6W-5

鍵不要のセキュアネットワークの提案 と Instant Message Web サービスへの実装

Proposal of a Secure Keyless Network and Implementing Instant Message Web Services

大矢 健太† 小瀬木 浩昭‡ 武田 正之†
Kenta Ohya Hiroaki Ozeki Masayuki Takeda

1. はじめに

ネットワークを通して秘密にしたい情報のやり取りをする際には暗号化によって秘密情報を秘匿する. しかし, 従来ある鍵暗号技術を用いたネットワーク通信では秘密情報がまるごと暗号化されるので, 鍵が解かれた場合や鍵が漏洩した場合に秘密情報が完全に漏洩してしまうという危険性がある. そのため, 鍵を安全に管理･配送することはもちろん, CPU の演算速度に対する解読時間を考慮した鍵寿命の設定や鍵の更新, 鍵の正当性の保証などさまざまなことが必要になる.

例えば, 電子メールの内容を暗号化することを考える場合, 既存の暗号化技術を用いると事前の相手との秘密裏な鍵の交換や第三者機関による鍵の正当性の保証などが必要になる. さらに, 毎回この鍵を使ってメール内容の暗号化, 復号化をしなければならない. これは困難であり, また手間がかかる.

本稿では秘密分散法を用いて秘密情報を単体では意味を持たない分散情報(これをシェアと呼ぶ)に分け, それを用いて通信を行うことにより利用者間で暗号化通信をするための鍵が必要なく, そのため鍵の管理や交換, 更新などが不要なセキュアなネットワークを提案する. さらに提案モデルの実現例として Instant Message への実装を示す.

2. 本提案モデル

提案モデルは,**(i)**秘密分散法の導入と, **(ii)**分散サーバの管理から構成される.

2.1. 秘密分散法の導入

秘密分散法は Shamir により 1979 年に提案され, 次のような性質を持つ[1]. **(a)**秘密情報を n 個のシェアに分割し, 任意の k 個以上 $(k \leqq n)$ のシェアの収集によって, 秘密情報の復元が可能になる(この k を閾値と言う). **(b)**k 個未満のシェアからでは, 秘密情報に関しての情報を全く得られない. これを(k,n)閾値法と言う.

2.2. 分散サーバの管理

利用者は, メッセージ送信など同じ機能を有した任意の複数個のサーバ S を選択して利用できるようにする. S の位置情報の管理は位置情報管理サーバ REG によって行われ, 閾値である k 個以上の S が結託しないとする. これにより, 秘密分散法の性質から秘密情報は全く漏洩しない. 利用者は REG へ認証によりログインすることで, 利用可能な S のリストを得る. また, REG は S からの問い合わせに対しては利用者の利用可能な S のリストを返す.

†東京理科大学 理工学部 情報科学科,
Dept. of Information Science, Tokyo University of Science
‡東京理科大学大学院 理工学研究科 情報科学専攻,
Graduate School of Science and Technology,
Tokyo University of Science

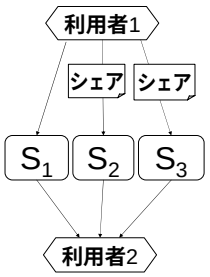


図 1 秘密分散法の導入による複数の通信路

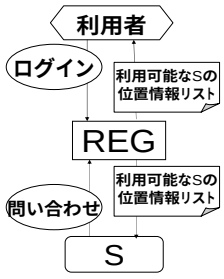
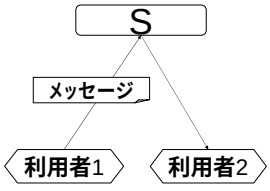


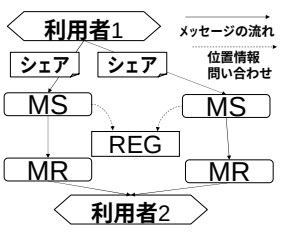
図 2 REG へのログイン

2.3. Instant Message への適用

Instant Message (IM) は, 相手の状態が事前にわかるプレゼンス通知機能と, 会話を行うメッセージ交換機能を有する. IM に提案手法を適用すると, 送信者はメッセージ送信サーバ MS, メッセージ受信サーバ MR を任意に選択利用できる. 送信者からシェアを送られた MS は REG に受信者の利用可能な MR を問い合わせ, そこへシェアを送る. また, 各自の REG へのアカウント名とメッセージの送信時刻を利用して各メッセージのシェアを識別する.



従来の IM モデル



提案手法を適用した IM モデル

図 3 従来のモデルとの比較

3. 実装に関する考察

利用者は 1 対 1 で会話を行うとし, アルゴリズムは閾値が分割数と同じ値である Shamir の(n, n)閾値法[1]を用いて, 会話内容となる文字数や分割数を変化させて評価を取った. なお, 実装言語として Java2 SDK, SOAP エンジンとして Apache AXIS, HTTP サーバ及び Servlet コンテナとして Apache Tomcat を用いて, SOAP1.1, WSDL1.1 準拠の Web サービスとして実装を行った. 各主体間の通信には SOAP/HTTP を採用した.

3.1. 分割数との関係

図 4 のグラフは分割数と文字数を変化させた場合の処理時間の変化である. 暗号化に必要な時間は文字数の増加, または分割数の増加に伴い長くなっていることがわかる. **図 5** のグラフは閾値の数だけ, つまり分割の数だけシェアがそろってから, 復号化に必要な時間を測定したものである. 暗号化の場合と同様に, 復号化に必要な時間は長くなっており, 閾値の増加とともに急激に増えている. 今回の実装では, 暗号化と復号化に必要な時間は 1 秒から 2 秒弱であった. 処理時間の上限を定めることにより, 分割数の取り得る範囲を定めることが可能である.

3.2. データサイズとの関係

3.2.1. 処理時間に関して

本手法では, データのサイズの増加によって処理に必要な時間が長くなってしまうので, ファイル交換やビデオチャットなどのような大きなデータのやり取りには有効とは言えない. そこで, 共通鍵を併用する方法を考える. 元データを X, 提案手法を用いての暗号化・復号化に必要な時間を E, D, さらに暗号化に用いる鍵を K, 鍵 K を用いての暗号化・復号化に必要な時間を E_k, D_k とすると, $E(X)+D(X)=E(K)+D(K)+E_K(X)+D_K(X)$ となるようなデータ X のサイズが求まる. これよりデータサイズごとに提案手法をそのまま用いるか, 鍵で暗号化してその鍵をやり取りするのかの適する方法が処理時間の面から決定できる.

3.2.2. 分割数と通信量に関して

秘密分散法を導入したことにより, 分割数を n として本手法を用いると通信量は n 倍になってしまう. 大きなデータをやり取りする場合にはこれは問題である. データ X と鍵 K のサイズを S で表し, さらに鍵 K による暗号化後のデータサイズを S_k とすると $S(X)\times n=S(K)\times n+S_k(X)$ となるようなデータサイズ X が求まる. これより, 処理時間の場合と同様に分割数・通信量に関して適する方法がわかる.

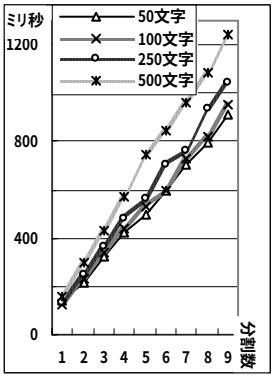


図 4 分割数ごとの暗号化に必要な時間

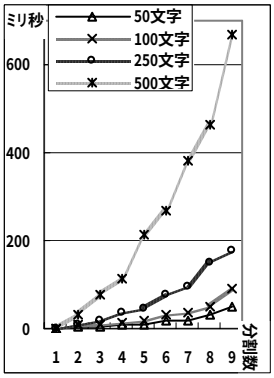


図 5 分割数ごとの復号化に必要な時間

3.3. サーバの不正・故障に対する検討

不正なサーバの割合と情報漏洩の危険性の関係を示した**図 6** より, 分割数を固定して閾値を増やすことは不正なサーバが多いときに有効であることがわかり, これより情報の重要度によって最適な閾値が設定可能である. 閾値の変化は暗号化時間に影響しないので, 名前・住所などのような重要度の高い情報は閾値を大きく, 重要度の低い情報は**図 7** から正常に届くように閾値を小さく暗号化することが有効である.

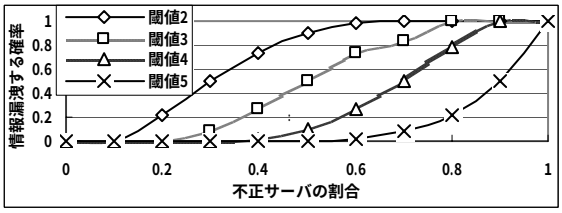


図 6 不正サーバの割合と情報漏洩の危険度の関係[注)]

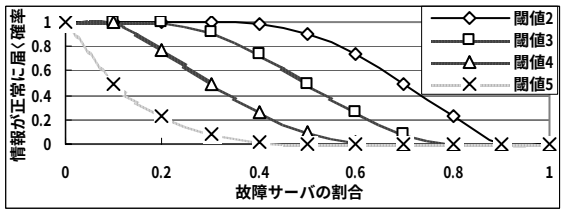


図 7 故障サーバの割合と情報が正常に届く確率の関係[注)]

3.4. REG についての考察

本手法では, REG の導入により複数の通信路を持つ動的なネットワーク構築が可能になり利用者間での鍵が不要になっている. この REG によって負荷がかかっていない S が動的に選択されれば, 一つの S で情報をやり取りする場合に比べて負荷の分散が期待できる. 耐故障性の向上も同様である. 今回の実装ではメッセージの送信の度に受信者の利用可能な S のリストを問い合わせているため, 毎回の REG へのアクセスが必要になるというデメリットがあるが, MS がリストをキャッシュすることにより, これを減らすことができると考えられる.

4. 関連研究

[2]では動的なグループ相手の暗号化通信の方法を提案・評価している. 提案している公開鍵利用方式の処理時間(30 人のグループで約 41 秒)と比べて, 本手法を人数分繰り返しグループに適用した場合の処理時間は同程度であり, さらに鍵の保証やグループメンバの離脱による鍵の変更も必要ないという優位性がある. しかし, 本手法では REG への問い合わせが常に必要である. 簡単に測定してみたところ, 全体に占める REG への問い合わせ回数は 20%程度であった. [3]のように日本語テキストをシェアとする秘密分散法と提案手法を組み合わせれば, 意味を持つ文がシェアとしてやり取りされることになり, 文章中に秘密情報が含まれているということを隠蔽できる.

5. まとめと今後の課題

本稿では, 秘密分散法を用いて情報を分割して通信を行うことにより, 鍵が不要なセキュアネットワークを提案した. 提案モデルの適用例として Instant Message への適用を紹介し, その実装に対する評価を行った. また, 他に提案手法が適用可能なものとして電子掲示板やメールなどのように蓄積されたデータが変化なくやり取りされるものが挙げられる. 今後は, そのような他のモデルへの適用やデータサイズ・処理時間・通信量に関しての効率的な手法と具体的な最適値を検討していきたい.

参考文献

[1] Shamir, A：How to share a secret, *Communication of the ACM*, Vol. 22, No. 11, pp. 612-613 (1979).

[2] 渡邉浩朗ら：P2P 環境下における動的グループ生成用暗号利用方式の評価, 情報処理学会論文誌, Vol. 44, No. 8, pp. 2155-2162(Aug. 2003).

[3] 滝澤修, 山村明弘：自然言語テキストを用いた秘密分散法, 情報処理学会論文誌, Vol. 45, No. 1, pp. 320-323(Jan. 2004).

注) 図 6, 図 7 の計算は全サーバ数を s, 図 6 では不正サーバを w, 図 7 では故障サーバを w としての次の式を用いた. なお, 分割数は 5 に固定して計算した.

図 6 $\sum_{i=0}({}_wC_{k+i}\cdot{}_{s-w}C_{n-(k+i)})\div{}_sC_n$

図 7 $\sum_{i=0}({}_{s-w}C_{k+i}\cdot{}_wC_{n-(k+i)})\div{}_sC_n$