Using GSM to Enhance E-Commerce Security

Vorapranee Khu-smith and Chris J. Mitchell
Information Security Group
Royal Holloway, University of London
Egham, Surrey TW20 0EX
United Kingdom
E-mail: {V.Khu-Smith,C.Mitchell}@rhul.ac.uk

ABSTRACT

Today, an e-commerce transaction is typically protected using SSL/TLS. However, there remain some risks in such use of SSL/TLS. These include that of information being stored in clear at the end point of the communication link and lack of user authentication. Although SSL/TLS does offer the latter, the security service is optional and usually omitted. This is because of the fact that users typically do not have the necessary asymmetric key pair. Since SSL/TLS protects data only while it is being transmitted, the merchant has access to sensitive information such as the debit/credit card number. The storage of unencrypted debit/credit card information at the merchant server therefore represents a risk that is not currently addressed by the use of SSL/TLS to secure electronic payment transactions.

In this paper, we propose a payment protocol in which the risk of having debit/credit card details stored at a merchant server is eliminated. User authentication is also provided. This is achieved by utilising the GSM data confidentiality service to encrypt sensitive information. The GSM security service is also used to provide user identity authentication. The additional security is realised in such a way that no management overhead is imposed on the user.

Categories and Subject Descriptors

H.4.3 [**Information Systems Applications**]: Communications Applications; C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design

General Terms

Security

Keywords

E-commerce security; mobile or Internet payment protocol; GSM security

WMC'02, September 28, 2002, Atlanta, Georgia, USA.

1. INTRODUCTION

The Internet is now widely employed for e-commerce. In an e-commerce transaction, a consumer typically makes a payment using a debit/credit card. The communications link between the consumer PC and the merchant server is commonly protected against eavesdropping using SSL/-TLS [2, 11]; even so, a number of security threats remain [5, 6, 7, 8, 10]. One reason for these remaining vulnerabilities is that SSL/TLS does not obligate client authentication. As a result, it is not easy to verify if the person who is making a payment is the legitimate cardholder. A malicious user, who may have obtained card details by some means, may then be able to use them to make payments over the Internet at the expense of the legitimate cardholder. Consequently, a way to reduce the risk of such frauds is to perform user authentication.

Apart from the lack of client authentication, using SSL/-TLS to protect an e-commerce transaction poses another threat. Since SSL/TLS was designed to secure the communications link, the information is available unencrypted at the destination. As a result, merchant servers have become a target for attackers who wish to obtain card details. Of course, all these problems could be avoided by use of the SET protocol [9], a scheme devised jointly by MasterCard and Visa to protect entire e-commerce transactions. Unfortunately, however, SET has not taken off, apparently for a variety of reasons. Foremost among these reasons are the major initialisation and implementation overheads it imposes on both e-consumers and merchants. Thus alternative ways of enhancing the level of security provided by SSL/TLS, and which do not impose such major overheads, are urgently needed.

If client authentication is to be provided by SSL/TLS, then the user must first establish a public key pair. A secure place will also be needed to store the private part of the key. Usually the key is stored in the user PC and hence the user has to use the particular machine every time a payment is to be made. Although a smart card could be employed to store the key and enhance mobility, not many user PCs are equipped with smart card readers. By contrast, very large numbers of users across the world now possess a GSM mobile phone.

Therefore, in this paper we propose a payment protocol in which a GSM mobile phone is used to provide cardholder authentication and card details confidentiality in a way that also supports user mobility. These security services are achieved by utilising mobile phone portability and the GSM data confidentiality service.

In this paper, the GSM data confidentiality service is first described, followed by the proposed protocol. A threat analysis, and a discussion of the advantages and disadvantages of the scheme are subsequently given.

2. DATA CONFIDENTIALITY

Three main security services are provided by the GSM air interface protocol. They are subscriber identity confidentiality, subscriber identity authentication, and data confidentiality. However, data confidentiality is the only security service used in the proposed protocol and hence will be the only service described here. Details of the other security services can be found in [3, 4, 12, 16].

GSM security is based on a long-term secret key K_i, shared between a Subscriber Identity Module (SIM) and the user's home network (which provided the SIM). Voice and signalling data sent between the mobile telephone (equipped with the SIM) and the visited network is encrypted using a secret session key K_c. This key is derived from the long term key K_i as a function of a random value *RAND* passed from the network to the SIM during subscriber identity authentication. The key K_c is computed within the SIM and made available to its host mobile telephone for data encryption (all data encryption is performed externally to the SIM). The key is also made available to the visited network by the subscriber's home network's Authentication Centre (AuC).

3. USING GSM FOR E-TRANSACTIONS

In this section, the proposed new protocol is described. In the scheme, a consumer is required to have a GSM Mobile Equipment (ME) and a SIM registered under the name that appears on his/her debit/credit card. It is important to note that the protocol does not need the SIM to be modified in any way. However, the ME does need some special capabilities, as described below.

In this section, the system components required are first described, followed by the transaction processing procedure.

3.1 System components

Five main system components are involved in our payment protocol. These are a User System, a merchant server, an acquirer, an issuer, and an AuC.

3.1.1 User System

The User System consists of a Mobile System (MS), which includes a SIM, an ME, and a PC. The MS (in fact the SIM) is responsible for outputing the key K_c. Therefore, although an ME is needed to interact with the SIM, the protocol can work without an ME if there is an alternative means for the SIM to communicate with the user PC.

The means of communication used between the MS and the user PC is not specified in this paper. However, Infrared, a cable, or Bluetooth[1] could be employed (such means of communication are becoming commonplace as mobile devices are increasingly being used for data transfer).

In the remainder of this paper the scheme is described in the context of a User System in which the PC provides the main platform for conducting user e-commerce, and the MS acts to support the additional security functions. However, in environments where the MS has sophisticated user interfaces and processing capabilities, e.g. a WAP or 3G phone, the MS could take on some or all of the PC's tasks.

Note that in this paper, we have proposed use of the key K_c for MAC computation where this key is normally used for data encryption. This is a breach of key separation principles although it may not be of significance here. However, if this does give rise to security concerns then the key could be modified, e.g. passed through a hash function, before being used to compute a MAC.

3.1.2 Merchant server

The merchant server is the component that interacts with the User System to support electronic transactions. The merchant server also interacts with the acquirer to request a payment authorisation. The choice of the communication link between the two is not an issue here. However, it could be the Internet, or a special-purpose link provided by the acquirer.

As discussed in Section 4.2.2, we suppose that the integrity of the merchant server/acquirer link is protected in some way, e.g. via MACs or signatures; however, the means by which this is achieved is outside the scope of the discussion here.

3.1.3 Acquirer, issuer and Authentication Centre

The acquirer interacts with the issuer via the financial network to support transaction authorisation. However, in the proposed protocol, the issuer has the additional roles of authenticating the cardholder, decrypting the card details, and verifying the authenticity of the payment details and card details.

The issuer interacts with the AuC of the user's home network in order to retrieve values necessary to utilise the GSM security service. The choice of the communication link between the issuer and the AuC is again outside the scope of this paper. However, it could be the Internet or a special-purpose link provided by the mobile network operator. As discussed in Section 4.2.3, we assume that the integrity and confidentiality of the issuer/AuC link are provided by some means.

The AuC is required to supply the issuer with values necessary for the GSM data confidentiality service. It takes inputs from the issuer and produces the values used for the additional security services.

3.2 Transaction processing

The proposed payment protocol starts after a consumer has decided to make a payment. The decision about which purchase to make is outside the scope of this paper — we simply assume that the consumer and the merchant wish to perform a specified transaction.

The consumer first fills in a typical Internet purchase form (excluding card details). In the protocol, the form is also required to contain a field for a GSM phone number. Upon receipt of the form, the merchant server initiates the proposed protocol. The procedure is illustrated in Figure 1.

In this figure:

- *RAND* denotes a randomly generated 64-bit value,
- $e_K(M)$ denotes message M encrypted (using symmetric encryption) with key K,
- K_c denotes a cipher key used for encryption and MAC computation,

[1] http://www.bluetooth.com

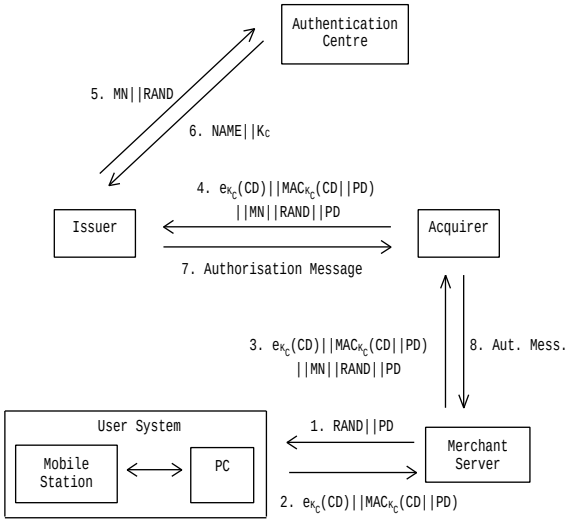

Figure 1: GSM-e-commerce payment protocol.

- 'CD' designates card details entered by a consumer,
- $X||Y$ denotes the concatenation of data items X and Y,
- $\text{MAC}_K(M)$ denotes a MAC computed on message M using the key K,
- 'PD' denotes payment details,
- 'MN' is a GSM phone number, and
- 'NAME' is the subscriber name.

Upon receipt of the form, the merchant server generates and sends a random number ($RAND$) and the payment details (PD) to the user PC, as shown in message 1. The user first checks the PD. If it is correct, the $RAND$ is then forwarded to the SIM, which in turn calculates the key K_c using the received $RAND$ and its stored key K_i as inputs to the key derivation algorithm shared with the AuC. The SIM then passes the generated K_c back to the ME, just as it would normally do in a GSM telephone (i.e. the SIM is not required to have any special functionality). The ME then forwards the encryption key to the user PC.

The user PC uses the key K_c to encrypt the card details entered by the user. Examples of card details include card number, expiry date, issue number, and card verification code (CVC). In addition to the encryption, a MAC is computed on a concatenation of the card details (CD) and the payment details (PD), again using the key K_c, to protect the integrity of the information. The PD must include (but is not limited to) the transaction value, date, and merchant and transaction identification number. The enciphered information and the MAC are then sent to the merchant server as shown in message 2 in the figure. Note that the encryption and MAC algorithms used here can be issuer-specific, and need bear no relationship to the GSM algorithms. The only requirement is that they are able to operate using a 64-bit GSM session key K_c.

The merchant server concatenates the received message with its own version of PD, the $RAND$, and the user mobile phone number (MN) extracted from the purchase form. The result is sent to the acquirer where it is forwarded to the issuer as shown in messages 3 and 4 respectively.

In order to decrypt the encrypted CD and verify the MAC, the issuer needs to contact the appropriate AuC to retrieve the key K_c. The issuer can either determine the identity of the user's home network (and hence the address of the AuC) from the mobile number, or, if necessary, an identifier for the AuC can be included in messages 2, 3 and 4. To enable the AuC to respond with the right information, the issuer sends the mobile number and the $RAND$ to request the AuC to respond with the subscriber name and the cipher key. This corresponds to message 5 in Figure 1. The AuC then responds with message 6 containing the name and key.

The issuer first decrypts the CD using the supplied key K_c. The issuer then verifies the MAC to check the integrity of both the CD and the information in PD, especially the transaction ID and merchant ID. The checking of PD is necessary in order to prevent replay attacks (see Section 4.3). The checking of the MAC is also important because if the MAC is valid, then the user must possess the valid SIM. If also the subscriber name matches the cardholder name, the cardholder is deemed to be the legitimate cardholder since he/she possesses the SIM.

If all these processes are successful, the issuer can now proceed with the 'normal' transaction authorisation. Otherwise, the transaction is declined. The decision of the issuer is reflected in the Authorisation Message (message 7) which is then sent to the acquirer where it is forwarded to the merchant server as shown in message 8. The protocol now ends.

Finally note that the protocol could be enhanced to ensure that a different key is used for every transaction, even if the merchant fails to generate a new $RAND$ every time. The user system could generate its own random number, $RAND^*$ say, and then derive a transaction key K_t as a one-way function of $RAND^*$ and K_c. The key K_t can then be used instead of K_c in the protocol ($RAND^*$ must also be sent).

4. THREAT ANALYSIS

In this section, we consider threats to the proposed protocol. The threats can be divided into four categories: threats to the User System, threats to the communications links (User System/merchant server, merchant server/acquirer, financial network, and issuer/AuC), threats in the merchant server, and threats in the acquirer, the issuer and the AuC.

4.1 Threats in the User System

As stated previously, the User System consists of a user PC and an MS. In this section threats to the MS are first described followed by threats to the user PC.

4.1.1 Threats to the Mobile System

If an attacker has stolen a SIM, although a valid cipher key K_c can be generated, he/she will not be able to complete a transaction. The attacker still needs card details and even if stolen card details are submitted, the fraud will be detected as soon as the transaction is processed by the issuer. This is because the card details, in particular the cardholder name, will not match the subscriber name sent by the AuC.

It is clear however that if the attacker has both a complete set of card details and a stolen SIM for the cardholder, then

the system cannot prevent an attack — unless, of course, the SIM has been reported stolen and blacklisted by the network.

If an attacker has stolen an ME, without a SIM, he/she will not be able to make a fraudulent payment, regardless of whether the corresponding card details have been obtained. The ME is only responsible for forwarding information between the SIM and the user PC. Without a SIM, a valid encryption key K_c cannot be generated and hence stealing an ME does not yield financial gains to an attacker.

4.1.2 Threats to the user PC

Since the user PC does not contain sensitive information, the threats arising from the PC are minimal. Although information that passes via the PC can be cached and attacked, this information is not confidential. A debit/credit card details and the payment details can be cached and compromised but the protocol still requires a corresponding SIM to make an electronic transaction. The cipher key can also be compromised, but since it is only a transient key and is a function of the *RAND* sent from the merchant server for each transaction, compromising this key is not a threat unless an attacker can impersonate a merchant server and force re-use of an old *RAND* value (and hence an old key K_c). This can be prevented by requiring the user system to authenticate the merchant server and by the provision of integrity protection for this link (see Section 4.2.1).

If the fact that the PC has access to the card details is considered an issue, alternative implementation scenarios are possible; in particular some of the functionality currently allocated to the user PC could be transferred to the ME. For example, if the ME has an appropriate user interface (and appropriate processing capabilities), the card details could be entered into the ME and encrypted there, denying the user PC any access to sensitive information.

4.2 Threats to the communication links

If any of the information transferred across any of the links is modified, then the protocol will fail. Hence, a theoretical denial of service attack exists, although there are many simpler ways to prevent the completion of a transaction. We now consider other threats arising to the three links (User System/merchant server, merchant server/acquirer, and issuer/AuC links). As stated before, the issuer/acquirer communication link is assumed to be the financial network. Therefore, its security is assumed here.

4.2.1 Threats on the User System/merchant server link

Threats on this link can be divided into two types, namely integrity threats and confidentiality threats. In this section each piece of information that is transmitted via this link, i.e. the *RAND*, the card details and the payment details, is considered in turn against both types of threat. However, threats to *RAND* will not be included since modifying or eavesdropping on this value do not enable attacks to be launched. As a result, only the card details and the payment details will be considered.

Integrity threats: It is important to ensure payment details (PD) integrity in order to prevent a malicious merchant from modifying the PD to gain financial advantage, such as charging the consumer more than is agreed upon. The PD is protected against unauthorised modification using a MAC. Without the key K_c, it is hard to generate a valid MAC for a modified PD.

Although modifying the card details (CD) does not yield any gain to an attacker, in our protocol the CD is included in the MAC computation to ensure its integrity. It is worth noting that including the CD in the MAC has no impact on the message length, and it only creates a small extra computational requirement.

As stated in Section 4.1.2 however, there are threats arising from forcing re-use of an old *RAND* for which the corresponding key K_c is known. In such a case, the MAC can be modified and/or the CD compromised. Although the likelihood of compromise of a key K_c by a malicious third party is likely to be relatively small, if the threat of compromise of the key K_c is a possibility, then integrity protection for this channel and merchant server authentication is required. This can be achieved using a secure channel such as SSL/TLS.

Confidentiality threats: It is essential to ensure the confidentiality of sensitive information, i.e. the CD. In the protocol, this is provided by symmetric encryption.

Unlike the CD, the PD contains no sensitive information and hence does not need protection against eavesdropping. Indeed, the PD is analogous to a Point of Sale (POS) receipt which typically contains only the store name, date, product description, transaction value and in some cases, the last four digits of the payment card used. Therefore, confidentiality of the PD is not provided.

However, as part of a purchase form, the consumer name along with other contact information, in particular his/her mobile number, will be entered. Consequently, confidentiality of the link is needed otherwise it would be possible for an attacker to passively eavesdrop on the link and obtain the (MN, consumer name) pair. Confidentiality protection for the User System/merchant server link can be provided using a secure SSL/TLS channel just as is normally the case for Internet transactions.

4.2.2 Threats on the merchant server/acquirer link

Threats to the CD, the *RAND*, and the PD are similar to those previously described. We now consider the remaining information, i.e. the MN and the authorisation message, in terms of confidentiality and integrity threats.

Confidentiality threats: By monitoring the link, a list of mobile phone numbers could be obtained. However, unlike the threat described in the previous section, the consumer name is not transmitted on this link. Therefore, having only a list of phone numbers without the corresponding names is not likely to be very valuable.

An authorisation message may contain information similar to that in a normal receipt. However, it is clear that it does not need to contain any card details since such information is not necessary for the merchant to complete the proposed payment protocol. Therefore, the authorisation message is not sensitive and hence is not protected against eavesdropping in this protocol.

Integrity threats: Modifying the MN can only make the protocol fail and does not yield gains to any party involved. On the other hand, the integrity of an authorisation message is important, since a malicious merchant could modify the authorisation message from reject to authorise, potentially causing a dispute. A way to prevent such a threat is to ensure the integrity of the message, e.g. to require the acquirer

to sign a message before sending it to the merchant.

4.2.3 Threats on the issuer/Authentication Centre link

Threats on this link can again be divided into two types, namely integrity threats and confidentiality threats.

Integrity threats: Modifying the *RAND* and the MN will only cause the protocol to fail. However, if the integrity of information sent via this link is not ensured, it would be possible for an attacker to manipulate this link in order to bypass the cardholder authentication check. The attacker could first use an arbitrary (but valid) GSM number and symmetric encryption key to encrypt the details of a stolen card (which, of course, will not match the GSM subscription name). In message 6 the AuC will provide a valid K_c and the name associated with the attacker's GSM subscription. An active attacker could then replace the contents of message 6 with the name associated with the stolen card details along with the arbitrary encryption key he/she used for the encryption. The issuer will accept the cardholder authentication because the key can be used to decrypt the card details successfully and the names match. It then will proceed with the payment authorisation process. The remainder of the protocol will complete correctly, and the account for which the details were stolen will be charged for the transaction. The existence of this attack means that it is vital that the integrity of the link between AuC and issuer is protected.

Confidentiality threats: As stated before, *RAND* is not sensitive and hence confidentiality threats to the data transmitted on this link are minimal. The key K_c is also not highly sensitive, although if the key can be intercepted and if an attacker also has access to the encrypted card details, then it would be possible for them to decrypt the card details. However, having only card details is not sufficient to make an electronic payment transaction in our protocol.

Confidentiality threats also arise from the fact that the mobile number and the corresponding subscriber name are sent across this link. Therefore, in the absence of confidentiality protection on the issuer/AuC link, an eavesdropper could find the subscriber name corresponding to any GSM number. This would be a significant breach of GSM subscriber confidentiality.

This attack means that it is important to provide confidentiality and integrity for this link, and this is why we assume throughout the paper that this link is both confidentiality and integrity protected.

4.3 Threats to the merchant server

In the protocol, the merchant server does not have access to some of the sensitive information, in particular the CD, that it would in traditional electronic transactions, since the information is encrypted with a key that the merchant server does not have. The protocol therefore reduces the threat of storing unencrypted card details at merchant servers which is one of the major security threats when SSL/TLS alone is used to protect electronic transactions.

The merchant server does have access to the *RAND*, PD, and the authentication message. However, this information is not sensitive. Therefore, there is no serious threat to data confidentiality in this system component.

It may be seen that a malicious merchant could replay message 3 in Figure 1 to re-capture a payment. However, recall that PD must contain the charging amount, date, and merchant and transaction ID (see Section *3.1.1*). Therefore if, for example, an unscrupulous merchant tries to re-submit message 3 to the acquirer, the fraud will be detected as soon as the issuer performs the authorisation. The issuer will be able to detect that the transaction ID of a certain merchant matches a previously submitted transaction. If the issuer maintains a record of the *RAND* values used for each payment card account, a matching *RAND* in two different transactions can also be an indication of merchant fraud. This is because the *RAND* must be re-used in the fraudulent transaction to enable the issuer to decrypt the replayed CD and hence be able to authorise the payment. The integrity of the CD and the PD is protected by use of the MAC.

Finally, the merchant server has access to large volumes of potentially sensitive GSM subscriber information. As part of the user authentication process, the merchant needs the user's mobile number. The merchant server also knows the name of the user since it is typically entered in the purchase form. As a result, the merchant server can collect mobile numbers and corresponding subscriber names. However, this is analogous to supplying personal contact information in a typical order form. Privacy laws then apply and may require order forms to contain a privacy statement or a section for user consent if their personal data is to be used for other purposes.

4.4 Threats to the acquirer and the issuer

Threats to the acquirer are minimal since it is responsible only for forwarding messages between the issuer and the merchant server. Moreover, the information that is transmitted via the acquirer is not sensitive.

Since the issuer is responsible for the identity authentication process, in particular the comparison of the names, it is important to protect the issuer against any attack which might cause the cardholder authentication process to be bypassed.

In the protocol, the issuer retrieves from the AuC the account holder name for any GSM telephone number. As a result, the same user privacy issue described in the previous section also exists here. Not only is this a sensitive privacy issue, but requiring the AuC to supply such information may potentially be in breach of its licence and/or data privacy legislation. It is therefore vital that the issuer is protected so that this information cannot be abused.

One way of mitigating this security issue is to make a slight modification to the protocol of Section 3.2. In the revised protocol, shown in Figure 2, in message 6 the AuC supplies the issuer only the encryption key K_c to enable the issuer to decrypt the card details. Subsequently, two more messages are required in the protocol. After successfully decrypting the CD, the issuer sends message 7 which contains the cardholder name as well as the mobile number. The AuC is then required to perform the matching between the name supplied in message 7 with the name it has associated with the GSM number. The AuC finally sends the result of the matching to the issuer (message 8).

This modified protocol has the advantage that the AuC retains control of sensitive subscriber information. However, it has the disadvantage of requiring two more communications and additional processing from the AuC.

Note that the revised protocol still allows the issuer and the acquirer to learn the phone number for the purchaser. To avoid such an issue, the merchant server could send an encrypted version of the MN to the acquirer (using a public

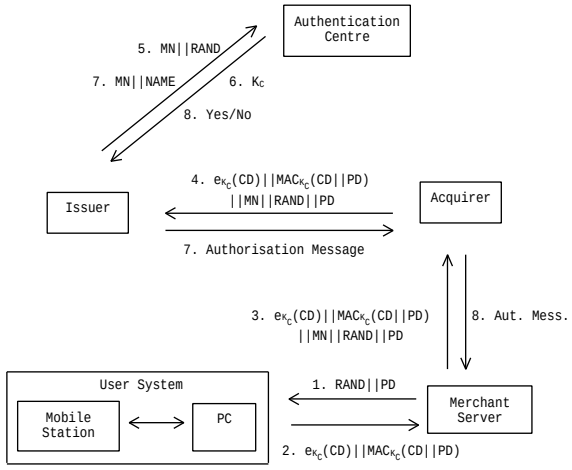

Figure 2: Revised protocol.

encryption key for the AuC), although messages 2, 3 and 4 would then need to contain an identifier for the AuC of the user's home network.

4.5 Threats to the Authentication Centre

If the integrity of the AuC could be compromised, then there are possible attacks to the security of the user authentication and encryption process and services. However, in such an event there are also many other serious attacks to the security of the GSM network itself, and so we assume that the AuC is well-protected.

5. ADVANTAGES AND DISADVANTAGES

In this section, the advantages and disadvantages of the proposed protocol are considered.

5.1 Advantages

The following advantages arise from use of the proposed GSM-based payment system.

1. The protocol supports user mobility. The additional security services, namely the user authentication process and card details encryption, require only the correct software to be loaded on the PC, an appropriately equipped ME and for there to exist a means to connect the MS to the PC. In particular, the new protocol has no key management overhead.

2. In the protocol, the PC is simply responsible for forwarding messages between the MS and the merchant server. Moreover, since the protocol does not involve storing any secrets on the PC, the risks in using untrusted PCs are minimised.

3. The protocol provides user authentication and card details confidentiality based on GSM data confidentiality. As a result, stolen card details can no longer be used to conduct a successful e-commerce transaction.

4. In the protocol, the merchant server has no access to the sensitive card details. As a result, the risks of storing unprotected card details in merchant servers are eliminated.

5. The protocol can work with a 'standard' GSM SIM. It simply requires an appropriate equipped ME and a user PC.

6. From the merchant point of view, the protocol will lessen fraudulent transactions and hence reduce the cost of 'card not present' chargebacks. The issuer can also reduce the cost of card frauds by using the protocol.

5.2 Disadvantages

The following disadvantages arise from use of the proposed GSM-based payment system.

1. Prior agreement is required between the issuer and mobile phone service provider so that the issuer can use the services of the AuC.

2. Issuers may be charged for the AuC services. This cost therefore has to be weighed against the cost of debit/credit card frauds. Of course, this is not a disadvantage for the GSM network provider, who may find this a useful additional revenue stream. Merchants may also be charged by banks in order to use the protocol. Again, the cost will have to be weighed against the cost of 'card not present' chargebacks.

6. RELATED WORK

There exist other GSM-based payment systems which we now briefly review.

- The payment scheme proposed by Claessens et al. [1] provides user authentication using GSM. However, unlike the scheme discussed above, it makes extensive use of SMS messaging.

- The GiSMo (G i(nternet) S M o(pen)) scheme was developed by Millicom International Cellular in 1999. In this scheme, consumers must first open an electronic wallet over the Internet and supply their mobile phone number. Every Internet transaction is then validated with a password sent over the mobile phone using an SMS message. The GiSMo project, however, ended in 2001.

- Mint[2] and Paybox[3] are both GSM-based payment systems. They too require consumers to first open an e-wallet. Transactions in the two protocols involve either making or receiving calls using the delegated mobile phone.

- The 3-D Secure Protocol has been developed by Visa [14, 15]. The protocol aims to provide cardholder authentication for merchants using a central server called the Access Control Service (ACS). The cardholder must enroll before using the service. When a transaction is to be made, he/she will be required to enter a Personal Account Number (PAN) in addition to other information used in a traditional purchase form. The merchant then has to contact the Visa Directory Server to determine whether authentication services are available for the cardholder. If such services are available,

[2]http://www.mint.nu
[3]http://www.paybox.co.uk

the response from the Visa Directory Server will instruct the merchant server how to contact the ACS of the associated issuer. The cardholder is then required to enter a password or PIN to authenticate him/herself to the ACS. The protocol can be extended to be used in mobile Internet devices such as a WAP phone [13], in which case the transaction flow remains similar to the one specified in [15].

Broadly speaking, the other GSM-based payment systems either use SMS messaging, require e-consumers to open an e-wallet, or require them to make or receive phone calls using a GSM phone. The protocol proposed here, however, does not use any such measures. It simply utilises the GSM data authentication session key, established during the subscriber identity authentication process. The Visa 3-D Secure Protocol is similar to the proposed protocol in the way that they both provide cardholder authentication. However, the Visa protocol requires both the Visa Directory Server and the ACS just to provide user authentication. The payment authorisation process then has to be performed separately. As as result, the proposed protocol appears to be considerably simpler and more powerful than the 3-D Secure scheme.

7. CONCLUSIONS

Today most e-commerce transactions are protected in a rather ad hoc way using SSL/TLS. This gives rise to threats partly because of the lack of user authentication, and partly because using SSL/TLS protects information only while it is transmitted. This latter property means that card details are stored unprotected at merchant servers, which gives rise to significant threats to their confidentiality.

In this paper, we have proposed the use of GSM data confidentiality to enhance e-commerce transaction processing security. The protocol provides user authentication and hence significantly reduces threats arising from misuse of misappropriated card details. It also eliminates the risk of storing card details in unencrypted form in merchant servers. The protocol works with a 'standard' GSM SIM and requires only an appropriately equipped Mobile Equipment and a user PC. It therefore imposes minimal overheads on the user, thus increasing the likelihood of successful use. The gains for the merchant in terms of reduced chargebacks and for the issuer in lessened card frauds also appear significant. The possibility of an increased revenue stream may also make the system attractive to GSM operators.

Acknowledgement

The authors would like to thank anonymous referees for valuable suggestions which have improved the paper.

8. REFERENCES

[1] J. Claessens, B. Preneel, and J. Vandewalle. Combining World Wide Web and wireless security. In B. De Decker, F. Piessens, J. Smits, and E. Van Herreweghen, editors, *Advances in Network and Distributed Systems Security*, Proceedings of IFIP TC11 WG11.4 First Annual Working Conference on Network Security, pages 153–171, Boston, 2001. Kluwer Academic Publishers.

[2] T. Dierks and C. Allen. *The TLS protocol version 1.0 — RFC 2246*. IETF, January 1999.

[3] ETSI. *Digital cellular telecommunications system (Phase 2+); Security aspects (GSM 02.09 version 8.0.1)*. European Telecommunications Standards Institution (ETSI), June 2001.

[4] ETSI. *Digital cellular telecommunications system (Phase 2+); Security related network functions (GSM 03.20 version 8.1.0)*. European Telecommunications Standards Institution (ETSI), July 2001.

[5] S. Garfinkel and G. Spafford. *Web Security & Commerce*. O'Reilly, 1997.

[6] A. Ghosh. *E-Commerce Security*. John Wiley and Sons, Inc., Third Avenue, New York, 1998.

[7] V. Hassler. *Security Fundamentals for E-commerce*. Artech House, 2001.

[8] D. O'Mahony, M. Peirce, and H. Tewari. *Electronic Payment System for E-commerce*. Architecture House Inc., 2001.

[9] SETCo. *Secure Electronic Transaction Specification — Books 1–4*. SETCo, May 1997.

[10] L. D. Stein. *Web Security: A step-by-step reference guide*. Addison Wesley, Reading Messachusetts, 1999.

[11] S. Thomas. *SSL and TLS Essentials — Securing the Web*. John Wiley and Sons, Inc., Third Avenue, New York, 2000.

[12] K. Vedder. GSM: Security, services, and the SIM. In B. Preneel and V. Rijmen, editors, *State of the Art in Applied Cryptography*, Lecture Notes in Computer Science 1528, pages 224–240. Springer-Verlag, 1998.

[13] Visa. *3-D Secure Protocol Specification: extension for mobile Internet devices version 1.0.1*. Visa International Service Association, November 2001.

[14] Visa. *3-D Secure Protocol Specification: system overview version 1.0.3*. Visa International Service Association, December 2001.

[15] Visa. *3-D Secure Protocol Specification: core functions version 1.0.2*. Visa International Service Association, July 2002.

[16] M. Walker and T. Wright. Security. In F. Hillebrand, editor, *GSM and UMTS: The Creation of Global Mobile Communication*, pages 385–406. John Wiley & Sons Ltd., 2002.