

## 成果の概要

# 移動コードを基本とした セキュアなプログラミング言語処理系

研究代表者 米澤 明憲 (東京大学)

## 1 参加研究者

### 研究代表者

米澤 明憲 東京大学 大学院 情報学環 教授

### 研究分担者

田浦 健次郎 東京大学 大学院  
情報理工学系研究科 講師  
住井 英二郎 東京大学 大学院  
情報学環 助手  
遠藤 敏夫 日本学術振興会 特別研究員

### 研究協力者

Benjamin ペンシルバニア大学 工学部  
C. Pierce 計算機情報科学科 助教授  
関口 龍郎 科学技術振興事業団 研究員  
後藤 礼史 東京大学 大学院 理学系研究科  
大岩 寛 東京大学 大学院  
情報理工学系研究科  
前田 俊行 東京大学 大学院 理学系研究科  
浜中 信行 東京大学 大学院 理学系研究科

### WWW

<http://www.yl.is.s.u-tokyo.ac.jp/>

## 2 研究概要

本研究班では、プログラミング言語の安全性にかかわる諸問題について、理論と実用の両面から研究をおこなっている。以下に、本年度の代表的な個別成果を要約する。

### 2.1 暗号 $\lambda$ 計算 [特記 1, 発表 1, 発表 4]

インターネットのように不特定多数が参加するオープンなネットワークにおいて、秘密情報を他者に漏洩しないようにやりとりするためには、通信の内容を暗号化するのが一般的である。そのような暗号系の安全性は、従来から活発に研究されてきた。

ところが、たとえ暗号系自体が数学的に安全であったとしても、それを利用するプログラムに欠陥があれば、やはり秘密情報が漏洩してしまう。実際に報告される漏洩の事例も、暗号系自体の問題ではなくプログラムの欠陥が原因である場合が大半である。

そこで本研究では、暗号を利用するプログラム全体の安全性を数理論理的に議論するために、プログラミング言語の代表的なモデルである  $\lambda$  計算を暗号操作で拡張した暗号  $\lambda$  計算の体系を定義した。さらに、 $\lambda$  計算の一種である多相  $\lambda$  計算におけるパラメタ性の理論を、自明でない方法で暗号  $\lambda$  計算に導入し、暗号を利用したプログラムについて、(暗号系が完全ならば) 実際に秘密情報が漏洩しないことを厳密に証明するための理論を確立した。

本研究は PPL 2001 (日本ソフトウェア科学会) および CSFW'14 (IEEE) にて、それぞれ査読をへて採択・発表され、PPL 2001 では論文賞を受賞した。

### 2.2 Fail-Safe な C 言語処理系 [発表 5]

C 言語は、1970 年代に提案された古典的なプログラミング言語の一つで、現在でも広範に使

用されている。UNIX という代表的なオペレーティングシステムを実装するために開発されたこともあって、もっとも原始的なプログラミング言語であるアセンブリ言語に近い、低水準の操作を詳細に記述することが可能である。

ところが、このような C 言語の特徴は一方でプログラマのミスを誘発しやすく、プログラムの複雑化ともあいまって、ソフトウェアの欠陥が多発する主要な原因の一つとなっている。アカデミックな分野では ML や Scheme といった、より安全な言語も利用されているが、プログラマを教育するコストなどの社会的要因により、産業界で普及するにはいたっていない。

そこで本研究では、たとえプログラムに欠陥があっても「コンピュータを乗っ取られる」「データが壊される」といった致命的影響がない (= Fail-Safe な) C 言語の実装方式を提案・実験した。具体的な方法は以下のとおりである。C 言語の仕様には不十分な部分があり、多くの誤った操作の結果がエラーではなく「未定義」とされている (バッファ溢れや誤った型変換など)。前述のような致命的影響は、そのような未定義の操作の、実際の結果である場合がほとんどである。そこで、結果が未定義とされているような危険な操作を実行時検査によりすべて検出し、致命的影響をおよぼす以前にプログラムを中断する、という方法である。

我々の実験によれば、上述のような実行時検査による速度低下は、単純な実現方式でも約 1 倍～十数倍以下であった。また、我々とは独立な研究者らの実験によれば、事前にプログラムを解析して無駄な実行時検査を削減することにより、平均で数十%以下、最悪の場合でも 2~3 倍程度に速度低下がおさえられたと報告されている。

### 2.3 Java におけるオブジェクト使用解析 [発表 8]

オブジェクト指向は、プログラムの内部に存在するデータ構造や、プログラムの外部に存在する計算資源 (ファイルやネットワーク通信手段

など) を、オブジェクトと呼ばれる単位で扱う、ソフトウェアの設計開発において現在では主流のパラダイムである。このパラダイムにもとづくプログラミングでは、各々のオブジェクトに対して不可能な操作を行わない (たとえば、開いていないファイルに対して読み取りや書き込みを行わない) という性質が必要とされる。

しかし、通常のオブジェクト指向プログラミング言語では、オブジェクトに対して可能な操作の集合を定義することはできても、それらの操作の順序は保証できなかった。そのために「ファイルを読み書きするには、まず開かなければならない」といった性質を十分に検査することができず、ソフトウェアの欠陥につながっていた。

このような問題を解消するために、本研究では代表的なオブジェクト指向プログラミング言語である Java を対象に、オブジェクトに対して可能な操作の集合のみならず順序をも解析・検査する体系を定義した。Java のような命令型言語における重要な問題 (エイリアシング、破壊的代入など) に対処しており、類似の研究と比較しても技術的に新規性がある。

本研究は A01-03 班 小林直樹 東京工業大学 助教授との共同研究である。

### 2.4 Linux/TAL [発表 7]

オープンソースの OS である Linux において、型により安全性が保証されたアセンブリのユーザプログラムを、カーネルモードで実行する方式を実装した。これにより、従来は CPU のメモリ管理ユニット (MMU) を利用した、カーネルモードとユーザモードという保護ドメインの切り替えを省くことができ、システムコールのオーバーヘッドが削減できる。カーネルのインターフェースとしては、従来のシステムコールの内部関数を利用しているので、ファイルアクセス制御等の機能は通常の Linux と同等である。実際に find や ping などの簡単なベンチマークプログラムで実験し、性能の向上を確認した。

## 2.5 分散計算におけるアクセス制御のための型システム [発表 6]

分散計算のモデルである  $\pi$  計算において、通信チャンネルの形で表された計算リソースに対するアクセス制御のための型システムを定義した。これにより、たとえばオンラインショッピングのサーバを  $\pi$  計算のプロセスとして表し、型検査を行うことによって、ある客が他の客のカートをのぞいたり商品を入れたりできない、といった性質を検証することができる。類似の型システムと比較すると、型検査はあくまで静的でありながら、アクセス制御における保護ドメインやその上下関係を動的に変化・生成できることが特長である。それにより、たとえばオンラインショッピングの例において、不特定多数 (無限) の客が存在しても、検証は有限時間で可能である。これは技術的には、動的に生成される名前に関する依存型を用いることにより可能となっている。

## 3 今後の展望

来年度は、上述の研究をそれぞれ発展させるとともに、それらを複合した研究も計画している。たとえば

- オブジェクト使用解析と同様の技術を、型付きアセンブリ言語におけるメモリ管理に応用する
- Linux/TAL にアクセス制御の型システムを導入し、メモリ安全性よりも高度なセキュリティを OS ではなく言語のレベルで静的に実現する
- Fail-Safe C から、OS の安全な拡張などに利用される proof carrying code へのコンパイラを開発する

といった組み合わせが考えられる。

## 4 平成 13 年度成果要覧

### 招待講演 / 招待論文

- [招待 1] Akinori Yonezawa. Overview of the Japanese Inter-University Research Project on Software Security. Fourth International Symposium on Theoretical Aspects of Computer Software (TACS 2001), October 2001.

### 特許申請 / 取得

- [特許 1] ソフトウェアの安全な自動組み込みシステム. 出願中.

### 特記事項

- [特記 1] 第 3 回プログラミングおよびプログラミング言語ワークショップ (PPL 2001) 論文賞 [発表 1].

### 発表論文

- [発表 1] 住井 英二郎, Benjamin C. Pierce. The Cryptographic  $\lambda$ -Calculus: Syntax, Semantics, Type System and Logical Relations. 第 3 回プログラミングおよびプログラミング言語ワークショップ (PPL 2001), 97–108 頁, 2001 年 3 月. 論文賞受賞 [特記 1].

- [発表 2] 関口 龍郎, 大岩 寛, 米澤 明憲. オブジェクト指向言語によって記述された、携帯電話・PDA のアプリケーションプログラム圧縮方式. 第 3 回プログラミングおよびプログラミング言語ワークショップ (PPL 2001), 121–126 頁, 2001 年 3 月. 「コンピュータソフトウェア」第 19 巻第 1 号 (2002 年 1 月) 1–9 頁掲載.

- [発表 3] 後藤 礼史, 田浦 健次朗, 米澤 明憲. Secure Shared Memory: オブジェクトを効率良く安全に共有するためのモデル.

第3回プログラミングおよびプログラミング言語ワークショップ (PPL 2001), 42-50 頁, 2001 年 3 月.

- [発表 4] Eijiro Sumii, Benjamin Pierce. Logical Relations for Encryption. 14th IEEE Computer Security Foundations Workshop, pp. 256-269, June 2001.
- [発表 5] 大岩 寛, 住井 英二郎, 米澤 明憲. 安全性を保障する ANSI-C 実行系の実装手法. 日本ソフトウェア科学会第 18 回大会, 2001 年 9 月. 「コンピュータソフトウェア」採録予定.
- [発表 6] Daisuke Hoshina, Eijiro Sumii, Akinori Yonezawa. A Typed Process Calculus for Fine-Grained Resource Access Control in Distributed Computation. the Fourth International Symposium on Theoretical Aspects of Computer Software (TACS 2001), Lecture Notes in Computer Science, vol. 2215, pp. 64-81, October 2001.
- [発表 7] 前田 俊行, 住井 英二郎, 米澤 明憲. Linux/TAL: 型付きアセンブリプログラムのカーネルモード実行方式. 第 4 回プログラミングおよびプログラミング言語ワークショップ (PPL 2002), 2002 年 3 月.
- [発表 8] 浜中 信行, 住井 英二郎, 小林 直樹, 米澤 明憲. Java バイトコードにおけるオブジェクト使用解析のための型システム. 第 4 回プログラミングおよびプログラミング言語ワークショップ (PPL 2002), 2002 年 3 月.