5ZB-9

PGPによるEメール暗号化のユーザーインターフェースの改善

小川 純平, 加藤 直樹
東京学芸大学

1. はじめに

今日，Eメールは，PCや携帯電話等の個人用端末において広く利用されている．しかし，多くのメールサーバーは必ずしも十分なメールの暗号化機能を提供しておらず，その場合は容易に通信内容の傍受が可能となる．

この問題を解決するため，Pretty Good Privacy (PGP)を用いたメールの暗号化が考案されており，既にそのための実装もいくつか存在するが，多くは複雑なユーザーインターフェース (UI) をしており，一般の利用者にとって難解であるだけでなく，技術者にとっても導入に手間がかかるため，普及していない．

このような現状から，メールの暗号化は依然として十分でなく，機密性の高い情報をメールに記述できないなどの問題の他，一部の企業では，機密性の高いファイルをパスワード付きのzipやexeファイルにまとめ，パスワードを別途メールで送るという，形式的手法を取っている場合もある．

本研究では，現状のPGP実装に存在するUI上の障害を取り除き，誰でも簡単に扱えるPGPのインターフェースを実装することで，PGPによるメールの暗号化をより普及させ，メールの暗号化が十分でないことに起因する諸問題の解決を目指す．

2. 既存の方式の問題

2.1. 様々な暗号化方式とその問題

現状Eメールにおける通信の暗号化はPOP over SSL，IMAP over SSL，SMTP over SSLが一般的である．これらは途中で経由するサーバーがこれらの方式に対応していない場合，暗号化が行われずにメールが送信されてしまう問題がある．

S/MIMEはPGPと近い方式だが，この方式では認証局が必要で，1メールアドレスにつき，5万から10万円程度の利用料がかかる．法人でも，あまり現実的な額ではない．

2.2. PGPによるEメール送信の構造と問題

今回バックエンドとして用いるPGPは，現在一般的な使い方では，いくつか問題がある．

図1に示すように，秘密鍵ファイルは，ローカルハードディスク上に配置されるのが一般的で，そのままでは複数の端末から，暗号化や署名を行うことはできない．

勿論，手動で秘密鍵を移動して共有することは可能であるが，やや手間がかかる．

また，公開鍵を送信者から安全に取得するためには，USBなど物理メディア経由で渡したり，安全な通信経路を用いて手動で渡す方法があるが，このような方法は手間がかかるため，公開鍵サーバーを経由して取得する方法が一般的である．公開鍵サーバーとは，誰もが公開鍵を登録することができ，誰もが登録された公開鍵を取得できるように配布を行うサーバーのことであるが，その特性上，配布されている公開鍵が，本当にそのメールアドレスの持ち主が作成したものか否かが保証されない．

2.3 PGPによるメール暗号化の実装例とその問題

PGPの実装としてはGPGが最も有名であるが，CUIであるため，一般ユーザーにとって使いやすいとは言えない．そのため，今回は検討の対象としない．

GUIの実装としては，Thunderbirdのアドオンである Enigmailがあるが，これはセットアップ時にウィザード形式で，鍵ファイルのパスフレーズや，署名や暗号化を行うか，失効証明書を作成するか，など，様々な項目についてユーザーに質問しており，極めて煩雑である．

KmailもPGPをサポートするメーラーの一つだが，PGP鍵をGPGコマンドなどを用いて予め生成し，生成された鍵の中から，使用する鍵を選択しなければならない．

これらに共通する問題点としては，PGP有効化の過程で，鍵の概念を意識しなければならないことがある．鍵のファイルを指定したり，鍵パスワードの設定などが要求され，鍵の概念がよくわからないユーザーには混乱が生じる．

また，送信時に暗号化するか否かをユーザー自身が行わなければならない．鍵を持っていないユーザーに対しては暗号化を行うことができないため，デフォルトで暗号化を行うか否かについては設定が不可能である．

3. 設計

前述の問題点を踏まえ，これらの問題を解消するためのシステムを設計する．

3.1 クライアント / サーバー方式

クライアント / サーバー方式を採用し，鍵ペアの生成や，メッセージの暗号化・復号化をクライアント側ではなくサーバー側で行う．秘密鍵はサーバーに置かれ，暗号化・復号化は，メーラーがメッセージをサーバーに送信し，サーバー側で暗号化と署名，または復号化と認証を行う．この時メッセージはhttpsで保護する．

これにより，秘密鍵ファイル自体を転送することなく，複数の端末からの暗号化が実現される．

User Interface Improvement of Email Encryption with PGP
Jumpei Ogawa, Tokyo Gakugei University
Naoki Kato, Tokyo Gakugei University

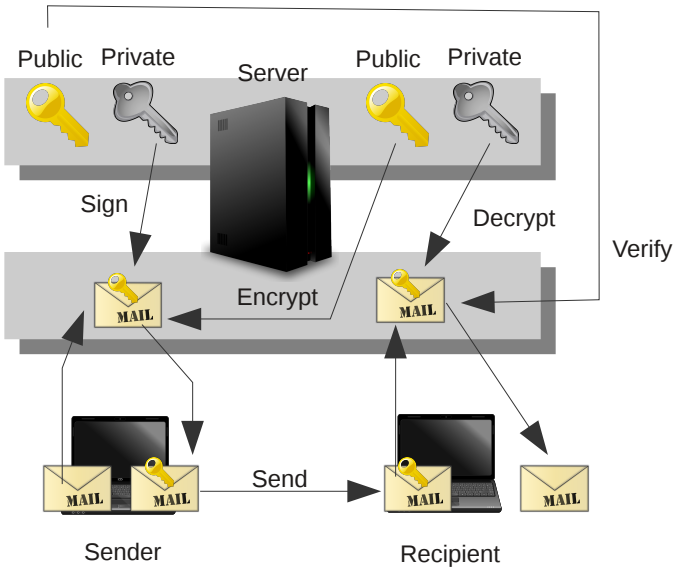

図 1: 本システムの構造

3.2 鍵の証明方式

メッセージの送受信の際には，そのユーザーの鍵を特定するために，メールサーバーを運営する企業・団体の提供する OpenID (Gmail なら Google アカウント，Yahoo! メールなら Yahoo! Japan ID など) を用いてユーザー認証を行う．その企業が OpenID を提供しない場合，別の OpenID を用いる．

これにより，登録者がそのメールアドレスの持ち主であることの証明が可能であるので，登録者間においては，その鍵の信頼性が保たれるため，公開鍵サーバーの鍵の信頼性の問題を部分的に解決することができる．

3.3 設定の簡略化

初期設定は，メールアドレスと，メールのパスワードの入力と，アドオンのインストールのみで，Thunderbird や KMail のような煩雑な初期設定作業は省く．

3.4 暗号化可否の表示

送信時に暗号化をするか否かは，メールアドレスを入力した時点で，そのアドレスの鍵が存在するか否か自動的に判別し，アドレスの左にアイコンを表示する．公開鍵が本研究で開発するサーバー上に存在する (つまり信頼できる鍵である) 場合は黄緑，公開鍵サーバー上に存在する場合は黄色，存在しない場合は赤にすることで，ユーザーはそのアドレスに対して暗号化が行われるか否か判別できる．

受信時にも同様に，そのメールに署名がなされているか否か確認され，送信者が正しいものか否か表示される．

4. 開発

サーバー側は Python 及び Django を用いた Web アプリケーションとして実装した．クライアント側は，Gmail，Yahoo! Mail，Outlook Live に対応させ，ブラウザーアドオンにより，メーラーのインターフェースを書き換える JavaScript を埋め込む形で実装した．

クライアントとの通信は Web API として行う．OpenID を用いてユーザー認証を行い，暗号化したいテキスト (つまりメールの本文) や関連する情報を POST で送信する．

例えばメールを送信する際には，メールのタイトル，本文，送信先，その他必要な情報がサーバーに送信され，そこから SMTP を経由してメールが送信される．

5. 評価実験と改良

評価実験を行い，実装したシステムと，対照実験のため Thunderbird / Enigmail を実際に利用してもらった．被験者は 10 名で，技術知識を持つテクニカルユーザー 5 名と，そうでない非テクニカルユーザー 5 名である．また，社会人経験者は 4 名，社会人未経験者 6 名である．

Thunderbird / Enigmail の評価実験では，質問紙調査の結果を見ると，テクニカルユーザーで難解と感じたのは 1 名だが，非テクニカルユーザーは 3 名が難解と感じ，1 名がどちらとも言えないと答えた．残りの 1 名は難解には感じないと答えたものの，「単に私がコンピューターが不得意なだけだから…(客観的には難しいというものではないのだろう)」と言って難解ではないと回答していた．「仕事でこのシステムを使うよう指示された場合負担に感じる」との質問には，非テクニカルユーザーは 4 名が，テクニカルユーザーも 2 名が「そう思」っていた．

次に行った，今回開発したシステムの評価実験の質問紙調査では，非テクニカルユーザー 1 名を除く 9 名が，このシステムの利用は困難でないと回答した．

一方で，アドオンインストールの際，インストール完了がわかりづらい，メールのパスワードを聞かれた時に何のパスワードなのかわからないなどの問題も判明したため，インストール後に終了したという画面を表示するようにしたり，メールのパスワードだけでなくアドレスも同時に訊くようにするなどの修正を行った．

送信時に暗号化がなされているかがわかりにくいという問題もあり，これについて，ある被験者から，このシステムはどういう動作をするのか，どこに何が表示されるのかわからないという指摘があったため，アドオンインストール終了後，簡単な説明を表示するよう変更した．

6. 終わりに

本研究では，PGP によるメール本文の暗号化のためのインターフェースの複雑さや手間を排除し，一般ユーザーでも扱うことが可能なシステムを開発した．評価実験の結果，多少わかりづらさが残るものの，どのレベルのユーザーも，概ね問題なく操作が可能であり，現在普及している PGP 実装の UI と比較して，大幅に改善できたと言える．今後更に評価実験を行い，ユーザーの手間をより減らすことを今後の課題としたい．