様式 C-19

科学研究費補助金研究成果報告書

平成２３年　６月１３日現在

機関番号：１２６０１
研究種目：特定領域研究
研究期間：2006～2010
課題番号：１８０４９０２７
研究課題名（和文）　情報爆発に対応する高度にスケーラブルでセキュアなソフトウェア構成・更新方式
研究課題名（英文）　Highly Scalable Software Security for Information Explosion Environments
研究代表者
柴山　悦哉（SHIBAYAMA ETSUYA）
東京大学・情報基盤センター・教授
研究者番号：８０１６２６４２

研究成果の概要（和文）：分散した多数のコンピュータを協調動作させるソフトウェアの安全性や信頼性を向上させるため、情報爆発時代に相応しい多重防御の枠組みを考案した。個々の対策技術が完全ではないことを前提に、ソフトウェアの開発段階における検証とテスト、運用段階におけるアップデートとモニタリングなどを組み合わせるものである。さらに、柔軟なソフトウェア構成を可能とするアスペクト指向などの考え方に基づき、スケーラブルな要素技術の開発も行なった。

研究成果の概要（英文）：We propose a Defense-in-Depth framework that is suitable for building secure and dependable software in the Info-plosion era, where software for large distributed computing systems is required.　Under the practical assumption that there are no perfect solutions, the framework is designed to integrate software verification and testing techniques in the development phase and dynamic update and monitoring in the operation phase.　In addition, we have developed scalable core technologies based upon the notions of Aspect-Orientation and other ideas for flexible software construction.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
２００６年度	11,900,000	0	11,900,000
２００７年度	16,400,000	0	16,400,000
２００８年度	11,200,000	0	11,200,000
２００９年度	15,400,000	0	15,400,000
２０１０年度	10,500,000	0	10,500,000
総　計	65,400,000	0	65,400,000

研究分野：計算機科学
科研費の分科・細目：情報学・ソフトウェア
キーワード：アスペクト指向，開発環境，ソフトウェア検証，ソフトウェアテスト，オーバーレイネットワーク，コンテンツ配信，分散計算，Web アプリケーション

１．研究開始当初の背景

人類が生み出すディジタル情報が爆発的に増加する情報爆発時代の到来により、その爆発する情報を処理する情報インフラおよびその情報インフラを制御するソフトウェアもまた爆発的に巨大化・複雑化することが予見されていた。そして、次の(1),(2)のような時代の要請が存在した。

(1) 今日の社会は、ソフトウェアに強く依存しており、爆発的に巨大化・複雑化するソフトウェアが暴走したり崩壊したりすると甚大な被害をもたらすおそれがある。そのため、大規模で複雑な情報インフラを制御する安全で信頼できるソフトウェアの構築方式が求められていた。

(2) 2000 年頃からソフトウェアのセキュリティホールに対する攻撃も多発するようになり、情報爆発時代の複雑で大規模なソフトウェアを悪意ある攻撃から守ることも重要であった。そのため、脆弱性のない（あるいは少ない）ソフトウェアの構築方式、脆弱性が皆無ではないソフトウェアを安全に運用する方式が求められていた。

２．研究の目的

この課題の大目標は、安全で信頼できる情報インフラを構築するためのソフトウェア技術を確立することである。そのために、ミドルウェアおよびアプリケーションソフトウェアのセキュアな構成・運用方法に関する研究が必要である。課題申請時には、特に重要と考えたアクセス制御、ホスティング、動的アプリケーション更新という三つの分野に注目し、利用シナリオに沿った技術開発を進めることとした。

長期的には、ソフトウェアのバグや脆弱性を完全になくす技術の開発を進めることが望ましいが、５年間の研究プロジェクトでその目標を達成するのは困難である。そこで、次善の策として、ソフトウェア検証、ソフトウェアテスト、動的ソフトウェアアップデート、モニタリングの４種類の防御策を組み合わせるスケーラブルな多重防御の基本的な方式と各防御策に関する要素技術の開発を目標として設定することとした。

３．研究の方法

次の４項目に関する研究を行い、さらに、これらの統合により、情報爆発時代に相応しい多重防御の基本的な枠組みを構築する。これは、ソフトウェアライフサイクルの各段階で適切な対策を行なうという考え方に従うものであり、(1) は設計から実装, (2) はテスト,(3)と(4)は運用の各段階での対策に相当する。

(1) ソフトウェアの形式検証により、ソフトウェアの正しさ（仕様を満たすように実現されていること）を機械的かつ網羅的にチェックする。これは、安全性と信頼性を保証するための究極の技法だが、コストがかさむため適用範囲は限られる。真に重要なソフトウェア部品に形式検証を適用し、これを第一の防御網とする。研究としては、ソフトウェア検証のスケーラビリティを向上させることで、その適用範囲の拡大を目指す。

(2) ソフトウェアのテストは、一般の開発現場でも使われている。本質的にサンプリング手法の一種であり、サンプル数が多いほど信頼性が高くなる傾向がある。これを第二の防御網とする。テストケースの準備、テスト環境の設定などのために人手で行なう作業が多数発生すると、スケーラビリティに問題が生じる。そこで、スケーラビリティを改善するために、主として自動化に関する研究を行なう。

(3) ソフトウェアアップデートは、ソフトウェアを最新の状態に保つことで、既知のバグや脆弱性の影響を抑えようとする対策技術である。これを第三の防御網とする。情報爆発時代においても、運用開始後にソフトウェアのバグが発覚し、あとから修正される可能性は高い。広域分散型のソフトウェアでは、全体をシャットダウンできないことも多いので、実行を継続しつつアップデートする技術が必要となる。そこでソフトウェアの実行時アップデートを可能とするソフトウェア構成法とアップデートの方式に関する研究を行なう。

(4)今日の広域分散システムはインターネットに依存しており、本来の計算に必要な情報や管理情報の交換を行っている。しかし、インターネット上で情報配信を安定的に行うのは容易ではない。フラッシュクラウドや DDoS 攻撃のような一時的アクセス集中により、性能低下や停止が起こりやすい。したがって、インフラの弱さを緩和することも大事になる。そこで、第四の防御網として、情報配信基盤の安定化を図る研究を行なう。

４．研究成果

「研究の方法」で述べた４項目のそれぞれに対応して以下のような成果を得た。

(1)スケーラブルな仕様記述方式：ソフトウェア検証を行なうためには、検証の対象となるソフトウェアの仕様をまず記述する必要がある。しかし、プログラムの規模が大きくなると仕様の記述量も爆発する。そのため、検証可能でかつ正しい仕様を記述すること自体が難しくなる。

この問題を解決するために、仕様記述にアスペクト指向の考え方を導入し、仕様の記述をモジュラに行なう方式を考案した。そして、その方式を採用したアスペクト指向表明記述言語 Moxa の設計と実現を行ない、さらに、Moxa の有効性を確認するために、Java 言語で記述されたプログラムを対象に、表明記述の実験を行なった。その結果、Java 言語用の標準的な表明記述言語 JML と比較して記述量の削減がはかれることを確認した。

表１は記述実験の結果の一部をまとめたものである。ここで Web は Java 言語で記述された Web アプリケーションを構成するクラス群であり、AST2J は抽象構文木の記述から

Visitor パターンに準じた Java クラスを生成するツールである。モジュール数が増え、総行数は減っている。つまり、より粒度の細かいモジュールを組み合わせ、全体としては冗長な部分が減っていることがわかる。

表１ Moxa の記述実験の結果（一部）

	JML		Moxa	
	Web	AST2J	Web	AST2J
モジュール数	15	1	21	5
表明数	525	32	295	24
行数	4250	720	1980	590

(2) ソフトウェアテストの自動化：ソフトウェア検証同様に、ソフトウェアテストにおいても、通常はテストすべきソフトウェアの仕様に基づきテストケースを作成する必要がある。これは仕様を「理解」していないとできない作業であり、一般には自動化が難しい。

そこで、この研究課題で特に注目している安全性や信頼性に関連した一部の性質に適用範囲を限定することで、テストケースの自動生成と問題の自動発見を行なう方式を考案した。そして、この方式を Web アプリケーションに適用した脆弱性検知ツール Volcano を設計・実装した。Volcano の方式では、Web アプリケーションに対する入力リクエストがセンシティブな操作（たとえば、SQL の呼び出し）に与える影響を一文字単位で追跡し、セキュリティポリシーに照らして不正な操作が行なわれていないかどうかを解析する。なお、このセキュリティポリシー自体は、アプリケーションには依存しない。

Volcano の有効性を確認するために、既知の脆弱性を含む数千行から数万行規模の Web アプリケーションを用いた評価実験を行なった。この評価実験では、比較対象として、ファズテスティング方式により自動化されたブラックボックステストを行なう脆弱性検知ツールとして有名な Paros を用いた。表２がその結果の一部である。Volcano の方が検知できる脆弱性が多く、また誤検知についても優れていることがわかる。

表２ Volcano の評価実験の結果（一部）

Web アプリ	行数	Paros	Volcano
		検知/誤検知	検知/誤検知
jobhut	2278	4/2	19/0
mycrocms	4256	0/0	9/0
pastelcms	4929	2/0	3/0
easymoblog	9996	1/0	10/0
phpaaCMS	13434	5/1	31/0

(3) 動的ソフトウェアアップデート方式：今日、OS やアプリケーションのバグを修正するために、ソフトウェアアップデートの技法がさまざまな箇所で使われている。Windows アップデートはその一例である。しかし、これらのソフトウェアアップデートの方式の中には、更新後にシステムやアプリケーションの再起動を要求するものが少なくない。

そこで、情報爆発時代に重要となる分散並列型アプリケーションを対象に、継続的に動作させたままでアップデートを行なえるソフトウェア構成方式を考案した。これはアスペクトと呼ばれる柔軟性の高いソフトウェアモジュールを定義し、これを既存のソフトウェアに動的に織り込むことを可能とする動的アスペクト指向の概念に基づくものである。そして、この方式を実現する言語として DandyJ を設計・実装した。DandyJ を設計するにあたっては、分散ソフトウェアにアスペクトを織り込む箇所を指定するリモートポイントカット、実行時に織り込むことができる動的アスペクト、初期化等の１回のみの動作を表現するワンタイムアスペクトの三つの概念を導入した。

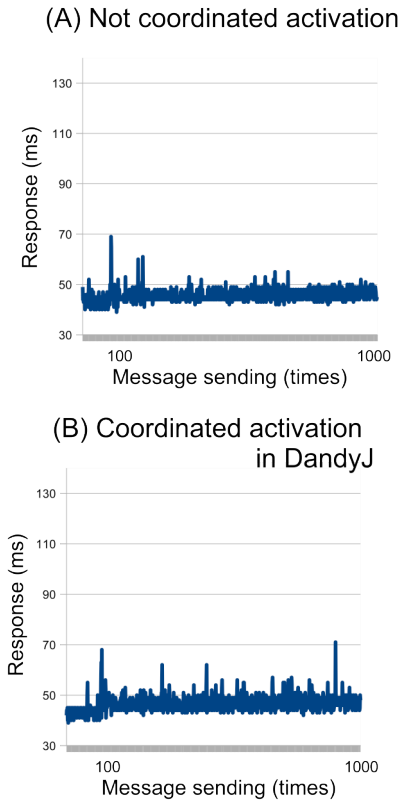

図１ DandyJ による同期織り込み

分散並列アプリケーションを対象とした場合，動的アスペクトを複数の計算ノードに同期的に織り込む方式が問題となる。全体を一時的に停止すれば、織り込みは比較的容易に実現できるが、パフォーマンスは低下する。一方、同期を取らなければ、ソフトウェア内

部の一貫性が崩れる危険性がある。DandyJ では、２段階の織り込みとワンタイムアスペクトを用いることで、この同期のオーバーヘッドを大幅に削減している。通信遅延が大きな環境でもこの方式が有効であることを確認するために、InTrigger 上での評価実験を行なった。その結果の一部を図１に示す。同期を行なった場合(B)に、同期を全く行なわない場合(A)と比べて遜色ない性能が達成できている。

(4)オーバーレイネットワークを用いた分散情報配信：分散アプリケーションの広域的なモニタリング等を行なうためには、安定した情報配信・交換の基盤をインターネット上に構築する必要がある。

そこで、オーバーレイネットワークによる方式を考案した。まず、各ノードに多次元空間内での位置座標を、ノード間の距離が両者間の通信遅延になるべく比例するように与える。この位置座標の情報を用い、ホップ数より予想遅延時間を重視した径路制御を行なう。さらに、コンテンツに対する需要が増加した場合にはミラーサーバを動的に起動する。これらの手法を組み合わせることで、通信遅延が小さく、需要の増大に耐えるオーバーレイネットワークが構築できる。

この方式の有効性をシミュレーションにより評価した。図２はその結果の一部であり、情報の更新処理に要する時間とメッセージ数を示している。提案したのは speculative update と呼ばれる方式であり、比較対象とした他の方式と比べ、高速であるか、やや遅くメッセージ数が大幅に少ないかである。

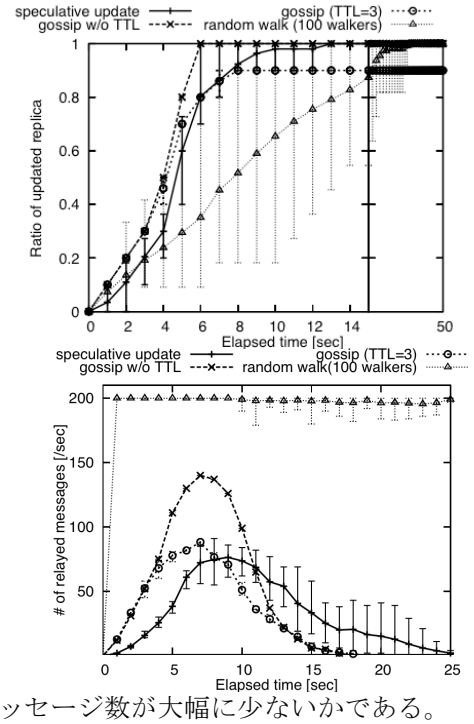

図２ Speculative Update の評価

５．主な発表論文等

（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計１４件）

① Toshinori Kojima、Masato Asahara、Kenji Kono、Ai Hayakawa、Practical Approach to Integrating Network Coordinates with Distributed Hash Tables 、IPSJ Transactions on Advanced Computing Systems、to appear

② 赤井駿平、千葉滋、コード領域を対象とする関心事を扱うためのアスペクト指向プログラミング言語の拡張、情報処理学会論文誌プログラミング、to appear

③ Masato Asahara、Kenji Kono、Ai Hayakawa、Toshinori Kojima、P2P-Based Approach to Finding Replica Server Locations for Alleviating Flash Crowds、IEICE Trans. on Information and Systems、Vol. E93-D、No. 11, pp. 3027-3037、2010

④ Ai Hayakawa、Masato Asahara、Kenji Kono、Toshinori Kojima 、A Strategy for Efficient Update Propagation on Peer-to-Peer based Content Distribution Networks 、IPSJ Transactions on Advanced Computing Systems、Vol. 3、No. 3、pp. 138-152、2010

⑤ Salikh Zakirov、Shigeru Chiba、Etsuya Shibayama 、How to Select Superinstructions for Ruby 、IPSJ Transactions on Programming、Vol. 2、pp. 1-8、2010

⑥ Kenichi Kourai、Hideaki Hibino、Shigeru Chiba 、Application-Level Scheduling Using AOP 、Transactions on Aspect-Oriented Software Development V、pp. 1-44, 2009

〔学会発表〕（計６５件）

① Michihiro Horie、Satoshi Morita、Shigeru Chiba、Distributed Dynamic Weaving is a Crosscutting Concern、Proceedings of the 26th Annual ACM Symposium on Applied Computing、to appear

② Thanh-Binh Dao 、Etsuya Shibayama 、Security Sensitive Data Flow Coverage Criterion for Automatic Security Testing of Web Applications、Proceedings of International Symposium on Engineering Secure Software and Systems、Lecture Notes in Computer Science、Vol. 6542、pp. 101-113、2011

③ Thanh-Binh Dao 、 Etsuya Shibayama 、 Coverage Criteria for Automatic Security Testing of Web Applications、Proceedings of International Conference on Information Systems Security、Lecture Notes in Computer Science、Vol. 6503、pp. 111-124、2010
④ Shigeru Chiba、Atsushi Igarashi、Salikh Zakirov、Mostly Modular Compilation of Crosscutting Concerns by C ontextual Predicate Dispatch、Proceedings of ACM OOPSLA、pp. 539-554、2010
⑤ Salikh Zakirov、Shigeru Chiba、Etsuya Shibayama、Optimizing Dynamic Dispatch with Fine -Grained State Tracking 、Dynamic Language Symposium、pp. 15-26、2010

〔図書〕（計１件）

① Ralf Lämmel, João Saraiva, Joost Visser 編、Dan S. Batory, Jean Bézivin, Shigeru Chiba 他 、 Generative and Transformational Techniques in Software Engineering, Springer, 2006.

〔その他〕

６．研究組織

(1)研究代表者

柴山 悦哉（SHIBAYAMA ETSUYA）
東京大学・情報基盤センター・教授
研究者番号：８０１６２６４２

(2)研究分担者

千葉 滋（CHIBA SHIGERU）
東京工業大学・大学院情報理工学研究科・教授
研究者番号：８０２８２７１３

渡部 卓雄（WATANABE TAKUO）
東京工業大学・大学院情報理工学研究科・准教授
研究者番号：２０２２２４０８

河野 健二（KONO KENJI）
慶應義塾大学・理工学部・准教授
研究者番号：９０３０１１１８

(3)連携研究者

（ ）

研究者番号：