If, the company chooses a web presence; they may choose to use COTS software to set up a "store front" describing the data they are selling. It should be noted that the store front software will likely require the data to be in particular formats (e.g. images in .gif format, etc). Thus, it will be the data transport software's responsibility to appropriately convert the telemetry data (after it is decoded and or post-processed) and move it to a location where the web store-front software can gain access to it.

![Diagram of Internet Telemetry System](Image)

Figure 2. Architecture of a simple Internet telemetry procurement system.

If the company chooses to have an e-mail presence (utilizing the S/MIME protocol for security) the COTS or custom software managing the e-mail interaction will again expect data to be in particular formats that the data transport software will have to manage. The
SET merchant software will have to be installed and integrated with the back end of the storefront web software (or e-mail management software) to coordinate order and pricing information. In addition a SET acquirer entity will have to be chosen by the company selling the telemetry data so authorizations, and captures can be processed. Finally, the entire system should be placed behind a firewall to insure the software and data is not vulnerable to "hacker" attack.

MANAGING THE PROTOCOLS

The secure SSL web server and secure S/MIME web server are used to guarantee the telemetry data product is not compromised during transit. The company selling telemetry data is assuming the customer will have access to SSL enabled web clients and/or S/MIME e-mail clients. This is a conservative assumption as both the Netscape Navigator (TM) and Microsoft Internet Explorer (TM) clients accommodate both SSL and S/MIME. Unfortunately, neither product supports SET at this time. Thus, the company selling the telemetry data will need to offer a plug in for the customer's client so the SET protocol can be used to communicate credit card sales information to the merchant.

It should be pointed out the S/MIME and SSL clients/servers require X509 certificates to be exchanged so clients and servers can be positively identified. Unfortunately, the format of these certificates is somewhat different than the X509 certificates utilized for SET transactions. Thus, generally the clients will be expected to manage two sets of X509 certificates; one to support SSL and S/MIME interactions; the other to manage their SET interactions.

CONCLUSION

The SET protocol is one of the final pieces of the Internet commerce puzzle. The protocol will facilitate streamlined and secure credit card transactions over the Internet while also encouraging more vendors to go on line with products to sell. A unique application for SET will be the buying and selling of telemetry data. SET merchant software is easily integrated into SSL or S/MIME based "storefront" systems; implying that those companies already utilizing SSL or S/MIME based servers to securely share telemetry data, may be closer than they think to setting up on-line telemetry data sales centers.
Appendix 1
Cryptography and X509 Certificates

The science of cryptography [6] seeks to disguise data and offer a mechanism for data recovery. A cipher algorithm can be used to both encrypt (disguise) or decrypt (unde-sign) data. The input to the cipher is usually the data (ciphered data or clear data) and a "key." Ciphers which can decrypt or encrypt data with the same key are known as symmetric key ciphers. There is another type of cipher that uses two keys. This type of cryptography is known as dual asymmetric key cryptography (also known as public key cryptography [7]). Here a public/private key pair is generated. The keys have a reciprocal relationship; that is, if one key is used to encrypt data, only the other can decrypt the data. Typically, one key is given to the public and one is kept private. The most popular public key algorithm is the RSA [8] algorithm.

An example will illustrate the public/private key concept. If Person A wishes to send a private message to Person B; Person A uses Person B's public key to encrypt the data to be sent to Person B. Person B would then have to use her private key to decrypt the data. She is the only person who could decrypt the data. If Person B sought to authenticate that Person A did indeed generate the message; she can request that Person A run a checksum of the encrypted message before transmission; then encrypt the checksum with his (Person A's) private key. If Person B can decrypt the checksum with Person A's public key (and the checksum matches Person B's generated checksum of the message); she can be assured herself that Person A truly did send the message; and further that it had not been tampered with.

Public key encryption algorithms are computationally very expensive; while symmetric key algorithms are quite fast. Thus, in practice a symmetric key is generated and used to bulk encrypt data to be disguised. The symmetric key is then encrypted using a public key algorithm. Public Key Cryptography Standards provide a syntax for organizing these data elements (e.g. the bulk encrypted data, the encrypted symmetric key, etc.) so they may be exchanged and understood by differing software implementations (conforming to the PKCS standards) running on dissimilar computer systems.

It should be noted that individuals typically don't release "raw" public keys to the public. Instead, they use a certification authority (CA) to imbed their public key into an X.509 [9] public key certificate. As part of this process the certification authority signs the certificate with its private key. If the certification authority can distribute its public key to a wide audience, users may validate public keys of individuals they do not directly know by verifying the CA's signature on a certificate.
References


