International Journal of Theoretical and Applied Science **3**(2): 79-87(2011) **ISSN No. (Print) : 0975-1718**
ISSN No. (Online) : 2249-3247

On Certain Attacks and Privacy — Protecting Coupon System: A Model

Santosh Kumar Yadav and Manoj Kumar

JJT University, (RJ)

(Received 17 August, 2011, Accepted 14 September, 2011)

ABSTRACT : Phishing is an attack in which victims are lured by official looking email to a fraudulent web-site that appears to be that of a legitimate service provide. The email also provides victims with a convincing reason to log-on to the site. If users are fooled into logging-on, then the attacker is provided with the victims' authentication information for the legitimate service provider, often along with personal information, such as their credit-card data, checking account information or social security data. Successful phishing attacks can result not only in identity and asset theft, but also in more subtle attacks that need not be directly harmful to the victim but which have negative consequences for society.

A coupon represents the right to claim some service which is typically offered by vendors. In practice, issuing bundled multi-coupons is more efficient than issuing single coupons separately. The diversity of interests of the parties involved in a coupon system demands additional security properties beyond the common requirements (e.g. unforgeability). Customers wish to preserve their privacy when using the multi-coupon bundle and to prevent vendors from profiling. Vendors are interested in establishing a long-term customer relationship and not to subsidise one-time customers, since coupons are cheaper than the regular price. We propose a secure multi-coupon system that allows users to redeem a predefined number of single coupons from the same multi-coupon. The system provides unlinkability and also hides the number of remaining coupons of a multi-coupon from the vendor. A method used in the coupon system might be of independent interest. It proves knowledge of a signature on a message tuple of which a single message can be revealed while the remaining elements of the tuple, the index of the revealed message, as well as the signature remain hidden.

Keywords: Phishing Attacks, Silver-bullet Technology, MITM Blind Signature, Coupon System.

I. INTRODUCTION

Phishing emails are now so convincing that even experts cannot tell what is or is not genuine; though one of my own quiz answering errors resulted from failing to believe that genuine marketeers could possibly be so clueless! Thus I believe that education of end users will be almost entirely ineffective and education of marketing departments- to remove "click on this" (and HTML generally) from the genuine material - is going to take some time. Providing end users with one-time passwords (pads of single-use numbers, SecurID tokens, PINs sent by mobile phone) can ensure that phishing only works when there is a real-time, Man-in-the-Middle (MITM), attack. This will immediately deter the bad guys if their technical expertise runs solely to copying websites. However, formal analysis of online banking protocols shows that only a handful of the "bag of bits" being passed around can be considered to be authenticated- and so a MITM can, unhindered, steal whatever they wish.

When a client attempts to interact with an online service provider that performs any form of financial transaction, the service provider requires the client to authenticate itself. This is normally done by having the client provide a user name and password that were previously agreed upon, through some procedure, the first time the client attempted to use the services provided by the provider. Asymmetrically, the client does not ask the provider for the same form of authentication. That is, the customer of the bank does not ask the web-page to somehow prove that it is really the bank's web-page. This asymmetry seems to come mostly from an attempt to port security models from the physical to the digital world: I would never expect a physical bank branch to authenticate itself to me through any form other than its branding. However, that is not to say customers don't implicitly authenticate their bank-branches, they do! However, it is a rather implicit authentication that is based on the use of branding and law-enforcement by the banks. Unfortunately, many of the security assumptions that hold in the physical world do not hold in the digital world: the costs of setting up an authentic looking but fraudulent web-page are low; the pay-off for successful phishing attacks is high; and digital law enforcement is weak to non-existent in the digital realm and so the risks are minimal. This makes phishing an attractive type of fraud, and has led to its growing popularity.

Today, coupons appear to be useful means for vendors to attract the attention of potential customers. Usually, coupons give the customer a financial incentive to purchase at a specific vendor. The purpose of coupons is many-fold. for instance, they can be used to draw the attention of customers to a newly opened shop or to prevent customers

from buying at a competitor's shop. Of course, coupons can also be purchased by kind of a coupon.

In general, a coupon is a representation of the right to claim some good or service, usually from the party that issued the coupon. The types of coupons mentioned before can, in general, be redeemed only once, i.e., the coupon is invalidated after the service or good has been claimed. However, there are also coupons which can be redeemed more than once, such as a coupon book of a movie theater, where customers pay, e.g., for 9 movies and are entitled to see 10. We call such coupons multi-coupons. In this paper, we are particularly interested in this type of coupons.

Typically, a real-world multi-coupon of value m is devalued by crossing out some field or by detaching a part of it. Offering such coupons can be beneficial for the issuing party, e.g., a movie theater. First, customers pay in advance for services or goods they have not claimed yet. Second, they are locked-in by the issuer/vendor, i.e., they are unlikely to witch to another vendor to purchase the same or similar service or good as long as they have not redeemed all their coupons [8]. Hence, multi-coupons can also be seen as a kind of loyalty program since they are specific to some vendor and induce loyalty, at least, as long as the customer has coupons left to spend.

Clearly, vendors are interested in creating loyalty and hence, it is likely that we are going to see such coupon systems in the Internet, too. In fact, introducing such a coupon system might be even more valuable to Internet vendors than to their real world counterparts. Since, from the customers' viewpoint, a priori all vendors, offering a certain good or service, look alike and can be reached as easily as their competitors.

II. PHISHING ATTACKS

A. Old Phishing' Hole

Professional studies that have attempted to estimate the direct losses due to phishing in 2004 have come up with widely varying figures: from $150-million to $ 2.4-billion U.S. dollars. However, all the studies agree that the costs will continue to rise in the foreseeable future unless something is done to educate users and/or technologies are introduced to defeat or limit such attacks. Further, these estimates measure only the direct costs, and do attempt to measure the indirect costs that result from the loss of consumer confidence in the Internet infrastructure and all o the services it can be used to provide. Our panel will look at a broad number of issues relating to the past, present and future of phishing, in order to better understand this growing problem. We will address topics that include the notion that phishing is a special case of "web-spoofing", an attack that was predicted and researched academically as early as 1996. We will look at the mutual progression of the research and practice of such attacks, and what we can learn from both. We will discuss the fact that phishing is currently a problem, and look at what information consumers are being given to mitigate their risk of exposure; we'll ask if the advice is practical and effective. We will see how the percentage of successful phishing attacks could dramatically increase if phishing attacks to make use of contextual information about their victims. It will argued that such attacks are easily automated, begging the question of how long it will take for such context sensitive attacks to appear in the wild. We will see that phishing-graphs can be used not only to model phishing attacks, but also to quantify the feasibility and economic costs of attacks. We will discuss the issue of mutual authentication, and how it relates to phishing attacks. It will be argued that easy to use mutual authentication protocols could mitigate many of the risks of phishing and we will discuss one such protocol. Finally, we will deliberate on the likelihood of the advent of a silver-bullet technology that will solve all of our phishing problems.

B. Preventing Phishing Attacks

We model an attack by a phishing graph in which nodes correspond to knowledge or access rights, and (directed) edges correspond to means of obtaining information or access rights from already possessed information or access rights - whether this involves interaction with the victim or not. Edges may also be associated with probabilities, costs, or other measures of the hardness of traversing the graph. This allows us to quantify the effort of traversing a graph from some starting node (corresponding to publicly available information) to a target node that corresponds to access to a resource of the attacker's choice. We discuss how to perform economic analysis on the viability of attacks. A quantification of the economical viability of various attacks allows a pinpointing of weak links for which improved security mechanisms would improve overall system security. This is a particularly threatening attack in that it is likely to be successful not only against the most gullible computer users (as is supported by experimental results we present). A context aware attack is mounted using messages that somehow-from their context -are expected (or even welcomed) by the victim. To draw a parallel from the physical world, most current phishing attacks can be described as somebody who knocks on your door and says you have a problem with your phone, and that if you let him in, he will repair it. A context aware phishing attack, on the other hand, can be described by somebody who first cuts your phone lines as they enter your home, waits for you to contact the phone company to ask them to come and fix the problem- and then knocks on your door and says he is from the phone company. We can see that observing or manipulating the context allows an attacker to make his victim lower his guards. As a more technical example, we show how to obtain PayPal passwords from eBay users that do not take unusual measures particularly intended to avoid this attack.

C. The Phish detection of Lure

In order to reduce the ability of phishes to launch successful attacks, we suggest that users request authentication from their service providers. In other words, we suggest that the client and service provider engage in mutual authentication. While such authentication is easily achievable with public-key cryptography and certificates, this solution is not appealing due to the historical difficulty users have had in understanding these concepts: currently many users automatically accept most certificates that are brought to their attention by web-browsers, regardless of their validity or origin.

We will discuss a protocol for mutual authentication that relies solely on a client being able to remember a password to authenticate him or herself to the service provider, and the ability to recognize- and not recall, as in the case of a password- a unique series of images (or other forms of stimuli, such as sound and touch) corresponding to the appropriate service provider. The client only needs to be educated to realize that if his or her appropriate sequence of images does not appear, then the site is not legitimate and should not be used, nor should any personal information be provided to it. Further, the protocol has the property that it is secure against man-in-the-middle attacks in the random-oracle model.

D. Phishing in Summit

Insisting on SSL (https) connections will prevent the use of random URLs for phishing websites and bring the focus back to control of the DNS. However, once the second level (fakebankname.com) is secured then the attackers will just move down a level (to bankname.plausible-second-world.com). I predict a lot of wasteful activity before the nature of DNS delegation is fully understood [10].

Insisting on client certificates prevents MITM attacks, but also stops me paying me gas bill from a holiday cybercafe- which is bad for business. But why do I need the same authority to pay the bill as the change the name of the gas company? A range of authentication systems is needed, chosen as the risk varies. The banks could learn from the activity monitoring systems of the credit card companies, and ensure that extra authentication is seldom necessary or onerous.

III. COUPON SYSTEM

At first, introducing a coupon system looks like a win-win situation, since both parties seem to benefit from such a coupon system. Vendors have a means to create a loyal customer base and customers value the financial benefit provided by coupons. However, since a customer normally redeems her coupons in different transactions, a multi-coupon can be used as a means to link transactions, and thus, to allow a vendor to create a record of the customer's past purchases. Such customer information might be exploited for data mining, to infer new customer data, customer profiling, promotion of new products, price discrimination, etc. Thus, if through usage of the coupon system customers expect a misuse of their personal data, *e.g.* by using it to create profiles for price discrimination, they are more likely to decline the coupon system. According to privacy is a concern to Internet users, especially when it comes to electronic commerce scenarios. Hence, a prudent vendor should take these concerns into account when planning to offer a coupon system [16, 17].

In order to rule out privacy concerns of customers from the start, vendors might want to introduce a coupon system that does not infringe their customers' privacy. Thus, a coupon should disclose as little information as possible. For instance, a multi-coupon should only give vendors an indication that it is still valid, *i.e.*, that at least one coupon is not spent, instead of disclosing the number of unspent coupons. Such a property could be useful in sensitive areas, *e.g.*, in health care scenarios, where a multi-coupon can be used as a prescription for a certain number of doses of some medicine. In this case, the pharmacist would deduct a single coupon from the multi-coupon and may only detect if the prescription has been used up. Also in welfare, paper-based checks or food stamps program with electronic benefits and debit cards replace their paper-based food stamp program with electronic benefits and debit cards. However, this electronic program does not protect the privacy of recipients, since the cards are processed similar to ordinary debit cards.

For vendors, in addition to common security requirements such as unforgeability, there are other requirements which are specific to a coupon system. As mentioned before, a vendor's driving reason for offering a coupon system is to establish a long term relationship with customers. However, customers may be interested in sharing a multi-coupon, *i.e.*, each customer obtains and redeems a fraction of the coupons in the multi-coupon. Moreover, this behaviour allows them, *e.g.*, to sell coupons on an individual basis for a cheaper price, *e.g.*, to one-time customers who otherwise would have purchased full-price services or goods. Thus, ideally, vendors wish to prevent customers from splitting their coupons.

The coupon system proposed here can be viewed as a digital counterpart to the real-world multi-coupon with non-detachable coupons, as mentioned before. In our coupon system, a multi-coupon M is a signature on a tuple X where $X = (x_1, ..., x_m)$. In the system specification, we denote a set of coupons by M and a single coupon by $x \in \{x_1, ..., x_m\}$.

In the coupon issue phase, a user first convinces a vendor that she knows X without revealing the values of X. then, the verifier issues the coupon M by "blindly" signing X, *i.e.*, $M := Sign(X)$, and sending M to the user. Here we make use of the Camenisch and Lysyanskaya (CL) signature scheme.

IV. MODEL REQUIREMENTS

The coupon system considered here involves mainly two parties, a customer *Y* (user) and a vendor ς. The sysem itself is comprised of an issue protocol and a redeem protocol which both are carried out between *Y* and ς. The output of the issue protocol is a multi-coupon *M* for *Y* and the result of the redeem protocol is a spent single coupon for ς and a multi-coupon devalued by one single coupon for *Y*. Next, we state the main security requirements for the involved parties.

In the following, we will use the notation *M N* the IEEE 802.11 MAC protocol allows the nodes to entre idle and sleep modes, which also consume energy. Since our analysis is a comparison of routing protocols independent of the MAC layer operation, we set the energies consumed in these modes to zero. We emphasize that we consider only the energy consumed in the RF interface during transmission and reception. It is remarked that mobile nodes also consumer energy for microprocessor operations, cache access, etc which has been neglected.

Unforgeability: It must be infeasible to create new multi-coupons, to increase the number of unspent coupons, or to reset the number of spent coupons.

Double-spending detection: A vendor must be able to detect attempts of redeeming 'old' coupons that have already been redeemed. This means, given two runs of the redeem protocol, where a single coupon *x* is deducted from multi-coupon *M* and *y* is deducted from *N*, the vendor must be able to decide if $x = y$.

Redemption limitation: An *m*-redeemable coupon *M* may not be accepted by the vendor more than *m* times.

Protection against splitting: A coalition of customers *Y*, should not be able to split an *m*-redeemable multi-coupon *M* into (disjoint) s_i-redeemable shares M_i with $\Sigma_i s_i < m$ such that M_i can only be redeemed by customer Y_i and none of the other customers $Y_j, j \neq i,$ or a subset of them is able to redeem the share M_i or a part of it. We call this property strong protection against splitting.

A weaker form of this property is all-or-nothing-sharing. This means that splitting is possible, however, only if customers trust each other not to spent (part of) the other's share M_i. Another way of putting this is to say that sharing *M* means sharing all *m* single coupons. We call this weak protection against splitting.

Unlinkability: It must be infeasible for vendors to link protocol runs of honest users. For this, we have to consider linking a run of an issue protocol to runs of corresponding redeem protocols and linking of any two redeem protocol runs.

(1) *Issue vs. redeem:* Given a run of the issue protocol with output a multi-coupon *M* and given a redeem protocol run with output a devalued multi-coupon *N*, the vendor must not be able to decide if $M \rightarrow N$.

(2) redeem vs. redeem: Given two runs of the redeem protocol with output two multi-coupons *M, N*. The vendor must not be able to decide if $M \rightarrow N$ or $N \rightarrow M$, *i.e.*, he cannot tell if *M* and *N* are related or unrelated.

Minimum Disclosure: As a result of a redeem protocol run, the vendor may only learn of the single coupon being redeemed but not the number of remaining coupons. This already follows from the unlinkability requirement but we make it explicit here, nevertheless.

A. Commitment Scheme

A commitment scheme is a two-party protocol between a committer *C* and a receiver *R*. In general, the scheme included a *Commit* process and an Open process. In the first process, *C* computers a commitment C_x with a message *x*, such that *x* cannot be changed without changing C_x. *C* then gives C_x to *R* keeps *x* secret. In the second process, C opens C_x by revealing *x*.

The commitment scheme we employ is due to Damgård and Fujisaki (DF) which is a generalization of the Fujisaki-Okamoto scheme. We skip the basic DF scheme for committing to a single value x and proceed to the scheme where the commitment is to a message tuple $(x_1, x_2, ..., x_m)$ [11].

Let $\langle h \rangle$ denote the group generated by $h \in_R QR_n$ and let $g_1, g_2, ..., g_m \in \langle h \rangle$. On secret input $X: = (x_1, x_2, ..., x_m)$, where $x_i \in (0, 2^{l_x})$ and public input $PK: = (g_1,..., g_m, h, n)$, the commitment is $C_X: = \prod_{i=1}^{m} g_i^{x_i} h^{\tau_x}$, where $\tau_X \in_R Z_n$ are chosen at random.

B. Signature Scheme

The signature scheme stated in the following is a variant of the Camenisch and Lysyanskaya (CL) signature scheme [9] for signing a block of messages which was used before. The signed message is denoted by a tuple $X: = (x_1, x_2, ..., x_m)$ where $x_i \in [0, 2^{l_x}), i = 1, ..., x_m)$ and l_x is a parameter for the message length.

Key Generation: Set the modulus *n* as described before. Choose $a_1, a_2, ..., a_m, b, c \in R\ QR_n$ and output a public key $PK: = (A, b, c, n)$ where $A: = (a_1, a_2, ..., a_m)$ and a secret key $SK: = (p, q, n)$.

Signing: On input $X: = (x_1, x_2, ..., x_m)$, choose a random prime number $e \in R[2^{l_e-1}, 2^{l_e-1}2^{l'_e-1}]$ and a random number *s* of length l_s, where l'_e is the length of the interval that the *e* values are chosen from, l_e is the length of the *e* values, l_s is the length of the *s* value. Both the values l_e and l_s are dependent on a security parameter l_ϕ, for the details see [5]. The resulting signature the tuple (u, e, s) such $C \equiv u^e a_1^{x_1} ... a_m^{x_m} b_1^s$ uea1xthat We denote this algorithm by: $ind \leftarrow Verify(A, b, c, n)(X, u, e, s)$, where $ind \in$ {accept, reject}.

Remark 1. The CL signature scheme is separable, *i.e.*, the signature (u, e, s) on X is also the signature on a sub-tuple of X if we change the public key accordingly. In the following, we use the notation $X\backslash(x_j)$ to denote the sub-tuple of X which is comprised of all of X's components but its j^{th} one, *i.e.*, $X\backslash(x_j) = (x_1, ..., x_{j-1}, x_{j+1}, ..., x_m)$. Now, the signature on X under the public key (A, b, c, n) is the same as the signature on $X\backslash(x_j)$ under the public key $[A\backslash(a_j), b, c/a_j^{x_j}, n]$ *i.e.*, $Sign_{(A,\ b,\ c,\ n,\ p)}(X) = (u, e, s) = Sign_{[A\backslash(a_j), b, c/a_j^{x_j},\ n,\ p]}[X\backslash(x_j)]$. This holds for any sub-tuple Y of X. We will use this property in our coupon system to redeem a single coupon from a multiple set of coupons.

Remark 2. As discovered the CL signature scheme has the property of randomisation, *i.e.*, the signature (u, e, s) can be randomised to ($T = ub^w$, e, $s^* = s - ew$) with an arbitrary w. From a verifier's point of view, (T, e, s^*) and (u, e, s) are equivalent since they both are signatures on X. This property benefits our scheme because a proof of knowledge of (T, e, s^*) can be done more efficiently than proving knowledge of (v, e, s) in an anonymous manner.

C. Relations Between Committed Numbers

PoKRep: A prove Π proves knowledge of a discrete logarithm representation (DL-Rep) modulo a composite to a verifier ς. Common inputs are a description of group Γ, $PK := (g_1, ..., g_m, h)$ with $h, g_i \in \Gamma$ and a commitment C. By this protocol, Π convinces ς of knowledge of $X := (x_1, ..., x_m)$ such that $C = \prod_{i=1}^{m} g_i^{x_i} h^{\tau}$ [12].

PoKEqRep: A proper Π proves to a verifier ς knowledge of equality of representations of elements from possibly different groups Γ_1, Γ_2. Common inputs are $PK_1 := (g_1, ..., g_m, h), g_i, h \in \Gamma_1, PK_2 := (g'_1, ..., g'_m, h), g'_i, h' \in \Gamma_2$, commitments $C_1 \in \Gamma_1$ and $C_2 \in \Gamma_2$. By running the protocol, Π convinces ς of knowledge of $X := (x_1, \ldots, x_m)$ such that $C_1 = \prod_{i=1}^{m} g_i^{x_i} h^{r_1}$ and $C_2 = \prod_{i=1}^{m} g'_i{}^{x_i} h^{r_2}$, *i.e.*, $\log_{g_i}(g'_i{}^{x_i}) = \log_{g'_i}(g'_i{}^{x_i})$ ($i = 1, ... m$) [15].

PoKInt: A proper Π proves to a verifier ς knowledge of x and r such that $C = g^x h^r$ and $a \le x \le b$. Common inputs are parameters (g, h, n), the commitment C, and the integers a, b. We use a sraightforward extension to the basic protocol, such that the proved knowledge is two tuples, instead of two values, and the interval membership of each component from a tuple, instead of one value. Within this extension, we denote $G := (g_1, g_2, ..., g_m)$, $H := (h_1, h_2, ..., h_l)$; $X := (x_1, x_2, ..., x_m)$, $R := (r_1, r_2, ..., r_1$ and $1C := \prod_{i=1}^{m} g_i^{x_i} \prod_{j=1}^{l} h_j^{r_j}$ [12]. By running the protocol, Π proves to ς knowledge of X and R, and the interval membership, $a \le x_i \le b$.

PoKOr: A prover Π proves to a verifier ς an or statement of a commitment C, such that $C := (C_1, ..., C_m)$, where $C_i := \prod_{j \in \alpha_i} g_j^{x_{ij}} h^{r_i}$ and $\alpha_i \subseteq \{1,...,m\}$ and Π knows at least one tuple $\{x_{ij} \mid j \in \alpha_i\}$ for some undisclosed i. We denote the or statement as $\vee_{i=1}^{m} C_i = \prod_{j \in \alpha_i} g_j^{x_{ij}} h^{r_i}$. Common inputs are C and parameters (G, n) where $G := (g_1, ..., g_m)$. By running the protocol, Π proves to ς knowledge of $\{x_{ij} \mid j \in \propto_i)$ without revealing the values x_{ij} and i. A number of mechanisms for proving the "or" statement.

PoK: Sometime, we need to carry out two or more of the above protocols simultaneously *e.g.*, when responses to challenges have to be used in more than one validity check of the verifier to prove intermingled relations among commitments. Instead of giving concrete constructions of these protocols each time, we just describe their aim, *i.e.*, what the verifier wants to prove. For this we apply the notation use *e.g.* for instance, the following expression

$$ind_V \leftarrow P \circ K\{(\alpha, \beta):$$

$$C = g^{\alpha} h^{\beta} \wedge D = \tilde{g}^{\alpha} \hat{h}^{\beta} \wedge 0 \le \alpha < 2^k\}$$

means that knowledge of α and β is proven such that $C = g^{\alpha} h^{\beta}$ and $D = \hat{g}^{a} \hat{h}^{\beta}$ holds and α lies in the integer interval $[0, 2^k)$ [11].

D. Blind Signature and Signature Proof:

BlindSign: Next we state a secure protocol for signing blinded tuple, shown in Fig. 1. In this protocol, a user Y obtains a signature from the signer Σ on a tuple $X := (x_1, x_2, ..., x_m)$ without revealing X to Σ. We assume that Σ has the public key $PK := (A, b, c, n)$, the secret key $SK := p$ and public length parameters and l_n, l_x, l_e, l'_e, l_s, and l_ϕ which are parameters controlling the statistical zero-knowledge property of the employed $P \circ K$. Y's input to the protocol is the message $X := (x_1, ..., x_m)$ for which Y wants to obtain a signature.

Among the first steps, Y computes the valued $D := \prod_{i=1}^{m} a_i^{x} b^{s'}$ and sends is to Σ. The next steps assure to Σ that Y indeed knows the discrete logarithms of D with respect to the basis (a_1, ..., a_m, b) respectively, and the interval of the committed values in D are selected correctly.

If all proofs are accepted, Σ chooses a prime e and computes $V := [c/(Db^{s''})]^{1/e} = \left[c / \left(\prod_{i=1}^{m} a_i^{x} b^{(s'+s'')} \right) \right]^{1/e}$. At the end, ς sends the resulting tuple (v, e, s'') to Y. Finally, Y sets $s := (s' + s'')$ and obtains (v, e, s) as the desired signature on X. We denote this protocol for blindly signing a tuple by (v, e, s) $BlindSign \leftarrow_{(PK)}(X)$ [13].

User Y	***Signer Σ***	
Common Input:	Verification key $PK := (A, b, c, n)$, $A := (a_1, \ldots, a_m)$	
	Length parameters l_x, l_e, l'_e, l_s, l_n, l_ϕ	
User's Inputs:	Message $X := (x_1, \ldots, x_m)$	
Signer's Input:	Factorisation of n : (p, q, n)	
choose $s' \in_R \{0,1\}^{l_n + l_\phi}$		
compute $D := \prod_{i=1}^{m} a_i^{x_i} b^{s'}$		
	Run *PoK* $\{(\xi_1, \ldots, \xi_m, \sigma) : D = \pm a_1^{\xi_1} \ldots a_m^{\xi_m} b^{\sigma} \wedge$	
	for $i = 1, \ldots, m$: $\xi_i \in \{0,1\}^{l_x + l_\phi + 2} \wedge$	
	$\sigma \in \{0,1\}^{l_n + l_\phi + 2}\} \rightarrow ind_S$	
		check $inds_\Sigma \stackrel{?}{=} accept$
		choose $\hat{s} \in_R \{0,1\}^{l_s - 1}$
		compute $s'' := \hat{S} + 2^{l_{s-1}}$
		chose $e \in_R (2^{l_e - 1}, 2^{e-1} + 2^{l_e - 1})$
compute s := s' + s'';	$\underline{(v, e, s'')}$	compute $v := [c/(Db^{s''})]^{1/e}$
check *Verify*(A, b, n) $(X, v, e, s) \stackrel{?}{=} accept$		
$\left[i.e.\ c = v^e b^s \prod_{i=1}^{m} a_x^{x_i} \right]$		
output (v, e, s)		

Fig. 1 Protocol for blindly signing a tuple : Blindsign.

PoKSign: The next protocol, shown in Fig. 2, is a zero-knowledge proof of a signature created in the *BlindSign* protocol. The idea of this protocol is to convince a verifier V that a prover Π holds a valid signature (v, e, s) on X satisfying $c \equiv v^e a_1^{x_1} \ldots a_m^{x_m} b^s$ without ς learning anything of the signature but its validity. The common inputs are the same as in the BlindSign protocol. Π's secret input is the message X and the corresponding signature (v, e, s).

The protocol works as follows: Π first randomises the signature components, v and s, by choosing w at random and computing $T := vb^w$ and $s^* = s - ew$. Π sends only T to ς. Then, Π proves to ς his knowledge specified in *PoK*.

As discussed in ς's view, (T, e, s^*) is a valid signature on X, as is (v, e, s). The difference between them is that we are allowed to reveal the value T to ς, but not the value v, because T is different in every proof. Therefore to prove the signature with $c \equiv v^e \prod_{i=1}^{m} a_x^{x_i} b^s$ becomes one with $c \equiv T^e \prod_{i=1}^{m} a_x^{x_i} b^{s''}$. Clearly, to prove the second equation is much simpler then the first one. *PoK* here performs the following three simple proofs in one go: (1) *PoKRep*: to prove knowledge of discrete logarithms of $c\left(\equiv T^e \prod_{i=1}^{m} a_x^{x_i} b^{s''}\right)$ with respect to the basis $(T, a_1, \ldots, a_m, b)$ respectively; (2) *PoKInt*: to prove the value $x_1, \ldots, x_m$ are within a right bound, *i.e.*, for $i = 1, \ldots, m$: $x_i \in \{0,1\}^{l_s + l_\phi + 2}$; (3) *PoKInt*: to prove the value e is also within a right bound, *i.e.*, is also within a right bound, *i.e.*, $(e - 2^{t_e}) \in \{0,1\}^{l'_e + l_\phi + 1}$ [14].

Prover Π	**Verifier** ς
Common Input:	Verification key $PK := (A, b, c, n)$, $A = (a_1, a_2, ..., a_m)$
	Length parameters l_x, l_e, l'_e, l_ϕ
Prover's Input:	Message $X := (x_1, ..., x_m)$, Signature (v, e, s)
choose $w \in_R \{0,1\}^{l_n + l_\phi}$	
compute $T := vb^w$;	$\underrightarrow{r}$
	Run PoK $\{(\xi_1,...,\xi_m,\sigma,\in) : c = \pm T^e a_1^{\in_1} ... a_m^{\in_m} b^\sigma \wedge$
	for $i = 1, ..., m$: $\xi_i \in \{0,1\}^{l_s + l_\phi + 2} \wedge$
	$(\in -2^{l_e}) \in \{0,1\}^{l'_e + l_\phi + 1}\} \rightarrow ind_\vee$
	check $ind_\varsigma \stackrel{?}{=} accept$

Fig. 2. Protocol for proving knowledge of a signature : *PoKSign*.

V. CONSTRUCTION

In this section we propose a concrete scheme for a coupon system that allows issuance and redemption of multi-coupons. The scheme is comprised of two protocols, Issue and Redeem, which are carried out between a used U and a vendor *Y* and an Initialisation algorithm.

Initialisation. ς initialises the system by generating a key pair $PK = (A, b, c, n)$ where $A = (a_1, a_2, ... , a_m)$ and $SK = (p, q, n)$. The vendor keeps *SK* secret and published *PK* with length parameters l_x, l_e, l'_e, l_n, l_s and the security parameter l_ϕ.

Issue. In the issue protocol, *Y* chooses serial numbers $x_i \in_R \{1,...,2^{l_s} - 1\}$ $(i = 1, ..., m)$ and sets $X := (x_1, ..., x_m)$. Then *Y* runs $(v, e, s) \leftarrow Blindsign_{(PK)}(X)$ with ς to obtain a blind CL signature (v, e, s) on *X*. The tuple $M := (X, v, e, s)$ will act as the user's multi-coupon.

Redeem. In the redeem protocol, *Y* (randomly) chooses an unspent coupon x_j from the tuple *X*, sets $x := x_j$ and [18]. The value *x* then becomes a common input to the random protocol. Next *Y* proves to ς that she is in possession of a valid multi-coupon (ς's signature on *X*) containing *x* without revealing the signature itself.

VI. PROPERTIES

We will analyse the security of the system assuming that the strong RSA assumption holds.

Unforgeability. The property of unforgeability of our coupon system follows from the unforgeability of the CL signature scheme. As described in the previous section, a set of multi-coupons is a single CL signature on a block messages.

Resetting the number of spent coupons requires to change some component in the tuple *X*, *e.g.*, replacing a redeemed coupon xi with since the vendor stores each spent single coupon x_i. However, replacing x_i by yielding tuple *X**, must be done such that $Sign_{(.)} = Sign_{(.)}(X^*)$. Suppose the latter can be done. Then, we get $v^e \prod_{i=1}^{m} a_x^{x_i} b^s \equiv v^e \prod_{j=1}^{i=1} a_j^{x_j} a_j x_i'' \prod_{j=i+1}^{m} a_j^{x_j b^s}$. Dividing by the right hand side yields $a_i^{x_i - x''_i} \equiv 1 \bmod n$. Since $x_i \neq x_i^*$ it must be the case that $x_i - x_i^* = .ord(Z_n)$ [20].

Now, choose any *e* such that $1 < e < (x_i - x_i^*)$ and $gcd\ (e, x_i - x_i^*) = 1$. By the extended Euclidean algorithm we can find *d* such that $ed + (x_i - x_i^*)t = 1$. Using this, we can compute e^{th} roots in Z_n. For this, let *u* be any value from Z_n^* and compute $w := u^d$. Since $u \equiv u^{eu + (x_i - x_i^*)t} \equiv u^{ed}\ u^{\alpha . ord(Z_n)t} \equiv (u^d)^e \equiv w^e \bmod n$, the value *w* is an e^{th} root of *u*. This means we would have found a way to break the strong RSA assumption [22]. Since this is assumed to be infeasible x_i cannot be replaced by $x_i^* \equiv x_i$ without changing the signature (v, e, s).

Double-spending detection. If cheating user tries to redeem an already spent single coupon x_i, she will be caught at the end of the redeem protocol, since the coupon to be redeemed must be disclosed and, thus, can easily be looked up in the vendor's database.

Redemption limitation. An *m*-redeemable coupon *M* cannot be redeemed more than *m* times (without the vendor's consent) [21]. Each multi-coupon *M* contains a signature on an m-tuple $(x_1, ..., x_m)$ of single coupons and in each run of the issue protocol a single coupon x_i is disclosed. Thus, after *m* honest runs using the same *M*, all x_i will be disclosed to the vendor. As argued under unforgeability and double-spending detection, already redeemed xi cannot be replaced by fresh x_i^* and any attempt to 'reuse' an already disclosed x_i will be caught by the double-spending check.

Weak protection against splitting. Suppose that two user Y_1 and Y_2 want to share multi-coupon $M := (X, v, e, s)$ such that Y_1 receives single coupons $i < j$. To achieve splitting, they have to find a way to make sure that Y_1 is able to redeem all $x_j \in X_1$ while not being able to redeem any coupon $x_j' \in X_2$ and analogously for Y_2. However, in the redeem protocol it is necessary to prove knowledge of the DLRep of C_X, which is X. Since proving knowledge of X while knowing only X_1 or X_2 would violate the soundness of the employed proof of knowledge *PoKRep* and hence violate the strong RSA assumption, this is believed to be infeasible. Again, the missing part of X, either X_1 or X_2, cannot be replaced by 'fake' coupons $X'_{1/2}$ since this violates the unforgeability property of the coupon system. Hence X cannot be split and can only be shared if both Y_1 and Y_2 have full knowledge of X which comes down to all-or-nothing sharing.

Unlinkability. For unlinkability, we have to consider two cases, unlinkability between issue and redeem protocol runs and between executions of the issue protocol.

(1) *Issue vs. redeem:* The issue protocol is identical to the protocol *BlindSign* and hence, the vendor ς, acting as singer, does not learn anything about the message X.

(2) *Redeem vs. redeem:* The redeem protocol mainly consists of the *PoKSign* protocol which only employs zero-knowledge proofs and statistically hiding commitments and hence, is unlinkable. However, in the redeem protocol the coupon value x is given to the vendor ς, *i.e.*, the verifier. In the following we sketch that ς cannot use this information to infer other information that helps him to link redemptions of single coupons ?

To see this, let r be the transcript of the redeem protocol where x is released. Since all proofs of knowledge applied in the redeem protocol perform challenge-response protocols, there will be some values u, containing x, which were formed by the user in response to challenges t chosen by ς. The general form of a response is $u = ty + r$, where t is ς's challenge, y is some committed value of which knowledge is proven, and r is a witness randomly chosen by the user. However, since x's index is not revealed (due to PoKOr) every response $u_i (i = 1, ..., m)$ is equally likely to contain x [24].

Now, if ς guesses x's index, say j, he would only learn the value r_j from the response u_j. However, this reveals no information of any other response u_i, $i \neq j$, from r, since for any value xi, contained in the response u_i, the witness r_i is randomly and uniformly chosen anew for each u_i. Hence, from ς's point of view u_i is a random value and may contain any value x^*_i and, thus, x_i is (still) statistically hidden in u_i.

Minimum Disclosure. A further consequence of the unlinkability of transactions in the coupon system, and due to the fact that no counter value is sent in any protocol, the number of unspent coupons cannot be inferred any redeem protocol run.

VII. CONCLUSION AND FUTURE TRENDS

Society may need a general solution to online security, but the banks only have to persuade the bad guys to move on to more attractive targets. However, the fixes must not be introduced one by one, allowing each to be overcome individually. What's need is a 'Kilimanjaro effect', where the security suddenly dominates the landscape and it will always seem to be a long way to the summit.

The coupon system presented in this work allows vendors to issue multi-coupons to their customers, where each single coupon of such a multi-coupon can be redeemed at the vendor's in exchange for some good, *e.g.*, an MP3 file, or some service, *e.g.*, access to commercial online articles of a newspaper. Issuing coupons is advantageous to vendors since coupons effectively retain customers as long as they have coupons left to spent. However, multi-coupons might be misused by vendors to link transactions of customers in order to collect and compile information from their transactions in a profile. To protect the privacy of customers in this respect, the coupon system that we proposed allows customers to unlinkably redeem single coupons while preserving security requirements of vendors.

REFERENCES

[1] Kahate. A., Cryptography and Network Security T.M.H. Second Edition.

[2] Chun Hung Liu & H Harry Asada, A source Coding and Modulation method for Power saving and Interference reduction in DS-CDMA sensor network systems, *Proceedings of American Control Conference*, pp 3003-3008, (2002).

[3] S. Brands. An efficient off-line electronic cash system based on the representation problem. CWI Report, CS-R9323, Centrum voor wiskunde en Informatica (CWI), (1993).

[4] S. Brands, Rethinking Public Key Infrastructure and Digital Certificates - Building in Privacy, Ph.D thesis, Eindhoven Institute of Technology. *The Netherlands*, (1999).

[5] G. Brassard, D. Chaum, C. Crepéau. Minimum disclosure proofs of knowledge, *Journal of Computer and System Sciences*, **37**, (1988).

[6] E. Brickell, J. Camenisch, L. Chen. direct anonymous attestation. Proc. 11th ACM Conference on Computer and Communications Security, pages 132-145, ACM press, (2004).

[7] J. Camenisch, Group Signature Schemes and Payment Systems Based on the Discrete Logarithm Problem. Ph.D thesis, ETH Zurich, Switzerland, (1998).

[8] J. Camenisch, J. Groth. Group signatures : better efficiency and new theoretical aspects. *Forth Int. Conf. on Security in Communication Networks*- SCN 2004, LNCS 3352, Springer, (2005).

[9] J. Camenisch, A. Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. EUROCRYPT '01', LNCS 2045. Springer, (2001).

[10] J. Camenisch, A. Lysyanskaya. A signature scheme with efficient protocol. Third Conference on Security in Communication Networks - SCN '02', LNCS 2576, Springer, (2001).

[11] J. Camenisch, M. Michels, Separability and efficiency for generic group signature schemes. Adv. in Cryptology - CRYPTO '99' LNCS 1666, Springer Verlag, (1999).

[12] D. Chaum. Privacy protected payments : Unconditional payer and/or payee untraceability. Smart Card 2000, Proceedings. North Holland, (1989).

[13] L. Chen. Access with pseudonyms. Cryptography : Policy and Algorithms, International Conference, Brisbane, Australia, July, 1995, Proceedings, LNCS 1029, Springer, (1996).

[14] R. Cramer, I. Damgard, B. Schoenmarkers. Proofs of partial knowledge and simplified design of witness hiding protocols. Adv. in Cryptology - CRYPTO '94', LNCS 838, Springer, (1994).

[15] I. Damgard, E. Fujisaki, A statistically hiding integer commitment scheme based on groups with hidden order. Adv. in cryptology - ASIACRYPT '02' LNCS 2501, Springer, (2002).

[16] M. Enzmann, M. Fischlin, M. Scheneider. A privacy friendly loyalty system based on discrete logarithms over elliptic curves. Financial Cryptography, LNCS 3110, Feb. (2004).

[17] F. Feinberg, A. Krishna, Z. Zhang. Do we care what others get? A behaviorist approach to targeted promotions. *Journal of Marketing Research*, **39**(3), Aug. (2002).

[18] N. Ferguson. Extension of single term coins. Adv. in Cryptology - CRYPTO '93', LNCS 773, Springer, (1993).

[19] E. Fujisaki, T. Okamoto, Statistical zero knowledge protocols to prove modular plynomial relations. Adv. in Cryptology - CRYPTO '97', LNCS 1294, Springer, (1997).

[20] D. Hoffman, T. Novak, M. Peralta. Building consumer trust online. *Communications of the ACM*, **42**(4), Apr. (1999).

[21] A. Kobsa. Tailoring privacy to use's needs User Modeling 2001 (UM 2001), LNAI 2109. Springer, (2001).

[22] G. Macintosh, L. Lockshin, Retail relationships and store loyalty : A multi-level perspective. *International Journal of Research in Marketing*. **14**(5), Dec. (1997).

[23] D. Maher. A platform for privately defined currencies, loyalty credits, and play money, *Financial Cryptography*, LNCS 1465, Springer, (1998).

[24] G.O'Connor. R. O'Keefe. The Internet as a new maketplace : Implications for consumer behaviour and marketig management. Handbook on Electronic Commerce. Springer, (2000).

[25] A. Odlyzko, Privacy, Economics, and Price Discrimination on the Internet. *5th International Conference on Electronic Commernce* (ICEC 2003). CM Press, (2003).