様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成　２９　年　　５　月　２４　日現在

機関番号：　８７１０３
研究種目：　基盤研究(C)（一般）
研究期間：　2014～2016
課題番号：　２６３３０１６９

研究課題名（和文）Androidアプリケーションのセキュリティ検証技術研究

研究課題名（英文）Analyze the security of Android applications

研究代表者
松本　晋一（Matsumoto, Shinichi）

公益財団法人九州先端科学技術研究所・その他部局等・研究員

研究者番号：８０６２４７７５

交付決定額（研究期間全体）：（直接経費）　　3,700,000円

研究成果の概要（和文）：世の中に出回っているスマートフォンアプリにはユーザーのプライバシ情報（ユーザーのメール内容や所在情報など）を漏洩させるなど悪性のものがあります．本研究では，スマートフォンの動作基盤であるフレームワークに動作ログの記録機構を組み込み，アプリ動作時のログの取得/分析手法を開発しました．ログを基に機械学習という手法を用い悪性アプリを分類する方法，また同じくログからアプリ内の広告表示の動作を調べる方法を開発しました．更に，全てのスマートフォンプラットフォームが備えるウェブブラウザがアプリの共通プラットフォームとなっていることに着目し，ブラウザに記録された情報をブラウザ外から取り出す手法を開発しました．

研究成果の概要（英文）：Nowadays, considerable part of malicious smartphone applications are distributed in marketplaces globally. Such applications leaks users' privacy (e.g. mail text, location) to outside of terminal. First, I developed a method to embed the code to perform API call logging mechanism into smartphone framework. This framework logs are recorded when some applications are running in the smartphone. Based on these application execution logs, I analyzed them and extracted the group of malicious applications with machine learning technique. Furthermore, I developed the method to analyze the behavior of advertising libraries linked with smartphone applications from these framework logs. In addition, I expected that all of smartphone platforms equip web browser and it is utilized as application execution foundation in smartphone. On that premise, I developed the digital forensics method to extract and analyze the information stored in web browser from without legitimate web programming interface.

研究分野：　情報セキュリティ

キーワード：　情報セキュリティ　モバイル　Android　フォレンジクス　プライバシ　機械学習

１．研究開始当初の背景

(1) 近年，Android などのスマートフォンは急速に普及しており，またスマートフォン上で利用できるアプリケーションも市場に多数流通している．しかしそれらの中にはユーザーのプライバシ情報を漏洩させる悪性のものや，脆弱性を持ったものも多々存在する．このようなアプリケーションはユーザーに直接的な被害を及ぼし，利便性を損なうことになる．

(2) またそのようなアプリケーションは，前記の脅威のみならず，スマートフォン市場の健全な成長の妨げとなっており，スマートフォンアプリケーション開発概査，アプリケーション市場運営者などの様々なステークホルダのビジネス機会を損なう状況となっている．

２．研究の目的

本研究は，スマートフォン用の悪性/脆弱なアプリケーションの流通を防ぐこと，またモバイル環境におけるプライバシ情報の漏えい/に関する技術の開発を目的としている．このために，

(1) Android アプリケーションの振る舞いから，(ユーザーのプライバシ情報の漏えいを引き起こすような)悪性/脆弱性を持つものを識別/分類する

(2) Android アプリケーションの開発において開発者の手の及ばないサードパーティライブラリ，特に広告ライブラリについて，同様にアプリケーションの振る舞いからその挙動を分析する．

(3) スマートフォンなどモバイル環境におけるプライバシ上の扱いについて明らかにするための，スマートフォンプラットフォームであるウェブフレームワークの保持する情報のフレームワーク外からの取得方法と，解析方法を実験，検討する．

３．研究の方法

(1) 前節第 1 項に関し，Android フレームワークのソースコードレベルに対して改造を施す．Android フレームワークとは Android のスマートフォンとしての主要なサービスを実現し，それらを アプリケーションに対して API として提供するソフトウェア階層であり，カーネル(Linux カーネル)の上部に位置するものである．

改造として，Android フレームワークにおける記録対象となる API のエントリポイントにおいてログ処理を追加する．当該改造を施された Android フレームワーク(以下，改造フレームワーク)により API の呼び出し履歴が記録可能となり，記録されたログは以降の解析に提供される．

改造フレームワークを用いてアプリケーション実行時のログを取得し，カーネルのシステムコールログと組み合わせ API/システムコール呼び出し履歴のベクトルを用い，K 平均法を用いることで，アプリケーションのユーザープライバシ漏洩といった悪性についての分類を行う．

(2) 前節第 2 項に関し，前項と同様にフレームワークに改造を施す．フレームワークにおけるネットワーク通信箇所と画面の描画箇所においてログを取得するよう改造したフレームワークを用いる．これは，広告ライブラリが，広告コンテンツをサーバーから読み出し，画面に(多くは上端もしくは下端)表示するという動作を，一定の時間間隔で行うという観測に基づくものである．

改造フレームワークを用いてアプリケーション実行時のログを取得し，HTTP 通信 API の呼び出し時刻と，画面再描画 API の呼び出し時刻，両時刻のフーリエ変換を行い，周波数領域での相関を求めることで，アプリケーションの広告ライブラリの挙動についての判別を行う．

(3) 前節第 3 項に関し，HTML5 ブラウザの備えるクライアントストレージ機能である webStorage に記録された内容を複数の経路で取得可能か，またどのようなフォーマット，コードで記録されているかの実験を行う．各種ブラウザについて webStorage に値を設定後，ブラウザを終了した後にメモリイメージをダンプし，主記憶内に残された当該値を探索した．webStorage は，いわゆる key-value ストアと呼ばれる形式で情報を保持し，またセキュリティ/プライバシの保全のため，URL 類似のオリジンと呼ばれる単位で情報の隔離を行う．

探索の結果 webStorage に記録された値がどのようなコード(文字コード)で収容されているか，どのようなフォーマットで key-value またオリジンの関係を記録しているかを分析し，これを主要な各種ウェブブラウザに対し明らかにした．

４．研究成果

(1) Android アプリケーションの悪性判断を，Android カーネルログを元に判断する手法は既に提案されている(引用文献①)．しかし当該手法では Android フレームワーク固有の機能の呼び出しは検知できない．ユーザーID，電話帳などスマートフォンユーザーのプライバシに関する機能は Android フレームワークが提供することから，当該フレームワークの呼び出し(API)を記録し，特徴量とすることでアプリケーションの悪性判断への影響を見た．ユーザーID 読み出しに関する API の呼び出し回数をロギングするよう改造したフレームワークを用い，当該回数を特徴量として K 平均法で判定した結果を表 1 に示す．

CASE1 は API/システムコール呼び出し回数を特徴量とした場合，CASE2 は API/システムコール呼び出し回数とエラーの回数を特徴量とした場合，CASE3 は API/システムコール呼び出し回数からエラー回数を引いたものを特徴量とした場合である．また特徴量の時限を主成分分析(PCA)で圧縮したものも加え比較している．

表からは，次元圧縮を行わない条件で，API 呼び出し回数を特徴量に加えることで判定精度が改善していることが分かる．特にエラー回数を引いた特徴量を用いた場合に 90%以上という高い精度が得られた．

	API あり		API なし	
	PCA なし	PCA あり	PCA なし	PCA あり
CASE1	87.5%	62.5%	83.3333%	58.3333%
CASE2	79.1667%	58.3333%	79.1667%	62.5%
CASE3	91.6667%	58.3333%	83.3333%	54.1667%

表 1 API の有無に対する判定精度

このような結果から，悪性の Android アプリケーションの検証技術について，既存の，カーネルログのみを用いた方式より優れた精度を達成することができた．

(2)Android フレームワーク内に API ログ取得のための改造を加える点は前項と同様の手法として，HTTP 接続に関する API と画面再描画に関する API の呼び出し記録を元に広告ライブラリの判定を行った．

アプリケーション実行後，ログより，HTTP コネクション API の呼び出し時刻と画面再描画 API の呼び出し時刻を抽出し，フーリエ変換を行うことで周波数領域での処理を行う．この概念を図 1 に示す．

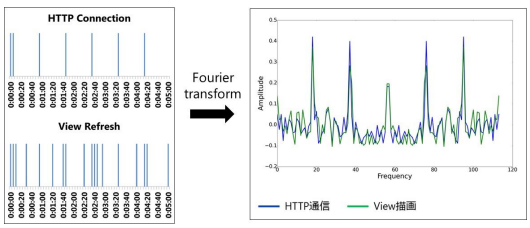

図 1 ログからのフーリエ変換

変換の結果得られた API 呼び出しタイミングに関する記録を，相関関数(式 1)にて処理し，二種類の API の呼び出しタイミングの相関を求める．

$$\frac{\sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\overline{y})^2}} \quad \cdots\cdots (1)$$

結果を表 2 に示す．検出率 80%以上に対し，偽陽性も 30%程度という結果が得られた．また API 呼び出し時刻のフーリエ変換に関わる時間幅(time slot width)は結果に有意な影響を及ぼさないことも判明した．

Time slot width(s)	Detection rate(%)	False positive(%)
1	81.42	30.64
2	82.17	29.79
5	81.44	32.33
10	78.25	33.50

表 2 API タイミングに基づく検出結果

このような結果から，Android アプリケーションの広告ライブラリの挙動について，アプリケーションバイナリ(実体)を検証することなく，その振る舞いから明らかにすることが可能になった．当該技術は特にアプリケーション市場の運営者が，アプリケーション開発者により登録されたアプリケーションの検証(契約内容に準拠しているかの判定など)に用いることができる．

(3) HTML5 の持つ webStorage 機能を用いクライアントサイドに情報を記録させた結果を，ファイルシステムとメインメモリと 2 つの手法(経路)から取得する実験を行った．まずファイルシステムからの取得について，ウェブブラウザ毎に異なるストレージ内容記憶のパスは，表 3 のようになった．

Browser	Supporting Version	Stored Location(Path)
Internet Explorer	8 or later	Fail to confirm
Firefox	3.6 or later	C:¥Users¥<username>¥AppData¥Roaming¥Mozilla¥Firefox¥Profiles¥<profile folder>
Google Chrome	8 or later	C:¥Users¥<username>¥AppData¥Local¥Google¥Chrome¥UserData¥Default
Opera	11 or later	C:¥Users¥<username>¥AppData¥Roaming¥Opera Software¥Opera Stable
Safari	5 or later	C:¥Users¥<username>¥AppData¥Local¥Apple Computer¥Safari

表 3 ブラウザ別のパス

またこれらの内容はファイルシステム内では JSON(JavaScript Object Notation)形式で収容されていることが判明した．これら記録内容の可視化を行うツールを，Firefox ブラウザについて実装した．当該ツールによる出力結果の例を図 2 に示す．

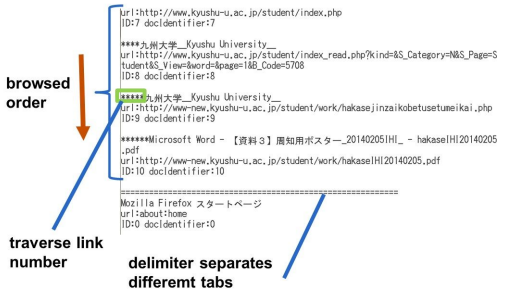

図 2 可視化ツール出力結果例

またメモリイメージからの取得のための実験装置を，図 3 のように構築した．ブラウザ外での情報が実験結果に影響を及ぼすのを防ぐため，記録する文字列は簡単なエンコードを施し，ブラウザ内でデコードを行い記録する．実験は webStorage の種類が二種類ある(sessionStorage，localStorage)ことに対応し，ブラウザ毎に二系列の実験を行った．

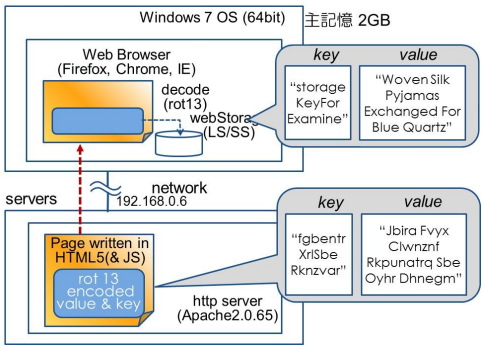

図 3 webStorage 実験装置

実験により得られた結果，ウェブブラウザやストレージの種別による違いをまとめたものを表 4 に示す．

Web browser	storage type	evidence			code	format
		key	value	origin		
	ls	5 of 5 (all)	5 of 5 (all)	none	ASCII	XMLエレメント
	ss	none	none	none	-	-
	ls	none	none	none	-	-
	ss	1 of 5	1 of 5	1 of 5	ASCII	JSON類似(但しLFで区切)
	ls	1 of 5	1 of 5	none	UTF-16	連結された文字列
	ss	none	none	none	-	-

表 4 記録形式，コードのまとめ

実験の結果，ブラウザ実装により，またストレージ種別により取得できる情報に違いがあること，またコードとフォーマットの違いを明らかにした．

当該成果は，ウェブブラウザの今後の HTML5 へ向けた移行においてプライバシがどのように扱われうるかを明らかにしたものであり，今後 HTML5 が特にモバイル環境におけるアプリケーション実行基盤として普及するにあたって重要になると考えている．この成果は直接的にはデジタルフォレンジクスのための技術として活用できるものと考えており，そのためにはウェブブラウザ上で動作するアプリケーションの種別毎のストレージ記憶内容の解析とあわせ，ユーザーの行動(アプリケーション処理情報)の解明に関わる技術に発展させることが可能なものと考えている．

＜引用文献＞

① Iker Burguera, Urko Zurutuza, Simin Nadjm-Tehrani, Crowdroid: behavior-based malware detection system for Android", In Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices. pp.15-26, 2011.

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 1 件）

① Shinichi Matsumoto, Kouichi Sakurai, Reconstructing and Visualizing Evidence of Artifact from Firefox SessionStorage, Information Security Applications, 査読有, Vol.8909, 2015, p.83-94
DOI: 10.1007/978-3-319-15807-1_7

〔学会発表〕（計 7 件）

① Shinichi Matsumoto, Kouichi Sakurai, Forensics Investigation from Residual Memory Image of WebStorage in HTML5 Web Browser, AsiaJCIS 2016, 2016 年 8 月 4 日～同 5 日, 西新プラザ(福岡市)

② 松本晋一, 櫻井幸一, ブラウザの HTML5 ウェブストレージに対するメモリフォレンジクス, 電子情報通信学会総合大会, 2016 年 3 月 15 日～同 18 日, 九州大学(福岡市)

③ 韓 燦洙, 松本晋一, 川本淳平, 櫻井幸一, Android マルウェアの API 処理ロギングによる個人情報漏えい検知及び分類, 火の国情報シンポジウム 2016, 2016 年 3 月 2 日～同 3 日, 宮崎大学(宮崎市)

④ Naoya Kajiwara, Junpei Kawamoto, Shinichi Matsumoto, Yoshiaki Hori, Kouichi Sakurai, Detection of Android Ad Library Focusing on HTTP Connections and View Object Redraw Behaviors, ICOIN 2015, 2015 年 1 月 12 日～同 14 日, Siem Reap(Cambodia)

⑤ 松本晋一, 櫻井幸一, HTML5 WebStorage 生成物のメインメモリイメージからの取得, コンピュータセキュリティシンポジウム, 2014 年 10 月 22 日～同 24 日, 札幌コンベンションセンター(札幌市)

⑥ Shinichi Matsumoto, Kouichi Sakurai, Acquisition of Evidence of WebStorage in HTML5 Web Browsers from Memory Image, AsiaJCIS 2014, 2014 年 9 月 3 日～同 5 日, Wuhan (China)

⑦ 松本晋一, 鬼塚雄也, 川本順平, 櫻井幸

二，デジタルフォレンジクスの為の Web 閲覧履歴可視化方式の提案，第 65 回コンピュータセキュリティ研究発表会，2014 年 5 月 22 日～同 23 日，ホルトホール大分(大分市)

〔その他〕
ホームページ等
http://www.isit.or.jp/lab2/matsumotoshinich/

６．研究組織

(1)研究代表者
松本 晋一（MATSUMOTO, Shinichi）
公益財団法人九州先端科学技術研究所・情報セキュリティ研究室・研究員
研究者番号： 80624775

(2)研究分担者
（ ）

研究者番号：

(3)連携研究者
櫻井 幸一（KOUICHI, Sakurai）
九州大学・大学院システム情報科学府・教授
研究者番号： 60264066

(4)研究協力者
川本 淳平（KAWAMOTO, Junpei）
梶原 直也（KAJIWARA, Naoya）
韓 燦洙（HAN, Chansu）
鬼塚 雄也（ONITSUKA, Yuya）