個人情報漏洩を防止するWebアンケートのセキュリティ強化

中里純二[†] 藤本賢司[††] 菊池浩明[†††]

本論文では，ロバスト性を保証して回答者のプライバシを守る，ウェブでのアンケートを行うセキュアプロトコルを提案する．提案プロトコルは，Cramer らによって提案された効率の良い秘匿性を満たしたセキュア電子投票プロトコルに基づいている．秘匿性と効率性は，選挙とアンケートの両方に共通する要求条件である．しかし，特にアンケートにおいては複数の選択肢を同時に選択することが許されていることがあり，選択肢数 n に対して，通信量が指数関数的に増加するという課題が生じることを本論文で指摘する．この課題を解決するために，正当性の証明コストを $\Theta(2^n)$ から $\Theta(n)$ に削減するプロトコルの提案を行う．また，試験実装に基づいたパフォーマンス評価を与える．

Security Enhancement Preventing Personal Information Disclosure in Web-based Questionnaire

Junji Nakazato,[†] Kenji Fujimoto[††] and Hiroaki Kikuchi[†††]

This paper proposes a secure protocol for web-based questionnaire that preserves the privacy of responders and ensures the robustness. The proposed protocol is based on secure electronic voting protocol proposed by Cramer et al. with confidentiality and efficiency, which are common requirements for both applications. This paper points out an issue particular in the secure questionnaire, that is, a requirement to deal with a multiple-choice question, and shows that the communication overhead grows exponentially with the number of choices n. To address the issue, the proposed protocol reduces the cost from $\Theta(2^n)$ to $\Theta(n)$. The performance based on the experimental implementation is also shown.

1. はじめに

Web による電子的なアンケート調査は，従来の紙ベースによるものに比べ，低コスト，即時性，集計の機械化，利用者の負担減などの特徴があり，現在広く普及している．たとえば，google で「Web アンケート」と検索すると，2004 年 11 月現在で，136 万件のサイトが表示される．また，様々なアンケートページを集めた会員制のアンケート調査ページなども存在する[1)]．

しかし，利便性の一方で，個人情報を含むデータベースが漏洩する事件が後を絶たず，利用者の不安を扇っている．**表 1** に示されるように，多くの企業から個人情報流出事件が起きて，大きな社会問題になった．

この個人情報漏洩の問題に対して，セッションの暗号化は 1 つの対策である．たとえば，JIS Q15001 に基づくプライバシマーク制度[2)] では，個人情報の収集の際には，SSL による暗号化とクロスサイトスクリプティング対策などを講じることを強く奨励している．それとともに，2005 年 4 月に個人情報保護法が本格施行されるため，情報収集者には，個人情報の扱いに関して十分注意を払う義務が生じる．そこで，Web を用いたアンケートページでも SSL を用いた暗号通信を行い，安全性を確保する動きが一般化してきている．

しかしながら，セッションの暗号化は，伝送中での第三者による盗聴を防止する技術であり，受信サイト

表 1 Web アンケートからの個人情報流出事件
Table 1 Private information disclosure incidents form Web-based questionnaire.

発生年	社名	漏洩件数	原因
2002/ 5	コンビニエンスストア	1,300	Web 設定ミス
2003/ 6	コンビニエンスストア	56 万	委託企業から
2004/ 2	インターネットプロバイダ	450 万	委託企業から
2004/11	通信会社	7,000	Web 設定ミス

† 東海大学大学院工学研究科電気工学専攻
Course of Electrical Engineering, Graduate School of Engineering, Tokai University
†† ウェブ・テック株式会社システム部
Web Tech Co., Ltd.
††† 東海大学電子情報学部情報メディア学科
Department of Information Media Technology, School of Information Technology and Electronics, Tokai University

でいったん復号された後は無防備である．そのため，そのサーバの管理者やアクセス可能な人物は簡単に個人情報を入手することが可能である．現に，表 1 にあげた例でも SSL を使っていながら生じているものもある．また，アンケートデータを保存しているような，通常外部からは参照不可能なはずのところが，人為的ミスにより参照可能になってしまっていることで，個人情報が漏洩することも多く発生している．つまり，いくら SSL を用いて，セッションを暗号化していたとしても，内部者の不正や人為的ミス，委託先での流失に対しては，十分な対処がなされていないのが現状である．また，個人情報漏洩のほとんどの原因がこれらにある．

この内部からの情報漏洩に対して，荒井らは，強制アクセス制御とファイル暗号化を導入することを提案している[3]．しかし，彼らのシステムでは，単一の閉じたシステムの上で，TCSEC B 以上の高いセキュリティレベルを仮定しており，本研究で対象とするようなオンラインでの一般的なウェブサーバの上での適用は困難である．

一般的なウェブベースでのアンケートを想定したセキュリティ強化の実装には，次の 2 つがあげられる．

(1) 横川らによる電子匿名アンケート機構[4]
ブラインド署名[5] と匿名通信路[6] によりアンケートフォームを安全に集計するシステム．匿名通信路は，ウェブのプロキシーサーバで代用している．

(2) 北川らによる講義評価の匿名アンケートシステム[7]
2 種類のブラインド署名とレシートを導入して，回答と未回答の識別を可能にしたアンケートシステム．匿名性は，登録時とアンケート回答時で異なる端末を用いることで回避している．

両方式とも，無資格者による不正な回答や二重投票を防止するために，登録時に回答権利に対するブラインド署名を発行し，回答時に提示する方式を採用している．それゆえ，アンケート形式に対する制約がなく，自由に意見を記述するようなアンケート形式にも適用できる．その反面，匿名性を保証するために匿名通信路を導入する必要性があるが，これには多重暗号化や不正防止のために多くのコストがかかる．文献 4) では，ウェブプロキシーサーバで簡易実装したり，文献 7) では，登録端末とは独立した共用端末から回答をさせるという運用上の工夫で匿名通信を代用したりしている．いずれも，匿名通信路を安全に実現しようとすると大きなコストがかかることが避けられない．

そこで，本研究では，Cramer らによって提案された電子投票プロトコル CGS97[13] に基づいたプライバシ保護を試みる．CGS97 は，準同型性を満たした公開鍵暗号で投票値を暗号化して投票し，準同型性により暗号化したままで投票値の集計処理を実現する暗号プロトコルである．CGS97 を採用したのには，主に次の理由があるからである．

(1) 集計（ウェブ）サーバにはデータが暗号化されたまま保存されるので，外部からの攻撃や内部犯に対してもデータの秘匿性が保証されること．集計サーバは，データを復号することなく集計処理を実現する．それゆえ，秘密鍵を管理する必要がないため，セキュリティコストを下げることができる．

(2) 最後の集計結果を復号する秘密鍵は，集計者とは独立のプライバシポリシ管理者に預託しておくことができること．さらに，秘密鍵をしきい値法で分散しておけば，複数のプライバシポリシ管理者で分散管理も可能である．

(3) コストのかかる匿名通信路が不要であること．シンプルな 1 ラウンドのプロトコルであるので，現行の HTTP との互換性が高く，汎用の SSL/TLS などの通信路を用いることもできるため，導入のコストが小さい．

このような特徴を持つ CGS97 は，準同型性による秘密関数計算による一連の研究☆の 1 つである．本研究の方式と従来方式[4),7)] との特徴を**表 2** に整理する．

ただし，CGS97 は電子選挙を想定したプロトコルであるために，これをウェブでのアンケートに適用するには問題が生じる．アンケート形式の多様性である．複数の候補者から 1 名を選ぶ選挙に対して，アンケートでは自由文の回答形式や複数の選択肢を同時に指定することが許される複数回答形式などが混在している．CGS97 で定められている投票値の正当性を担保するゼロ知識証明をここに直接適用すると，大幅なオーバヘッドが生じてしまう．そこで，本論文では，ウェブアンケートの調査を行い，このオーバヘッドの大きさの統計値を同定する．さらに，この複数回答形式に対応する効率の良い方法を提案する．

実際には，提案する方式が現実の範囲内で動作可能

☆ 年齢や性別などの個人情報を秘匿したまま統計値だけを導出する試みがいくつか研究されている．Sako は，Trusted Third Party（TTP）を分散して集計するプロトコルを提案した[8]．Nakanishi らは，グループ署名を用いて，属性証明者を導入するプロトコルを提案している[9]．Franklin らや，Naor ら，Ogata らは，Web での視聴者らが，ある一定以上集まったことだけを証明する視聴度測定法を提案した[10)〜12)]．

表 2 ウェブアンケートシステムと特徴
Table 2 Features of some Web-based questionnaire systems.

システム	文献 4)	文献 7)	本提案
匿名性	匿名通信路	運用上の工夫	準同型性を持つ公開鍵暗号による秘密関数計算
安全性	ブラインド署名	ブラインド署名（2種）	ゼロ知識証明
信頼性の仮定	プロキシーサーバ	共用投票端末	（複数の）プライバシポリシ管理者
複数回答投票	対応可能	対応可能	問題点

であるか，多種多様なアンケート形式に少ない運用コストで対応するにはどうしたらよいかどうか，運用にあたって多くの課題が残っている．これらの疑問点に応え，提案方式の有効性を検証するために，Java アプレットの上で暗号化処理を実現する試験システムを実装し，その上でのパフォーマンス評価を行った．本論文では，これらについても報告する．

本論文の構成は次のとおりである．2 章では，Web ベースのアンケートの現状を調査した結果を示し，電子投票と，アンケートでの条件の違いを示す．3 章で，アンケートへの電子投票プロトコルの説明を行う．電子投票プロトコルとしては文献 13) を用いる．また，ここで，2 章で示した条件を考慮した正当性の証明の効率化を説明する．4 章では，実装システムについて説明し，5 章で，評価を行う．最後に 6 章でまとめる．

2. Web アンケートの現状

2.1 調 査 方 法

検索エンジン google で，次のようにしてアンケートページの調査を行った．

「企業」「アンケート」の 2 つのキーワードで検索し，上位からアンケートページだけを 30 件収集し，調査した．このキーワードを選んだ理由は，いくつかのキーワードを試みたうえで，最も代表的なアンケートサンプルが得られていたためである．

2.2 用 語 定 義

1 つのアンケートは，一般に複数の質問から構成されている．この質問の数を**質問数**と呼び，m で表す．また，各質問に対する回答の選択肢の数を**選択肢数**と呼び，n で表す．i 番目の質問の選択肢数を n_i と書く．たとえば，$Q1$.「はい」「いいえ」，$Q2$.「思う」「どちらともいえない」「思わない」というようなアンケートは，$m = 2$，$Q1$ に対する選択肢数は $n_1 = 2$，$Q2$ に対する選択肢数は $n_2 = 3$ となる．

n 個の選択肢の中から 1 つだけを選ぶ形式を **1-out-of-n**，複数回選ぶことが許されている形式を**複数回答**と呼ぶ．アンケートの中には，1-out-of-n の中に「その他」が用意されていて，選択肢を自由に追加できるものがある．これを，**自由書き込み形式**と呼ぶ．

表 3 アンケートページの統計情報
Table 3 Statistics of questionnaire pages.

		数	比率 [%]
サンプルページ数		30	
質問数 (m)	平均	8.10	
	最大	32	
	標準偏差	6.78	
選択肢数 (n)			
1-out-of-n	平均	4.73	
	最大	19	
	標準偏差	2.60	
複数回答	平均	9.30	
	最大	35	
	標準偏差	6.19	
1-out-of-n の質問		112/243	46.09
複数回答の質問		61/243	25.10
自由書き込みの質問		70/243	28.81
記名アンケート		29/30	96.67

表 4 アンケート形式
Table 4 Type of questionnaire.

アンケート形式	件数	比率 [%]
HTML + Mail	7	23.33
HTML + CGI	10	63.33
PHP	1	3.33
ASP	3	10.00

2.3 アンケート結果

アンケートの構成や質問形式に関する調査結果を**表 3** に示す．

30 ページの質問数の合計は 243 である．記名アンケートとは名前や e-mail などの個人情報を特定したアンケートである．

表 4 は，アンケートを実現している技術についての調査結果である☆．

☆ ここで，HTML+CGI とは HTML から CGI に送る形式で，HTML+Mail とは指定したメールを用いてメールアドレスに回答が送られる形式である．ASP とはすべての処理をサーバサイドで行い結果を HTML で表示させるものである．

2.4 考　　察

表 3 より，アンケート構成では 1-out-of-n の形式が 46.09%を占めており，自由書き込みと複数回答が残りの半分ずつを占めている．ほとんどのアンケートページには，住所などの個人情報を入力するための記名式の項目があることが分かる．

63%以上のページが CGI を用いたアンケート処理を行っている．また，残りの 36%のものでも，サーバサイドで回答データの処理を行う方式であり，SSL を用いた通信を行っているものは多くはなかった．さらに，ページによっては SSL を用いているため，“個人情報の漏洩はいっさいしません” と明記されているものも見受けられたが，たとえ SSL 通信を用いた場合でも，表 1 に示したように，情報漏洩は通信中ではなく，サーバ到達後に起きていることから，安全であるとはいいがたい．

本論文では，多くのアンケートページが HTML によって構成されているため，HTML のレイアウトを崩さず，変換の工程を最少化するには，JavaScript を用いて，Java Applet と組み合わせる方式が最適と考える．

3. 電子投票プロトコルの応用

本研究では，Cramer らによって提案された電子投票システム[13] を利用することによって，アンケートデータを暗号化したまま保管，管理し，さらに，選択問題などの決められた回答が用意されている設問に対する集計を，暗号化したまま実行することを試みる．これにより，万が一委託企業や，Web などの設定ミスにより情報が漏洩したとしても，安全性を保つことが可能になると考える．

ここでは，文献 13) の概要を述べる．また，電子投票と，電子アンケートの相違点に着目し，検証方法の改良を検討する．

3.1 CGS 97 概要

投票内容を G または G^{-1} とし，投票者 $i = 1, \ldots, l$ について，準同型性を満たした暗号アルゴリズムを用いる．暗号化を $E[\cdot]$ とすると，$C_i = E[G]$ または $C_i = E[G^{-1}]$ を投票する．ここで，G は乗法群 Z_p^* の位数 q となるような生成元である．集計者はすべての投票の積 $S = \prod_{i=1}^{l} C_i$ を計算することによって，投票内容を秘密にしたまま集計する．ここで，賛成者数を h とすると，集計結果は $S = E[G^h]$ となる．これを復号することで，$D[S] = G^h$ が得られる．

一方，複数の値から 1 つを選択する 1-out-of-n の場合は，生成元を G_1 から G_n までの n 個用意し，同様のプロトコルを実行することによって実現する．

	知識の証明	べき乗回数
暗号文	$c_1 = g^r,\ c_2 = G_1 y^r$	2
正	$a_1 = g^\alpha,\ b_1 = y^\alpha$	2
当	$a_2 = g^\beta c_1^{d_2}$	2
性	$b_2 = y^\beta (c_2/G_2)^{d_2}$	2
証	$d_1 = h(a_1 \| b_1 \| a_2 \| b_2) \oplus d_2$	0
明	$\gamma = \alpha - r d_1$	0
send $c_1, c_2, a_1, b_1, a_2, b_2, d_1, d_2, \beta, \gamma$		
検	$d_1 \oplus d_2 \stackrel{?}{=} h(a_1 \| b_1 \| a_2 \| b_2)$	
証	$a_1 \stackrel{?}{=} g^\gamma c_1^{d_1}$	2
	$b_1 \stackrel{?}{=} y^\gamma (c_2/G_1)^{d_1}$	2
	$a_2 \stackrel{?}{=} g^\beta c_1^{d_2}$	2
	$b_2 \stackrel{?}{=} y^\beta (c_2/G_2)^{d_2}$	2

図 1 正当性の証明の計算コスト
Fig. 1 Computation cost of proof of validity.

3.2 正当性の証明

送信内容がすべて暗号化されていると，正しいアンケート内容が送信されているか否かを検証する必要性が出てくる．1-out-of-n 形式の場合，回答結果の暗号文 C が $G_1, \ldots, G_n$ のいずれかの暗号文になっているということを示すために，以下の正当性の証明（Proof of knowledge: PK）をする．

$$PK\{C = E[G_1] \vee \cdots \vee C = E[G_n]\}$$

文献 13) では，文献 14) のプロトコルを用いることによって，問題を解決している．

図 1 に，選択肢数 $n = 2$ としたときの証明と検証を示す．ここでは，G_1 と G_2 の選択肢の中から，G_1 を回答することを考える．ここで，α，β，d_2 は乱数とし，y，g を公開鍵とする．

3.3 複数回答形式の問題

前節まででは，1-out-of-n の場合の説明を行った．しかし，表 3 でも示したように，多くの Web アンケートページでは，複数回答形式の質問が用いられている．そこで，文献 13) を応用することで，複数回答形式を実現する．n 個の選択肢からの複数回答形式は，$\{1, \cdots, n\}$ の部分集合 A によって特徴付けられる．$G_1, \cdots, G_n$ を独立する n 個の生成元とすると，暗号文は

$$C = E[\prod_{i \in A} G_i]$$

で表現される．

ここで，アンケート回答が正しい形式であることの検証には以下の方法が考えられる．

(1) 選択可能なすべての組合せを考えて 1-out-of-2^n の知識の証明を行う．
(2) 暗号文 C のメッセージが，G_1 から G_n の値の

表 5 方式 (1) と (3) のべき乗の回数
Table 5 Computation cost in number of modular exponentiations.

	方式 (1)	方式 (3)
暗号文数	1	n
暗号化	2	$2n$
PK	$4 \cdot 2^n - 2$	$6n$
検証	$4 \cdot 2^n$	$8n$
計	$8 \cdot 2^n$	$16n$

範囲に入っていることを区間のゼロ知識[16] で証明する．

(3) 各選択項目を 2 択の独立な n 個の質問へ展開して証明を行う．暗号文は $C_1, \cdots, C_n$ の n 個が必要になるが，知識の証明は，$i = 1, \cdots, n$ につれて，

$$PK\{C_i = E[G] \vee C_i = E[1]\}$$

を実行すればよい．

(1) では，とりうるメッセージのすべての組合せは 2^n 個存在する．(3) の検証では，検証のオーダは $O(n)$ となる．

ここで，アンケート回答全体での効率を (1) と (3) の場合で比較してみる．アンケートに回答するためには，選択肢の暗号化，暗号文に対する知識の証明の生成，回答結果の送信，検証，集計となる．(1) と (3) の場合で異なるのは，暗号文数，知識の証明の計算コスト，送信量，検証の計算コストである．ここで，計算量をべき乗剰余演算の回数で見積もる．文献 14) による知識の証明を行うとき，方式 (1) と (3) の各コストを**表 5** で整理する．ただし，n は選択肢の数を表すため，2^n 通りの回答が許されていることに注意してほしい．

表より，$n = 1$ と $n = 2$ のときは，方式 (1) と (3) のコストがまったく等しいことが分かる．$n \geq 3$ では，明らかに，方式 (3) の方が小さい．方式 (3) では暗号文数が n に比例して増えるが，方式 (1) では PK のサイズが $O(2^n)$ のオーダで増えるため，通信コストの面でも方式 (3) が優れている．

4. 実装システム

4.1 要求条件

本実装システムでは，アンケート回答データから得られる個人情報の内部からの流出を防ぐ，セキュアなアンケートページの実現を目的としている．そのため，すでに多く存在している，HTML や CGI などで書かれているページに若干の変更を加えることでセキュアなアンケートページに移行できることが望ましい．そこで，以下に実装システムにおける要求条件をあげる．

- プライバシ強化の実現（内部からの情報漏洩の防止）
- アンケート妨害の防止（決められた値以外の回答を送信することの防止）
- Web 管理者へ負担をかけず，セキュアページへの移行が可能（特別な知識を必要とせず，本ツールを利用可能にすること）

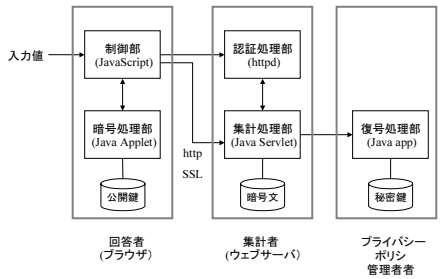


図 2 システムのブロックダイアグラム
Fig. 2 Block diagram for the system.

4.2 システム構成

図 2 に本システムのブロックダイアグラムを示す．本システムは，ウェブブラウザで実行される回答者とウェブサーバで実現される集計者，そして，プライバシポリシ管理者から構成されている．集計者と回答者間は，HTTP/SSL 通信を用いることで安全な通信路を構築する．さらに，Signed Applet を用いているため，偽造 Applet の利用や，フィッシング詐欺などのなりすましを防止することが容易である．

回答者では，各質問においてどの回答を選んだかを入力として，暗号化処理を行う Applet を呼び出し，回答に対応した平文の暗号化と正当性の証明を行う．暗号化にはプライバシポリシ責任者の公開鍵を用いる．Basic 認証や SSL クライアント認証により回答者の確認を行い，SSL 通信を用いて集計者（Java Servlet）に送信する．これらの処理はブラウザの中で実行され，全体の制御は JavaScript が司る．ここで，選択問題に対しては，3 章で説明したように暗号化および，正当性の証明を行う．また，多くのアンケートページで見られる自由書き込み形式の場合には，共通鍵暗号を用いて暗号化する．暗号化し，暗号化された回答データとともに送信する．

集計処理部（Servlet）では受信した値を検証し，正しい暗号文なら集計するとともに，暗号文データベースへ保存する．集計サーバには回答者個々の情報がファイルとして保存されるが，暗号化された状態で保存している．

最後に，全回答が完了した後に，プライバシポリシ責任者が集計結果を復号し，結果を公開する．この暗

表 6 実行および開発環境
Table 6 Execution and implementation environment.

CPU	PowerPC G5 2.5G Dual
Memory	512 M
開発言語	J2SDK 1.4.2_02
servlet	Tomcat 5.0.26
通信フォーマット	XML（独自フォーマット）
暗号化アルゴリズム	ElGamal 暗号
暗号化ビット数	1,024 bit
共通鍵暗号	AES (128 bit)

表 7 基本クラス
Table 7 Fundamental classes.

ppsq	本 Applet のメインクラス．JavaScript からのデータを入力とし，複数回答形式の質問を提案プロトコルに従い展開し，選択問題以外の値はすべて AES クラスにより暗号化を行う．最後に，集計サーバに出力する．
Read	HTTP プロトコルを用いてデータや公開鍵の読み込みを行う．
CGS	2 つの選択子と，回答結果を入力とし，ElGamal クラスを用いて暗号化，CDS クラスを用いて正当性の証明の作成を行う．
ElGamal	回答値を入力として，暗号文を出力する．
CDS	選択子および暗号文，暗号化に用いた乱数を入力とし，プロトコルに従い，正当性の証明を出力する．
Hash	入力値に対して，MD5 値を出力する．
AES	入力値を 128 bit AES 暗号化を行う．
XMLWriter	各クラスからの値を入力とし，独自フォーマットの XML データを出力する．

号文を復号することが可能なのは，プライバシポリシ責任者のみのため，外部に個人情報が漏れる心配はない．

ここで，JavaScript と Applet を用いた理由は，Applet だけでは，ソースの変更とコンパイルが必要になることを避けるためである．JavaScript を組み合わせることで，質問数や選択肢数が増えても Applet のソースの変更はいらず，JavaScript を変更するだけで柔軟に対応できる．

実行および開発環境を**表 6** に，本実装を実現する基本クラスとその機能を**表 7** にまとめる．

4.3 多重回答防止機能

アンケートには回答者を無記名として，不特定多数の回答者を対象とすることが多い．しかし，文献 7) のように在学生のみを対象とするアンケートや同一回答者による多重回答を防止するためには，次にあげるような対応が考えられる．本システムでは，これらの中で最も多用されており，応用範囲も広いパスワードによる Basic 認証を実装している．

(1) IP アドレスによる多重投票防止機能
同一の IP アドレスの投票を排除し，不注意による多重投票を防止する．

(2) ウェブのユーザ認証機能との併用
パスワードによる Basic 認証や公開鍵証明書による SSL クライアント認証を用いる．会員制のアンケート投稿フォームなどにも適応できる．本試験実装では，集計者のウェブサーバの機能によりこの認証処理を実現している．

(3) 暗号文の開示をさせる方式[17)]
有長らは，票をランダムに開示させることで，その結果の統計的性質に基づき，ある程度の誤差を許しながら不正を検出する方法を提案している．

4.4 プライバシポリシ管理者の権限分散

プライバシポリシ管理者の権限は，次のようにして分散管理することができる．公開鍵対の復号鍵を n 個中 k 個集まると復元できるような (k, n) しきい値法で秘密分散し，n 人の管理者に秘密に配布する．任意の k 人以上の管理者が同意すれば，各々の（分散された）秘密鍵の情報を互いに露見することなく，与えられた暗号文を復号化することが可能である．しきい値復号化プロトコルについては，文献 18) に詳しい．

現実には，信頼できる消費者団体や監査組織，および，国際間の調停組織などがこのプライバシポリシ管理者の役割を果たしてもよい．アンケートを実施して集計結果を復号するときだけこれらの組織に介入してもらう．これによって，アンケート回答者の安心感を得ることが期待できる．

4.5 通信フォーマット

本システムでは通信フォーマットに独自フォーマットの XML を用いた．XML を用いることで，管理者による，アンケート回答データの再利用が容易になると考える．暗号化された氏名などを取り除き，アンケート回答値のみのデータを作成したり，住所のみの暗号化データベースを構成したり，様々な応用が考えられる．

ElGamal 暗号[15)] や，AES 暗号のフォーマットは XML で定義された独自フォーマットとした．付録 A.1 に，本システムで用いている XML Schema を示す．

アンケートデータ全体は `<ppsq>` により定義し，`<ppsq>` 内には 1 つ以上の `<question>` と，`<Key>` が存在する．ここで，`<question>` は，各質問に対応し，選択肢数 n，質問番号，質問のタイプ（`radio`，`checkbox` など），`FORM` での名前を属性に持つ．`<Key>` は，共通鍵暗号方式のアルゴリズム名を属性に持ち，自由書き込み形式のデータを暗号化するために用いた共通鍵を格納している（ElGamal 暗号によって暗号

化された値)．

<question> には 1 つ以上の <cipher-text> または，<CGS> により構成されている．自由書き込み形式のデータの場合，AES により暗号化し，<cipher-text> に格納する．また，それ以外の選択形式の質問の場合，選択した選択肢の暗号文を，暗号化データを属性に持つ <ElGamal> に格納し，その暗号文に対する．正当性の証明を <CDS> に定義する．<CGS> は，この組を格納する．ここで，複数回答形式の場合，2 択の n 個の質問に展開するため，<question> には <CGS> が n 個格納される．

<CDS> は，暗号文に対する正当性の証明を構成する．<commit> には図 1 中の a，b を ElGamal 暗号の形で格納する．<challenge> には，図 1 中の d を格納し，<response> には，γ を格納する．

5. 評　　価

5.1 パフォーマンス

ここでは，実装システムのパフォーマンスの評価を行う．**図 3** に選択肢数 n を変化させたときの 1-out-of-n の通信量の変化，**図 4** に選択肢数 n を変更させたときの複数回答形式の通信量の比較を行う．ここで，図 4 では，図 3 より得られた結果より，とりうるすべての選択肢の組合せの中から 1 つを選ぶ 1-out-of-2^n 方式の通信量を試算し，2 択の n 個の質問に展開する提案方式との比較を行う（3.3 節の検証方法 (2) を参照)．

図 3，図 4 より，1-out-of-n 形式と，複数選択化方式の提案方式では，ともに線形に増加していることが分かる．これは，1-out-of-n 形式の場合は n に比例して正当性の証明が増え，提案方式の場合は n 個の 1-out-of-2 形式の質問に展開するためである．図 4 より，n 個の 1-out-of-2 の質問に展開することで，通信量の改善がされていることが確認できる．たとえば，複数選択化形式の平均選択肢数である $n = 10$ のときには，約 97%データ量を削減することができた．

次に，10 件のサンプルページを実際にセキュリティ強化し，通信量，およびアンケート回答者の処理時間を測定した．**図 5** に，アンケートページ全体の選択肢数による通信量の変化，**図 6** に，ページ全体の選択肢数による処理時間の変化を示す．また，**表 8** にセキュリティ強化を行った 10 件のページの平均値を示す．

表 8 より，セキュリティ強化にともなう通信量の平均増加率は 125.5 倍となった．また，送信ボタンを押してからの平均待ち時間は 3.5 秒となった．通信量の増加は，暗号化と正当性の証明を行うためのオーバ

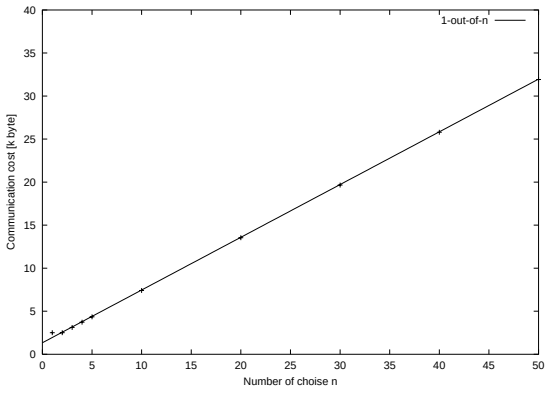


図 3 質問数による通信量の変化（1-out-of-n）
Fig. 3 Communication cost with respects to n (1-out-of-n).

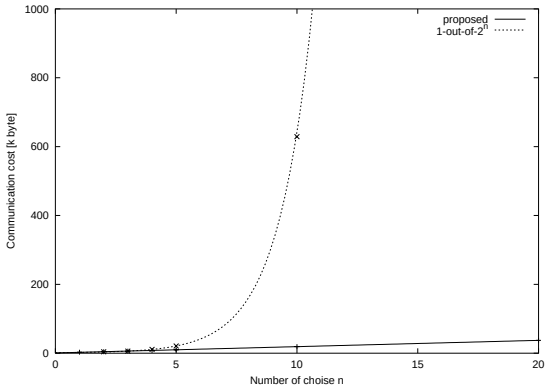


図 4 質問数による通信量の変化（複数選択化問題）
Fig. 4 Communication cost with respects to n (multiple choice).

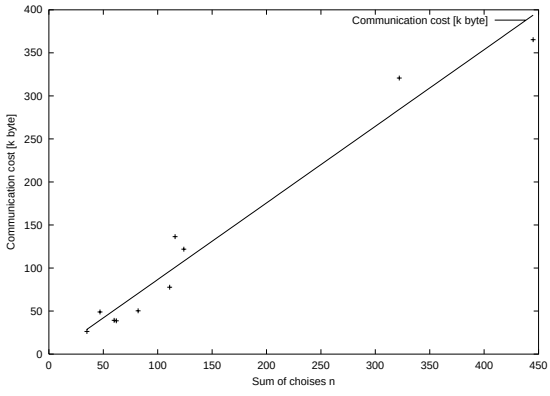


図 5 選択肢数による通信量の比較
Fig. 5 Communication cost with respect to sum of choices.

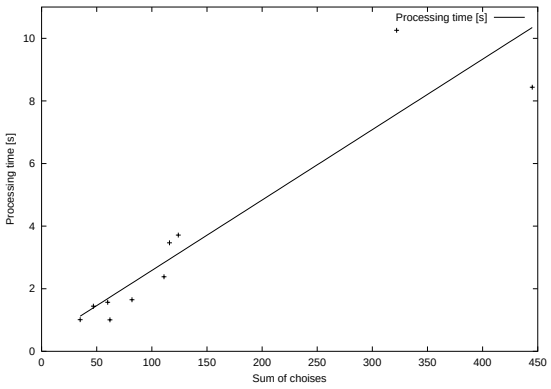


図 6 選択肢数による計算量の比較
Fig. 6 Processing time with respect to sum of choices.

表 8 セキュリティ強化による平均オーバヘッド
Table 8 Average overhead of security enhancement.

	average
Processing time [s]	3.5
Communication cost [kbyte]	
secure page	125.5
original page	1.0
Number of choices n	
1-out-of-n	94.2
multiple	28.6
free	17.6
total	140.4

ヘッドから生じるためである，

図 5 より，選択肢数 n に比例して通信量が増加することが分かる．また，図 6 より，計算量も比例する．ここで，表 3 より，アンケートページ 1 ページあたりの平均選択肢数は，約 39 個となることが分かる．そこで，図 5 より，通信量は，約 33.1 kbyte，図 6 より，計算量は，約 1.2 秒程度であることが分かる．

5.2 セキュリティ強化の工程

ここでは，本システムを用いた，既存のアンケートページのセキュリティ強化方法を説明する．本システムではアンケートページのセキュリティ強化に 3 Step を必要とする．以下に移行方法を示す．

Step 1. `HTML` のヘッダ部分（`<HEAD></HEAD>`）に外部の JavaScript の呼び出しを宣言する．

Step 2. `<BODY>` へ `applet` を埋め込む．

Step 3. `<FORM>` タグ内の送信先の指定 `action`，`name` を埋め込んだ Applet へデータを送るように変更する．

図 7 のような簡単なアンケートページは，**図 8** のようにセキュア化される．

2 行目に用意した JavaScript で，`FORM` 内で入力さ

```
<html><head></head>
<body>
<FORM method=post action=count.cgi>
<input type="radio" name=q1>賛成
<input type="radio" name=q1>中立
<input type="radio" name=q1>反対
<INPUT type=submit name=Submit value=送信>
</FORM></body></html>
```

図 7 元のアンケートページ
Fig. 7 Original questionnaire page.

```
 1 <html><head>
 2 <script type="text/javascript" src="ppsq.js"
  name="ppsq"></script>
 3 </head><body>
 4 <APPLET height=0 width=0 code=ppsq.class
   archive=ppsq.jar name="Question"
   mayscript></APPLET>
 5 <FORM method=post action=SERVER_ADDRESS
  onSubmit=ppsq(ppsq_form) \
   name="ppsq_form">
 6 <input type="radio" name=q1>賛成
 7 <input type="radio" name=q1>中立
 8 <input type="radio" name=q1>反対
 9 <INPUT type="submit" value=送信>
10 </FORM></body></html>
```

図 8 変更を加えたアンケートページ
Fig. 8 Modified questionnaire page.

れたデータが，何番目の質問に対する回答の選択肢か符号化されて Applet に渡される（Step 1）．4 行目では，実際に暗号化や，正当性の証明を行う `Applet` を挿入している（Step 2）．5 行目で，`FORM` の名前を指定し，`action` で，集計サーバのアドレスを指定する（本実装では，Servlet による集計を可能としている）．また，`onSubmit=ppsq(ppsq_form)` を指定することで，`submit` ボタンが押されたときに `JavaScript` を実行するようにする（Step 3）．

5.3 セキュリティ向上のためのコストの妥当性

アンケートページのセキュリティ強化を実現するためには，ユーザ側と管理者側の各々の立場からコストを算出しなくてはならない．

まず，ユーザ側では，通信時間と計算時間が増加する．ただし，この増加量は，サンプルページでの評価に基づくと通常の 125 倍の通信量と 3.5 倍の計算量である．ブラウザの上で選択肢を選び終わって送信のためのボタンを押した後での処理なので，ここでのコストはそれほど利便性を妨げないと考えられる．Java Applet と JavaScript という一般的なブラウザ環境だけで実現できるので，特殊なソフトウェアをインストールする必要がない利点がある．

次に，管理者側には集計用のソフトウェアをインストールする工程が必要になる．ただし，図 7 と 8 を

見れば明らかなように，質問 1 つあたりの変化量はたかだか 2 行しかない．Applet などは汎用的に設計されており，アンケートごとにコンパイルしなおす必要もない．暗号文を格納するための記憶容量の増加やプライバシポリシ管理者へ復号要求を出さなくてはならないコストがかかるが，個人情報を持つことによるリスクに比べれば，十分な対価と考えられる．

以上により，提案方式のセキュリティ強化のコストは十分に妥当であると結論づける．

6. おわりに

本論文では，個人情報漏洩を防止するためにセキュリティ強化した実用的なアンケートシステムの提案実装を行った．セキュア電子投票プロトコルに代表される文献 13) を用いて，電子投票では起りにくい複数回答形式に対する，効率の良い正当性の証明の方法を提案した．

電子投票プロトコルを用いることによって，暗号化されたままでのアンケート集計を可能とし，暗号化したままでの個人情報の管理を実現した．そのため，SSL を用いて通信を行った場合でも防ぐことのできない，内部からの個人情報の漏洩を防ぐことを可能とした．

実際に 10 件のサンプルページにおいて，セキュリティ強化を施した結果，平均 125.5 倍の通信量の増加が生じたが，処理時間は 3.5 秒と実用的な結果が得られた．通信量の増加は，昨今のネットワーク回線の高速化などでそれほど問題にならないと考える．

本システムでは，セキュリティ強化を行うために，アンケートページの管理者が自らソースに変更を加える必要があるため，今後これらの自動化が課題である．

参考文献

1) いーこえ・net．http://www.e-koe.net/（2004 年 11 月現在）
2) 財団法人日本情報処理開発協会. http://privacymark.jp/（2004 年 11 月現在）
3) 荒井正人，甲斐　賢，永井康彦，冨田　理：情報漏洩防止システムの提案，情報処理学会研究報告，2004-CSEC-24, pp.61–67 (2004).
4) 横川典子，菊池浩明，村井　純：電子匿名アンケート機構の設計と実装，情報処理学会研究報告，1996-DPS-20, pp.73–78 (1996).
5) Chaum, D.: Blind signatures for untraceable payments, *Proc. Cypto'82*, pp.199–203 (1982).
6) Chaum, D.: Untraceable electronic mail, return addresses and digital pseudonyms, *Comm. ACM*, Vol.24, No.2, pp.84–88 (1981).
7) 北川　隆，岡　博文，楫　勇一：大学における講義評価のための匿名アンケートプロトコルとその試作，情報処理学会論文誌，Vol.44, No.9, pp.2353–2362 (2003).
8) Sako, K.: Generating Statistical Information in Anonymous Surveys, *IEICE Trans. Fundamentals*, Vol.E79-A, pp.507–512 (1996).
9) Nakanishi, T. and Sugiyama, Y.: An Efficient Anonymous Survey for Attribute Statistics Using a Group Signature Scheme with Attribute Tracing, *IEICE Trans. Fundamentals*, Vol.E86-A, No.10, pp.2560–2568 (2003).
10) Franklin, M. and Malkhi, D.: Auditable Metering with Ligthweigth Security, *Proc. Financial Cryptography'97*, LNCS 1318, pp.151–160 (1997).
11) Naor, M. and Pinkas, B.: Secure and efficient metering, *Proc. EUROCRYPT'98*, LNCS 1403, pp.576–589 (1998).
12) Ogata, W. and Kurosawa, K.: Provably Secure Metering Scheme, *Advances in Cryptography ASIACRYPT2000*, LNCS 1976, pp.388–398 (2000).
13) Cramer, R., Gennaro, R. and Schoenmakers, B.: A Secure and Optimally Efficient Multi-authority Election Scheme, *Advances in Cryptology EUROCRYPT'97*, LNCS 1233, pp.103–118 (1997).
14) Cramer, R., Damgård, I. and Schoenmakers, B.: Proofs of Partial Knowledge and Simplefied Design of Witness Hiding Protocols, *Advances in Cryptology CRYPTO'94*, LNCS 839, pp.174–187 (1994).
15) ElGamal, T.: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, *IEEE Trans. Information Theory*, Vol.IT-31, No.4, pp.469–472 (1985).
16) Boudot, F.: Efficient proofs that a committed number lies in an interval, Proc. *EUROCRYPT'00*, LNCS 1807, pp.431–444 (2000).
17) 有長伸吾，土井　洋，辻井重男：統計的手法を利用した票の正当性証明に関する一考察，情報処理学会研究報告（CSEC），Vol.2004, No.129, pp.17–22 (2004).
18) Desmedt, Y.: Some Recent Research Aspects of Threshold Cryptography, *Information Security, 1st International Workshop ISW '97*, pp.158–173 (1997).

付　録

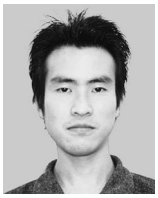
A.1 XML Schema

```
<?xml version="1.0" encoding="UTF-8">
<xs:schema xmlns:xs="http://www.w3c.org/2001/XMLSchema">

<xs:element name="ppsq">
<xs:complexType>
```

```
<xs:sequence>
<xs:element ref="question" minOccurs="0" maxOccurs="unbounded"/>
<xs:element ref="Key" />
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="question">
<xs:complexType>
<xs:sequence>
<xs:element ref="cipher-text" minOccurs="0" maxOccurs="1" />
<xs:element ref="CGS" minOccurs="0" maxOccurs="unbounded" />
</xs:sequence>
<xs:attribute name="choice-n" type="xs:integer" use="required"/>
<xs:attribute name="question-no" type="xs:integer" use="required"/>
<xs:attribute name="type" type="xs:string" use="required" />
<xs:attribute name="name" type="xs:string" use="required" />
</xs:complexType>
</xs:element>
<xs:element name="cipher-text">
<xs:simpleType>
<xs:restriction base="xs:string" />
<xs:sttribute name="Algorithm" type="xs:string" default="AES"
 use="required" />
</xs:simpleType>
</xs:element>
<xs:element name="CGS">
<xs:complexType>
<xs:restriction>
<xs:element ref="ElGamal" minOccurs="1" maxOccurs="unbounded"/>
<xs:element ref="CDS" minOccurs="1" maxOccurs="unbounded" />
</xs:restriction>
</xs:complexType>
</xs:element>
<xs:element name="ElGamal">
<xs:complexType>
<xs:restriction />
<xs:attribute name="C1" type="xs:string" use="required" />
<xs:attribute name="C2" type="xs:string" use="required" />
</xs:complexType>
</xs:element>
<xs:element name="CDS">
<xs:complexType>
<xs:restriction>
<xs:element ref="commit" />
<xs:element ref="challenge" />
<xs:element ref="response" />
</xs:restriction>
</xs:complexType>
</xs:element>
<xs:element name="commit">
<xs:complexType>
<xs:restriction>
<xs:element ref="ElGamal" minOccurs="1" maxOccurs="unbounded"/>
</xs:restriction>
</xs:complexType>
</xs:element>
<xs:element name="challenge">
<xs:complexType>
<xs:restriction>
<xs:element ref="d" minOccurs="1" maxOccurs="unbounded" />
</xs:restriction>
</xs:complexType>
</xs:element>
<xs:element name="d">
<xs:simpleType>
<xs:restriction base="xs:string" />
</xs:simpleType>
</xs:element>
<xs:element name="response">
<xs:complexType>
<xs:restriction>
<xs:element ref="r" minOccurs="1" maxOccurs="unbounded" />
</xs:restriction>
</xs:complexType>
</xs:element>
<xs:element name="r">
<xs:simpleType>
<xs:restriction base="xs:string" />
</xs:simpleType>
</xs:element>
<xs:element name="Key">
<xs:simpleType>
<xs:restriction base="xs:string"/>
<xs:sttribute name="Algorithm" type="xs:string" default="AES"
 use="required" />
<xs:restriction>
<xs:element>
</xs:schema>
```

(平成 16 年 11 月 29 日受付)
(平成 17 年 6 月 9 日採録)



中里 純二（学生会員）

平成 13 年東海大学工学部電気工学科卒業．平成 15 年同大学院博士前期課程修了．現在，東海大学大学院工学研究科博士後期課程在学中．コンピュータセキュリティ，暗号・情報セキュリティの研究に従事．電子情報通信学会会員．

藤本 賢司

2004 年東海大学工学部電気工学科卒業．2004 年ウェブ・テック株式会社入社，現在に至る．WEB システム構築に興味を持ち，現在の業務に就く．

菊池 浩明（正会員）

1988 年明治大学工学部電子通信工学科卒業．1990 年同大学院博士前期課程修了．1990 年（株）富士通研究所入社．1994 年東海大学工学部電気工学科助手．1995 年同専任講師．1999 年同助教授，1997 年カーネギーメロン大学計算機科学学部客員研究員．2000 年電子情報学部情報メディア学科助教授，現在に至る．博士（工学）．ファジィ論理，多値論理，ネットワークセキュリティに興味を持つ．1990 年日本ファジィ学会奨励賞，1993 年情報処理学会奨励賞，1996 年 SCIS 論文賞，2004 年情報処理学会研究開発奨励賞受賞．電子情報通信学会，日本ファジィ学会，IEEE，ACM 各会員．