平成22年度 情報工学コース卒業研究報告要旨

坂部 研究室	氏　　名	青山知由
卒業研究題目	動的監視によるプログラムの実行時安全化	

情報システムによって扱われる情報の量が増えるにつれて，機密性の高い情報を情報システムで扱う機会も増えてきている．このため，プログラムにおける非干渉性の研究が行われている．非干渉性とはプログラムの出力が機密性の高い変数に全く依存せず，出力の変化から機密性の高い変数の値を知る事ができない事を言う．

手続き型プログラムの非干渉性を実現するための手法の一つとしてオートマトンによる機密性監視手法が Guernic et al., によって提案されている．このオートマトンによる機密性監視手法は機密性の高い変数を条件式にもつ条件分岐内での出力を全て削除する．なぜなら，これらの出力は機密性の高い変数値の変化からくる実行経路の変化によって，出力の有無が変わるため機密性の高い変数に依存していると言えるからである．本研究の目的は，プログラムの非干渉性を保ちつつ機密性の高い変数を条件式に持つ条件分岐内での出力を可能な限り出力する手法の開発である．

本研究ではインタープリタを用いたプログラム実行時の動的監視手法を提案する．このインタープリタはプログラムを受け取り，プログラムの出力が機密性の高い変数に依存しているかどうかを監視しながらプログラムを実行する．そのために，このインタープリタは実行する命令毎に下記のような動作をする．出力命令では機密性の高い変数の値を出力する場合その出力の値を関係のない値に変更する．代入命令では代入によって機密性の高い変数に影響を受ける変数があるならばその変数を機密性の高い変数として記憶し，逆に機密性の高い変数の影響が無くなる変数があるならばその変数を機密性の低い変数として記憶する．条件分岐命令ではその条件式に機密性の高い変数が含まれるならば，インタープリタ自身を再帰的に呼び出し条件分岐節を実行させる．そして，条件分岐節を実行したインタープリタの出力を受け取り機密性の高い変数に依存しない出力を計算し，条件文の出力とする．例えば，簡単な条件分岐命令文 if(c){S_{then}節 }else{S_{else}節 } の実行を考える．c が機密性の高い変数であるとき，この if 文を実行するインタープリタは再帰的に2つのインタープリタを呼び出し S_{then} 節と S_{else} 節を別々のインタープリタに実行させる．呼び出し元のインタープリタは2つのインタープリタの実行結果である出力系列をそれぞれ受け取り，それらの最長共通部分系列を if 文の出力とする．この最長共通部分系列は機密性の高い変数 c の変化による実行経路変化に依存していない．前述の命令以外の命令はインタープリタによって命令通りに実行される．

本手法が正しいこと，すなわち，プログラムの非干渉性を保つことをプログラムの構造に関する帰納法により証明した．さらに，オートマトンによる機密性監視手法よる出力は本手法でも必ず出力される事も同様に証明した．また，本手法のインタープリタとオートマトンによる機密性監視手法のインタープリタを実装した．これらの実装を用いてオートマトンによる機密性監視手法に比べ本手法が多くの出力を得られる例を発見した．

本手法では機密性の高い変数に依存した条件分岐内での出力を一部可能にするために条件分岐の真文と偽文両方をインタープリタが実行するのに対して，オートマトンによる機密性監視手法では真文のみ実行し偽文は静的な解析を行うだけにとどめている．このため本手法のインタープリタは機密性監視手法のインタープリタに比べてプログラムの実行が遅くなっている．インタープリタの実行速度の向上が課題として挙げられる．