様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　３ 年　 ６ 月 １０ 日現在

機関番号：１２６０８
研究種目：基盤研究(C)（一般）
研究期間：2017～2020
課題番号：１７Ｋ００１８１

研究課題名（和文）行動系列類似度によるマルウェア検知手法の開発

研究課題名（英文）Malware detection scheme using behavioral sequence similarity

研究代表者
一色　剛（Isshiki, Tsuyoshi）

東京工業大学・工学院・教授

研究者番号：１０２８１７１８

交付決定額（研究期間全体）：（直接経費）　　3,500,000円

研究成果の概要（和文）：人手によるマルウェア動作解析作業を自動化する新たな技術基盤の構築を目的として、「行動系列類似度によるマルウェア検知手法の開発」を行った。具体的には、エミュレータ駆動型プログラム構造解析手法、マルウェア行動系列による全行動パターン列挙方法、及びプログラム構造解析処理を高速実行する専用プロセッサの開発を行い、これらの研究成果によって、マルウェア動作解析作業を高精度で自動化するための技術構築を行なった。

研究成果の学術的意義や社会的意義
近年、マルウェア（悪意ソフトウェア）による情報システムの障害が急増しており、重要な情報インフラ・社会インフラへの深刻な攻撃が現在でも大きな社会問題になっている。増大するマルウェア攻撃は近年益々巧妙化してきており、これらの攻撃に迅速に対処するためには、多様なマルウェアの特徴を自動的に解釈し、未然に検知・防御する体系的な仕組みが必須であり、本研究は、この課題解決に対し、有効な技術要素の蓄積を行ったものである。

研究成果の概要（英文）：Behavioral analysis and detection of malwares has mostly been performed by experienced engineers, which is too time consuming for the drastic increase in malware attack incidents nowadays. For automating the analysis and detection of malwares, we have developed a new set of techniques combining emulator-driven program structure analysis scheme, enumeration of malware API call sequences for profiling malware behaviors, and a custom processor for accelerating malware program analysis.

研究分野：集積回路設計

キーワード：マルウェア対策　プログラム解析　プロセッサ設計

１．研究開始当初の背景

近年、マルウェア（悪意ソフトウェア）による情報システムの被害が深刻化しており、以下に示す既存のマルウェア検知技術を巧妙に回避する新種マルウェアも次々と出現している。

【静的パターン解析法】マルウェアの特徴的データ系列と、検査対象ソフトのデータ系列の類似度計測による検知技術であり、パターンマッチング法と静的ヒューリスティック法が存在する。パッキング（実行コードの圧縮・暗号化）や、実行コード改変などの検知回避手段がある。

【動的振舞い監視法】マルウェアの実行動作（振舞い）のコンピュータ上の監視によるマルウェア判別で、仮想実行環境上の監視法と実機環境上の監視法が存在する。外部指令で動作する「ボット型」や、監視環境を察知し不活性化するマルウェアの振舞い検知は困難である。

新種マルウェアの検知回避手段を含めた動作原理の解明には、最終的に人手による解析作業に依存するのが現状である。近年のパッキング対策を講じた静的解析法を応用したマルウェア分類法や、監視回避機能を一部抑止できる先進的監視法の研究は、人手解析作業の効率化を目的とする側面が強く、今後の IoT 時代到来でマルウェアの攻撃対象が爆発的に増加する中、人手によるマルウェア動作解明作業を自動化する新たな技術基盤の構築が急務であると考える。

２．研究の目的

当研究室はプロセッサの設計手法やプロセッサエミュレーション高速化手法に関する研究を展開しており、高速なプロセッサ実行時間推定手法として開発した「トレース駆動型ワークロードシミュレータ」 [研究業績(2)(9)(16)(17)(29)]では、以下の重要な技術を確立した。

- 分岐履歴ビット系列：プログラム実行履歴を小データ量で表現する実行トレースデータ形式。
- プログラムトレースグラフ (PTG)：プログラム制御グラフを縮退化したグラフ構造であり、分岐履歴ビット系列からプログラム全命令実行履歴を完全復元できる大きな特長を持つ。

そこで本研究では、当研究成果をマルウェア検知に応用することを着想し、プログラム構造解析とエミュレータを融合した高精度プログラム構造解析手法と、ファイル・ネットワーク等の資源に対するマルウェアの一連の不正動作の因果関係を「行動系列」として大局的・網羅的に列挙することで、その動作原理解明の自動化に道筋をたてる新規のマルウェア検知方法論を構築することを研究目的とする。

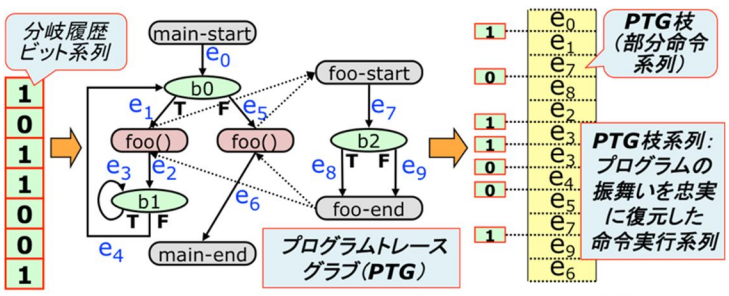

図１:トレース駆動型ワークロードシミュレータの動作原理

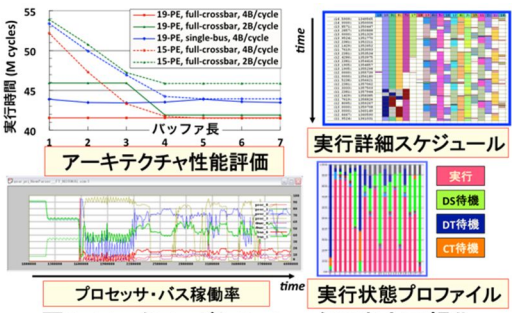

図２:ワークロードシミュレータの出力可視化

３．研究の方法

本研究は、(1)エミュレータ駆動型プログラム構造解析手法、(2)マルウェア行動系列による全行動パターン列挙方法の開発、(3)プログラム構造解析処理を高速実行する専用プロセッサの開発、の３部で構成されている。

(1) エミュレータ駆動型プログラム構造解析手法

(a) プログラム構造解析部：検査対象プログラムのプログラム構造解析は、エントリーポイントから到達可能な命令番地を再帰的に命令解釈する recursive traversal 法を応用することで前記 PTG 生成が実現される。この手法の欠点は、パッキング（暗号化）された隠蔽コードが解析できないことと、関数ポインタによる動的呼出し先関数が解決できないことである。

(b) エミュレータ駆動による隠蔽実行コード構造解析と動的関数呼出の解決：エミュレータ上で書込みメモリ領域の命令実行時に、そのメモリ領域をダンプする機構による暗号化隠蔽コード抽出法（Kang, et.al, “Renovo: A hidden code extractor for packed executables”, ACM WORM’07）を発展させ、書込みメモリ領域への命令実行を検知した際、その命令番地から隠蔽コードの PTG 生成を行うプログラム構造解析機能をエミュレータ上で実現する。さらに、エミュレータ上で動的関数呼出しを検知した時に、その呼出し番地の PTG が未生成の場合に、PTG 生成を行う機構を実現する。

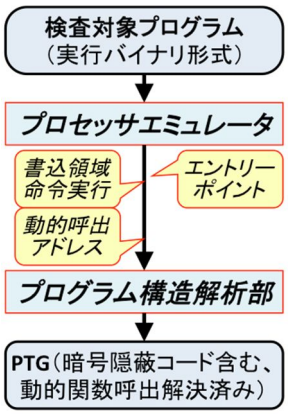

図３：エミュレータ駆動型プログラム構造解析処理

(2) マルウェア行動系列による全行動パターン列挙方法の開発

(a) マルウェア行動系列の自動生成手法：マルウェアの攻撃対象となるファイル・ネットワーク等の資源へのアクセスは、外部ライブラリ（DLL などの API）やシステムコールを介して実行され、その資源アクセスの発生系列を「マルウェア行動系列」と呼ぶ。そこで、検査対象プログラムから(1)の手法で生成される PTG から、グラフ縮退変換によりマルウェア行動系列を有向グラフとして生成する手法を開発する。

(b) マルウェア行動系列の類似度計測手法：マルウェア行動系列を表す有向グラフを NFA（非決定性有限オートマトン）と解釈し、これを正規表現に変換し、正規表現上での類似度計測手法を開発する。

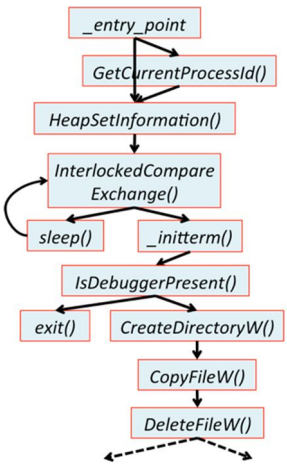

図４：外部API呼出し等の発生系列のグラフ表現（マルウェア行動系列）

(3) プログラム構造解析処理を高速実行する専用プロセッサの開発

(a) 標準マイクロプロセッサ命令セットを実行可能な簡易プロセッサの開発：マルウェアの大部分の攻撃対象ある Intel-X86 命令セットを実行する簡易プロセッサの開発を行う。ここで、当研究室で開発した C 言語ベースのプロセッサ設計環境[研究業績(4)(7)(8)(10)]を使用する。

(b) プログラム構造解析処理用アクセラレータ回路の開発：プログラム構造グラフのデータ構造の高速な生成・改変処理を行うための専用回路（アクセラレータ）を簡易プロセッサに搭載する。

４．研究成果

(1) エミュレータ駆動型プログラム構造解析手法

X86 用命令セットシミュレータ（エミュレータ）とプログラム構造解析の連携機構として、書込みメモリ領域への命令実行検知機構と動的関数呼出し検知機構と、前記検知時に、その実行アドレスの PTG が未生成の場合に、PTG 生成を行う機構を実装した。このことにより、暗号化されたマルウェアの高精度なプログラム構造解析（PTG 生成）を自動的に実行できるようになった。また、マルウェア行動系列の類似度計測手法として、縮退 PTG として表現されたマルウェア行動系列を NFA（非決定性オートマトン）状態遷移図への変換を介して、マルウェア行動系列の正規表現を生成し、この正規表現の類似度計測手法として、文字編集距離計算による手法を実装し、類似したサンプルプログラムにおいて類似度計測値の相関を確認した。また、外部ライブラリの呼び出し命令をメタ記号として扱う類似度計測手法を実装した。また、マルウェア行動系列の類似度計測手法として、ラベル付きグラフ構造の類似度計測法に基づく、より直接的で高精度なマルウェア類似度計測手法の検討を行った。機械学習に基づくプログラム構造グラフの類似度計測手法を実装したが、採用したディープラーニング手法には、様々なパラメータが存在し、パラメータチューニングによるマルウェアの識別精度のさらなる向上が必要であることが確認できた。

以上の研究成果を踏まえ、より大規模なマルウェアデータベースを使い、追加実験を行うとともに、精度向上のための手法の改良を行なった。これまでは、機械学習に基づくプログラム構造グラフの類似度計測手法を実装してきたが、採用したディープラーニング手法には、様々なパラメータが存在し、パラメータチューニングによるマルウェアの識別精度のさらなる向上が必要であることが確認できたため、グラフマッチング手法を応用した新たな手法の研究を行なった。具体的には、マルウェアの行動系列として、API 呼出系列を生成するための API 推移グラフをプログラム構造解析により生成し、API 部分パスのマッピング処理を行うことで、API 部分パスの類似度を計算することで、マルウェアが生成する API 呼出系列の類似度を測定する。その結果、比較的小規模のマルウェアデータベースでは、100%の精度でマルウェアの亜種分類ができることが確認できた。一方で、大規模なマルウェアデータベースについては、API ライブラリの不一致等による原因により、亜種分類精度が低下するケースも確認された。今後の課題として、API ライブラリの不一致を解消するために、同一機能を実現する別 API 関数について、API 機能の類似性を考慮した API シンボル距離を導入することで、異なる API ライブラリを使用したマルウェアについても高精度な亜種分類が可能になると考えられる。

(2) マルウェア行動系列による全行動パターン列挙方法の開発

マルウェア行動系列の自動生成手法として、外部ライブラリ（DLL 等）やシステムコールを介したマルウェア攻撃対象資源（ファイル、ネットワーク）のアクセスの出現系列を、解析されたプログラム構造から全列挙する機構を実装した。具体的には、プログラム構造グラフ（PTG）を縮退することで、DLL・システムコールと、命令分岐構造からなる簡易 PTG を生成し、この簡易 PTG の実行可能経路から DLL・システムコールの出現系列を全列挙した。このことにより、マルウェア類似度計測のためのデータ生成が可能になった。また、マルウェア行動系列の自動生成手法として、外部ライブラリ（DLL 等）を介したマルウェア攻撃対象資

源（ファイル、ネットワーク）のアクセスの出現系列を、グラフ表現可視化する機能を実装し、多数のマルウェアプログラムのグラフ構造を調査した。一部のマルウェアプログラムは、非常に大規模なグラフ構造となるため、グラフ構造縮退処理を実装することで、より直感的な視覚化機能が実現でき、大規模なグラフ構造にも対応できるようになった。また、既知のマルウェアとその亜種について、DLL 出現系列やグラフ構造が非常に類似していることを確認した。また、時限的もしくは外部指令に従って解凍処理を実行する機構を含んだマルウェア暗号化手法による暗号解凍処理の検知回避手段への対応方法として、シミュレータ上で実行されなかったコード領域(解凍処理を含む可能性がある)を実行させるために擬似的に条件分岐命令の挙動を変える仕組みのエミュレータ実装を行った。強制的に分岐方向を変える事により、計算結果の整合性が取れずにシミュレーションが継続できないケースが観測され、条件分岐の変更に伴う計算結果の補正処理、または、計算結果の不整合に伴う例外処理を迂回するための仕組みが必要であることが確認できた。

(3) プログラム構造解析処理を高速実行する専用プロセッサの開発

標準マイクロプロセッサ命令セットを実行可能な簡易プロセッサの開発の準備として、RISC-V 命令セットのプロセッサ開発を行った。RISC-V 命令セットは、次世代組込みプロセッサとして、世界中の大学・研究機関・企業が開発に乗り出しており、RISC-V 用のソフトウェア開発環境も充実しており、プログラム構造解析処理を高速実行するための専用命令を追加するための命令セット拡張性にも優れている。また、関連して、画像認識処理用の専用プロセッサ開発も行い、SW 処理を高速化するための回路設計に関する技法・知見を蓄積した。Intel-X86 命令セットのプログラム構造解析処理の RISC-V 命令セットの命令拡張による高速化処理の設計を行った。X86 命令のデコード処理には、様々なコードパターンに対する命令種別判定回路（通常命令、条件分岐命令、サブルーチン呼出命令、復帰命令）を設計する事により、SW 実行に比べて 10 倍程度の速度向上が達成できた。

５．主な発表論文等

〔雑誌論文〕　計2件（うち査読付論文　2件／うち国際共著　0件／うちオープンアクセス　2件）

１．著者名 Shanlin Xiao, Tsuyoshi Isshiki, Dongju Li, Hiroaki Kunieda	４．巻 Vol. E100.A, No.7
２．論文標題 Design of an Application Specific Instruction Set Processor for Real-Time Object Detection Using AdaBoost Algorithm	５．発行年 2017年
３．雑誌名 IEICE Trans. Fundamentals	６．最初と最後の頁 1384-1395
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

１．著者名 Shanlin Xiao, Tsuyoshi Isshiki, Dongju Li, Hiroaki Kunieda	４．巻 Vol. E100.A, No.12
２．論文標題 HOG-Based Object Detection Processor Design Using ASIP Methodology	５．発行年 2017年
３．雑誌名 IEICE Trans. Fundamentals	６．最初と最後の頁 2972-2984
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 －

〔学会発表〕　計4件（うち招待講演　4件／うち国際学会　3件）

１．発表者名 Tsuyoshi Isshiki
２．発表標題 CNN Training HW Architecture Design Using C2RTL SoC Synthesis/Verification Framework
３．学会等名 19th International Forum on MPSoC for Software-defined Hardware (MPSoC '19)（招待講演）（国際学会）
４．発表年 2019年

１．発表者名 Tsuyoshi Isshiki
２．発表標題 C2RTL SoC Synthesis/Verification Framework For IoT Edge Devices
３．学会等名 18th International Forum on MPSoC for Software-defined Hardware (MPSoC '18)（招待講演）（国際学会）
４．発表年 2018年

１．発表者名 一色剛
２．発表標題 C2RTLフレームワークによるRISC-VベースSoCモデルの論理合成とシステム検証
３．学会等名 Design Solution Forum（招待講演）
４．発表年 2017年

１．発表者名 Tsuyoshi Isshiki
２．発表標題 C++ Object-Oriented RTL Modeling for System-Level Synthesis/Verification on the C2RTL Framework
３．学会等名 17th International Forum on MPSoC for Software-defined Hardware (MPSoC ’17)（招待講演）（国際学会）
４．発表年 2017年

〔図書〕　計0件

〔産業財産権〕

〔その他〕

―

６．研究組織

	氏名 （ローマ字氏名） （研究者番号）	所属研究機関・部局・職 （機関番号）	備考

７．科研費を使用して開催した国際研究集会

〔国際研究集会〕　計0件

８．本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関