Vol.2013-DPS-154 No.2
Vol.2013-CSEC-60 No.2
2013/3/14

Web アプリケーションから利用者端末内情報へのコンテキストアウェアなアクセス制御手法の提案

海野雪絵† 野田敏達† 大久保隆夫† 金谷延幸†

GPS やカメラ，アドレス帳などのスマートフォンならではの情報・機能を使いこなした，より利便性の高い Web アプリケーションが実装されている．一方，スマートフォンなどの携帯端末から Web アプリケーションに渡しても良いプライバシー情報(位置情報やアドレス帳情報等)は，時間・場所・使用中の Web アプリケーションといったユーザの状態によって異なるため，Web アプリから携帯端末に格納されたプライバシー情報へのアクセスを制御する必要がある．本稿では，Web アプリサーバと携帯端末の通信を監視し，動的に生成するコンテキストに応じて Web アプリケーションにおけるプライバシー情報の利用可否を判断するコンテキストアウェアなアクセス制御手法を提案する．

A Proposal of Context-Aware Access Control Method for Privacy Data Accessed from Web Application

YUKIE UNNO† BINTATSU NODA†
TAKAO OKUBO† NOBUYUKI KANAYA†

Web applications are now very convenient with user's privacy data including functions of smartphones such as s GPS receiver, camera and contact lists. Otherwise, users may restrict the web applications to access their privacy data according to their context such as using applications, location, time, and connection network. Then, the context-aware access control mechanism based on the context is required. Using this mechanism, users can give web applications only their privacy data that they allow web applications to access according to the context. In this paper, we propose context-aware access control method that monitor the communication between servers and device, generate context from user's condition, and control the access to privacy data from web applications based on the context.

1. はじめに

スマートフォンやタブレットといった個人向け高機能携帯端末の急速な普及に伴い，携帯端末向けアプリケーションが続々と開発されている．アプリケーションの中には，携帯端末に搭載されたセンサーや携帯端末内に蓄積されたユーザのアドレス帳, 写真, 秘密情報等の様々な情報(以下，プライバシー情報)を利用してサービスを提供するものも登場しており，ユーザはこうしたアプリケーションを携帯端末から利用することで，いつでもどこでも便利なサービスを利用できるようになってきた．

しかし，携帯端末に様々なプライバシー情報を蓄積しているユーザにとって，いつ，どんな時でもアプリケーションに対して携帯端末内のプライバシー情報を渡しても良いとは限らない．例えば，BYOD において，ユーザが業務用アプリケーションで機密情報を扱っている時に，私用アプリケーションに機密情報が渡ってしまうことを制限する必要がある．このように，アプリケーションに渡しても良い情報は使用中の Web アプリケーション・場所・時間などのユーザの状態(以下，コンテキスト)によって異なることがあると考えられる．

そこで本稿では，Web アプリケーションからプライバシー情報へのコンテキストアウェアなアクセス制御手法を提案する．本手法では，使用中の Web アプリケーションの情報をコンテキストとして用いて，コンテキストアウェアに Web アプリケーションがアクセス可能な API を制御することで，コンテキストに応じて Web アプリケーションが取得できる携帯端末内のプライバシー情報を制御できる．このような制御を行うことで，ユーザは必要な時にのみ Web アプリケーションにプライバシー情報を渡してサービスを利用することができ，不要な時は Web アプリケーションへのプライバシー情報をブロックすることでプライバシー情報の流出を防ぐことができる．さらに，あらゆる端末にセキュリティ機能を提供する個人用のセキュリティアプライアンス PSER(Pocketable Security Enforcement Router) [1]に本制御手法を組み込むことで，携帯端末からの PSER が搭載するセキュリティ機能や保持するプライバシー情報の利用をコンテキストアウェアに制御できるようにする手法を提案する．

2. 背景と課題

2.1 HTML5

現在，W3C が 2014 年中の勧告を目指し HTML5[2]の策定を進めている[3]．HTML5 およびその周辺 API では，WebSocket や WebIntent 等の新たな要素が加わり, 双方向通信やアプリケーション連携等が可能となる．また，Device

† (株)富士通研究所
FUJITSU LABORATORIES LTD, 4-1-1, Kamikodanaka, Nakahara-ku, Kawasaki 211-8588, Japan

APIs Working Group などでは，Web アプリケーションからカメラやアドレス帳などの端末固有の機能を使用するための API の議論が進められている[4].

今後, 携帯端末に搭載されている Web ブラウザが HTML5 に対応することで, Web アプリケーションは Web ブラウザを経由して端末固有の機能へアクセスすることが可能になる．そのため，Web アプリケーションについても携帯端末向けネイティブアプリケーションと同様に，携帯端末内のプライバシー情報の扱いについて慎重に考慮していく必要があると考えられる．

2.2 Android のパーミッション

Android[5]には携帯端末固有の機能や端末内の情報を利用するための豊富な API が用意されている．Android アプリケーションがこれらの API へのアクセス権を得るためには，パーミッションを要求して承認される必要がある．パーミッションの例を表 1 に示す[6]．パーミッションは各 Android アプリケーションの AndroidManifest.xml に記載されており，アプリケーションのインストール時にユーザに提示される．ユーザが要求された API のパーミッションを承認すると，アプリケーションは API を利用できるようになる．アプリケーションのインストール時に承認されたパーミッションを後から変更することはできない.

表 1　Android のパーミッション例
Table 1　Example of Android permission

パーミッション名	説明
INTERNET	完全なインターネットへのアクセス
ACCESS_FINE_LOCATION	GPS の使用を許可
CAMERA	カメラ機能へのアクセスの許可
READ_CONTACTS	連絡先データの読み取りの許可
READ_CALENDAR	カレンダーデータの読み取りの許可

2.3 課題

いつでもどこでもユーザが持ち歩く携帯端末にはプライバシーのリスクがある．その理由の一つは，携帯端末にはユーザの位置情報やアドレス帳，画像，アプリケーションの利用履歴等の様々なプライバシー情報が蓄積されているためである．そして，もう一つの理由は，カメラやマイク，センサー等の携帯端末が搭載する機能を利用して，ユーザのプライバシー情報を取得することができるためである．近年は，私物のスマートフォンを業務で利用する業務形態(BYOD)も登場しており, ユーザ個人の携帯端末にはプライベートの情報だけでなく，仕事上の機密情報も格納されている．そのため，私用での携帯端末利用時に機密情報が流出することや，仕事での携帯端末利用時にプライベートの情報が流出するといったリスクも発生する．また，HTML5 の普及により，Web アプリケーションは従来は不可能であった，ネイティブアプリケーションのように携帯端末内のプライバシー情報を利用することや，アプリケーション連携等が可能になるため，Web アプリケーションから携帯端末内の情報が様々な形でサーバに送信されることが考えられる(図 1). したがって，Web アプリケーションにおけるプライバシー情報の利用を適切に制御する仕組みが必要である.

ここで以下のようなシーンを考える．田中さんは，仕事で携帯端末を使用している．田中さんは，顧客 A 社との資料共有サイト(以下，資料共有サイト A)へ携帯端末内の A 社向け資料をアップロードしようと考えていた．その時，顧客 B 社から B 社との資料共有サイト(以下，資料共有サイト B)から資料をダウンロードして内容を確認してほしいという急な連絡があった．田中さんは資料共有サイト B にアクセスし，資料をダウンロードした．ダウンロードした資料を確認した後，A 社向け資料をアップロードすることを思い出し, 資料をアップロードしようとした. この時，資料共有サイト B からダウンロードしたデータを誤って資料共有サイト A へアップロードしてしまうことを防ぐ必要がある.

また，次のようなシーンも考えられる．HTML5 を使うと携帯端末のカメラやマイクを使ってビデオチャットが可能になる．田中さんは携帯端末を仕事とプライベートで使用しており，プライベートでビデオチャットをすることがある．この場合，田中さんが仕事用 Web サイトを使用して仕事をしている時に，機密資料がビデオチャットのカメラへ映り込んで流出することを防ぐ必要がある.

ハイブリッドアプリケーションの場合はインストール時にパーミッションを指定することで，Web ブラウザから使用する Web アプリケーションの場合は, プライバシー情報にアクセスする際にその旨を Web ブラウザがユーザに確認することで，不要なプライバシー情報へのアクセスを防ぐことができる．しかし，これらの方法では，上記シーンにあるように, ある Web アプリケーションを使用中はプライバシー情報へのアクセスを制限するというようなユーザのコンテキストに応じた制御を行うことはできなかった.

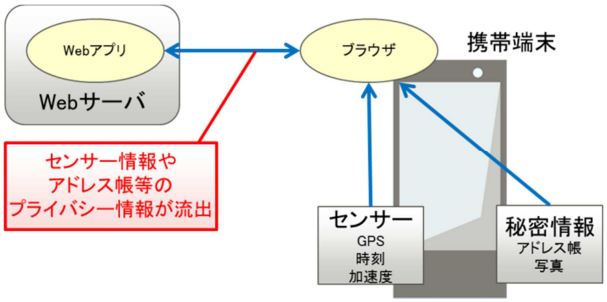

図 1　携帯端末内のプライバシー情報の流出
Figure 1　Flows of privacy data in mobile device

3. 提案手法

3.1 本提案のコンセプト

我々は携帯端末におけるプライバシーのリスクを解決するため，Web アプリケーションが取得できる携帯端末内の情報をユーザのコンテキストに応じて制御するコンテキストアウェアなアクセス制御手法を提案する．本手法は以下を特徴とする．

- コンテキストとして，使用中の Web アプリケーション情報を用いる
- コンテキストに応じて Web アプリケーションがアクセスできるプライバシー情報を制限する

使用中の Web アプリケーションを含むコンテキストに応じた制御を行うことで，ユーザがある Web アプリケーションを使用している状況において，端末内の情報が他のアプリケーションに不要に流出することを防ぐ．

例えば，2.3 節で挙げた一つ目のシーンの場合，ユーザが資料共有サイト B を使用している時は，資料共有サイト A に対してプライバシー情報へのアクセスを制限する．資料共有サイト B を使用している時は，携帯端末内に B 社関連資料が存在すると考えられる．そこで本提案手法を用いて，資料共有サイト B を使用している時は，資料共有サイト A へのデータのアップロードを制限する．こうした制御を行うことで，A 社に対して B 社の情報が流出することを防ぐことができる．二つ目のシーンの場合は，ユーザが仕事用 Web サイトを使用している時は，ビデオチャットに対してカメラ機能へのアクセスを制限する．ユーザが仕事用 Web サイトを使用している時は仕事中と考えられる．そこで本提案手法を用いて，仕事用 Web サイトを使用している時は，友人とのビデオチャットでのカメラの利用を制限する．この制御により，仕事の機密資料がカメラに写り込むことを防ぐことができる．

3.2 コンテキストに応じた制御

コンテキストアウェアな制御を行うためには，Web アプリケーションがプライバシー情報へアクセスする時のユーザのコンテキストが必要となる．本提案手法では，「CGRule (Context Generation Rule)」に基づいてコンテキストの生成を行う．また，「CACRule (Context based Access Control Rule)」を用いて，Web アプリケーションがプライバシー情報を利用しても良いかどうかを判断する．各ルールは以下の特徴を持つ．

- CGRule (Context Generation Rule)

  CGRule には，各コンテキストの生成に必要となるコンテキスト内容が定められている．Web アプリケーションがプライバシー情報の利用を要求する時に，本ルールをもとにコンテキストが生成される．コンテキスト内容には以下が含まれ，これらの AND/OR でコンテキストが定義される．

  - 使用中の Web アプリケーションの情報
  - センサー機能を利用して取得した，位置や時刻等の情報
  - 監視している通信の接続先ネットワークの情報

  CGRule の例を表 2 に示す．表 2 のコンテキスト A は，時間が業務時間であり，かつ携帯端末のある場所が社内，接続先のネットワークが社内 LAN，使用中の Web アプリケーションがアプリ 1 であるコンテキストである．CGRule には，表 2 のよう複数のコンテキストを記述しておくことができる．

- CACRule (Context based Access Control Rule)

  CACRule には，あるコンテキストにおいて，各 Web アプリケーションに利用を許可する API について記してある． CACRule の例を表 3 に示す．例えば，表 3 の 2 行目は，コンテキストがコンテキスト A である時，Web アプリ X に GPS とカメラの API の利用を許可することを意味している．

表 2　CGRule (Context Generation Rule)の例
Table 2　Example of CGRule

コンテキスト	コンテキスト内容
コンテキスト A	時刻=業務時間 and 場所=社内 and ネットワーク=社内 LAN and 使用中の Web アプリ=アプリ 1
コンテキスト B	時刻=業務時間 and 場所=社外 and 使用中の Web アプリ=アプリ 1
コンテキスト C	時刻=非業務時間 and 使用中の Web アプリ=(アプリ 2 and アプリ 3)
その他	その他

表 3　CACRule (Context based Access Control Rule)の例

Table 3　Example of CACRule

コンテキスト	Web アプリケーション	利用許可 API
コンテキスト A	Web アプリ X	GPS, カメラ
コンテキスト A	Web アプリ Y	GPS, アドレス帳
コンテキスト B	Web アプリ X	カメラ

3.3 実現方法

我々は，本提案手法の実現方法として図 2 に示す制御モジュール CCM(Context based Control Module)を提案する．なお，図 2 では CCM が携帯端末内に組み込まれている状態を示しているが, CCM はユーザの近傍にあるならば携帯端末から分離されていても良い．分離されているケースについては 4 章で説明する．CCM の動作を以下に示す．

1. 携帯端末とサーバの通信を監視して，使用を開始した Web アプリケーションの情報と接続先ネットワークの情報を取得する．
2. Web アプリケーションが携帯端末内の API を呼び出す際, CGRule に基づいてコンテキストを生成する．
3. コンテキスト生成後，CACRule に基づき呼び出し元アプリケーションに対する API の利用可否を判断する．利用可の場合は API の処理を行い呼び出し元アプリケーションに結果を返し，利用不可の場合はエラーを返す．なお，API 利用制御の動作箇所としては，(a)Web ブラウザで JavaScript API 等の利用を制御する方法や，(b)Android フレームワーク等を拡張するなどして，OS の API の利用を制御する方法が挙げられる．(b)に関しては，文献[7]が Android の API 単位でフックを行い，API 利用のリアルタイムな動的制御を行う方式を提案しており，この方式を適用することで本手法を実現できると考える．

本手法において重要な点として考えなければいけないのが，使用中の Web アプリケーションに関する情報の取得方法である．本手法では，使用中の Web アプリケーションの情報を，プロセス名/ID と接続先 Web サーバの URL とすることを考えている．これらの取得方法としては，プロキシで Web アプリケーションとサーバの通信を監視する方法が挙げられる．しかし，プロキシによる通信監視では，HTTPS 通信時のリクエスト内容を把握できないという問題がある．そこで，我々は API 利用の監視による情報取得を検討している．サーバとの通信開始時に使用される API(例えば Android の java.net など)を監視し，Web ブラウザがそれらの API を使用して通信を開始する時に，そのプロセス名/ID と接続先の URL を取得する. 取得した情報は，プロセスが終了するまで使用中の Web アプリケーションの情報として CCM に保持される．

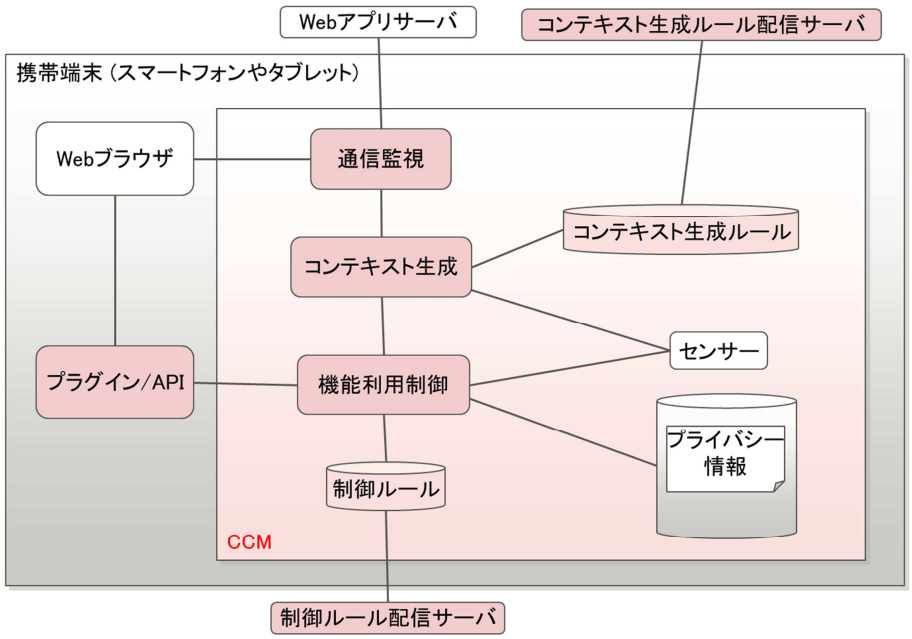

図 2　CCM の構成

Figure 2　CCM architecture

4. PSER への実装

4.1 目的

我々が研究を行っている PSER は，ユーザの携帯端末からセキュリティ機能を分離した，個人用のセキュリティアプライアンスである．PSER は，携帯端末で使用する暗号機能，署名機能，ウイルス検知機能等を携帯端末に代わって実行する．ユーザは PS ER を常に携帯することで，どんな OS のどんな携帯端末からでも PSER のセキュリティ機能を利用できる．また，様々な端末に散らばるプライバシー情報を PSER 上で一括して管理することで，プライバシー情報をより安全に利用・保持できる．

本提案手法の CCM を携帯端末から分離し，この PSER に搭載することで， PSER 内のセキュリティ機能やプライバシー情報へのアクセスをコンテキストアウェアに制御することが可能となる．

具体的には，まず PSER が搭載するセキュリティ機能を本 CCM の制御対象 API とする．本利用シーンを PSER の暗号機能を例に説明する．PSER の暗号機能は携帯端末からサーバへアップロードされるデータを暗号化し，サーバから携帯端末へダウンロードされるデータを復号する機能である．この暗号機能を使用しているユーザが，機密情報の不要な流出を防ぐために，「ある Web アプリケーションを使用している時は機密情報をダウンロードしても復号しない」と考えるかもしれない．このような場合，本提案手法を用いて暗号機能の利用可否を制御することで，ユーザの状況に応じて機密情報を復号するか否かを制御できるようになる．暗号機能以外のセキュリティ機能も同様にコンテキストに応じて利用の制御が可能となる．

さらに，Web アプリケーションが携帯端末のデータやカメラ等の利用を要求する場合，携帯端末内ではなく PSER 内のプライバシー情報を Web アプリケーションに返すことで，Web アプリケーションからプライバシー情報へのアクセスを制御することができる．このような仕組みにすることで，ユーザのプライバシー情報を PSER で安全に管理できると同時に，コンテキストに応じたアクセス制御が可能となり，プライバシー情報の不要な流出を防ぐことができる．

4.2 実現方法

現在，我々が考えている実現方法 2 パターンについて以下に示す．また，その構成例を図 3 に示す．

- (パターン 1)PSER 対応型携帯端末を使用

  パターン 1 は，携帯端末に専用モジュールを組み込み，Web アプリケーションからプライバシー情報へのアクセスを，PSER へのアクセスに書き変える方法である．動作の流れとしては，まずユーザが携帯端末上の Web ブラウザから Web アプリケーションにアクセスする．Web アプリケーションから携帯端末内のプライバシー情報へのアクセスが実行される時，専用モジュールがそのアクセスを PSER 内の CCM へのアクセスに書き変える．CCM はコンテキストに応じて PSER 内の適切なプライバシー情報を携帯端末に返す．コンテキストとして用いる使用中の Web アプリケーションに関する情報は，専用モジュールが CCM へ送信する．

- (パターン 2)PSER 対応型 Web アプリケーションを使用

  パターン 2 は，Web アプリケーションを PSER に対応した構成とすることで，CCM を呼び出す方法である．Web アプリケーションが PSER に対して，使用中の Web アプリケーションに関する情報を渡し，さらにプライバシー情報を利用する際は，携帯端末内ではなく，PSER 内の API を CCM 経由で呼び出す．CCM は Web ブラウザへ適切なプライバシー情報を返す．

パターン 1 の利点としては，従来の Web アプリケーションから本 CCM を利用できる点が挙げられる．パターン 2 の利点は，携帯端末に手を加える必要がないため，携帯端末を限定することなくどんな携帯端末からでも PSER の CCM を利用できる点である．

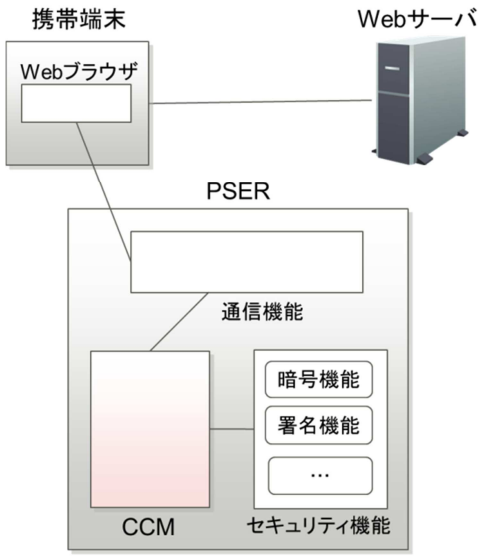

図 3　CCM を組み込んだ PSER
Figure 3　CCM in PSER

5. 関連研究

携帯端末内の情報の流出を防ぐことを目的とした研究は複数行われている．文献[8]は，端末内のプライバシー情報の流れを監視し，情報が端末の外に出たことを記録してユーザに提示するシステムを提案している．この提案では，ユーザはアプリケーションがどんな動作をしているかを知ることができ，プライバシー情報の流出を確認できるが，ユーザのコンテキストに応じて情報流出の制御を行うこと

はしていない．文献[9]では，Android 端末上のアプリケーションの通信を監視し，通信内容に端末内の情報が含まれていた場合，情報フロールールに基づき情報の送信可否を判断することで，情報の外部送信を制御している．この情報フロールールには，本提案で示したユーザのコンテキストは含まれていない．

また文献[10]は，Android のパーミッション機構を拡張し，アプリケーションの構成やコンテキスト等のセキュリティポリシーによってパーミッションの付与を動的に制御することを提案している．この提案では，Android パーミッション単位での制御を行っており，また使用中の Web アプリケーションをコンテキストとして用いることについては記されていない．我々の提案手法は，使用中の Web アプリケーションを含むコンテキストに応じて，API 単位でアクセス制御を行うため，よりきめ細やかな制御を行うことができる．

6. おわりに

我々は，Web アプリケーションが取得できる携帯端末内のプライバシー情報をユーザのコンテキスト(使用中の Web アプリケーション，接続先ネットワーク，位置，時間等)に応じて制御できるアクセス制御手法を提案した．本提案手法では，制御モジュール CCM でコンテキストを生成し，生成したコンテキストに応じたルールに基づいて Web アプリケーションにおけるプライバシー情報利用可否の制御を行う．本提案手法を用いることで，使用中の Web アプリケーションやユーザの位置などに応じて，Web アプリケーションに渡されるプライバシー情報を制限できる．さらに，本提案手法を PSER に組み込むことで様々な端末から本制御を利用でき，PSER が搭載するセキュリティ機能やプライバシー情報のコンテキストアウェアな利用制御ができることを示した．

今後の課題としては，本稿で示した実現方法に基づいて本提案手法の実装を行い，効果を検証することが挙げられる．また，携帯端末や PSER に CCM を組み込むことで，Web アプリケーション利用時にコンテキスト生成処理や，API 利用可否の判断のための処理等が新たに加わるため，これらの処理速度の評価を行って実用性を検証する必要がある．

参考文献

1) 野田敏達, 海野雪絵, 大久保隆夫, 金谷延幸: 個人用セキュリティアプライアンスの提案，第 60 回　CSEC 研究発表会, 2013
2) W3C HTML5,
http://www.w3.org/TR/html5/
3) W3C FAQ,
http://www.w3.org/html/wiki/FAQs
4) W3C Device APIs Working Group,
http://www.w3.org/2009/dap/
5) Android,
http://www.android.com/
6) タオソフトウェア株式会社, Android Security 安全なアプリケーションを作成するために, 2011
7) 川端 秀明，磯原 隆将，竹森 敬祐，窪田 歩，可児 潤也，上松 晴信，西垣 正勝: Android OS における機能や情報へのアクセス制御機構の提案, Computer Security Symposium 2011, pp.161-166, 2011
8) 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", 9th USENIX Symposium on Operating Systems Design and Implementation, 2010
9) 葛野弘樹: Android アプリケーションに対する情報フロー制御機構の提案, Computer Security Symposium 2011, pp.155-160, 2011
10) Machigar Ongtang, Stephen McLaughlin, William Enck, Patrick McDaniel : Semantically Rich Application-Centric Security in Android, 2009 Annual Computer Security Applications Conference, pp.340-349, 2009