平成２３年度研究開発成果概要書
通信プロトコルとその実装の安全性評価に関する研究開発」
副題：「形式手法によるプロトコル実装の検証技術と
形式仕様に基づく網羅的ブラックボックス検査技術の開発」

（１）研究開発の目的

インターネット標準は現実の実装においては必ずしも仕様がその通りに反映されていない場合が多く、暗号通信プロトコルなどにおいて、その実装の誤りが安全性の脆弱性に繋がるケースが複数見受けられている。このような誤りはもちろん実装の作成者の注意不足に起因する場合も多いが、一方で誤りの一端が標準そのものにある場合もある。

本プロジェクトでは、形式的手法による通信プロトコルの規格の記述を元にその実装の様々な評価・検証手法を統一的に扱うことのできる手法および、ブラックボックスになっている実装においてもテストケースの網羅性などを保証し、仮想マシンモニタを利用した実行のトレース及びロールバックにより脆弱な実装を解析できる環境の開発を行う。

（２）研究開発期間

平成２２年度から平成２４年度（３年間）

（３）委託先企業

独立行政法人産業技術総合研究所＜幹事＞、株式会社レピダム

（４）研究開発予算（百万円）

平成２２年度	４１（契約金額）
平成２３年度	３９（　〃　）
平成２４年度	３６（　〃　）

（５）研究開発課題と担当

課題ア：形式的手法によるプロトコル実装の検証技術の開発

1.　（産総研）

平成２３年度は平成２２年度に作成したTLSの実装の形式検証基盤を拡張して、Ｃ言語実装をデータとして扱う定理群のライブラリ化を行う。この定理を用いて、抽出した関数の形式仕様を記述し、その形式検証も行う。また、TLSのプロトコルの形式仕様検証については平成２２年度にパケットの形式仕様を作成したが、これを拡張して状態遷移系の形式仕様として使えるものにする。この形式仕様は課題イ）の網羅的テストジェネレータで使えるようにする。

2.　（レピダム）

平成23年度は、産総研が定理証明器Coq上に定義・構築を進めているC言語の

サブセット言語と、それで記述されたプログラムに対する検証技術に対し、既存の TLS の実装の一部を実際に検証可能とする。さらに形式検証基盤を利用して､既存の TLS の実装の一部のコードに対する静的検証を行う環境を整備する。同時に、産総研によって定理証明器 Coq 上に構築が進められている形式検証基盤に、産総研と協力してこの拡張を反映させる。この作業は網羅的テストで活用されることを念頭に作業を進める。

課題イ：網羅的テストケース生成による実装のブラックボックス解析技術の開発

イー１）網羅的汎用テストジェネレータおよび実行環境

1.　（産総研）

平成２２年度に課題ア）と連携して行ったパケットの形式仕様検証を応用して、網羅的汎用テストジェネレータに活用できるようにする。平成２３年度は自動的な疑似攻撃の挿入などのための付加情報をプログラム中に記述するできる基盤を作成する。具体的には、平成２２年度に調査したプログラム変換の理論を元に、変換が容易な言語の作成とプログラム変換技術の実装を行う。この実装によって自動的な疑似攻撃の挿入を行えるようにする。さらに、平成２４年度の最終統合に向けて、課題（イー２）で開発する実際のプロトコル処理のトレース・実行状態ロールバックと連携できるようにし、プロトコルの分岐可能性を網羅的に試験できるかを検証する。

2.　（レピダム）

平成 23 年度は、昨年度から産総研と協議しながら実装を進めているプロトコル中間言語の処理系を、TLS など複雑なプロトコルの記述・実行ができるように拡張する。そのプロトコル言語で TLS 等のプロトコルを主要な部分から順に記述を行い、既存の実装との間でやりとりができるようにする。更に、ファジングなど疑似攻撃を含む通信を生成する機構を試験的に組み込み、既存の実装の不具合の発見を目的としたテスト環境を整える。また、課題(ア)における形式仕様によるテスト環境および、課題(イー2)における実行状態制御の仕組みとの連携方式について、検討・その一部の試作を行う。

イー２）仮想マシンベースとしたコントロールフロー解析（産総研）

平成２３年度は、平成 22 年度からの継続課題である仮想マシンベースの ICE を進めると共に、仮想マシン上のメモリ操作とロールバックの開発に着手し、ブラックボックスのＯＳおよびプロトコル処理の解析に利用できる技術を開発する。

仮想マシンベースの ICE は平成２２年度に開発した各プロセスの仮想メモリ・実メモリマッピングトレース技術を発展させ、ブラックボックスのＯＳ内のプロトコル処理の実行トレースおよび解析が行える技術の研究開発を行う。想マシン上のメモリ操作とロールバック機能は平成２４年の最終統合に向けて、課題（イー１）と連携し、プロトコル実行の分岐点に巻き戻ってテストできる環境に使える機能を試作する。

（６）これまで得られた研究開発成果

		（累計）件	（当該年度）件
特許出願	国内出願	0	0
	外国出願	0	0
外部発表	研究論文	4	2
	その他研究発表	0	0
	プレスリリース	0	0
	展示会	0	0
	標準化提案	0	0

具体的な成果

(1) 形式検証基盤の構築と形式検証実験

平成２２年度に作成したC言語実装の形式検証基盤を拡張した。本課題では形式仕様は逐次的なプログラミング言語に相応しい分離論理で書く。その形式仕様を実際に検証できるように、定理群のライブラリを標準的な補題で補充し、下記の実験を行った。

平成２２年度にケーススタディとして選んだ既存の TLS 実装のパーザの一部の形式仕様を改良し、検証作業を開始した。実用的に扱える形式仕様を得るために、依存型による独特の記述方法を提案し、パケットフォーマットの形式仕様を改良した。その記述方法により、TLS の正式の仕様(RFC 5246)の曖昧なところを発見し、堅固な形式仕様を得た。その形式仕様を用いて、TLS 実装のパーザの一部の検証作業を開始した。具体的に、パケットの一種類のヘッダーのパーザの形式検証で我々のC言語実装の形式検証基盤の実用性を確認できた。最後に、我々のC言語実装の形式検証基盤の最終的な改良として、ポータビリティの概念の導入方法を提案し、形式検証基盤の拡張実験を行った。その改良により、C言語実装とアーキテクチャーの関係をモデル化され、無条件で低レベルの通信プロトコルの実装の形式検証を行える。

(2) 形式仕様を記述する言語の設計と処理器の実装

昨年に Coq 上で行った文法のパーザ・プリンタの双方向生成の形式化の知見をもとに、プロトコルの通信文の仕様（文法）と両端末のプロトコル状態遷移の仕様を同時に記述する使用記述言語の第１版を設計し、プロトタイプとして実際に参照実装生成器の施策を行い TLS の初期化処理の最初の部分に関して通信文の自動生成と解釈を行うことができた。形式仕様記述言語は通常のモデル検査などで用いる抽象的な仕様と異なり、具体的な通信文の内容に含まれる情報を自動的に解釈し内部状態に自動的に反映する機構と、逆に内部状態から自動的に通信文を生成する機構を備えている。また文法記述は IETF の標準 ABNF 記法をもとにしつつ、通常はあいまいな形でしか記述できない前後データの相互依存関係などを正確に記述できるように拡張されている。

(3) プロトコル検査用の VM 制御プロトコルの設計

プロトコル網羅検査のために必要となる、外部から仮想マシンと通信状態を

制御するプロトコルを設計し、(2) のプロトコル実行側と (4) の仮想マシン側を試作した。このプロトコルは特に、仮想機械のスナップショット取得などと、プロトコルの通信状態を正確に同期させ、ロールバック後の通信再開時に仮想ネットワークによるデータの欠落や重複を引き起こさないために必要となっている。仮想機械側の実装に必要な詳細を備えつつ、プロトコル自体は将来の関連した他の実装手法(たとえばプロセス制御や言語処理機構による巻き戻し) にも応用できるよう汎用的に設計されている。

(4) ネットワークの接続を保ったまま仮想マシンがロールバックする技術

通常の仮想マシンのスナップショット機能ではネットワークの接続が保てないが、二重の仮想化とプロキシを使うことで実現した。外側の VM 内にプロキシと内部 VM を置き、内部 VM の通信はプロキシを通ってプロトコルジェネレータと通信する。この際に外側の VM のスナップショット機能を使うと内側 VM とプロキシの通信状態は維持されたまま状態が保存される。ロールバックする際には、プロキシとプロトコルジェネレータの通信は切断され、状態を戻す必要があるが、プロキシがその状態を認識して再接続する。このような実装にすることで、内側の VM に変更することなく、ネットワークの接続を保ったまま仮想マシンがロールバックすることができる。また、内側の VM 物理 CPU と異なる CPU エミュレータを採用した場合、異なる CPU を想定する OS のブラックボックステストも可能にする。

（７）研究開発イメージ図

下記の図で研究課題ア）とイ）の関係を示す。

課題ア）において形式的手法によるプロトコル実装の検証技術の開発を行う。検証結果を元に課題イ）において網羅的テストケース生成技術の作成、および仮想マシンを活用してブラックボックス実装でも解析できる技術の開発を行う。