6L-2

静的解析による Android パーミッションの利用目的の可視化方法

坂下卓弥† 小形真平† 海谷治彦† 海尻賢二†

信州大学工学部情報工学科†

1. はじめに

近年，Android の普及が進み，年間数十万の Android アプリケーション(以後，単にアプリ)が公開されている．それに伴い，Android 向けのマルウェアが急速に増加している．Android にはユーザがアプリを安全に利用するために，個人情報を漏えいする等の動作を防ぐためのパーミッションシステムがある．ユーザは本システムを通して，アプリが「連絡先データの読み取り」や「完全なインターネットアクセス」等のパーミッションを利用することをインストール時に許可する．しかし，本システムはパーミッションが何の目的に利用されるかを通知しないため，ユーザがその目的を理解せずに許可し，マルウェアによる個人情報漏えい問題が生じている．

本研究ではユーザがパーミッションの利用目的を理解しやすいように，アプリのソースコードを静的解析し，その目的を可視化する方法を提案する．

2. パーミッションシステムと問題点

Android には，個人情報やセキュリティに関わる情報へのアプリのアクセスを制限するパーミッションシステムがあり，これらの情報へアクセスするためには，アプリ内でパーミッションを宣言する必要がある．パーミッションの種類としては「インターネットへのアクセス」，「連絡先の読込，書込」等があり，ユーザはアプリのインストール時にパーミッションが利用されることへの同意を求められる(図１)．

しかし，図１の「完全なインターネットアクセス」表記に見られるように，ユーザは何の機能が使われるかは理解できるが，何の目的に機能が使われるかは理解できず，マルウェアをインストールしてしまう危険性がある．

そこで我々は，アプリが個人情報を漏えいさせるマルウェアかどうかをユーザがより判断しやすいように，アプリのソースコードからパーミッションの利用目的を解析・可視化する方法を提案する．

図 1. パーミッションのユーザへの通知

3. 静的解析による利用目的の可視化方法の提案

個人情報を漏えいさせるマルウェアの場合，例えば電話帳の情報が取得され，その情報を外部に送信する可能性がある．本研究では，図 2 のようにある URL に対して送信されているデータを可視化することにより，ユーザはパーミッションの利用目的を理解できると仮定する．そこで，Eclipse の構文解析器である ASTParser を用いてソースコードの静的解析を行い，可視化を行う．正確な情報を得るには動的解析が有用であるが，Android を対象とした自動的かつ網羅的な動的解析は難しく，十分な手法がない．そこで，動的解析すべき箇所を限定するため，静的解析により可能な限りの解析を行う．

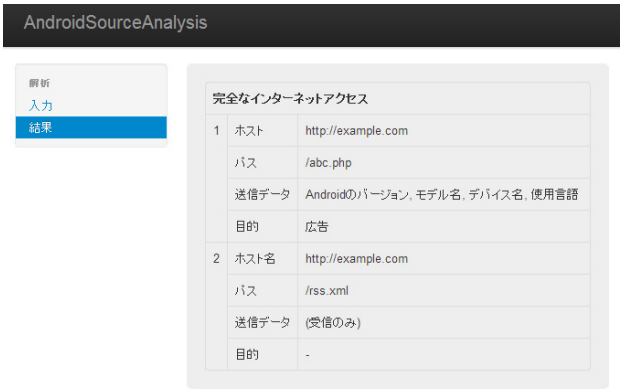

図 2. 解析結果のイメージ図

解析の流れは以下の通りである．

(1) エントリポイントのクラスを取得し，1 行ずつ評価する．

(2) 外部に送信されるデータを特定するために

Visualization of what applications are doing with permissions in Android by static analysis.

†Takuya Sakashita, Shinpei Ogata, Haruhiko Kaiya, Kenji Kaijiri. Department of Information Engineering, Faculty of Engineering, Shinshu University.

は，変数と API の関係を紐付けることが不可欠である．そこで，次の関係を紐づける．(A)情報入力部となるパーミッション(READ_CONTACTS や READ_SMS 等)が必要な API とそのパーミッション，(B)代入関係にある変数と固定値，(C)代入関係にある変数と変数，(D)代入関係にある変数と API．(E)API の引数に用いられる変数とその API．

(3) 最後に，外部へ送信する情報を特定する．(2)にてパーミッションや固定値に紐付いた変数が，情報出力部のパーミッション(INTERNET や SEND_SMS 等)が必要な API の引数の場合，外部への送信内容とみなす．また，接続先 URL が広告 URL リストに一致する場合，図 2 の目的に”広告”を出力する．ここで，広告 URL リストとは，広告目的の既知の URL をリスト化したものであり，解析への入力とするものである．

次節から，ソースコードを解析する際の Android 特有の課題と対処法について述べる．

3.1 パーミッションの必要な API の特定

パーミッションの必要な API を特定するためには，API とパーミッションの対応関係のデータが必要となるが，公式のリファレンスにはこのデータに不足や誤りがあることが既存研究[2]で述べられている．そのため，提案手法では，既存研究[2]により得られる permission map を利用する．表 1 にその対応関係の例を示す．例えば，java.net.URL.getContent()の呼び出し時に，INTERNET のパーミッションが必要なことを表す．

表 1. Permission map(一部抜粋)

API	パーミッション
java.net.URL.getContent()	android.permission.INTERNET
android.telephony.TelephonyManager.getLine1Number()	android.permission.READ_PHONE_STATE

3.2 インテントへの対処

Android には他のアプリとデータをやりとりするためにインテントと呼ばれる機能があり，情報入出力部として解析する必要がある．Android には，本機能を利用するための API が用意されており，3 章にて述べた方法により対処できる．

3.3 隠蔽されているメソッドへの対処

Android にはアプリの画面を表すアクティビティがあり，このアクティビティには固定のサイクルにて呼び出される onCreate や onStart 等の API がある．同様に，非同期処理を行う AsyncTask クラスがあるが，execute メソッドを呼び出すと onPreExecute， doInBackground，onPostExecute メソッドが順に呼び出されるため，これらのような API の解析に対処する必要がある．提案手法では，これらの API の解析順序をハードコーディングにより決定しているが，柔軟な解析を実現できるように，このような点に絞り動的解析を導入する方法を今後検討する．

4. 適用事例

INTERNET パーミッションを用いて取得した RSS をリスト化して表示するアプリに対し，作成した可視化ツールを適用した．結果として，接続先 URL の可視化に成功した．

5. 関連研究

ユーザがインストール前にマルウェアを適切に判断するための支援として secroid[1]がある．secroid は Google Play の公開アプリを解析し，リスクレベル判定とリスクの一覧表示を行う．ここで，リスクとは，電話帳を外部に送信する等，ユーザに不利益となりうる動作を指す．リスクレベルは SAFE から DANGER までの 5 段階に判定されるため，ユーザはアプリのリスクを直感的に理解しやすいが，正当なアプリのリスクレベルが高く判定されてしまう問題点がある．我々は，アプリの挙動に関して情報漏えいの観点から，よりきめ細かい判断材料をユーザに提供できるように可視化方法を提案した．

6. おわりに

本論文では，静的解析によりアプリの利用目的を可視化する方法を提案した．そして，アプリのソースコードを静的解析した場合の Android 特有の課題と対処法及び適用事例の解析結果について述べた．

今後の課題として，さらに柔軟な解析が行えるようにツールの改良を行うとともに，静的解析における情報取得の限界を調査し，動的解析を併用して精度を上げる．また，ユーザがパーミッションの利用目的を理解し，マルウェアを見過ごし難くなっているかを評価する．

参考文献

[1] ネットエージェント株式会社, Android アプリの潜在リスクチェック secroid(セキュロイド), http://secroid.jp/, 2013 年 1 月 5 日.

[2] A.P.Felt, E.Chin, S.Hanna, D.Song, and D.Wagner, Android Permissions Demystified, Proc. of the 18th ACM Conference on Computer and Communications Security, CCS '11, pp.627-637 (2011).