情報漏洩防止のためのアクセス制御機構を備えた DF-Salvia の開発

樫山 武浩† 大石 達也†
瀧本 栄二† 毛利 公一†

1. はじめに

近年，USB メモリなどの記憶媒体やネットワークを通じた情報漏洩事件が頻繁に発生している．情報漏洩は，プライバシ侵害につながるだけでなく，企業に対しイメージダウンや補償問題による金銭的な負担を与える．日本ネットワークセキュリティ協会「2010 年情報セキュリティインシデトに関する調査報告書[1)]」によると，情報漏洩事件の原因として,「誤操作」,「盗難」,「管理ミス」,「紛失・置き忘れ」といった，正当なアクセス権限を持つユーザの人為的ミスや持ち出しなどの割合が高いことがわかる．これらは，暗号化や認証などの外部からの攻撃を防ぐことを目的としたセキュリティ技術で防ぐことは難しい．また，Windows や SELinux 等に実装されるファイルアクセス制御（以降，アクセス制御）では，情報漏洩防止の目的からすると，その機密性に適したアクセス制御ができないため，ユーザビリティが低下する．

以上の背景から，我々は，人為的ミスを要因とした個人情報漏洩を防止するためのアクセス制御機構を備えるオペレーティングシステム *DF-Salvia* の開発を行っている．*DF-Salvia* では，コンパイラと OS を協調させることで,「なに」が「どこ」に出力されようとしているのか、つまり保護データの機密度を考慮したアクセス制御を実現する．

2. DF-Salvia の概要

DF-Salvia では，データの機密度をデータ保護方針（以降，ポリシ）[2)] として定義し，それをファイル（以降，保護ファイル）単位に設定することを可能とする．そして，プロセスの動作を監視し，データが外部に出力されようとするとき，出力データの源となったファイルに設定されたポリシに従ってアクセス制御を行う．特徴となる技術を次に示す．

- ポリシの設定
  保護データを計算機資源に出力しようとしているプロセスに対するアクセス制御の内容と，その制御が行われる条件をファイル単位に設定可能とする．条件には，コンテキストと呼ばれるユーザ ID，時刻，計算機の位置，送信先計算機の IP アドレスなど，プロセスや計算機の状況を指定する．これにより，例えば「19:00〜7:00 の出力を禁止する」,「USB フラッシュメモリ内へのコピーのみ許可する」,「ネットワーク上への送信を禁止する」といった設定が可能となる．
- アクセス制御
  システムコールの発行を監視することで，プロセスによる外部へのデータの出力を検知する．そして，データが外部に出力されようとするとき，出力されるデータの源である保護ファイルを特定し，それに設定されたポリシに従ってアクセス制御を課す．これにより，人為的ミスが発生したとしても，保護ファイルに設定されたポリシに従い，情報漏洩を防止することが可能となる．
- データフローの静的解析
  上記のアクセス制御を実現するためには，出力されようとするデータから，その源である保護ファイルを特定するため，プロセス内のデータフローを把握する必要がある．これには，コンパイラでプログラムを静的解析することでデータフロー情報を求め，それを OS に提供することで実現する．このアプローチでは，プログラムの実行前にデータフロー解析を完了させることで，アクセス制御におけるオーバヘッドを軽減し，*DF-Salvia* の導入によるユーザビリティの低下を抑えることができる．

† 立命館大学

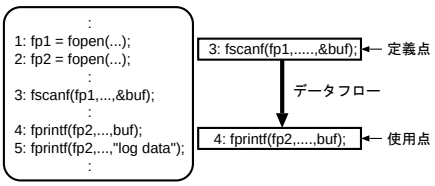

図 1 データフローの静的解析

3. データフローの静的解析

DF-Salvia のアクセス制御のためのデータフロー情報として，プログラムから定義使用連鎖を静的解析する．一般的な最適化では，変数の代入文を定義点とし，その変数を使用する文を使用点とする定義使用連鎖が解析される．一方，本解析では，保護ファイルから読み込まれたデータが外部に出力されるまでのデータフローを把握することを目的とすることから，ファイルからデータを読み込むライブラリ関数コールを定義点とし，定義点で読み込まれたデータを計算機資源に出力するライブラリ関数コールを使用点とする定義使用連鎖を解析する．

図 1 のプログラム例では，3 行目の fscanf が定義点，4 行目の fprintf が使用点とする定義使用連鎖を解析する．

4. アクセス制御手順

DF-Salvia におけるアクセス制御手順を図 2 に示すとともに，各手順の内容を次に示す．

(1) プロセスの実行前に静的解析したデータフロー情報を受け取る．

(2) read システムコールが発行された時，読み込み対象が保護ファイルなら，読み込まれるデータが属するデータフローに対して，対象保護ファイルのポリシを適用する．

(3) write システムコールが発行された時，書き出されるデータが属するデータフローに対して適用されたポリシの有無を確認する．

(4) ポリシが適用されている場合は，そのポリシに従ってシステムコールの実行の可否を判断する．

上記の処理を行うためには，手順 (2) , (3) において，システムコールが発行された時点で使用されているデータが属するデータフローを OS が特定できなければならない．データフローは，システムコールの発行元であるライブラリ関数コールの命令アドレスから求める．事前にプログラム解析するデータフロー情報として，read システムコールを発行するライブラリ関数コールを定義点，write システムコールを発行するライブラリ関数コールを使用点とする定義使用連鎖を使用する．また，それぞれの定義点，使用点について，ライブラリ関数コールの命令アドレスを求めておく．そのため，システムコールにおいて取得した命令アドレスとデータフロー情報に含まれる命令アドレスを比較することで，データフローを特定できる．なお，OS 実行時のライブラリ関数コールの命令アドレスは，システムコール発行時にプロセスのスタックを解析することで求める．

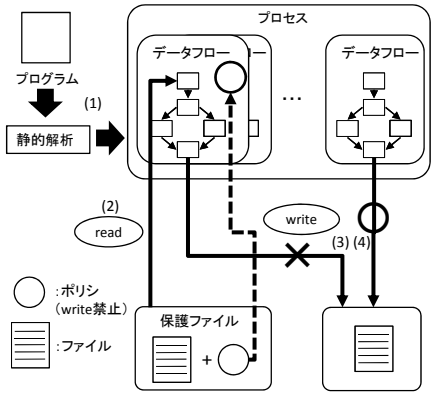

図 2 アクセス制御モデル

5. おわりに

本論文では，人為的ミスを要因とした個人情報漏洩を防止するためのアクセス制御機構を備える *DF-Salvia* について述べた．*DF-Salvia* は，プログラムから静的解析したデータフロー情報に基づくことで，出力されるデータに応じて決定したポリシに従ってアクセス制御を課すことを可能とする．

今後は，*DF-Salvia* のアクセス制御の精度を検証するとともに，その性能向上に取り組むことで，実環境への適応性の向上を図る．また，データフロー情報の自動生成を実現するセキュアコンパイラの開発を行う．

参 考 文 献

1) NPO 日本ネットワークセキュリティ協会：2010 年情報セキュリティインシデントに関する調査報告書 Ver.1.4 `http://www.jnsa.org/result/incident/data/2010incident_survey_PIL_v1.4.pdf`

2) 鈴来 和久，他，Privacy-Aware OS *Salvia* におけるデータアクセス時のコンテキストに基づく適応的データ保護方式，情報処理学会論文誌:コンピューティングシステム，Vol.47，No.SIG3(ACS 13)，pp.1-15，情報処理学会（2006）