SCIS 2015 The 32nd Symposium on
Cryptography and Information Security
Kokura, Japan, Jan. 20 - 23, 2015
The Institute of Electronics,
Information and Communication Engineers

スマートフォン向け安心・安全ソフト流通フレームワークの提案
Proposal of Secure and Safe Distribution Framework for Smartphone

竹中 萌 * 山内 利宏 † 才所 敏明 ‡
Hajime Takenaka Toshihiro Yamauchi Toshiaki Saisho

あらまし 利用者は，様々な経路からスマートフォン向けアプリケーションを入手できる．しかし，流通するアプリケーションには様々なリスクが存在し，正規の経路であっても安全性が十分とはいえない．また，現在の流通方式では，利用者はアプリに存在するリスクを利用者の責任で判断する必要がある．しかし，利用者がリスクを判断するだけの知識を持っているとは限らない．また，十分な知識のない利用者は，リスクの存在するアプリケーションを利用してしまう可能性がある．このため，利用者の知識に依存せず，安心かつ安全にアプリケーションを利用できる環境が必要である．本稿では，信頼できる第三者機関の検証によってアプリケーションの安全性を保障する流通フレームワークを提案する．提案するフレームワークでは，第三者機関において，アプリケーションの不正な動作と脆弱性の解析や利用者情報の利用に関する検証を行い，検証結果を蓄積する仕組みを持つ．利用者は，アプリケーションを入手する際，第三者機関で蓄積された検証結果を確認することで，アプリケーションの安全性を確認できる．本稿では，Android OS を対象とした提案フレームワークの設計について述べる．

キーワード スマートフォン，Android アプリケーション，流通フレームワーク，第三者検証

1 はじめに

スマートフォンを利用する利用者は，様々な経路からスマートフォン向けアプリケーション (以降，アプリと呼ぶ) を入手できる．例えば，Google 社が提供する Android では，Google 社が運営する Google Play や携帯端末事業者が運営するマーケットといった公式のマーケットに加え，その他の事業者や個人が運営するマーケットのような非公式マーケットからもアプリを入手できる．しかし，不正アプリや利用者情報の取り扱いが十分でないアプリが様々なマーケットで配布されており，公式のマーケットであっても安全性が十分とはいえない．

トレンドマイクロの調査 [1] では，2014 年 3 月時点で，不正アプリであると判定されたアプリが 200 万を超えたと報告されている．多くの不正アプリは，非公式マーケットで配布されている [2]．加えて，公式のマーケットに導入されているセキュリティシステムを回避することにより，公式のマーケットで配布されている不正アプリも現れている [3]．不正アプリでなくとも，攻撃者に意図せず利用されてしまう脆弱性の存在するアプリも配布されている．2013 年 10 月版の「Android アプリ脆弱性調査レポート」[4] では，6,170 個のアプリのうち，脆弱性が存在するアプリは 96%存在すると報告されている．このように，公式のマーケットで配布されているアプリも含め，現在配布されているアプリは，不正アプリである可能性や脆弱性が存在する可能性がある．

また，近年は，利用者情報の取り扱いが問題となっている．現在配布されているアプリには，アプリのサービスのために利用者情報を利用するアプリが存在する．利用者情報の利用例としては，利用者の現在地を取得することで，地図サービスを提供することが挙げられる．また，利用者情報を収集するモジュール (以降，情報収集モジュールと呼ぶ) を組み込み，広告や情報分析に利用する場合もある．しかし，利用者情報の取り扱いに関する情報の提示が十分でないアプリが多い [5]．情報の提示が十分でない場合，利用者は，利用者情報が適切に取り扱われているか否か判断が難しい．また，情報収集モジュールが組み込まれたアプリは，第三者に利用者情報を提供するため，プライバシの観点で問題となる．他にも，利用者情報を不正に取得する不正アプリも存在する [6]．情報の提示が十分でない場合，正規のアプリと不正アプリ

* 岡山大学工学部, 〒 700-8530 岡山県岡山市北区津島中 3-1-1, Faculty of Engineering, Okayama University, 3-1-1, Tsushima-naka, Kita-ku, Okayama, 700-8530, Japan
† 岡山大学大学院自然科学研究科, 〒 700-8530 岡山県岡山市北区津島中 3-1-1, Graduate School of Natural Science and Technology, Okayama University, 3-1-1, Tsushima-naka, Kita-ku, Okayama, 700-8530, Japan, yamauchi@cs.okayama-u.ac.jp
‡ (株) IT 企画, 〒 158-0083 東京都世田谷区奥沢 6-18-10, Advanced IT Corporation, 6-18-10, Okusawa, Setagaya-ku, Tokyo 158-0083 Japan, toshiaki.saisho@advanced-it.co.jp

の判断も難しい．

このような状況の中，現在の流通経路では，利用者はアプリに存在するリスクを利用者の責任で判断する必要がある．しかし，一般の利用者がリスクを判断できる知識を持っているとは限らない．また，利用者情報の取り扱いに関する情報を含め，リスクを判断するための情報が利用者に十分に提示されているとはいえない．このため，一般の利用者がアプリに存在するリスクを正しく理解し，判断することは困難である．

この問題を解決するためには，利用者の知識に依存せず，安心かつ安全にアプリを利用できる環境が必要である．そこで，本稿では，信頼できる第三者機関の検証によって，アプリの安全性を保障する流通フレームワークを提案する．提案するフレームワークは，6 つの関係者から構成されており，第三者機関による客観的な検証と検証結果の蓄積・提示を行う仕組みを持つ．利用者は，第三者機関にアプリの検証結果を要求し，提示された検証結果を確認することで，知識に依存せず，アプリの安全性を確認できる．また，提案するフレームワークは，セキュリティ分野の専門技術者によるアプリの安全性を検証する技術とセキュリティには必ずしも精通していない利用者によるアプリの安全性を確認する技術が分離しており，それぞれの技術の独自の発展を可能としている．

本稿では，オープンソースソフトウェアであり，アプリを入手するための様々な流通経路が存在する Android OS を搭載したスマートフォンを研究対象としたフレームワークの設計について述べる．

2　Android アプリが入手可能な経路の安全性

2.1　Google Play のセキュリティシステム

2.1.1　Bouncer

Google 社が運営している公式の Android 用アプリマーケットである Google Play へアプリを登録する場合，アプリの提供者は Google アカウントの取得と登録料の支払いのみを行えばよい．また，Google Play は，アプリへの事前審査が存在しない．このため，Android 向けアプリを配布することが容易になる反面，マルウェアを組み込んだ不正アプリを配布することも容易になる．2011 年 9 月の時点で，これまで発見されたマルウェアのうちの 29%が Android Market（現在の Google Play）で公開されていた [2].

これに対し，Google 社は，Google Play にマルウェアスキャン機能「Bouncer」を導入した [7]．Bouncer は，アプリが新規に登録されると，既知のマルウェアが組み込まれていないか分析する．アプリの登録後も，Google Play 上の全てのアプリを Google 社のクラウドインフラ上で動作させ，Android 端末上での動作をシミュレーションし，登録時に検出されていない不正動作を検出する．さらに，新規開発者アカウントを分析し，悪質な開発者の再登録を防ぐ．Google 社は，Bouncer を導入することにより，2011 年の上半期から下半期にかけて不正アプリのダウンロード数を 40%減少させた [7].

しかし，Bouncer では検出できないアプリが配布された．例として，Bouncer を回避する「時間差攻撃」を利用したアプリが配布されたことが挙げられる [3].「時間差攻撃」とは，攻撃者が無害なアプリをマーケットに登録し，利用者がそのアプリをインストールした後，アプリの更新機能によって利用者に不正アプリをインストールさせる手法である．このため，Bouncer が導入されたとしても，利用者は，不正アプリであるか否かを判断する必要がある．

2.1.2　アプリの権限の確認

アプリの権限の確認 [8] は，インストールするアプリが利用者情報と端末上の機能にアクセスする場合，利用者からアクセス許可を得る機能である．アプリは，利用者からアクセス許可を得られた場合のみ，利用者情報と端末上の機能にアクセスできる．また，利用者は，インストールしようとしているアプリに対して利用者情報と端末上の機能へのアクセスを全て許可した場合のみ，アプリをインストールできる．広告機能や情報収集モジュールが組み込まれているアプリも，広告機能や情報収集モジュールが利用者情報と端末上の機能を利用する場合，インストールする際に許可を得る必要がある．

しかし，アプリの権限の確認では，アプリがアクセスする利用者情報と端末上の機能のみが示されており，利用方法や利用目的は示されていない．利用方法や利用目的に関しては，それらを記載したプライバシポリシが提示されている．利用者は，アプリがアクセスする利用者情報と端末上の機能の利用方法や利用目的をプライバシポリシから確認する．しかし，プライバシポリシにアプリのサービスに関する利用者情報と端末上の機能の利用方法や利用目的が示されておらず，アプリを開発した企業の利用者情報の取り扱いに関する方針が記載されている場合がある [9]．このため，利用者は，利用者情報と端末上の機能がアプリのサービスに適切に利用されているか否かの判断が難しい．また，利用者は，利用者情報と端末上の機能が，広告機能や情報収集モジュールへ利用されるのかアプリのサービスへ利用されるか否かの判断も難しい．

2.2　非公式マーケットの問題

非公式マーケットは，Google 社と KDDI 社のような携帯端末事業者が運営する Android 用アプリマーケット以外のマーケットを指す．利用者は，Android の設定で

提供元が不明であるアプリのインストールを許可することにより，非公式マーケットからアプリをインストールできる．非公式マーケットには，公式のマーケットに登録されているアプリに限らず，登録されていないアプリも配布されている．さらに，非公式マーケットには，不正アプリを検知するセキュリティシステムや配布されるアプリに対する事前検証が存在しないものがある [2]．このため，多くのマルウェアや正規アプリを不正コピーしたアプリが非公式マーケットで配布されており [2]，利用者が安全にアプリを利用できる環境であるとはいえない．

2.3 Android アプリの流通経路の問題点

これまでに述べたことから，現在の Android アプリの流通経路における問題点は以下の通りである．

(問題 1) 公式のマーケットであっても，不正アプリか否かを利用者が判断

(問題 2) 利用者情報の取り扱いも含め，利用者への情報の提示が不十分

(問題 3) 事前検証が未導入のマーケットが存在

また，利用者の安全性を重視するだけでなく，アプリ開発者の負担の軽減やアプリ開発の支援も必要であると考える．

本稿では，上記の問題点を解決し，様々な利用者が安心かつ安全に Android アプリを利用できる流通フレームワークを提案する．

3 提案する流通フレームワーク

3.1 フレームワークの構成

図 1 に，我々が提案する流通フレームワークを示す．提案するフレームワークは，ソフト開発事業者，ソフト検証事業者，検証事業者認定組織，ソフト情報提供組織，ソフト配布事業者，および利用者の 6 つの関係者から構成されている．

(a)ソフト開発事業者
アプリを開発する企業や個人を指す．

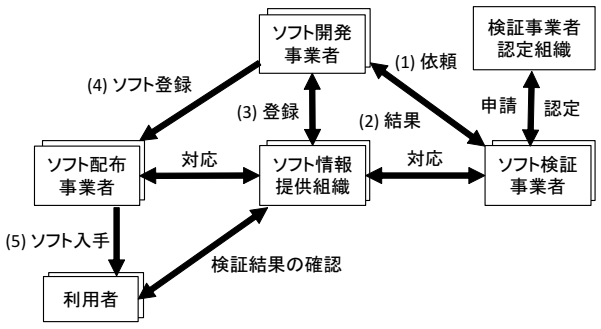

図 1 提案する流通フレームワークの全体像

(b)ソフト検証事業者
ソフト開発事業者が開発したアプリに対して，アプリの安全性を検証する．

(c)検証事業者認定組織
提案するフレームワークのソフト検証事業者として役割を担うことのできる事業者を認定する組織である．検証事業者認定組織は，認定しているソフト検証事業者をソフト開発事業者に向けて公表する．

(d)ソフト情報提供組織
ソフト検証事業者で検証された検証結果を蓄積・提供する．

(e)ソフト配布事業者
ソフト開発事業者が開発したアプリを利用者に配布する．

(f)利用者
アプリを利用するユーザを指す．

3.2 フレームワークにおけるアプリ流通の流れ

提案するフレームワークにおいて，アプリが利用者に配布されるまでの流れを以下で述べる．

(1) 検証の依頼
ソフト開発事業者は，ソフト検証事業者へ開発したアプリの検証を依頼する．検証を依頼する際，アプリの挙動に関する情報や利用者情報の利用に関する情報をソフト検証事業者へ提供する．

(2) 結果の提供
ソフト検証事業者で，アプリを検証後，ソフト開発事業者へ検証結果を提供する．ソフト開発事業者は，提供された検証結果をもとに配布の判断やアプリの修正を行う．

(3) ソフト情報提供組織へ情報の登録
ソフト開発事業者は，アプリを配布する場合，開発者に関する情報と検証結果をソフト情報提供組織に登録する．

(4) ソフトの登録
ソフト開発事業者は，ソフト配布事業者に検証済みのアプリを登録する．

(5) ソフトの入手
利用者はソフト配布事業者からアプリを入手する．

ただし，(1) で依頼できるソフト検証事業者は，検証事業者認定組織が認定している事業者のみである．認定されているソフト検証事業者は検証事業者認定組織により公開されており，ソフト開発事業者はその中から，検証を依頼する事業者を選ぶ．

また，アプリをアップデートする際も，同様の手順である．

3.3 第三者機関での技術的検証

3.3.1 安全性検証の観点

ソフト検証事業者では，利用者に配布されるアプリの安全性を技術的に検証する．安全性の観点として以下の

3 つがある．

(a) 不正な動作の有無

既存の技術と新規の技術を組み合わせることにより，不正な動作を検出し，利用者に不正なアプリが配布されることを回避する．検出する不正な動作の例としては，root 権限の奪取やマルウェアをインストールする URL への誘導が挙げられる．また，Android 端末を狙ったマルウェアには，プライバシ情報を搾取するマルウェアも存在する．このようなマルウェアは，後述する利用者情報の取り扱いに関する検証と併用して，検出する．

(b) 攻撃者に悪用される脆弱性の有無

多くの脆弱性の存在原因は，アプリの開発者が Android のセキュリティを意識せずに開発していることである．ソフト検証事業者でアプリの脆弱性を検出し，検出された場合，アプリの開発者に結果と対策を提供する．アプリの開発者は，受け取った結果と対策を参考にアプリを開発する．これにより，開発されたアプリが攻撃者に悪用されることを防ぐ．

(c) 提供された利用者情報の取り扱い情報と実際の動作との整合性

アプリが利用者情報を適切に取り扱うか検証する．ソフト開発事業者がアプリの検証を依頼する際，ソフト検証事業者に利用者情報の取り扱いに関する情報を提供する．具体的な例としては，スマートフォン プライバシ イニシアティブ（SPI）[6] の 8 項目に準拠したアプリのプライバシポリシ，定義しているパーミッション，および利用している情報収集モジュールや広告機能の情報が挙げられる．検証対象アプリを静的解析・動的解析し，実際の情報取得や情報送信の動作を確認する．その後，提供された情報と実際の動作の整合性を確認する．

提案するフレームワークにおいて，配布されるアプリは第三者機関の検証により，客観的に安全性を判断される．これにより，(問題 1）を解決できる．

また，ソフト配布事業者としてフレームワークに適用された場合，第三者機関でアプリが事前検証される．検証の存在しないマーケットであっても，ソフト配布事業者としてフレームワークに適用されることにより，(問題 3）を解決できる．

3.3.2 情報の蓄積・管理

ソフト開発事業者は，検証済みアプリの情報，検証結果，および開発者情報をソフト情報提供組織へ登録する．ソフト情報提供組織へ登録された情報は，利用者やソフト検証事業者へ提供される．利用者は，情報を提供されることにより，アプリの安全性を確認できる．また，ソフト検証事業者は，技術的検証へ情報を利用する．

また，ソフト情報提供組織は，ソフト配布事業者と連携することにより，ソフト配布事業者のマーケットで公開されている検証済みアプリの数を増やす．

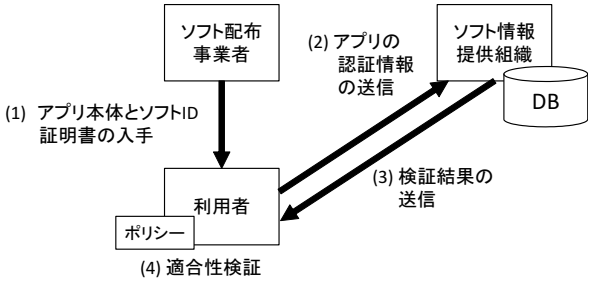

図 2 利用者環境でのアプリ利用までの流れ

3.4 利用者環境での安全性確認検証

3.4.1 アプリを利用するまでの流れ

利用者環境における，アプリの利用までの流れを図 2 に示し，以下で述べる．

(1) アプリ本体と認証情報の入手

ソフト配布事業者に登録されている検証済みアプリには，ソフト ID 証明書が付与されている．ソフト ID 証明書は，ダウンロードしたアプリの認証情報が含まれており，検証結果を取得する際に必要である．利用者は，ソフト配布事業者からアプリを入手した際，ソフト ID 証明書も入手する．

(2) 認証情報の送信

利用者端末は，入手したアプリのソフト ID 証明書をもとにアプリの認証情報をソフト情報提供組織に送信する．

(3) 検証結果の送信

ソフト情報提供組織では，アプリの認証情報を受信し，DB に蓄積されている当該アプリの検証結果を利用者端末に送信する．

(4) 検証結果の提示

利用者側では，アプリの検証内容に対する安全性ポリシー（以降，ポリシーと呼ぶ）が定義されている．ポリシーの内容としては，広告機能が組み込まれていない，利用者情報を第三者に提供しない等が挙げられる．利用者端末は検証結果を受信後，ポリシーと検証結果の適合性を検証する．ポリシーと検証結果が適合する場合，アプリをインストールする．そうでない場合，適合しない項目を利用者へ提示する．利用者は，提示された適合しない項目を確認し，利用可否を判断する．

アプリをアップデートする際も，同様の手順である．

アプリを入手した際，ソフト情報提供組織よりアプリの検証情報が入手できる．検証情報には，利用者情報の取り扱いに関する情報も含まれている．これにより，(問題 2）が解決できる．ただし，提案するフレームワークにおいては，利用者の負担を軽減するため，利用者端末が検証情報を確認し，利用可否を判断している．

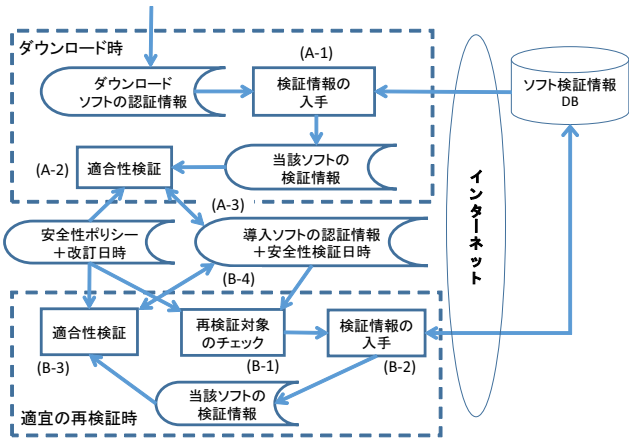

図 3　利用者端末内での安全性確認検証の流れ

3.4.2　利用者端末内の安全性確認検証

3.4.1 節の (4) の安全性確認検証における利用者端末内での処理の流れとして，図 3 を検討している．アプリをダウンロードした際の処理の流れは以下の通りである．

(A-1) 検証情報の入手
アプリをダウンロードした際，アプリの認証情報をもとに，ソフト検証情報 DB より検証情報を入手する．

(A-2) 適合性検証
ポリシーと入手した検証情報が適合するか確認する．適合していることが確認された場合，利用者にその旨を提示し，インストールを開始する．適合していない項目がある場合，利用者にその項目を提示する．利用者は，提示された適合していない項目をもとに，利用可否を判断する．

(A-3) 認証情報と検証日時の保存
アプリがインストールされた場合，インストールしたアプリの認証情報と安全性を検証した日時を端末内に保存する．

また，ポリシーが変更されると，インストールされているアプリの安全性を再確認する．安全性の再確認は，アプリの起動時に行うことを検討している．安全性の再確認の流れは以下の通りである．

(B-1) 再検証アプリの確認
ポリシーの改定日時と起動するアプリの安全性検証日時を確認する．ポリシーの改定日時以前にアプリの安全性検証が行われていた場合は，再検証を行う．そうでない場合は，アプリの起動処理に移る．

(B-2) 検証情報の入手
アプリの認証情報をもとに，再検証するアプリの検証情報を，ソフト検証情報 DB より入手する．

(B-3) 適合性検証
検証情報をもとに，ポリシーとの適合性を確認する．検証情報がポリシーと適合している場合，利用者にその旨を提示する．そうでない場合は，利用者に適合していない項目を提示する．利用者は，提示された適合しない項目をもとに，利用可否を判断する．

(B-4) 検証日時の更新
安全性検証日時を更新し，アプリの起動処理に移る．

3.4.3　未検証アプリへの対処

利用者は，入手したアプリが検証済みか否かも，アプリの利用可否を判断する情報として利用できる．このため，未検証アプリを入手した場合，利用者は，そのアプリを利用することができ，また，入手したアプリのインストールを中止することもできる．

また，未検証アプリを入手した場合，利用者がソフト情報提供組織へ検証を依頼することも対処として考えられる．

4　期待される効果

提案するフレームワークを運用することにより，以下に示す効果が期待される．

(a) 客観的な検証によるアプリの信頼性の向上
信頼できる機関の検証により，アプリの安全性に関する客観的な証拠が得られる．得られた客観的な証拠とともにアプリの安全性を示すことができるため，アプリの信頼性が向上する．また，信頼性の向上により，アプリ開発者にとって，開発したアプリの利用を促進することが期待できる．

(b) 利用者に対する安全性の向上
利用者にアプリが配布される前に，アプリの安全性が検証されるため，リスクを事前に回避することができる．また，利用者端末は，検証情報をもとにアプリの安全性を確認する．このため，利用者は，少ない負担で安全にアプリを利用できる．

(c) 公式のマーケット以外の発展の促進
提案するフレームワークにおけるソフト配布事業者は，公式のマーケットに限らず，非公式マーケットも含む．事前検証の存在しない非公式マーケットが提案するフレームワークのソフト配布事業者として役割を担うことにより，マーケットに登録されているアプリの安全性が保障できる．このため，非公式マーケットの利用促進につながり，様々なマーケットが発展できると考える．

(d) 専門技術者が利用する技術と利用者が利用する技術とが独自に発展可能
提案するフレームワークは，セキュリティ分野の専門技術者によるアプリの安全性を検証する技術とセキュリティには必ずしも精通していない利用者によるアプリの安全性を確認する技術が分離している．これにより，例えば，新たな検証技術を研究開発する際に，情報工学や情報セキュリティの専門家が利用することを前提とした検証技

術を開発できる．また，利用者端末には，様々な利用者に向けた可視化方法や安全性確認方法の開発が可能である．このように，提案するフレームワークは，情報工学や情報セキュリティの専門家が利用する技術とそうでない一般利用者が利用する技術で，それぞれの技術開発が可能であると考える．

5　第三者検証に関する取り組み

5.1　既存研究

竹森らの研究 [10] では，マーケットが第三者検証機関として審査役を担い，正確で分かりやすいアプリのプライバシポリシを生成・提示することで，利用者判断を仰ぐフレームワークを提案している．本フレームワークでは，アプリ開発者がマーケットにアプリを登録する際に，送信情報に関して申告し，第三者検証役であるマーケットの運営者が申告内容に関して検証する．検証においては，アプリに対して静的・動的解析を行い，申告内容と実際の情報送信に関する動作との整合性を確認する．申告内容に不備がない場合，マーケット側でアプリのプライバシポリシの概要版と詳細版を生成し，アプリを公開する．また，本フレームワークでは，利用時にアプリからプライバシポリシを参照する機能を持たせており，任意のタイミングでプライバシポリシを参照できる．

高橋らの研究 [11] では，利用者がインストール・アップデートしようとしている Android アプリケーションパッケージ (APK) のリスクを分析・可視化するシステムを検討している．本システムでは，ネットワーク上に知識ベースを設置しており，リスク評価に必要な情報を蓄積している．リスク評価では，脅威評価，脆弱性評価，プライバシ評価の 3 つの評価項目により評価する．既知 APK の場合，蓄積されている情報をもとにリスクを評価する．未知 APK の場合は，APK を解析し，解析結果を蓄積する．本システムは，情報を蓄積し，既知 APK を増やしていくことで迅速なリスク評価が可能であること，既存の技術を利用可能であり，幅広いリスク評価技術に対応することが可能である．

我々の研究では，情報送信やプライバシポリシに限らず，アプリの安全性を広く検証する要素技術を採用しうるフレームワークを提案している．アプリの安全性の評価に関しては，高橋らの研究が手法の一つとしてなりえると考える．

5.2　au Market の取り組み

KDDI 社が製作･販売している Android 端末は，KDDI 社が運営する au Market を利用できる．au Market では，アプリを開発する前に，アプリの企画内容の審査が存在する [12]．また，アプリ開発後も企画内容が実現されているか検証する品質検証や個人情報の取り扱いを検証する KDDI 検証が行われる．このため，アプリの安全性が確認された場合でも，企画内容に沿って実現できていない場合は，マーケットに登録できない．au Market に登録されているアプリは，審査や検証を通過したアプリであり，安全性が保障されたアプリと言える．利用者端末では，au Market からアプリをダウンロードした際，外部送信される利用者情報や送信目的が確認できる説明画面が表示される．利用者は，表示された説明画面から利用される利用者情報を確認できる [5]．

提案するフレームワークの運用においては，au Market のような独自の審査･検証を行うマーケットとの共存が必要となる．例えば，au Market で審査･検証されたアプリの検証情報をソフト情報提供組織に登録し，au Market からアプリをダウンロードした際，利用者情報に関するポリシーをもとに利用者端末が安全性確認検証を行うことができると考える．

5.3　セキュリティ関係事業者の取り組み

トレンドマイクロ株式会社は，モバイルアプリの安全性や快適さを評価する Mobile App Reputation(MAR) を運用している [5,13]．MAR は，モバイルアプリの不正な動作の有無，プライバシの観点での問題の有無，およびリソース消費の観点での問題の有無についてそれぞれ分析し，評価する．評価結果をデータベースとして管理し，スマートフォン向けセキュリティソフトへの利用やアプリ開発者への提供を行っている．

タオソフトウェア株式会社は，Tao RiskFinder を運用している [5,14]．Tao RiskFinder は Android アプリの脆弱性を診断するウェブサービスである．本サービスは，アプリの実行ファイルを静的解析し，解析結果を表示する．Tao RiskFinder は，脆弱性に限らず，マルウェアとして検知される可能性のある挙動も解析する．同様の取り組みとして，ソニーデジタルネットワークアプリケーションズ株式会社の Secure Coding Checker [15] や情報処理推進機構の AnCoLe [16] があり，開発したアプリに脆弱性が存在するかアプリ開発者自身で確認できる．

セキュリティ関係事業者の取り組みは，ソフト情報提供組織とソフト検証事業者の取り組みの要素の一つとなり得ると考える．例として，アプリの脆弱性を診断するウェブサービスをソフト検証事業者が提供することにより，ソフト開発事業者自身でアプリの安全性を事前に確認できると考える．また，セキュリティ関係事業者が，ソフト情報提供組織とソフト検証事業者の役割を担うことができると考える．

5.4 アプリ紹介サイトの取り組み

Android 向けアプリ紹介サイトの「アンドロイダー」[17] では，独自の基準に基づき，安全性が確認されたアプリのみを推薦している [5]．アンドロイダーでは，アプリ開発者と開発されたアプリに対して安全性を確認する．アプリ開発者に対しては，物理的，運用的に実在していることを確認する．確認できた場合，そのアプリ開発者を「公認デベロッパー」としている．また，公認デベロッパーが開発したアプリに対して，利用する利用者情報の危険度や利用目的の正当性を確認する．安全性が確認された場合，そのアプリを「公認アプリ」としてサイトに掲載している．その他，アンドロイダーでは，アプリ開発者が，公認アプリに関して，パーミッションの列挙と利用者情報の利用目的を記載する仕組みも取られている．

アンドロイダーのような独自の審査を行っている事業者が存在しており，我々の提案するフレームワークにおけるソフト検証事業者として役割を担うことが可能な事業者として期待できる．

6 おわりに

信頼できる第三者機関の検証によってアプリの安全性を保障する流通フレームワークを提案した．提案するフレームワークは，ソフト開発事業者，ソフト検証事業者，検証事業者認定組織，ソフト情報提供組織，ソフト配布事業者，および利用者の 6 つの関係者で構成されている．第三者機関が，アプリケーションに存在する不正な動作と脆弱性の解析，利用者情報に関する検証を行うことで，利用者に配布される前にアプリに存在するリスクを回避し，安全性を保証する．利用者環境では，利用者端末が安全性ポリシーと検証情報との適合性を判断することにより，利用者はアプリの安全性を確認できる．

提案するフレームワークは，客観的な検証を行うことで，アプリの信頼性の向上や利用者の安全性の向上につながるフレームワークである．また，様々なマーケットの発展につながるフレームワークでもある．さらに，専門技術者によるアプリの安全性を検証する技術と一般利用者によるアプリの安全性を確認する技術とが分離しているため，それぞれの技術を独自に開発することが可能である．

今後の課題として，提案するフレームワークのプロトタイプを開発し，実証実験を行うことがある．また，提案するフレームワークの運用に向けた課題の検討がある．

参考文献

[1] トレンドマイクロ，“モバイル端末を狙った不正アプリおよび高リスクアプリが 200 万を突破，”トレンドマイクロセキュリティブログ，http://blog.trendmicro.co.jp/archives/8808.

[2] 日経エレクトロニクス/日経コミュニケーション（編），“Android セキュリティ・バイブル : マルウェアの動向，”pp.66-77，日経 BP 社（2012）.

[3] ITmedia Mobile，“不正アプリは「Google Play」の中にアリ トレンドマイクロがサイバー犯罪の最新手口を解説，”http://www.itmedia.co.jp/mobile/articles/1312/03/news127.html.

[4] ソニーデジタルネットワークアプリケーションズ株式会社，“Android アプリ脆弱性調査レポート 2013 年 10 月版，”http://www.sonydna.com/sdna/topics/2013/pressrelease_20131030.html.

[5] 総務省，“スマートフォンプライバシイニシアティブ Ⅱ，”http://www.soumu.go.jp/main_content/000236366.pdf.

[6] 総務省，“スマートフォンプライバシイニシアティブ，”http://www.soumu.go.jp/main_content/000171225.pdf.

[7] Google，“Android and Security，”Google Mobile Blog，http://googlemobile.blogspot.jp/2012/02/android-and-security.html.

[8] Google，“アプリの権限の確認，”https://support.google.com/googleplay/answer/6014972?p=app_permissions&rd=1.

[9] 総務省，“スマートフォンプライバシアウトルック，”http://www.soumu.go.jp/main_content/000298928.pdf.

[10] 竹森 敬祐，磯原 隆将，川端 秀明，窪田 歩，高野 智明，可児 潤也，西垣 正勝，“アプリ/コンテンツ向けプライバシーポリシーの第三者検証フレームワーク，”情報処理学会研究報告，Vol. 2013-CSEC-62，No.62，pp.1-8，(2013).

[11] 高橋 健志，高野 祐輝，中尾 康二，太田 悟史，金岡 晃，坂根 昌一，松尾 真一郎，“Android 端末のリスク判定フレームワークとそのプロトタイプ構築，”2014 年暗号と情報セキュリティシンポジウム（SCIS2014），(2014).

[12] KDDI 株式会社，“au スマートパス au スマートパスでのサービス提供について，”http://www.au.kddi.com/developer/android/teikyo.html.

[13] トレンドマイクロ，“Mobile App Reputation，”http://www.trendmicro.co.jp/jp/why-trendmicro/spn/features/mobileapp/index.html.

[14] タオソフトウェア株式会社，“Tao RiskFinder，”http://www.taosoftware.co.jp/services/riskfinder/.

[15] ソニーデジタルネットワークアプリケーションズ株式会社，“Secure Coding Checker，”http://www.sonydna.com/sdna/solution/scc.html.

[16] 独立行政法人情報処理推進機構，“AnCoLe，”http://www.ipa.go.jp/security/vuln/ancole/index.html.

[17] アンドロイダー株式会社，“アンドロイダー，”https://androider.jp/.