Available online at www.sciencedirect.com
SCIENCE DIRECT®
Electronic Commerce Research and Applications 4 (2005) 413–426
Electronic Commerce Research and Applications
www.elsevier.com/locate/ecra

Securing credit card transactions with one-time payment scheme ☆

Yingjiu Li [a,*], Xinwen Zhang [b]

[a] School of Information Systems, Singapore Management University, 80 Stamford Road, Singapore 178902
[b] Lab for Information Security Technology, George Mason University, Fairfax, VA 22030, USA

Received 5 October 2004; received in revised form 7 January 2005; accepted 1 June 2005
Available online 21 July 2005

Abstract

Traditional credit card payment is not secure against credit card frauds because an attacker can easily know a semi-secret credit card number that is repetitively used. Recently one-time transaction number has been proposed by some researchers and credit card companies to enhance the security in credit card payment. Following this idea, we present a practical security enhancement scheme for one-time credit card payment. In our scheme, a hash function is used in generation of one-time credit card numbers with a secret only known to the card holder and issuer. Compared with related work, our scheme places less burden on credit card issuers, and can be easily deployed in on-line or off-line payment scenarios. Analysis and simulation show that the time and space complexity is affordable to the card issuer with desired security features.

Keywords: Credit card transaction; Credit card fraud; Security

1. Introduction

Credit card frauds have caused millions of dollars loss each year and exposed the security weaknesses in traditional credit card processing system [2]. In such system, a customer (i.e., credit card holder) repetitively uses a fixed credit card number as well as personal identifying information in all transactions. Because this credit card number is "sticky", it is relatively easy for an attacker to steal

☆ A preliminary version of this paper appeared in the 14th International Workshop on Research Issues on Data Engineering: Web Services for E-Commerce and E-Government Applications, pages 40–47, Boston, USA, March 2004.

* Corresponding author. Tel.: +65 6828 0913; fax: +65 6828 0919.
E-mail addresses: yjli@smu.edu.sg (Y. Li), xzhang6@gmu.edu (X. Zhang).

1567-4223/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.elerap.2005.06.002

it with intention to commit illegal activities. Some common ways to commit credit card fraud include:

- *Shoulder surfing*: An attacker watches a customer from a nearby location as the customer punches in his credit card number. If the customer is giving his credit card number over the phone (e.g., to a hotel or car rental company), the attacker may listen to the conversation so as to get credit card information.
- *Dumpster diving*: An attacker goes through a customer's garbage cans or trash bins to obtain copies of credit card statements.
- *Packet intercepting*: An attacker sniffs some e-commerce packets during on-line credit card payment. In some cases, the attacker does not need to break down the possibly encrypted packets (e.g., over Secure Socket Layer), but fools the customer into thinking that he or she is visiting an intended site but actually the attacker's spoofing one.
- *Database stealing*: To encourage purchasing, many merchants (who provide services to customers) choose to store their customers' credit card information in online databases. Recent news reported that attackers could break into merchants' web sites and steal millions of credit card numbers [1].

Not only does the credit card fraud cause millions of dollars loss each year, but also causes significant worry among customers. According to a recent study conducted by Opinion Research Corporation, it causes more worry than the war in Iraq in terms its impacts on customers' awareness of security issues [5].

1.1. Evaluation criteria

Many efforts have been made so far to thwart credit card fraud. Before we look into them, we summarize the evaluation criteria that has been proposed by Shamir [10], Rubin and Wright [7] for secure credit card payment.

- *Ease of deployment*: The system should be easy to deploy in real-world settings. There should be few additional requirements on current infrastructure and communication protocols. Card issuers should be able to handle most of the deployment tasks, without placing unreasonable burdens on merchants and customers. Even for the card issuers, the additional requirements on equipment should be tolerable compared with the benefits obtained from security enhancement.
- *Ease of use*: The importance of ease of use cannot be overstated since it is the customers who use the payment system. The customers should feel convenient in all payment scenarios.
- *Security*: A secure payment system should address real security concerns thus overcome customers' psychological fear due to credit card fraud. The security of the system may not be perfect, but it should be good enough to protect customers in all payment scenarios.

1.2. Related work

A previous effort to thwarting credit card fraud led to the development of Secure Electronic Transactions (SET) protocol [9]. SET was designed to protect credit card information from various attacks in on-line environment. Unfortunately, SET never succeeded in the marketplace because of its high overhead and additional requirement of public key infrastructure (PKI).

Among hundreds of other solutions [4] that have been proposed (most of them failed or remain untested), credit card payment over Secure Socket Layer (SSL) is the only one that is widely used in e-commerce nowadays. SSL [3] provides encryption channel to transmit credit card numbers; it also provides server authentication to identify merchants. While a flawless implementation of SSL thwarts packet intercepting, it has no effect on other credit card frauds such as shoulder surfing, dumpster diving and database stealing.

In terms of those evaluation criteria mentioned in Section 1.1, SET addresses the security concerns but fails to satisfy the requirements on ease of deployment and ease of use. On the other hand, SSL solution satisfies the requirements on ease of deployment and ease of use; however, it does not address the security concerns in all scenarios.