RO-004

SSL における暗号設定確認ツールの提案と評価
A Proposal and Evaluation of a Tool to Check Cryptographic Settings in SSL

佐藤 亮太† 吉田 勝彦† 知加良 盛† 関 良明† 神田 雅透† 栢口 茂† 平田 真一†
Ryota Sato Katsuhiko Yoshida Sakae Chikara Yoshiaki Seki
Masayuki Kanda Shigeru Kayaguchi Shinichi Hirata

1．まえがき

近年，インターネット技術の発展と共に，我々のくらしの中でインターネットを用いた情報流通の重要さが増大している．インターネットは，単なる通信網としてだけではなく，組織体や社会の活動に必要な情報の収集・処理・伝達・利用にかかわる仕組みを包括する情報システムとして機能している．その中でも特に，生活に密着した機密性の高い情報のやりとりが必要なサービスを，インターネットを介して利用／提供する場合に，暗号通信はなくてはならない技術となっている．

一方で，GRID やクラウドのような計算機能力の向上や暗号解読技術の進展等に伴い，その暗号通信を支える暗号アルゴリズムの安全性が次第に低下することが懸念されており，これは暗号危殆化と呼ばれている．米国標準技術研究所(NIST: National Institute for Standards and Technology)は，その暗号危殆化対策の一つとして，米国政府が利用する共通鍵暗号や公開鍵暗号，ハッシュ関数などの暗号アルゴリズムを，2010 年末までに，より安全性の高いものへ移行させる方針を示している[1]．また，日本においても，内閣官房情報セキュリティセンター(NISC: National Information Security Center)が，2008 年に情報セキュリティ政策会議にて決定した「政府機関の情報システムにおいて使用されている暗号アルゴリズム SHA-1 及び RSA1024 に係る移行指針」を公表している[2]．これらの方針もあり，政府機関のみならず，多くの企業にとっても暗号危殆化への対策の必要性が生じている．そして，この対策に伴い生じる各種問題は「暗号の 2010 年問題」と総称され，議論やサービスを提供する企業への啓発活動がされている[3]．

暗号危殆化対策の実施ステップは，現状調査，対策立案，対策実施となる．その最初のステップでは，対象システムにおいて，どこでどのような暗号アルゴリズムが利用されていて，その暗号アルゴリズムが移行すべき対象であるか否かを調査する必要がある．しかし，利用されている暗号アルゴリズムの調査は，そのシステムを構成する機器や設定等により調査方法が異なるため，実施に時間と人手がかかる．また，暗号アルゴリズムの危殆化の評価には専門的な知識が必要とされるため，一般的なシステム管理者等では，その実施が非常に難しい．

そこで本稿では，主に現状調査のステップをサポートするツールの提案，評価を通じて，暗号危殆化対策の促進させることを目的とする．2 章で関連研究を概観し，3 章にて，危殆化対策を実施する暗号技術のうち，インターネット上で広く利用されている SSL/TLS プロトコル（以下，単に SSL プロトコル）に着目し，日本の政府・公共系と金融系のサーバにおける暗号危殆化の状況について予備調査を実施する．4 章では，現状調査のステップをサポートするツールに必要な要件について整理する．5 章で，これら要件を満たす暗号設定確認ツールを提案し，その具体的な実装方法について述べる．6 章では，暗号設定確認ツールを要件に照らしながら評価を実施し，7 章で本稿をまとめる．

†NTT 情報流通プラットフォーム研究所，
NTT Information Sharing Platform Laboratories

2．関連研究

本章では，暗号危殆化の原理について改めて整理し，その対策として，SSL プロトコル内で利用される共通鍵暗号と公開鍵暗号のハイブリッド暗号やサーバの正当性を検証するデジタル証明書の先行研究について概観する．

2.1 暗号危殆化の原理

現代暗号の安全性は暗号アルゴリズムの解読に必要な計算量が莫大であり，現在の計算機では現実的な時間で解読できないことに立脚している．例えば，広く利用されている公開鍵暗号 RSA は桁数が大きい素因数分解の困難性を基にしている．しかし，GRID やクラウドのような「計算機能力の向上」や素因数分解問題の解法や離散対数問題の解法の効率化等による「暗号解読手法の進歩」，また量子コンピュータの完成等の「計算機モデルの変化」により，計算量的に安全とされていた暗号アルゴリズムの安全性が低下していく．これが，暗号が危殆化する際の原理である[4].

素因数分解に対する RSA の安全性を DES の安全性と対比させた著名な研究が行われている[5]．RSA 危殆化の進行を踏まえ DES の安全性がきわめて高かった時期を 1982 年と想定し，その時点での DES の安全性と等価であるためには RSA の鍵サイズがどれだけ必要であるかを，安全性の基準，単体コストあたりの計算量，解読にかける予算，解読アルゴリズムの進化の四つの評価指標を定義して分析している．

2.2 暗号危殆化対策

暗号の危殆化は必然性を有し，その影響が広範に及ぶことが予想されるが，専門家および政府における認識にばらつきがあり，技術的および制度的対策が不十分である．そこで，暗号の危殆化問題に対して，専門家および政府を始めとして共通認識をはかり，主に電子政府における暗号アルゴリズムの危殆化の影響範囲を把握することを目的として調査が行われている[5].

暗号を利用したサービスの提供者と利用者間において，暗号危殆化問題を考慮したうえで暗号利用における合意をはかることを目指し,暗号 SLA が提案されている[6][7]. 暗号 SLA では，暗号危殆化状態を容易に把握するための指標として暗号 SLA レベルを定義している．暗号 SLA レ

ベルは，関係者間で合意を目指す際のガイドラインの立場をとっている．また，暗号 SLA を利用した関係者間での合意を支援するためのサポートツールも開発し，暗号 SLA の有用性について評価することを目的としてサポートツールを利用したロールプレイングシミュレーションを実施している．これからの情報社会において暗号危殆化を適切に理解し，新しい暗号系へスムーズに移行計画を検討していくことが急務な課題であるとしている[6].

2.3 デジタル署名の危殆化対策

デジタル署名の利用が増加する傾向にあり，デジタル署名の長期利用を考慮すると，危殆化を確認した際の影響を考慮せざるをえない．デジタル署名の安全性は公開鍵暗号に依存しているので，公開暗号危殆化時の署名付き文書への影響の分析とその対策の検討が不可欠である．そこで，公開鍵暗号の危殆化が近く生じることが明確になった場合に，既存の署名付き文書の証拠性を確保するために必要な対策の最適な組合せを費用とリスク低減効果のバランスを考慮して求める方法が提案されている[8].

また，既に広く利用されている SSL 及び SSL 証明書の事例を示すことで移行問題の複雑さと重要さを説明し，今後の取り組むべき課題について考察を行っている[9].

さらに，基本的な暗号アルゴリズムの安全性を立証できなければ，公開鍵認証基盤のすべての安全上の問題も解決できないため，コンポーネントを取り替えるためにフェールセイフの概念が提案されている[10].

3．予備調査

前章で述べた通り，暗号危殆化問題やその対策に関する多くの関連研究が存在する．本章では，これら研究成果の，暗号通信サービスを提供する企業等への浸透状況を，対策の進捗状況をもって確認する．具体的には，インターネット上で広く利用されている SSL プロトコルを利用した暗号通信を行うサーバについて，暗号危殆化の状況を調査する．そして，その対策にあたっての調査手順について考察する．

3.1 政府・公共系サーバ，金融系サーバの実態

インターネット上に公開されている日本の金融系のサーバと政府・公共系のサーバを対象として，そのサーバで利用可能となっている暗号アルゴリズムについての調査結果を付図 1（Appendix A を参照）に示す．調査対象サーバの数は，2008 年は金融：138，政府・公共：147，2009 年は金融：136，政府・公共：142，2010 年は金融：117，政府・公共：130 である．各年で同じサーバを調査対象としているが，対象サーバ数が年毎に異なる理由は，当該サーバの統廃合などが発生したためである．

金融系サーバ（付図 1 の左図）では AES256-SHA など，暗号危殆化の観点から安全性の高い暗号アルゴリズムにやや上昇傾向が見られ，輸出規制対応暗号や SSL2.0 利用などの安全性の低い暗号は利用可能設定率が下がっている．しかし，危殆化の懸念がある RC4-MD5 [11]の使用率は依然として高いままである．また，政府・公共系サーバ（付図 1 の右図）でも金融系と同じ傾向が見られ，RC4-MD5 が利用可能な設定のままである [12],[13].

さらに，これら調査対象サーバにおけるサーバ証明書で使用している暗号アルゴリズムも同様に調査を実施した．その結果を付図 2（Appendix A を参照）に示す．

付図 1 の結果とは対照的に，2009 年まで使用されていた MD5 with RSA 1024[bit]の使用が，金融系サーバ，政府・公共系サーバ共に，大幅に減少している．また，この結果と同調するように，SHA1 with RSA 2048[bit]の使用が増加していることが分かる．なお，金融系サーバにおいては，EV 証明書の使用率の上昇傾向が見られた．EV 証明書は従来のサーバ証明書に比べて取得認定を厳密にしたものであり，暗号危殆化というよりはサーバ証明書の信頼性向上を目的としている．

サーバ証明書の結果は，利用可能な暗号アルゴリズムの結果とは対照的に，暗号危殆化対策の進行を示している．対策が進んでいる要因は，サーバ証明書を発行する組織が，安全性の低い暗号アルゴリズムを用いた証明書を新規発行停止したためである[3]．つまり，これら暗号危殆化に対する知識や意識が高い組織のリードにより，企業等にとっては，ある種，受動的に暗号危殆化対策がされたものと推察される．

サーバ証明書は 1 年から数年単位で更新されることが一般的であり，それを発行する組織の対策により，企業等への受動的な対策実施が可能である．しかし，SSL プロトコルにおける暗号アルゴリズムは，サーバ証明書の中だけでなく，通信暗号化にも利用されている．ここで利用される暗号アルゴリズムについては，サーバ証明書のように，定期的な更新契機が一般的には設けられていないため，企業等による能動的な対策実施が求められる．

3.2 暗号危殆化対策に向けた調査手順と課題

前節の実態調査により，インターネット上に存在する SSL プロトコルを利用した暗号通信を行うサーバの多くで危殆化した暗号が使用可能となっており，企業等による能動的な対策の必要性が確認された．そこで本節では，具体的な対策手順を検討し，その実施にあたっての課題を明らかにする．

企業等において暗号危殆化対策を実施するステップは，大きく分けて，現状調査，対策立案，対策実施となる．本稿では，この最初のステップである現状調査の実施に焦点を絞る．ここでは，以下の 2 項目の実施が考えられる．

①サーバで利用可能となっている暗号の把握

②その暗号が危殆化したものであるか否かの判断

これら実施項目を確認する際の具体例として，Web サーバのうち最もシェアの大きい apache と OpenSSL を用いた暗号通信を提供するサーバを想定する．

実施項目①は，サーバの管理者等がサーバへログインをし，その設定等ファイルや下記コマンドにて確認することが想定される．

```
> openssl␣ciphers␣-v
```

上記コマンドの実行結果例を図 1 に示す．”DHE-RSA-AES256-SHA”などの文字列が列挙されており，これらは CipherSuites と呼ばれている．例えば，前節の実態調査で議論した”RC4-MD5”も CipherSuites の一つであり，SSL プロトコルにおいて，公開鍵暗号に RSA，共通鍵暗号に RC4，ハッシュ関数に MD5 を利用していることを意味している．

```
[root@venus ryota]# openssl ciphers -v
DHE-RSA-AES256-SHA      SSLv3 Kx=DH       Au=RSA  Enc=AES(256)  Mac=SHA1
DHE-DSS-AES256-SHA      SSLv3 Kx=DH       Au=DSS  Enc=AES(256)  Mac=SHA1
AES256-SHA              SSLv3 Kx=RSA      Au=RSA  Enc=AES(256)  Mac=SHA1
KRB5-DES-CBC3-MD5       SSLv3 Kx=KRB5     Au=KRB5 Enc=3DES(168) Mac=MD5
KRB5-DES-CBC3-SHA       SSLv3 Kx=KRB5     Au=KRB5 Enc=3DES(168) Mac=SHA1
EDH-RSA-DES-CBC3-SHA    SSLv3 Kx=DH       Au=RSA  Enc=3DES(168) Mac=SHA1
```

図 1　CipherSuites の一覧(一部抜粋)

実施項目②は，①で明らかとなったサーバ上で利用可能な各 CipherSuite 関して，そこに含まれる暗号アルゴリズムそれぞれに対し，危殆化に関する調査を行う．

上述した実施項目①，②の具体的な調査方法に基づき，その実施にあたっての課題を考察する．①に関しては，利用している Web サーバ毎にその設定ファイル等の記載や調査する際に使えるコマンドなど，個別に調査方法を確認する必要がある．さらに，サーバ管理者等がサーバに対し，コンソールなどから直接調査を実施することが必要となり，調査者や調査場所，端末等の制限がある．また②については，暗号アルゴリズムの危殆化に関する知識は専門性が高く，調査実施者がその知識の習得をすることは非常にコストが高くなると考えられる．

従って，これらの課題を解決し，企業等において暗号危殆化対策の第一歩を踏み出すためには，上記①，②の実行をサポートするためのツールが有効となる．

4．要求仕様

前章では，暗号危殆化対策をサポートするツールが，その対策促進に有効であることを述べた．本章では，このツール（以後，暗号設定確認ツール）に必要な要件を，予備調査を実施した知見に基づいて整理する．

4.1 機能要件

現状調査の実行性向上の観点から，暗号設定確認ツールの機能要件として下記 3 点を挙げる．

要件1. 調査対象システム個別のアプリケーションや設定等に依らず調査できること．

要件2. 調査対象システムを外部から調査できること．

要件3. 暗号アルゴリズムの危殆化に関する情報を用いて，危殆化状況の評価を実施できること．

要件 1 では，各システム固有のアプリケーションやその設定等により調査方法が異なる場合でも，画一的に調査可能とする．これにより，調査を容易にすると共に，調査に係る時間や質の平準化を目的としている．

要件 2 では，直接サーバへログインし，操作できない場合でも調査可能とする．これにより，調査実施に関する人や場所，端末等に関する制限の減少を目的としている．

要件 3 では，専門家による判断が必要となる暗号の危殆化情報を別途用意し，調査実施者と危殆化の判定を行う者を分離する．これにより，調査実施者が必ずしも暗号の危殆化情報についての十分な知識を習得する必要はなく，効率的な調査実施が想定される．

次に，現状調査の網羅性の観点から，要件 2 点を挙げる．

要件4. 対象システムで利用が想定される暗号を網羅的に調査できること．

要件5. 利用シーンにおいて，対象システムで最終的に選択/利用される暗号の利用状況を把握できること．

要件 4 に関しては，IANA(Internet Assigned Number Authority)に登録されている SSL プロトコルで利用可能な CipherSuites は，番号だけの予約も含め 551 個存在する．これら全て CipherSuites に関する危殆化状況を調査し，対象システムで危殆化した暗号が利用可能ではないことを保障する．

要件 5 に関しては，SSL プロトコルに限らず，多くの暗号プロトコルにおいて，ある 2 者間で暗号を利用した通信等を確立する際に，互いの利用可能な CipherSuites の一覧から，互いのもつ優先順位に基づき一つを選択するネゴシエーションを行う．この優先順位には，必ずしも暗号の危殆化が考慮されていない．そこで，サービス提供対象の端末との間で採用される CipherSuite とその危殆化状況を確認し，暗号危殆化対策の要否の判断に利用する．

4.2 性能要件

暗号設定確認ツールの調査時間を性能要件として規定する．調査の実行性の観点から，調査時間はサーバ管理者が直接サーバへログインして調査する時間との比較を行う．具体的には，利用可能な CipherSuites の一覧を取得し，それらと各 CipherSuites の危殆化状況を記載したリスト（以後，危殆化リスト）との照合をする時間より短くあるべきである．実際に apache+OpenSSL のサーバを用いて実施すると，サーバへのログインからリストとの照合まで含めた調査速度は，1 サーバあたり 5 分程度かかった．ただし，この場合は OpenSSL が実装している CipherSuites 103 個が対象であり，IANA が規定する 551 個を対象としたた場合は，その数に比例して調査時間は長くなる．つまり，5 倍程度の調査時間がかかると推定し，性能要件としては 25[分/サーバ]を目標とする．

5．実装

前章で機能要件を 5 点と性能要件 1 点を整理した．これに基づき，暗号設定確認ツールを実装する．実装の全体像を図 2 に示す．

図 2　暗号設定確認ツールの全体構成

大きく四つに構成に分けることができ，入出力 I/F と危殆化リスト，調査実行コマンドである．以降では，各構成要素の実装内容について詳細に述べる．

5.1 調査実行コマンド

調査実行コマンドは調査対象サーバを外部から調査するためのコマンドである．ベースには OpenSSL を利用し，そのコマンドの一つである s_client を利用する．調査対象サーバと NW で繋がった OpenSSL がインストールされている端末から下記コマンドを実行すると，指定した CipherSuite での暗号通信の接続を試みることができる．

> openssl␣s_client␣-connect␣*www.example.com:443*␣-cipher␣*cipherlist*

この時，”␣”はスペースを表し，サーバのアドレスとポートをそれぞれ *www.example.com*，*443* で指定し，CipherSuite を *cipherlist* で指定する．この結果の例を図 3 に示す．

```
[root@venus ryota]# openssl s_client -connect localhost:443 -cipher RC4-MD5
CONNECTED(00000003)
depth=0 /C=JP/ST=Berkshire/L=Koto-ku/O=ryo/OU=ryo/CN=ryo.dnsalias.com
verify error:num=18:self signed certificate
verify return:1
```

図 3　s_client による調査結果(一部抜粋)

この例では，CipherSuite として RC4-MD5 で接続を試み，CONNECTED の結果が得られている．つまり，この調査対象サーバでは，RC4-MD5 が利用可能であると分かる．従って，このコマンドを利用し，CipherSuite を順次変化させ，その結果を集計すると，図 1 に示すような調査対象サーバで利用可能な CipherSuite の一覧が得られる．

ただし，前述の通り OpenSSL では IANA が規定する全ての CipherSuites を実装していない．そこで我々はこの s_client コマンドを拡張した．

> openssl␣s_client␣-connect␣*www.example.com:443*␣-cipherid␣*cipher-ID*

上記の *cipher-ID* 部分に IANA が規定する CipherSuite の ID 全てを指定可能とする．この時，その ID が示す CipherSuites の実体を OpenSSL へ組み込むことをせず，SSL プロトコルの初期のハンドシェイクのみを実行するよう工夫を行っている．これにより，その ID をもつ CipherSuites が利用可能か否かの結果を得ながらも，s_client への拡張を最小限にとどめることが出来ている．

また，個別に CipherSuite を調査する機能に加えて，複数の CipherSuites を一括で調査する機能も実装した．コマンドを以下に記す．

> openssl␣s_client␣-connect␣*www.example.com:443*␣-cipheridfile␣*cipher-IDFile*

このように，cipheridfile オプションを指定し，*cipher-IDFile* 部分には IANA が規定する CipherSuite の ID が複数記載された外部ファイルを指定することで，その中に少なくとも一つ利用可能なものがあるか否か，を確認できる．これにより，利用されている可能性が低いと思われる CipherSuites については，一括で利用可否を判別でき，調査の効率性と網羅性を担保できる．

5.2 危殆化リスト

危殆化リストは CipsherSuite 個々の危殆化状況を予め評価したリストである．各 CipherSuite に対し，使用可能，採用自粛，使用停止の 3 段階での評価フラグを設けている．この評価にあたっては，CipherSuites に含まれる暗号アルゴリズムの危殆化状況に基づく評価だけでなく，危殆化対策の実施の要否の判断を含めてもよい．つまり，危殆化した暗号であっても，そのリスクを受容するとの経営判断があれば，使用可能と判定することもできる．このように，危殆化リストは，各企業等やサービスの種類によって適宜変化するものであるため，外部ファイル化した実装をしている．

5.3 入力 I/F

入力 I/F は設定状況確認ツールの利用者からの入力を受け付け，その入力に従い調査実行コマンドを実行する．調査者のスキルレベルが様々であっても利用できるよう，Excel をベースとし，簡易な入力項目としている．付図 3（Appendix A を参照）に入力 I/F を示す．

ホスト指定欄では，調査対象サーバの URL や IP アドレスとポート番号をコロンで繋いで入力する．このアドレスとポートが調査実行コマンドへと渡される．また，調査対象サーバが複数の場合は，それらアドレスの一覧が記載されたファイルを参照指定することもできる．

次に接続間隔指定欄では，一つの調査対象サーバに対して，調査実行コマンドが CipherSuite を一つ一つ確認する際に，その実行する間隔を秒単位で指定できる．これにより，調査対象サーバへ短期間に多くのリクエストを投げることによる負荷の軽減が可能となる．

最後に Proxy やそのポート，Proxy 認証がある場合はそのユーザ ID やパスワードの入力欄も設けており，これらも調査実行コマンドへと渡される．

5.4 出力 I/F

調査結果の出力 I/F も Excel を利用しており，CipherSuites，各ブラウザで選択される CipherSuite，証明書の三つの情報を出力する．

5.4.1 CipherSuites

CipherSuites の利用可否一覧とその危殆化状況の判定結果を出力する．例を付図 4（Appendix A を参照）に示す．

各 CipherSuite が，調査対象サーバにおいて利用可能であれば ON，利用不可能であれば OFF として表示する．また，危殆化リストに従って，使用可能，採用自粛，使用停止をそれぞれ青，黄，赤の 3 色で色分けて評価をしている．この結果から，黄色や赤色の CipherSuites が ON である場合は，危殆化対策に向けた検討が必要がとなる．

また，前節で述べた cipheridfile オプションを利用した一括での CipherSuites の調査結果を付図 5（Appendix A を参照）に示す．OFF と表記されている場合は，*cipher-IDFile* に記載された全ての CipherSuites が利用不可であることが確認できる．

5.4.2 各ブラウザで選択される CipherSuite

ブラウザがサーバとネゴシエーションを実施する際に提示する CipherSuites とその優先順位を記したファイルを cipheridfile として指定し，結果を取得した例を図 4 に示す．

		site
		www.example.com:443
ブラウザ毎の CipherSuite	IE8-Win7	TLS_RSA_WITH_AES_128_CBC_SHA
	IE8-WinXP	TLS_RSA_WITH_RC4_128_MD5
	IE7-WinXP	TLS_RSA_WITH_RC4_128_MD5
	FireFox3.6-Win7	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA
	Chrome8-Win7	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA

図 4　ブラウザ毎の CipherSuite の調査結果（一部抜粋）

各ブラウザ名とそのバージョンだけでなく，利用 OS も記載している．これは多くのブラウザにおいて，それらで利用可能な CipherSuites が，その端末の OS にも依存しているためである．選択される CipherSuite を危殆化リストと照らし合わせ，3 色で評価している．

このように各種ブラウザ（OS）を模擬し，一つのツールで実行可能であるため，実際に複数のブラウザを用意

する必要がない．従って，調査対象サーバとの間で選択される CipherSuite とその危殆化状況を効率よく評価することでき，調査コストの削減が期待される．

5.4.3 証明書

調査対象サーバがもつ証明書の情報を出力する．これは下記コマンドを利用した結果を整形して出力する．

> openssl␣s_client␣-connect␣*www.example.com:443*␣-showcerts

		site
		www.example.com
サーバ証明書	version	3 (0x2)
	issuer	C = JP, O = NTT, OU = EX Servers CA
	subject	C = JP, O = NTT, OU = EX Servers, CN = example1.com
	notBefore	May 18 05:39:35 2009 GMT
	notAfter	May 18 05:39:35 2029 GMT
	public key algorithm:size	rsaEncryption:1024
	signature algorithm:size	sha1WithRSAEncryption:2048

図 5　サーバ証明書の調査結果（一部抜粋）

図 5 は調査結果例である．Version, Issuer, Subject, notBefore, notAfter はそれぞれ証明書のバージョン，発行者，対象者，利用開始日，利用終了日を表す．また，public key algorithm:size には証明書に利用されている公開鍵暗号アルゴリズムとその鍵長を，signature algorithm:size には署名アルゴリズムとその鍵長を記し，危殆化リストを用いて危殆化状況を評価し，色分けしている．

6．評価検証

前章では，4 章で挙げた要件に基づき，暗号設定確認ツールの実装方法について述べた．本章では，このツールが各要件を満たしていることを確認/検証する．

6.1 機能要件の検証

要件1. 個々の CiphserSuite での接続を順次試行し，その接続可否を確認する手法を用いることで，調査対象システム個別のアプリケーションや設定等に依らない調査を実現した．

要件2. OpenSSL の s_client コマンドをベースに実装することによって，調査対象システムの遠隔調査を実現した．また，様々な NW 環境を想定し，Proxy や Proxy 認証にも対応できるようにしている．

要件3. 危殆化リストを外部ファイル化することによって，暗号アルゴリズムの危殆化に関する情報を用いて，危殆化状況の評価を実現した．

要件4. s_client を拡張し，IANA で規定された全ての CipherSuites を調査可能として，対象システムで利用が想定される暗号の網羅的調査を実現した．

要件5. 各ブラウザの CipherSuites とその優先順位に従って調査する機能によって，ある利用シーンにおいて，対象システムで最終的に選択/利用される暗号の利用状況把握を実現した．

以上のように要件 1〜5 を満たした暗号設定確認ツールを実装することが出来た．さらに，このツールを用いた調査を実施した際に，サーバに与える影響についても考察した．このとき，二つの観点から影響を評価している．

一つ目は，調査対象サーバにおけるログの出力である．通常の HTTPS アクセスとしてのログがサーバ側に残るだけであれば一般的に問題はないと考えられる．しかし，今回のように SSL プロトコルのシーケンスを中断することで，エラーや異常値として判断されると，場合によってはサーバに多量のログが出力される可能性もある．これにより，サーバ運用時の解析業務等で必要なログを埋没，消失させてしまうことも想定される．そこで，我々はシーケンスを中断した場合に調査対象サーバの Web サーバに記録されるログを調査した．Web サーバが apache の場合の結果が図 6 である．

```
[Fri Oct 15 11:41:13 2010] [info] [client 127.0.0.1] Connection to child 3
established (server www.example.com:443)
[Fri Oct 15 11:41:13 2010] [info] Seeding PRNG with 136 bytes of entropy
[Fri Oct 15 11:41:13 2010] [debug] ssl_engine_kernel.c(1866): OpenSSL:
Handshake: start
      (中略)
[Fri Oct 15 11:41:13 2010] [debug] ssl_engine_kernel.c(1903): OpenSSL: Exit:
error in SSLv3 read client certificate A
[Fri Oct 15 11:41:13 2010] [info] [client 127.0.0.1] (104)Connection reset by
peer: SSL handshake interrupted by system [Hint: Stop button pressed in
browser?!]
[Fri Oct 15 11:41:13 2010] [info] [client 127.0.0.1] Connection closed to
child 3 with abortive shutdown (server www.example.com:443)
```

図 6　Web サーバのログ出力例

apache のログレベルは debug, info, notice, warn, error, crit, alert, emerg の順で緊急度が設定されており，今回は debug 以上全てのログが出力されるよう設定した．図 6 から，info 以下に設定されている場合，シーケンスが中断されたことが記録されることが分かる．しかし，apache のデフォルト設定では，ログレベルが warn 以上に設定されており，この s_client の拡張による影響は少ないと考えられる．なお，Web サーバが IIS の場合は，ログレベルの設定に依らずシーケンス中断によるログが出力されないことを確認している．

二つ目は，調査対象サーバにかかる負荷である．調査対象のサーバのマシンスペックやそれが接続されたネットワークの帯域などが，このようなツールによるアクセスに耐えられない場合も想定される．その場合には，このようなツールの利用者側で意図しない形で，そのサーバが提供するサービスの妨害をしてしまう可能性もある．

この点に関しても，本ツールでは s_client を拡張し cipheridfile オプションを実装したことで，利用頻度の低い CipherSuite は一括調査できるため，調査対象サーバへのアクセス回数を減らすことができる．また，入力 I/F に接続間隔指定欄を設けているため，個別に CipherSuites を調査する際でも，そのアクセス間隔を空けることによって，サーバにかかる負荷を時間的に分散できる．

以上より，調査対象サーバへの影響を考慮しながら，暗号設定確認ツールに必要な機能要件 1〜5 を実装できていることが検証された．

6.2 性能要件の検証

性能要件について評価検証を実施する．目標性能としては 25[分/サーバ](=1,500[秒/サーバ])を設定している．

調査対象サーバ(OS: CentOS 5.5, CPU: Athlon 2.6GHz, Mem: 1.0GB)と暗号設定確認ツールを搭載した端末（OS: Windows7, CPU: Intel Core i7 1.2GHz, Mem: 4GB）を Hub 経由で接続した系にて実験した．個別に調査する CipherSuite の数を変化させながら調査を実施し，その調査時間を測定した結果を図 7 に示す．

代表的な 3 点をプロットすると，調査時間は個別に調査する CipherSuite の数に比例している．従って，IANA で規定されている全ての CipherSuites を個別に調査した場合でも，性能要件の値より低く，要件を満たしていることが検証できた． 調査対象サーバ数の増加と共に，このツールによる効率化の影響は大きくなると考えられる．

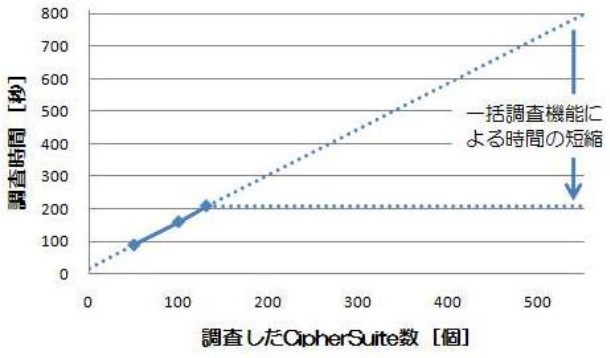


図 7　個別調査する CipherSuite の数と調査時間

7．まとめ

暗号危殆化対策は，現状調査，対策立案，対策実施の三つステップから構成される．その最初のステップである現状調査ステップでは，対象システムにおける，暗号アルゴリズムが利用状況と，その暗号アルゴリズムが移行すべき対象であるか否かを調査する必要がある．しかし，利用されている暗号アルゴリズムの調査は，そのシステムを構成する機器や設定等により調査方法が異なるため，実施に時間と人手がかかる．また，暗号アルゴリズムの危殆化の評価には専門的な知識が必要とされるため，一般的なシステム管理者等では，その実施が非常に難しい．

そこで，本稿では暗号危殆化対策を促進させることを目的として，主に現状調査のステップをサポートするためのツールを提案し評価を行った．前準備として，暗号危殆化対策の先行研究を整理した上で，政府/公共系や金融系の Web サーバを対象に危殆化に関する予備調査を実施し，世の中の暗号危殆化対策が進んでいないことを明らかにした．現状調査ステップでは，時間と人手がかかり，専門的な知識も必要とされるため，一般的なシステム管理者等では，その実施が非常に難しいことから，現状調査ステップをサポートするための暗号設定確認ツールを提案することの有用性を確認した．

暗号設定確認ツールに必要な以下の五つの機能要件を整理するとともに，性能要件を設定した．

要件 1. 調査対象システム個別のアプリケーションや設定等に依らず調査できること．

要件 2. 調査対象システムを外部から調査できること．

要件 3. 暗号アルゴリズムの危殆化に関する情報を用いて，危殆化状況の評価を実施できること．

要件 4. 対象システムで利用が想定される暗号を網羅的に調査できること．

要件 5. 利用シーンにおいて，対象システムで最終的に選択/利用される暗号の利用状況を把握できること．

これらの要件を基に，SSL を利用した HTTPS サーバの暗号危殆化対策をサポートする暗号設定確認ツールを開発した．実装においては，汎用的で使用実績も高い OpenSSL の s_client コマンドをベースに，IANA で規定されている SSL プロトコルで利用可能な全ての CipherSuites を調査可能となるように，性能要件を満たす工夫を施しながら拡張した．開発した暗号設定確認ツールが，機能要件 1～5 および性能要件を満たしていることを実装により動作確認するとともに，調査対象サーバへ不要なログを残さないことの確認や，多くの通信を発生させ負荷をかけない配慮も可能であることも運用によって確認した．

現状調査を実施する際に本ツールを利用することで，対象システムの構成や設定に依らず，また外部から調査可能であるため，調査に係る時間や質を統一し，実施時の人や場所，時間等に係るコストや制限を削減できる．調査の実行性を向上させることで，企業等は暗号危殆化対策の計画，管理，実行が容易となり，その推進に大きく寄与することができる．

参考文献

1) Elaine Barker, William Barker, William Burr, William Polk, and Miles Smid, “Recommendation on Key Management SP800-57-Part-1-revised2 Mar08-2007, ” NIST, http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf, 参照 Oct.26,2010.

2) NSIC(内閣官房情報セキュリティセンター), “政府期間の情報システムにおいて使用されている暗号アルゴリズム SHA-1 及び RSA1024 に係る移行指針”, 情報セキュリティ政策会議, Apr.2009.

3) VeriSign, “WHITE PAPER「暗号アルゴリズムにおける 2010 年問題」対応ガイドー 問題なく移行するために抑えておきたい, サーバ管理者のためのチェックリスト 10 項目,” VeriSign, https://www.verisign.co.jp/cgi-bin/mf.cgi?n=wp2010.

4) 佐々木良一: 公開鍵暗号危殆化対策のためのリスク評価, オペレーションズ・リサーチ, Vol.54, No.3, pp.155-160 (2009).

5) 独立行政法人情報処理機構: 暗号の危殆化に関する調査 報告書, (2005).

6) 猪俣敦夫, 大山義仁, 岡本栄司: 暗号危殆化に対する暗号 SLA の提案と支援ツールの実現, 情報処理学会論文誌, Vol.48, No.1, pp.178-188 (2007).

7) 猪俣敦夫, 岡本栄司: 我々をとりまく情報社会と暗号危殆化のかかわり, 情報処理, Vol.51, No.5, pp.528 (2010).

8) 藤本 肇, 上田祐輔, 佐々木良一: デジタル署名付文書への公開鍵暗号危殆化対策の組合せ最適化法の提案と一適用, 情報処理学会論文誌, Vol.49, No.3, pp1105-1118 (2008).

9) 島岡政基, 松本 泰: SSL 証明書の事例にみる暗号アルゴリズムの移行問題 -収束しない 2010 年問題-, 電子情報通信学会論文誌 B, Vol J94-B, No.1, pp.1-13 (2011).

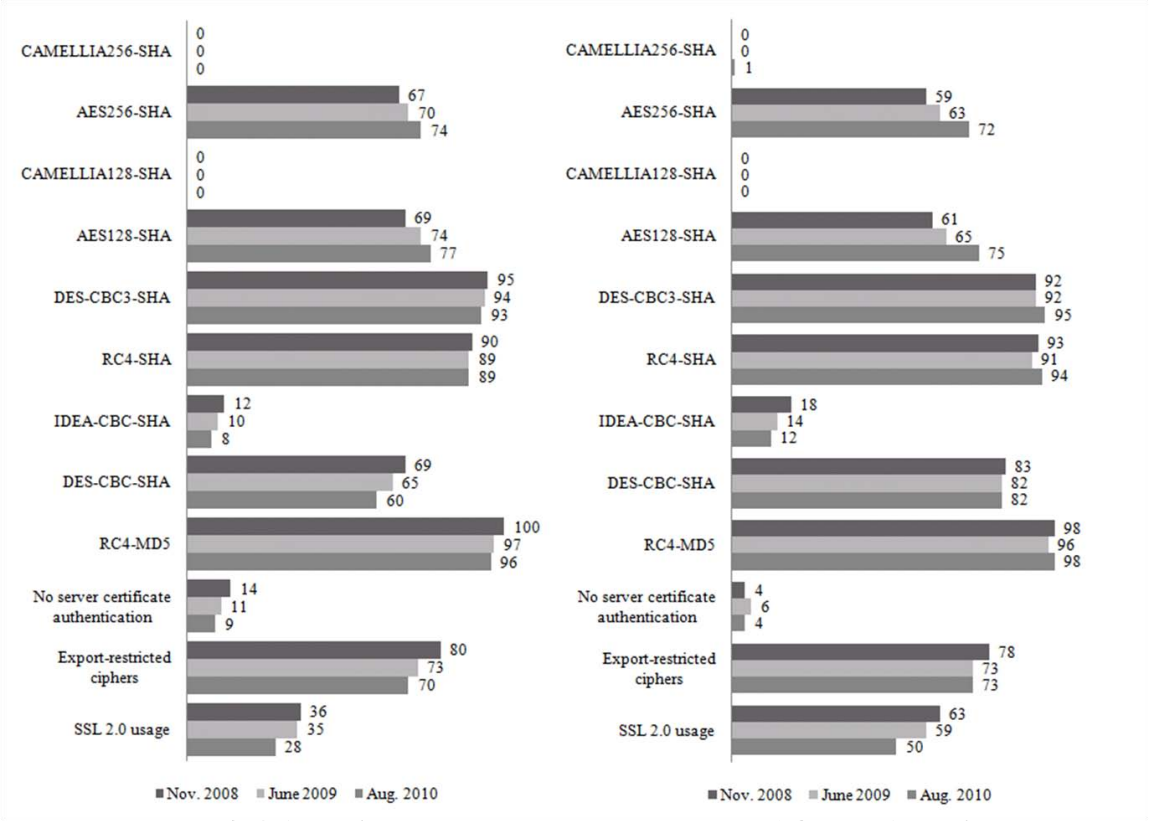
10) Michael Hartmann, Sonke Maseberg: Replacement of Components in Public Key Infrastructures, IECON’01, The 27th Annual Conference of the IEEE Industrial Electronics Society, pp.2012-2016 (2001).

11) 安田 幹, 佐々木悠: 暗号学的ハッシュ関数 -安全神話の崩壊と新たなる挑戦, IEICE Fundamentals Review, Vol.4, No.1 http://w2.gakkai- web.net/gakkai/ieice/vol4no1pdf/vol4no1_57.pdf.

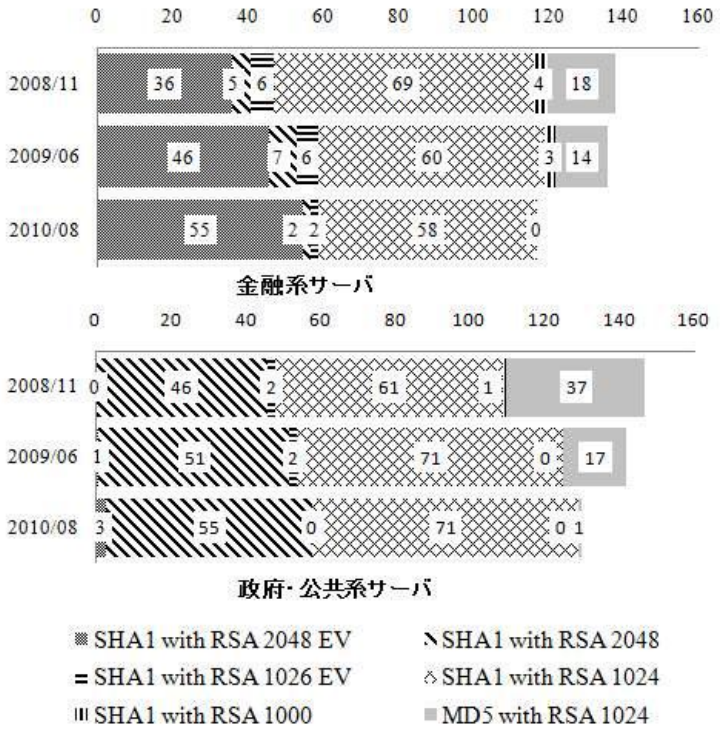
12) 高野誠士, 佐藤亮太, 武藤健一郎, 知加良盛, 神田雅透, 関 良明: SSL における暗号危殆化サンプル調査とその考察, 電子情報通信学会技術報告, LOIS2010-38 (ISEC2010-59), pp.65-72 (2010).

13) 佐藤亮太, 関 良明, 吉田勝彦, 栢口 茂, 平田真一: セキュリティプロトコルにおける暗号アルゴリズム調査手法の提案-SSL で利用可能な暗号アルゴリズムの暗号危殆化対策-, 電子情報通信学会技術報告, LOIS2010-39 (ISEC2010-60), pp.73-80 (2010).

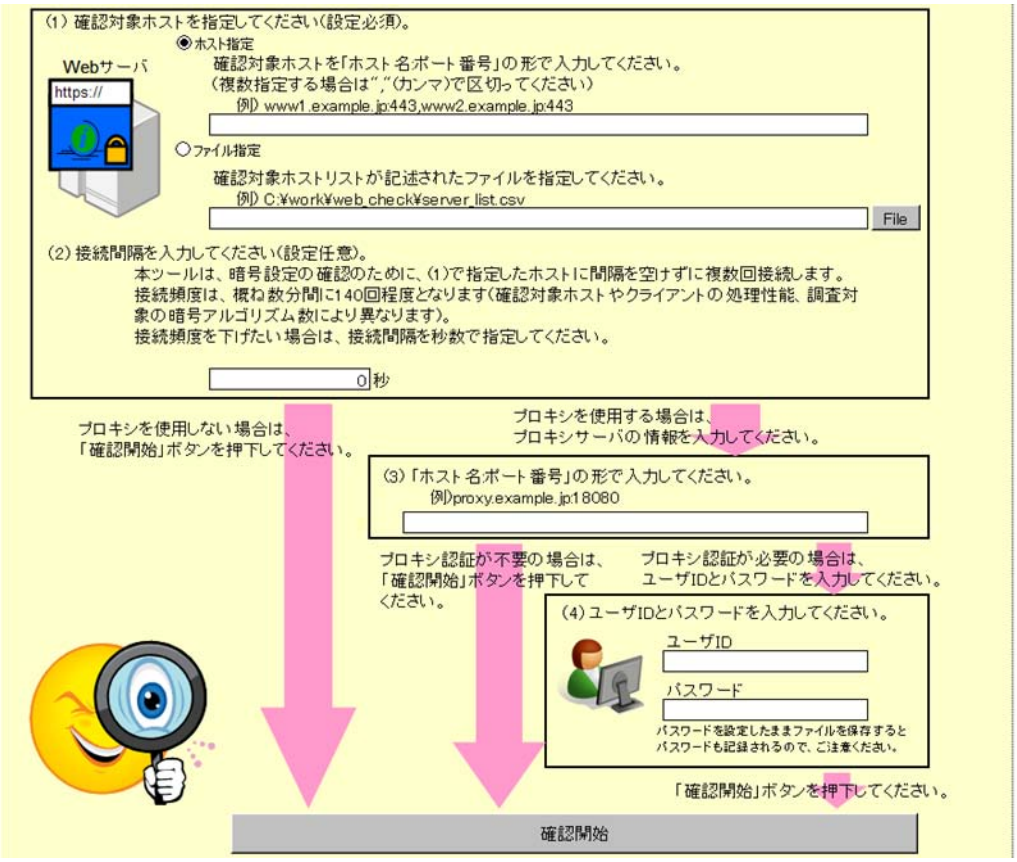
Appendix A



付図 1　金融系，政府・公共系サーバの暗号設定



付図 2　金融系，政府・公共系サーバの暗号設定（サーバ証明書）



付図 3　暗号設定確認ツールの入力 I/F

			site
		CipherSuites	www.example.com:443
CipherSuitesの設定状況	tls1(使用可能)	TLS_RSA_WITH_3DES_EDE_CBC_SHA	ON
		TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA	OFF
		TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA	OFF
		TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	OFF
		TLS_RSA_WITH_CAMELLIA_128_CBC_SHA	ON
		TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA	OFF
		TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA	ON
	tls1(採用自粛)	TLS_RSA_WITH_RC4_128_SHA	ON
		TLS_DHE_DSS_WITH_RC4_128_SHA	OFF
		TLS_ECDH_ECDSA_WITH_RC4_128_SHA	OFF
		TLS_ECDHE_ECDSA_WITH_RC4_128_SHA	OFF
		TLS_ECDH_RSA_WITH_RC4_128_SHA	OFF
		TLS_ECDHE_RSA_WITH_RC4_128_SHA	OFF
	tls1(使用停止)	TLS_RSA_WITH_NULL_MD5	OFF
		TLS_RSA_WITH_NULL_SHA	OFF
		TLS_RSA_EXPORT_WITH_RC4_40_MD5	ON
		TLS_RSA_WITH_RC4_128_MD5	ON
		TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5	ON
		TLS_RSA_WITH_IDEA_CBC_SHA	ON
		TLS_DH_anon_EXPORT_WITH_RC4_40_MD5	OFF

付図 4　CipherSuites の個別調査結果（一部抜粋）

		site
		www.example.com:443
その他のCipherSuites	本項目が"ON"になっている場合、一般的に使われていないCipherSuiteのうち1つ以上のCipherSuiteが使用できるようになっています	OFF

付図 5　CipherSuites の一括調査結果（一部抜粋）