安全な Android アプリの提供を実現する アプリ開発・管理方式 ADMS の提案

上松晴信† 可児潤也†† 名坂康平† 川端秀明††† 磯原隆将††† 竹森敬祐††† 西垣正勝††††

†静岡大学大学院情報学研究科，432-8011 静岡県浜松市中区城北 3-5-1

††静岡大学情報学部，432-8011 静岡県浜松市中区城北 3-5-1

†††株式会社 KDDI 研究所，356-8502 埼玉県ふじみ野市大原 2-1-15

††††静岡大学創造科学技術大学院，432-8011 静岡県浜松市中区城北 3-5-1

あらまし 本稿では，アプリ開発者とマーケット運営者の二人三脚の運用によって，安全なAndroidアプリの提供を実現する仕組み（ADMS）を提案する．ADMSでは，(i) OSにセキュリティマネージャという機構を設けた上で，(ii) アプリ開発者に，セキュリティ上重要であるイベントを発生させる場合には，必ずセキュリティマネージャにそのイベントを通知するようにアプリを開発することを義務付けるとともに，(iii) その通知を実装していないアプリは不正アプリであるとみなしてマーケットから削除するというルールを適用する．

ADMS: an application development and management system to realize provision of secure Android applications

Harunobu Agematsu† Junya Kani†† Kohei Nasaka† Hideaki Kawabata†††

Takamasa Isohara††† Keisuke Takemori††† Masakatsu Nishigaki††††

†Graduate school of Informatics, Shizuoka University, 3-5-1 Johoku, Naka, Hamamatsu,Shizuoka, 432-8011 Japan

††Faculty of Informatics, Shizuoka University.

†††KDDI R&D Laboratories, Inc. 2-1-15 Ohara, Fujimino, Saitama, 356-8502 JAPAN

††††Graduate School of Science and Technology, Shizuoka University.

Abstract. To realize provision of secure Android applications, this paper proposes an application development and management system, or ADMS for short, that is operated and maintained by application developers and market manager. ADMS requires (i) Android OS to be equipped with a “security manager”, (ii) all application developers to embed an API for event notification into applications to tell every events to the security manager whenever application launches any security-related event, and (iii) market manager to delete all such applications that don’t include the event notification API.

1　はじめに

近年，Android OS が搭載された Android フォンが爆発的に普及している．Android フォンの特徴として，Android フォンを持つユーザは Android マーケットにアクセスし，アプリケーションソフトウェア（以下，アプリ）を自由にダウンロードしてインストールできることが挙げられる．また，Android アプリの開発環境は無償で提供されているため，誰でも自由にアプリを作成し，Android マーケットにアップロードすることができる．

しかし，このAndroidマーケットのオープンな性質により，近年，トロイの木馬と呼ばれる正規のアプリを装った不正アプリ(Gold Dream[1],Droid Dream[2]等)が Android マーケット上に紛れ込み，様々な被害をもたらしている．これらの不正アプリはインストールされると，フォアグラウンドでは正規のアプリと同様に振舞いながら，バックグランドでユーザの個人情報(SMS メッセージや通話記録など)を取得しそれらを外部のサーバへ送信することで，ユーザに気づかれることなく情報漏洩を引き起こす．また，これらの不正アプリはユーザの許可なく他のユーザに電話をかけることや，SMS メッセージを送ることも可能である．

不正アプリの被害を減らすためのアプローチとして，ユーザ側で対策を行う方法とアプリ提供側で対策を行う方法が考えられる．

ユーザ側での対策としては，アプリインストール時に表示されるパーミッションを各ユーザが注意深く確認し不正アプリかどうかの判断を行うことや，アンチウイルスソフトを活用するなどの方法が考えられる．しかし，Androidフォンのユーザのセキュリティに対する意識や知識は様々であり，ユーザ側で完璧な対策を行うことはきわめて困難である．

アプリ提供側での対策としては，提供するアプリをあらかじめ検査しておき，安全であると判断されたアプリのみをマーケットにて提供する方法が考えられる[3][4]．アプリ提供側で対策を行うことにより，ユーザのセキュリティ意識などに依存せず，各ユーザに対して安全なアプリのみを提供することが可能となる．しかしながら不正アプリの種類は膨大であり，すべての不正アプリを完全に漏れなく検出することは現実的には困難である．そのため，不正アプリの検査のすり抜けを可能な限り防止する仕組みを設ける必要がある．

そこで本稿では，アプリ開発者とマーケット運営者の二人三脚の運用によって，安全なAndroidアプリの提供を実現する仕組みであるApplication Development and Management System（以下ADMS）を提案する．ADMSでは，(i) OSにセキュリティマネージャという機構を設けた上で，(ii) アプリ開発者に，セキュリティ上重要であるイベントを発生させる場合には，必ずセキュリティマネージャにそのイベントを通知するようにアプリを開発することを義務付けるとともに，(iii) その通知を実装していないアプリは不正アプリであるとみなしてマーケットから削除するというルールを適用する．不正アプリは，(iii)のルールによる削除を逃れるためには，(ii)のイベント通知を行わざるを得ない．この結果，不正アプリの挙動は(i)のセキュリティマネージャに筒抜けとなり，セキュリティマネージャがポリシに則って実行の可否を判定することが可能となる．

この仕組みをアプリ開発者の負荷なく実現させるため，(ii)のイベント通知には ADMS API と呼ばれる専用の API 群を用意する．また，セキュリティマネージャがアプリにおける「不正な行為」を判別するための知識ベースを用意する．ただし，ADMS API と知識ベースの具体的な実現方法については，本稿では詳述しない．

2　既存研究

不正な Android アプリに対する対策は既にいくつか提案されている．

Enck らは，機密情報(SIM,IMEI 等)を taint(色づけ)し，この情報の流れを捕捉することによって情報漏洩をリアルタイムに検知する TaintDroid という仕組みを提案している[5]. 文献[5]では TaintDroid を実際に実装しての評価も行っており，そのオーバヘッドは最大 29%程度であったと報告されている.

竹森らは，業務に必要かつ安全なアプリのみを配信するホワイトリスト方式を提案している[6]. これは，管理者が予めインストールを許可するアプリのホワイトリストを作成しておき，ホワイトリストに含まれないアプリはインストール直後に削除する機構を Android フォンに組込むことで不正アプリの侵入を防ぐ方式である.

川端らは，文献[7]で，サーバ側からダウンロードされる JavaScript によってアプリのメソッドが実行され，その結果様々な不正行為が引き起こされる可能性があることを指摘した上で，JavaScript を利用するアプリに対し，静的解析を行うことで脅威を推定する手法を提案している.

3 Android のセキュリティ機構とその課題

Android OS では独自のセキュリティ機構が設けられている. 具体的には，図 1 に示したように Dalvik と呼ばれる仮想マシンが搭載されており，全てのアプリはこのサンドボックス上で実行される. Android では，アプリがサンドボックスの外の情報(端末の機密データなど)にアクセスするためには，アプリのインストール時にパーミッション(図 2)を提示し，ユーザの承認を得る必要がある.

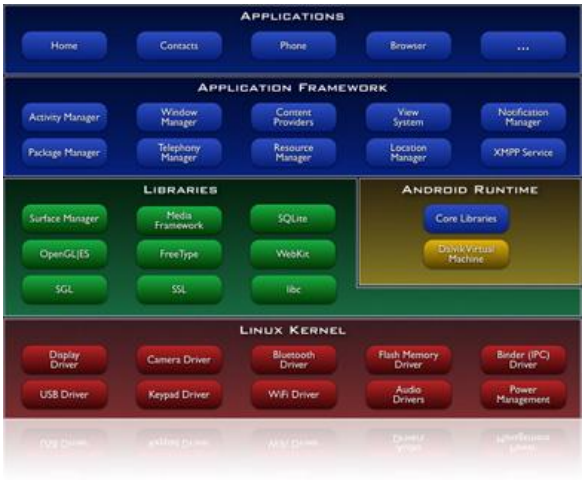

図 1. Android OS の構成[8]

アプリはパーミッションを得ることでさまざまな機能を提供する API(Application Program Interface)を使うことができる. この API は図 1 の Application Framework 層に定義されている. 一例として，READ_PHONE_STATE パーミッションによって使用することができる API の一覧を表 1 に示す.

図 2 アプリのパーミッション

表 1. READ_PHONE_STATE パーミッションによって使用可能な主要 API の一覧[9]

API	取得する情報
getLine1Number	電話番号
getDeviceId	デバイス ID
getSimCountryIso	SIM の国コード
getSimOperator	MCC+MNC (mobile country code + mobile network code)
getSimOperatorName	サービスプロバイダ名
getSimSerialNumber	SIM のシリアル番号
getSimState	SIM の状態(通信可能か, PIN ロックされているかなど)
getVoiceMailNumber	ボイスメール番号

ここで, 図 2 で示されるパーミッション情報は非常に抽象的であり, この情報だけではユーザは具体的にアプリがどのような機能を有し, どのリソースにアクセスするのか判断することができず, アプリが有する潜在脅威を理解することは難しい.

4 アプリ開発・管理方式 ADMS

本稿では安全な Android アプリの提供を実現するアプリ開発・管理方式 ADMS(Application Development and Management System)を提案する.

ADMS では, まず, (i)セキュリティマネージャを図 1 の Application Framework 内に組み込む. そして, (ii) アプリ開発者に対し, セキュリティ上重要であるイベントを発生させる場合には, 必ずセキュリティマネージャにそのイベントを通知するようにアプリを開発することを義務付ける. 合わせて, (iii) (ii)のイベント通知を実装していないアプリは不正アプリであるとみなし, マーケットから削除するというルールを適用する.

不正アプリは, (iii)のルールによる削除を逃れるためには, (ii)のイベント通知を行わざるを得ない. この結果, 不正アプリの挙動は(i)のセキュリティマネージャに筒抜けとなり, セキュリティマネージャがポリシに則って端末内でアプリの動作を抑制することができる. 必要によっては, 不正なアプリを削除するとともに, Android マーケットの管理者へ通知することも可能である.

Android アプリ開発者が(ii)のセキュリティマネージャへの通知機能を含んだアプリを開発する際の負荷を取り除くために, ADMS API と呼ばれる API 群を用意する. ADMS API は, Android OS に用意されているオリジナルの API の内, セキュリティ上重要であると思われるイベントを発生させる API(例えば getLine1Number 等)を抽出し, それらの API のそれぞれにセキュリティマネージャへの通知を行う処理を付加するという改造を加えたものである. これによって, アプリ開発者は(ii)のイベント通知を意識せずに従来通りの方法でアプリを開発しておけば, API を ADMS API に入れ替えるだけでセキュリティマネージャへのイベント通知を備えたADMS準拠アプリを作成することが可能となる.

また, ADMS を実効的な仕組みとするためには, すべてのアプリからセキュリティマネージャに通知されるイベント情報をセキュリティマネージャがリアルタイムに検査し, アプリの不正な挙動を動的に検知する方法の確立が肝要となる. この実現にあたっては, イベント情報からアプリの「不正な行為」を判別するためのデータベース(検査用知識ベース)を構築する. セキュリティマネージャは, この知識ベースを参照してアプリから通知されるイベントを検査することにより, アプリの不正なイベントを発見する. 本稿ではこの知識ベースの実装方式については特に詳述しないが, ブラックリストとのパターンマッチングやアノマリ検知によるビヘイビアブロッキング等, 種々のアプローチによる実装が考えられる.

セキュリティマネージャの全体図を図 3 に示す.

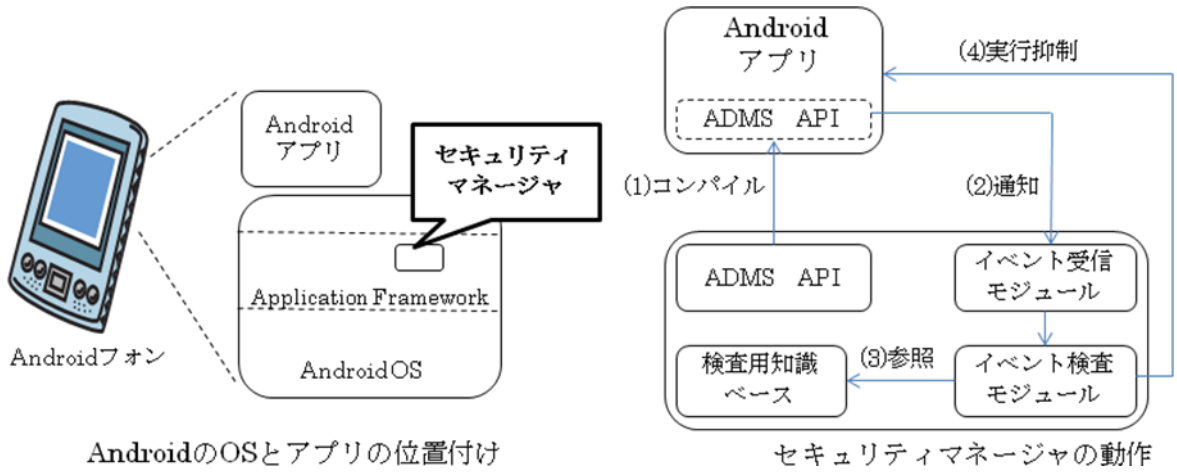

図 3.セキュリティマネージャ

セキュリティマネージャは，Android OS の Application Framework 内に実装されており，各アプリからのイベントを受信するモジュール（イベント受信モジュール），イベントを検査するモジュール（イベント検査モジュール），イベント検査のためのルールが蓄積されているデータベース（検査用知識ベース），および，イベント通知機能を有する API（ADMS API）から構成される．

ADMS API は Android フォンの中で(1)アプリがコンパイルされた際にアプリの中にライブラリとして組み込まれる．アプリが ADMS API 群に含まれるいずれかの API を call すると，そのイベントが(2)セキュリティマネージャに通知される．セキュリティマネージャは，この通知をイベント受信モジュールによって受信する．受信されたイベントはイベント検査モジュールに渡され，(3)イベント検査モジュールが検査用知識ベースを参照しながら，そのイベントが不正であるか否か（不正である可能性があるか否か）の判定を行う．万一，不正であると判定されたイベントが発見された場合は，セキュリティマネージャは，アプリに指示を発行し，(4)当該イベントの実行を抑制する．また，必要に応じては，当該アプリを Android フォンから削除したり，Android マーケットの管理者へその旨の通報を行う．

5 まとめ

本稿では，安全な Android アプリを提供するためのアプリ開発・管理方式 ADMS を提案した．ADMS ではアプリ開発者とマーケット運用者が規定された手順を実施し，定められた役割を果たすことによって，ユーザに安全なアプリを提供する．ADMS は安全なアプリの提供のための手順・運用・技術を統合的に管理する「仕組み」であり，ISMS と同様の思想に基づく management system の一つである．

本稿はまだ ADMS のコンセプト提案の段階であるため，今後，早急にセキュリティマネージャの実装を行い，ADMS の有効性を実証していく予定である．また，アプリの不正な挙動を検出する手法を確立し，その検知精度に対しても評価を行っていく．

参考文献

[1] CA Security Advisor Research Blog : http://community.ca.com/blogs/securityadvisor/archive/2011/07/07/dynamic-analysis-of-golddream-a-trojan.aspx

[2] Lookout Blog : “Security Alert: DroidDream Malware Found in Official Android Market” http://blog.mylookout.com/2011/03/security-al

ert-malware-found-in-official-android-market-droiddream/
[3]au one Market:
http://www.au.kddi.com/seihin/ichiran/smartphone/app/index.html
[4] App Store Review Guidelines - App Store Resource Center:
http://developer.apple.com/jp/appstore/guidelines.html
[5] William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, Anmol N. Sheth : "TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones", Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI'10) , Canada, 2010
[6] 竹森敬祐 川端秀明 磯原隆将 窪田歩 池野潤一 : "Android フォンのアプリ管理 ～ホワイトリスト方式～", 情報処理学会研究報告, 2011-CSEC-53 , 2011.5
[7] 川端秀明 磯原隆将 竹森敬祐 窪田歩 : "Android パーミッションを悪用するScriptの脅威と静的解析",情報処理学会研究報告, 2011-CSEC-53 , 2011.5
[8]Android:
http://developer.android.com/guide/basics/what-is-android.html
[9] Techbooster:
http://techbooster.jpn.org/andriod/device/1528/