社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2005－DPS－122（12）
2005－CSEC－28（12）
2005／3／22

通信相手のセキュリティ対策を保証する セキュリティ保証基盤モデルの実装および評価

磯原 隆将 †　石田 千枝 †　北田 夕子 †　竹森 敬祐 ‡　笹瀬 巌 †

† 慶應義塾大学理工学部情報工学科
223-8522 神奈川県横浜市港北区日吉 3-14-1
{isohara,ishida,yuko,sasase}@sasase.ics.keio.ac.jp

‡(株)KDDI 研究所
356-8502 埼玉県上福岡市大原 2-1-15
takemori@kddilabs.jp

現在，通信相手の身元を認証する技術として PKI(Public Key Infrastructure) 等があるが，通信自体の安全性を保証する技術ではない。信頼すべき通信相手のサーバから，意図せず受け取ったウイルスや攻撃による影響は大きくなる．そこで，サーバのセキュリティ対策の状況を第三者機関であるセキュリティ保証局が確認して証明書を発行し，クライアント側でその署名を検証するセキュリティ保証基盤モデルを提案し，その基本動作について検討されている．本研究では，セキュリティ保証基盤を実現するにあたり，安全性，柔軟性，即時性に関する課題を整理して，通信プロトコルと各種処理モジュールについて提案・設計する．即時性については，本技術の普及のポイントとなるサーバ・クライアント間の証明書の提示処理について実装・評価を行い，十分な高速性を達成できていることを確認する．

Implementation and evaluation of Security Certificate Platform for Verification of Safety Communication

Takamasa Isohara †　Chie Ishida †　Yuko Kitada †　Keisuke Takemori ‡
Iwao Sasase †

†Dept. of Info. & Computer Science, Keio University
3-14-1 Hiyoshi, Kohoku, Yokohama Kanagawa,
223-8522, Japan
{isohara,ishida,yuko,sasase}@sasase.ics.keio.ac.jp

‡KDDI R&D Laboratories Inc.
2-1-15 Ohara, Kamifukuoka,
Saitama, 356-8502 Japan
takemori@kddilabs.jp

Abstract Recently, the authentication technology to attest the identity of the other party of the communication such as PKI is used. However, that technology cannot confirm even the introduction situation of other party's security countermeasures. Therefore, the user on the network has faced the threat of a viral infection and an intrusion. Then, the new concept that was called a security certificate platform that guaranteed other party's of the communication safety measures situation was proposed. In this paper, the requirement for safety, flexibility, and the feasibility to achieve the security certificate platform is clarified. We have also implementation the proposed scheme, and evaluated its feasibility in view of swift response.

1　はじめに

ユビキタス社会の実現に向けて，インターネットへのアクセス回線の普及，PDA や携帯電話などの通信端末の小型化，P2P やアドホックネットワークなどの接続形態の多様化が進む中，ネットワークを通じたウイルス感染や不正侵入による被害が増大している．これらの被害を防止するため，ウイルス感染を防止するウイルス検知システム (VDS: Virus Detection System)，ネットワーク攻撃を監視する侵入検知システム (IDS: Intrusion Detection System)，および端末の安全性を監査する検疫システムに関する研究が行われている [1, 2].

VDS や IDS は保護対象の端末自身を脅威から守ることを目的とした技術であるが，全ての端末に設定する場合の導入費用の問題やや，そもそも小型化された端末に設定できない問題がある．P2P サービスやアドホックネットワークにおいて，送受信や中継されるパケットに攻撃コードが含まれていることも多く，自衛手段を持たない端末にとって通信相手の安全性を把握することは重要である．通信相手の本人性を確認する技術として，公開鍵基盤 (PKI: Public Key Infrastructure) があるが [3]，通信相手が実施しているセキュリティ対策状況までは保証し

ていない．信頼すべき通信相手のサーバから，意図せず受け取ったウイルスや攻撃による影響は大きい．また，検疫システムは利用者自身の端末がセキュリティ対策についての監査を受ける技術であり，監査結果を通信相手に通知する手段が欠落している．

そこで [4] では，サーバのセキュリティ対策の状況を第三者機関であるセキュリティ保証局が確認して証明書を発行し，クライアント側でその署名を検証するセキュリティ保証基盤に関する基本モデルが提案されている．この [4] では，端末において VDS/IDS が正しく設定され，それらが確実に稼働していることを，セキュリティ証明書を用いて保証し，対策状態を相手に通知する枠組の必要性について検討されている．

本研究では，[4] で提案した基本モデルを実現するために，安全性，即時性および柔軟性の要件を明確にし，通信プロトコルと各種処理モジュールについて提案・設計する．即時性については，本技術の普及のポイントとなるサーバ・クライアント間の証明書の提示処理について実装・評価を行い，セキュリティ保証基盤を導入してもサーバがクライアントの要求に迅速に返信できることを確認する．

以降，2 章でセキュリティ保証基盤の基本モデルについて説明し，3 章で実現のために検討すべき課題を整理する．4 章で各システムモジュールの設計を行い，5 章で通信プロトコルについて設計する．6 章で，実装したシステムにおける通信処理に関する速度評価を行い，最後に 7 章でまとめる．

2　セキュリティ保証基盤の概要

本節では，[4] で提案されているセキュリティ保証基盤について，動作に注目して概要を述べる．

サーバ・クライアントモデルによるセキュリティ保証基盤の基本モデルを図 1 に示す．基本モデルは第三者機関として位置付けられるセキュリティ保証局，各種サービスを提供するサーバ，およびサービスを受けるクライアントから構成される．セキュリティ保証局は，サーバの OS やアプリケーションに最新のパッチが適用されていることや，VDS/IDS が適切に稼働していることを検証し，セキュリティ証明書を発行する．サーバはクライアントからの要求に従ってセキュリティ証明書を提示する．クライアントは，サーバの公開鍵証明書を用いてサーバの相手認証を行い，セキュリティ保証局の公開鍵を用いて証明書のディジタル署名の検証を行ったうえで，通信相手のセキュリティ対策の状態を確認する．

3　システムの設計における課題

セキュリティ保証基盤モデルを実現するために，安全性，柔軟性，即時性に関する課題について，明確にする．

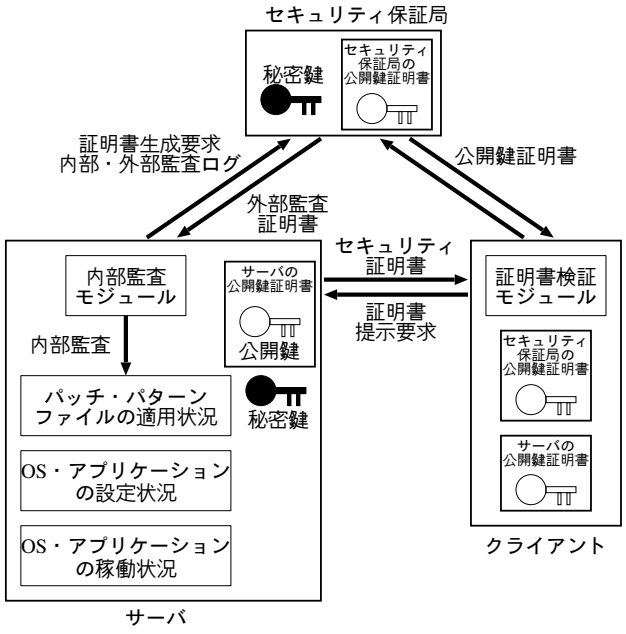

Figure 1: セキュリティ保証基盤の基本モデル

3.1　安全性に関する課題

セキュリティ保証基盤では，セキュリティ証明書の発行，提示，検証処理を行うため，セキュリティ保証局とサーバ間，サーバとクライアント間，クライアントとセキュリティ保証局間において相互認証を伴う通信が必要である．さらに，サーバからセキュリティ保証局に送付される監査結果などを盗聴されることで，脆弱性の漏洩を防止しなければならない．

悪意の侵入者やサーバの管理者によりサーバ OS の奪取や内部監査モジュールを偽造することで，偽の内部監査ログを生成してセキュリティ証明書の取得を試みる脅威が想定される．したがって，サーバ OS や内部監査モジュール自体の信頼性を確保する必要がある．

また，悪意のユーザがセキュリティ証明書に記載されている情報から，サーバの脆弱性を知り得てしまうことを防止しなければならない．

課題 1 相互認証を伴う通信プロトコルであること

課題 2 セキュリティ保証局とサーバ間の通信は暗号化されていること

課題 3 サーバ内部に信頼の基点を置くこと

課題 4 セキュリティ証明書に具体的なサーバの対策情報を含まないこと

3.2　柔軟性に関する課題

OS やアプリケーション，VDS や IDS には様々な種類が存在する．これらのパッチ情報やパターンファイル，設定情報として，数値や文字列といった多種

多様なデータ構造が混在する．さらに，数値については時刻としての情報やバージョン番号としての情報といったように，同じデータ形式における異なる意味表現を取り扱うことを考慮しなければならない．

課題 5 様々な形式の情報を交換できること

3.3 即時性に関する課題

セキュリティ保証基盤におけるサーバとクライアント間でのセキュリティ証明書の提示および検証にかかる時間は，それ自体が本来の通信目的ではないため，通信におけるオーバーヘッドとみなされる．クライアントにとって，証明書の取得と検証処理に要する時間が短いことが，本技術の普及のポイントとなる．

課題 6 サーバから迅速に証明書を提示できること

課題 7 クライアント側で迅速に署名検証できること

4 処理モジュールの設計

本節では，セキュリティ保証基盤を構成する各種処理モジュールの設計について述べる．

4.1 セキュリティ保証局の設計

図 2 にセキュリティ保証局のシステム構成を示す．

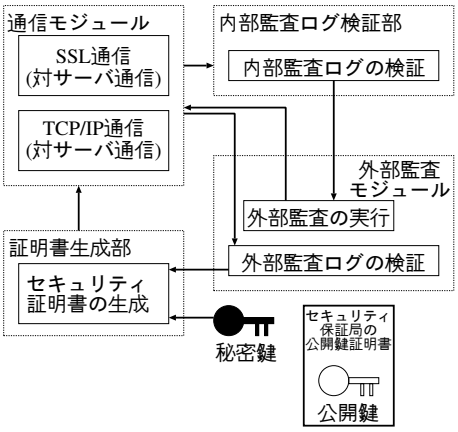

Figure 2: セキュリティ保証局のシステム構成

課題 1 と課題 2 を考慮して，セキュリティ保証局とサーバ間の通信における相互認証と暗号化を実現するための SSL 通信モジュールを設置する．これにより，セキュリティ保証局とサーバがそれぞれ所有する公開鍵証明書を用いた相互認証と標準化された SSL 通信を利用できるようになる．

内部監査ログ解析部は，サーバの内部監査モジュールが実行した内部監査のログから，サーバで稼働している OS やアプリケーションの名称，バージョン，設定ファイルの情報などを取得する．

外部監査コマンド生成部は，内部監査ログ解析部により検出されたアプリケーションの稼働を検証するためのコマンドを生成し，サーバに対して外部監査を行う．

外部監査ログ検証部は，受信した外部監査ログを検証し，外部監査を行ったサーバにおいて監査対象のアプリケーションの稼働状態を検証する．

証明書生成部はセキュリティ証明書を生成し，サーバに送付する．

4.2 サーバの設計

図 3 にサーバのシステム構成を示す．

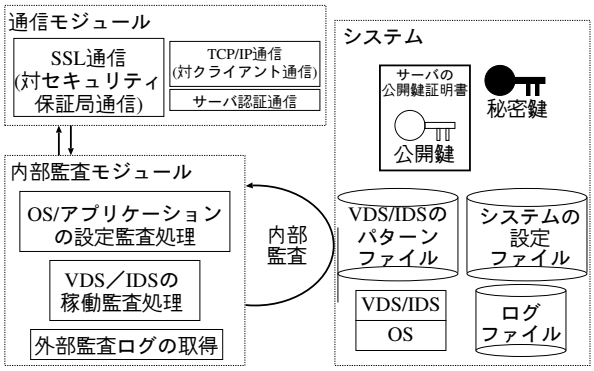

Figure 3: サーバのシステム構成

サーバ側でも，課題 1 と課題 2 を考慮してセキュリティ保証局と同等の SSL 通信モジュールを設ける．

OS/通信アプリケーションの設定監査処理は，サーバが導入している OS の名称，バージョン情報およびファイアウォールの稼働状態を取得する．

VDS/IDS の稼働監査処理部は，サーバで稼働する VDS/IDS の名称，バージョン情報，設定ファイルの情報，および稼働状態を取得する．

外部監査ログの取得部は，外部監査により監査対象のアプリケーションが生成したログをシステムから取得する．

上記に示す各種処理モジュールやサーバ OS への攻撃対策となる課題 3 を考慮して，サーバには Trusted Platform Module (TPM) に代表される耐タンパモジュールを設置して，信頼の基点を設ける．図 4 に，TPM による信頼性の確保の仕組みを示す．ここで TPM とは，信頼できるコンピュータプラットフォームを構築するための業界標準仕様の開発，普及を目的とした団体である Trusted Computing Group (TCG) が提唱する耐タンパモジュールであり [5, 6]，BIOS，OS，ソフトウエアの改ざんを発見するためのハッシュ値を管理する機能，公開鍵と秘密鍵を生成して秘密鍵を管理する機能などを持つ．TPM の代わりに，スマートカードなどの技術も適用できる．TPM による信頼の基点によって守るべき情報と守られる情報を表 1 にまとめておく．

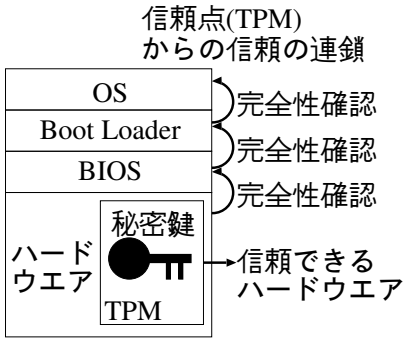

Figure 4: TPM による信頼性の確保

Table 1: TPM による信頼の基点によって守るべき情報と守られる情報

守るべき情報	守られる情報
・BIOS，Boot Loader OS のハッシュ値 ・公開鍵と秘密鍵を生成する機能 ・秘密鍵 ・内部監査モジュールのハッシュ値 ・通信インターフェイスのハッシュ値	・BIOS，Boot Loader，OS ・OS のパッチファイル ・内部監査モジュール ・通信インターフェイス ・セキュリティ証明書 ・各種アプリケーション

4.3 クライアントの設計

図 5 にクライアントのシステム構成を示す.

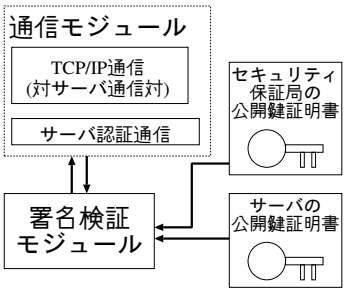

Figure 5: クライアントのシステム構成

通信モジュールはサーバ認証を行う．サーバ認証にあたって，クライアントは乱数を生成し，サーバに送付する．サーバは保有する秘密鍵で乱数から生成されるディジタル署名を計算し，クライアントに送付する．クライアントはサーバの公開鍵証明書を用いてディジタル署名の検証を行う.

サーバ認証に成功した場合，セキュリティ証明書の提示要求を行う．サーバは保有するセキュリティ証明書をクライアントに送付する．署名検証モジュールはセキュリティ保証局の公開鍵証明書を用いてセキュリティ証明書のディジタル署名を検証し，証明書の発行元を確認する.

4.4 セキュリティ証明書の設計

図 6 にセキュリティ証明書の構成を示す.

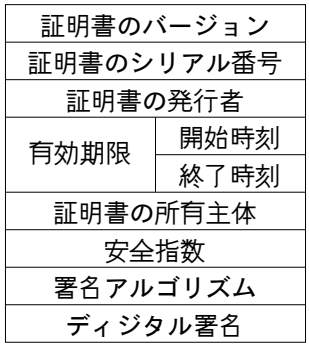

Figure 6: セキュリティ証明書の構成と失効処理

セキュリティ証明書の構成は，PKI で用いられる X.509 を参考に，セキュリティ保証基盤で必要となる情報を持っている [7]．ここで，課題 4 を考慮してセキュリティ証明書にはサーバのセキュリティ対策に関する詳細な情報を隠蔽するために，対策の度合を示す指数として安全指数を導入する.

セキュリティ保証基盤モデルでは，既存の PKI モデルに比べて頻繁に証明書の更新・失効処理が繰り返されるため，証明書の有効期限は短くなる．失効証明書リスト (CRL: Certificate Revocation List) を用いた管理では，セキュリティ保証局における負荷が高くなり，迅速な署名検証を行えない．したがって，課題 7 を考慮して，セキュリティ証明書の失効処理は行わないこととし，証明書に記載されている有効期限が失効状態を表す指標とみなす.

5 通信プロトコルの設計

本節では，セキュリティ保証局とサーバ，サーバとクライアント間，クライアントとセキュリティ保証局間の通信プロトコルの設計について述べる.

5.1 データグラムの構造

図 7 に，セキュリティ保証局がサーバにセキュリティ証明書を送付するときのデータグラムを示す．データグラムは，ヘッダ部とデータ部から構成される.

ヘッダ部には，あらかじめ規定されたメッセージが格納される．セキュリティ保証基盤の各モジュールは，受信したデータグラムのヘッダのメッセージを解読し，監査や証明書の生成などの処理を行う.

データ部では，各モジュールにおける処理に応じた様々なデータが格納される．ここで課題 5 を考慮

して，データの記述方式は，あらかじめ規定したタグを用いてデータを囲む，マークアップ言語の形式を採用する．マークアップ言語を用いることで，様々なデータ形式を，テキスト形式で表現することが可能となり，また，それぞれのデータの意味表現をタグで示すことができる．

ヘッダ	
メッセージ	

```
<certificacte>
  <verion>1.0</version>
  <serial>1008</serial>
  <sca>TestSCA</sca>
  <validity>
    <notBefore>
      Feb 19 12:18:40 2005 GMT
    </notBefore>
    <notAfter>
      Feb 20 12:18:40 2005 GMT
    </notAfter>
  </validity>
  <server>TestServer</server>
  <secure_level>5</secure_level>
  <signature algorithm>
    md5WithRSAEncryption
  </signature algorithm>
  <signature>
    YWthbWFzYSBJ
  <signature>
</certificate>
```

Figure 7: データグラムの構造

5.2 サーバとセキュリティ保証局間の通信手順

図 8 に，SSL 通信を適用した場合のサーバとセキュリティ保証局間の通信手順を示す．システムの実装にあたっては SSL の代わりに IPSec などの相互認証・暗号化プロトコルを選択することも可能である [8]．セキュリティ証明書発行の通信は，相互認証や暗号化と独立した設計とする．

サーバとセキュリティ保証局間の通信においては，TCP と UDP の選択肢があり，TCP では通信の信頼性が保証され，UDP では迅速な処理が可能となる．ここではモジュール間で確実にデータグラムが送受信されることを重視して，TCP を採用する．TCP の 3 ウェイハンドシェイクに続き，課題 1 を考慮して SSL による相互認証，通信路の暗号化が行われ，セキュリティ証明書の発行処理が行われる．

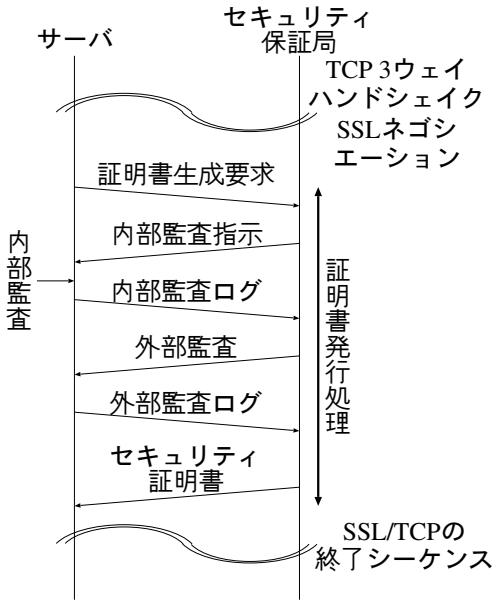

Figure 8: SSL 通信を適用した場合のサーバとセキュリティ保証局間の通信手順

5.3 サーバとクライアント間の通信手順

サーバとクライアント間の通信では，目的とする通信に先立ってセキュリティ証明書の提示処理が行われる．図 9 に，サーバとクライアント間の通信手順を示す．

課題 1 を考慮して，サーバとクライアント間の通信には，サーバ認証を行うためのクライアントから送られてきた乱数に対してサーバ側の秘密鍵による署名処理を行う．サーバ認証を終えた後に，サーバから証明書を送付する．クライアントで証明書の検証に成功した場合，サーバとクライアント間の所望の通信が開始される．

ここで，課題 6 を考慮して，内部監査はクライアントからの証明書の提示要求に対して非同期に実行することとし，クライアントからの要求を受け次第，上記の署名処理を実施してセキュリティ証明書を提示する．

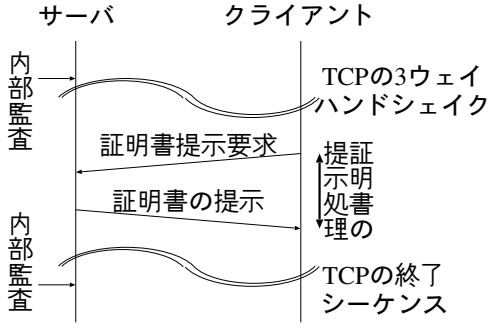

Figure 9: サーバとクライアント間の通信手順

6　通信に関する速度評価

ここでは課題 6 を確認するために，サーバとクライアント間における証明書の提示処理の通信速度を評価する．

6.1　評価環境

評価に用いたサーバの諸元は，CPU が Pentium4 3.4GHz，メモリ容量が 2GBytes，OS は FreeBSD-5.3，NIC は 100Base-TX である．また，クライアントの諸元は，CPU が PentiumM 1.5GHz，メモリ容量が 768MBytes，OS は FreeBSD-5.3，NIC は 100Base-TX である．

6.2　サーバ・クライアント間の通信速度の評価

ここでは，証明書提示要求に内部監査を非同期に行う方式の即時性を確認するために，証明書提示要求を受けるたびに内部監査を実施する同期監査方式と比較，評価を行う．クライアントからの要求を増加させながら，証明書提示要求を発してから証明書を取得するまでの時間について，ネットワーク上でパケットをキャプチャすることで計測した．結果を図 10 に示す．

図 10 より，いずれの監査方式においても同時接続数が増加するにともない，処理速度も増加していることがわかる．

両方式を比較した場合，非同期監査方式は同期監査方式よりも迅速に処理を終えており，その差異は同時接続数が増加するほど拡大している．

以上より，証明書の提示処理を迅速に行うには非同期監査方式が適しており，課題 6 に示す迅速性を達成できていることがわかる．

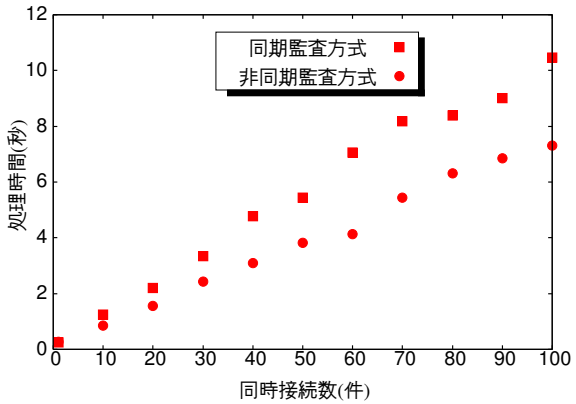

Figure 10: サーバとクライアント間の通信速度評価

7　おわりに

本研究では，通信相手のセキュリティ対策を保証するセキュリティ保証基盤を実現するための各種処理モジュール，および通信プロトコルに関して，安全性，柔軟性，迅速性に関する要件を整理して，それぞれの設計，実装を行った．

本基盤の通信の安全性を確保するために，モジュール間の相互認証および暗号化を行うシステムを設計した (課題 1，課題 2 の達成)．また，耐タンパハードウエアを用いることでサーバの内部監査モジュールの安全性を確保する設計を行った (課題 3 の達成)．セキュリティ証明書に関しては，悪意のユーザがセキュリティ証明書から脆弱性を得る攻撃を想定し，証明書に具体的なセキュリティ対策を記載しない証明書の設計を行った (課題 4 の達成)．また，通信データグラムはヘッダ部とデータ部から構成されるデータグラムを設計し，データ部に格納するデータはマークアップ言語の形式を用いることで，複数のデータ構造を統合的に取り扱うことができ，同時にそれぞれのデータの意味情報を容易に定義できることを示した (課題 5 の達成)．サーバとクライアント間における証明書提示処理に関する通信速度を評価した結果，内部監査を非同期で行う場合であれば，迅速に証明書を提示できることを示した (課題 6, 7 の達成)．

References

[1] 山口 英, 鈴木裕信, 「情報セキュリティ」, 共立出版, 2000 年 9 月.

[2] 三輪信介, “持ち込み PC 検疫機構の提案”, 情報処理学会, コンピュータセキュリティシンポジウム 2003(CSS2003), pp.265-270, 2003 年 10 月.

[3] 塚田考則, 「企業システムのための PKI」, 日経 BP 社, 2001 年 12 月.

[4] 竹森敬祐, 三宅 優, 田中俊昭, 磯原隆将, 笹瀬 巌, “通信相手の安全性を確認するためのセキュリティ保証基盤の提案”, 情報処理学会, コンピュータセキュリティシンポジウム 2004(CSS2004), 2004 年 10 月.

[5] Trusted Computing Group: Home, https://www.trustedcomputinggroup.org/home.

[6] TCG, “TPM Main Part1 Design Principles”, TCG PUBLISHED, 2003.

[7] Housley R., Ford W., Polk W. and Solo D., “Internet X.509 Public Key Infrastructure Certificate and CRL Profile”, IETF, RFC2459, January 1999.

[8] Eric Rescorla, 「マスタリング TCP/IP SSL/TLS 編」, オーム社, 2003 年 11 月.