(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6650439号
(P6650439)
(45)発行日　令和2年2月19日(2020.2.19)　(24)登録日　令和2年1月22日(2020.1.22)

(51)Int.Cl.		F I		
G06F 21/56	(2013.01)	G06F	21/56	360
G06F 9/48	(2006.01)	G06F	9/48	300J
G06F 11/07	(2006.01)	G06F	11/07	151
G06F 11/34	(2006.01)	G06F	11/34	147
G06F 11/30	(2006.01)	G06F	11/07	140E

請求項の数 15　(全 43 頁)　最終頁に続く

(21)出願番号　特願2017-513221(P2017-513221)
(86)(22)出願日　平成27年8月28日(2015.8.28)
(65)公表番号　特表2017-536594(P2017-536594A)
(43)公表日　平成29年12月7日(2017.12.7)
(86)国際出願番号　PCT/US2015/047489
(87)国際公開番号　WO2016/040015
(87)国際公開日　平成28年3月17日(2016.3.17)
審査請求日　平成30年8月13日(2018.8.13)
(31)優先権主張番号　14/483,800
(32)優先日　平成26年9月11日(2014.9.11)
(33)優先権主張国・地域又は機関
米国(US)

(73)特許権者　507364838
クアルコム，インコーポレイテッド
アメリカ合衆国　カリフォルニア　９２１２１　サン　ディエゴ　モアハウス　ドライブ　５７７５
(74)代理人　100108453
弁理士　村山　靖彦
(74)代理人　100163522
弁理士　黒田　晋平
(72)発明者　マストゥーレ・サラジェゲ
アメリカ合衆国・カリフォルニア・９２１２１－１７１４・サン・ディエゴ・モアハウス・ドライヴ・５７７５

最終頁に続く

(54)【発明の名称】モバイルデバイス挙動のアグリゲートマルチアプリケーション挙動分析のための方法およびシステム

(57)【特許請求の範囲】
【請求項１】
コンピューティングデバイスの挙動を分析する方法であって、前記方法は、前記コンピューティングデバイスのプロセッサによって実行され、
前記コンピューティングデバイス上の複数のソフトウェアアプリケーションの活動を監視して、挙動情報を収集するステップと、
前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの集合的挙動を特徴付ける複数の数値を含む挙動ベクトル情報構造を生成するステップと、
前記生成された挙動ベクトル情報構造をマルチアプリケーション分類器モデルに適用して、分析情報を生成するステップであって、前記マルチアプリケーション分類器モデルが、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動が非良性であるかどうかを決定することに最も関連があるテスト条件を各々が評価する判定ノードを含む、適用するステップと、
前記生成された分析情報を使用して、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を評価するステップと
を含むことを特徴とする、方法。
【請求項２】
前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を特徴付ける前記複数の数

値を含む前記挙動ベクトル情報構造を前記生成するステップが、前記複数の数値によって、前記複数のソフトウェアアプリケーション内で、前記ソフトウェアアプリケーションの前記集合的挙動を特徴付ける情報構造を生成するステップを含む、請求項1に記載の方法。

【請求項３】

前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を特徴付ける前記複数の数値を含む前記挙動ベクトル情報構造を前記生成するステップが、前記複数の数値によって前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの間の関係を特徴付ける情報構造を生成するステップを含む、請求項1に記載の方法。

【請求項４】

前記コンピューティングデバイス上の前記複数のソフトウェアアプリケーションの前記活動を前記監視して、前記挙動情報を収集するステップが、前記複数のソフトウェアアプリケーション間の対話を監視するステップを含み、

前記生成された分析情報を前記使用して、前記複数のソフトウェアアプリケーションの前記集合的挙動を評価するステップが、グループとして一緒に評価されるべき2つ以上のソフトウェアアプリケーションを特定するステップを含む、

請求項1に記載の方法。

【請求項５】

前記特定された2つ以上のソフトウェアアプリケーションの追加の活動を監視して、追加の挙動情報を収集するステップと、

前記収集された追加の挙動情報に基づいて、前記特定された2つ以上のソフトウェアアプリケーションの前記集合的挙動を特徴付ける集合的挙動ベクトルを生成するステップと、

前記生成された集合的挙動ベクトルを前記マルチアプリケーション分類器モデルに適用して、追加の分析情報を生成するステップと、

前記追加の分析情報を使用して、前記特定された2つ以上のソフトウェアアプリケーションの前記集合的挙動が非良性であるかどうかを決定するステップと

をさらに含む、請求項4に記載の方法。

【請求項６】

各々が前記特定された2つ以上のソフトウェアアプリケーションの前記挙動を特徴付ける挙動ベクトルを前記マルチアプリケーション分類器モデルに適用して、追加の分析情報を生成するステップと、

前記挙動ベクトルの各々に関して生成された前記追加の分析情報を集約するステップと、

前記集約された分析情報を使用して、前記特定された2つ以上のソフトウェアアプリケーションの前記集合的挙動が非良性であるかどうかを決定するステップと

をさらに含む、請求項4に記載の方法。

【請求項７】

前記生成された挙動ベクトル情報構造を前記マルチアプリケーション分類器モデルに前記適用するステップが、

前記マルチアプリケーション分類器モデル内に含まれた各テスト条件を評価するステップと、

前記マルチアプリケーション分類器モデル内のテスト条件を評価した各結果の加重平均を計算するステップと、

前記加重平均に基づいて、前記集合的挙動が非良性であるかどうかを判定するステップと

を含む、請求項1に記載の方法。

【請求項８】

前記生成された分析情報を前記使用して、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を評価するステップが、

前記監視された複数のソフトウェアアプリケーションを類別するステップと、

前記複数のソフトウェアアプリケーションの各カテゴリを特徴抽出するステップと、

前記複数のソフトウェアアプリケーションの各カテゴリに関する性能番号を生成するステップと

を含む、請求項1に記載の方法。

【請求項９】

コンピューティングデバイス上の複数のソフトウェアアプリケーションの活動を監視して、挙動情報を収集することと、

前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの集合的挙動を特徴付ける複数の数値を含む挙動ベクトル情報構造を生成することと、

前記生成された挙動ベクトル情報構造をマルチアプリケーション分類器モデルに適用して、分析情報を生成することであって、前記マルチアプリケーション分類器モデルが、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動が非良性であるかどうかを決定することに最も関連があるテスト条件を各々が評価する判定ノードを含む、適用することと、

前記生成された分析情報を使用して、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を評価することと

を含む動作を前記コンピューティングデバイス上のプロセッサに実行させるように構成されたプロセッサ実行可能ソフトウェア命令が記憶されたことを特徴とする、非一時的コンピュータ可読記憶媒体。

【請求項１０】

前記記憶されたプロセッサ実行可能ソフトウェア命令が、前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を特徴付ける前記複数の数値を含む前記挙動ベクトル情報構造を前記生成することが、前記複数の数値によって、前記複数のソフトウェアアプリケーション内で、前記ソフトウェアアプリケーションの前記集合的挙動を特徴付ける情報構造を生成することを含むような動作を前記プロセッサに実行させるように構成される、請求項９に記載の非一時的コンピュータ可読記憶媒体。

【請求項１１】

コンピューティングデバイス上の複数のソフトウェアアプリケーションの活動を監視して、挙動情報を収集するための手段と、

前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの集合的挙動を特徴付ける複数の数値を含む挙動ベクトル情報構造を生成するための手段と、

前記生成された挙動ベクトル情報構造をマルチアプリケーション分類器モデルに適用して、分析情報を生成するための手段であって、前記マルチアプリケーション分類器モデルが、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動が非良性であるかどうかを決定することに最も関連があるテスト条件を各々が評価する判定ノードを含む、適用するための手段と、

前記生成された分析情報を使用して、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を評価するための手段と

を含むことを特徴とする、コンピューティングデバイス。

【請求項１２】

前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を特徴付ける前記複数の数値を含む前記挙動ベクトル情報構造を前記生成するための手段が、前記複数の数値によっ

て、前記複数のソフトウェアアプリケーション内で、前記ソフトウェアアプリケーションの前記集合的挙動を特徴付ける情報構造を生成するための手段を含む、請求項１１に記載のコンピューティングデバイス。

【請求項１３】

前記収集された挙動情報に基づいて、前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの前記集合的挙動を特徴付ける前記複数の数値を含む前記挙動ベクトル情報構造を前記生成するための手段が、前記複数の数値によって前記複数のソフトウェアアプリケーションのうちの2つ以上のソフトウェアアプリケーションの間の関係を特徴付ける情報構造を生成するための手段を含む、請求項１１に記載のコンピューティングデバイス。

【請求項１４】

前記コンピューティングデバイス上の前記複数のソフトウェアアプリケーションの前記活動を前記監視して、前記挙動情報を収集するための手段が、前記複数のソフトウェアアプリケーション間の対話を監視するための手段を含み、

前記生成された分析情報を前記使用して、前記複数のソフトウェアアプリケーションの前記集合的挙動を評価するための手段が、グループとして一緒に評価されるべき2つ以上のソフトウェアアプリケーションを特定するための手段を含む、

請求項１１に記載のコンピューティングデバイス。

【請求項１５】

前記特定された2つ以上のソフトウェアアプリケーションの追加の活動を監視して、追加の挙動情報を収集するための手段と、

前記収集された追加の挙動情報に基づいて、前記特定された2つ以上のソフトウェアアプリケーションの前記集合的挙動を特徴付ける集合的挙動ベクトルを生成するための手段と、

前記生成された集合的挙動ベクトルを前記マルチアプリケーション分類器モデルに適用して、追加の分析情報を生成するための手段と、

前記追加の分析情報を使用して、前記特定された2つ以上のソフトウェアアプリケーションの前記集合的挙動が非良性であるかどうかを決定するための手段と

をさらに含む、請求項１４に記載のコンピューティングデバイス。

【発明の詳細な説明】

【背景技術】

【０００１】

セルラー通信技術およびワイヤレス通信技術は、過去数年の間に爆発的な成長をとげている。ワイヤレスサービスプロバイダは現在、情報、リソースおよび通信に対するかつてないレベルのアクセスをユーザに提供する、多岐にわたる機能およびサービスを提供している。これらの強化と歩調を合わせるために、消費者向け電子デバイス(たとえば、セルラーフォン、時計、ヘッドフォン、遠隔制御装置など)は、今までよりも強力かつ複雑になり、今や一般的に、強力なプロセッサ、大規模メモリ、および各自のデバイス上で複雑で強力なソフトウェアアプリケーションを実行することを可能にする他のリソースを含む。これらのデバイスによって、それらのユーザは、アプリケーションダウンロードサービス(たとえば、Apple(登録商標) App Store、Windows(登録商標) Store、Google(登録商標) playなど)またはインターネットから様々なソフトウェアアプリケーションをダウンロードし、実行することができる。

【０００２】

これらおよび他の改善により、ますます多くのモバイルデバイスおよびワイヤレスデバイスのユーザは、現在各自のデバイスを使用して、機密情報(たとえば、クレジットカード情報、連絡先など)を記憶し、および/またはセキュリティが重要であるタスクを達成する。たとえば、モバイルデバイスユーザは、商品を購入するために各自のデバイスを頻繁に使用し、機密の通信を送受信し、請求を支払い、預金口座を管理し、他の機密のトランザクションを行う。これらの傾向により、モバイルデバイスは、急速にマルウェア攻撃お

よびサイバー攻撃に対する次の開拓分野になっている。したがって、たとえばモバイルデバイスおよびワイヤレスデバイスなど、リソース制約のあるコンピューティングデバイスをより良く保護する、新しい、および改善されたセキュリティソリューションが消費者に有益である。

【発明の概要】

【課題を解決するための手段】

【０００３】

様々な実施形態は、コンピューティングデバイス上で動作する2つ以上のソフトウェアアプリケーションの集合的挙動を評価するために挙動分析技法または機械学習技法を使用する方法を含む。この方法は、コンピューティングデバイスのプロセッサ内で、複数のソフトウェアアプリケーションの活動を監視するステップと、複数のソフトウェアアプリケーションの各々の監視された活動に関する挙動情報を収集するステップと、収集された挙動情報に基づいて挙動ベクトルを生成するステップと、分析情報を生成するために、生成された挙動ベクトルを分類器モデルに適用するステップと、複数のソフトウェアアプリケーションの集合的挙動を評価するために分析情報を使用するステップとによって、コンピューティングデバイス内の挙動を分析するステップを含み得る。

【０００４】

ある実施形態では、収集された挙動情報に基づいて挙動ベクトルを生成するステップは、複数のソフトウェアアプリケーションの集合的挙動を特徴付ける情報構造を生成するステップを含み得る。さらなる実施形態では、収集された挙動情報に基づいて挙動ベクトルを生成するステップは、複数のソフトウェアアプリケーション間の関係を特徴付ける情報構造を生成するステップを含み得る。さらなる実施形態では、複数のソフトウェアアプリケーションの活動を監視するステップは、複数のソフトウェアアプリケーション間の対話を監視するステップを含んでよく、複数のソフトウェアアプリケーションの集合的挙動を評価するために分析情報を使用するステップは、グループとして一緒に評価されるべき2つ以上のソフトウェアアプリケーションを特定するステップを含んでよい。

【０００５】

さらなる実施形態では、この方法は、追加の挙動情報を収集するために、特定された2つ以上のソフトウェアアプリケーションの追加の活動を監視するステップと、収集された追加の挙動情報に基づいて、特定された2つ以上のソフトウェアアプリケーションの集合的挙動を特徴付ける集合的挙動ベクトルを生成するステップと、追加の分析情報を生成するために、生成された集合的挙動ベクトルを分類器モデルに適用するステップと、特定された2つ以上のソフトウェアアプリケーションの集合的挙動が非良性であるかどうかを決定するために追加の分析情報を使用するステップとを含み得る。さらなる実施形態では、この方法は、追加の分析情報を生成するために、各々が特定された2つ以上のソフトウェアアプリケーションの挙動を特徴付ける挙動ベクトルを分類器モデルに適用するステップと、各挙動ベクトルに関して生成された追加の分析情報をアグリゲートするステップと、特定された2つ以上のソフトウェアアプリケーションの集合的挙動が非良性であるかどうかを決定するために分析結果を使用するステップとを含み得る。

【０００６】

さらなる態様では、分析情報を生成するために、生成された挙動ベクトルを分類器モデルに適用するステップは、生成された挙動ベクトルをマルチアプリケーション分類器モデルに適用するステップを含み得る。さらなる実施形態では、収集された挙動情報に基づいて挙動ベクトルを生成するステップは、各々が複数のソフトウェアアプリケーションのうちの1つの挙動を特徴付ける複数の挙動ベクトルを生成するステップを含んでよく、生成された挙動ベクトルをマルチアプリケーション分類器モデルに適用するステップは、分析情報を生成するために、挙動ベクトルの各々をマルチアプリケーション分類器モデルに適用するステップを含んでよい。

【０００７】

さらなる実施形態では、生成された挙動ベクトルをマルチアプリケーション分類器モデ

ルに適用するステップは、マルチアプリケーション分類器モデル内に含まれた各テスト条件を評価するステップと、マルチアプリケーション分類器モデル内のテスト条件を評価した各結果の加重平均を計算するステップと、加重平均に基づいて、集合的挙動が非良性であるかどうかを決定するステップとを含み得る。さらなる実施形態では、複数のソフトウェアアプリケーションの集合的挙動を評価するために分析情報を使用するステップは、監視された複数のソフトウェアアプリケーションを類別するステップと、複数のソフトウェアアプリケーションの各カテゴリをプロファイリングするステップと、複数のソフトウェアアプリケーションの各カテゴリに関する性能番号(performance number)を生成するステップとを含み得る。

【０００８】

さらなる実施形態は、複数のソフトウェアアプリケーションの活動を監視することと、複数のソフトウェアアプリケーションの各々の監視された活動に関する挙動情報を収集することと、収集された挙動情報に基づいて挙動ベクトルを生成することと、分析情報を生成するために、生成された挙動ベクトルを分類器モデルに適用することと、複数のソフトウェアアプリケーションの集合的挙動を評価するために分析情報を使用することとを含む動作を実行するためのプロセッサ実行可能命令で構成されたプロセッサを有するコンピューティングデバイスを含み得る。

【０００９】

ある実施形態では、このプロセッサは、収集された挙動情報に基づいて挙動ベクトルを生成することが、複数のソフトウェアアプリケーションの集合的挙動を特徴付ける情報構造を生成することを含むような動作を実行するためのプロセッサ実行可能命令で構成され得る。さらなる実施形態では、このプロセッサは、収集された挙動情報に基づいて挙動ベクトルを生成することが、複数のソフトウェアアプリケーション間の関係を特徴付ける情報構造を生成することを含むような動作を実行するためのプロセッサ実行可能命令で構成され得る。さらなる実施形態では、このプロセッサは、複数のソフトウェアアプリケーションの活動を監視することが、複数のソフトウェアアプリケーション間の対話を監視することを含むような、かつ複数のソフトウェアアプリケーションの集合的挙動を評価するために分析情報を使用することが、グループとして一緒に評価されるべき2つ以上のソフトウェアアプリケーションを特定することを含むような動作を実行するためのプロセッサ実行可能命令で構成され得る。

【００１０】

さらなる実施形態では、このプロセッサは、追加の挙動情報を収集するために、特定された2つ以上のソフトウェアアプリケーションの追加の活動を監視することと、収集された追加の挙動情報に基づいて、特定された2つ以上のソフトウェアアプリケーションの集合的挙動を特徴付ける集合的挙動ベクトルを生成することと、追加の分析情報を生成するために、生成された集合的挙動ベクトルを分類器モデルに適用することと、特定された2つ以上のソフトウェアアプリケーションの集合的挙動が非良性であるかどうかを決定するために追加の分析情報を使用することとをさらに含む動作を実行するためのプロセッサ実行可能命令で構成され得る。

【００１１】

さらなる実施形態では、このプロセッサは、追加の分析情報を生成するために、各々が特定された2つ以上のソフトウェアアプリケーションの挙動を特徴付ける挙動ベクトルを分類器モデルに適用することと、各挙動ベクトルに関して生成された追加の分析情報をアグリゲートすることと、特定された2つ以上のソフトウェアアプリケーションの集合的挙動が非良性であるかどうかを決定するために分析結果を使用することとをさらに含む動作を実行するためのプロセッサ実行可能命令で構成され得る。さらなる態様では、このプロセッサは、分析情報を生成するために、生成された挙動ベクトルを分類器モデルに適用することが、生成された挙動ベクトルをマルチアプリケーション分類器モデルに適用することを含むことが可能であるような動作を実行するためのプロセッサ実行可能命令で構成され得る。

【００１２】
さらなる実施形態では、このプロセッサは、収集された挙動情報に基づいて挙動ベクトルを生成することが、各々が複数のソフトウェアアプリケーションのうちの1つの挙動を特徴付ける複数の挙動ベクトルを生成することを含むことが可能であり、生成された挙動ベクトルをマルチアプリケーション分類器モデルに適用することが、分析情報を生成するために、挙動ベクトルの各々をマルチアプリケーション分類器モデルに適用することを含むことが可能であるような動作を実行するためのプロセッサ実行可能命令で構成され得る。
【００１３】
さらなる実施形態では、このプロセッサは、生成された挙動ベクトルをマルチアプリケーション分類器モデルに適用することが、マルチアプリケーション分類器モデル内に含まれた各テスト条件を評価することと、マルチアプリケーション分類器モデル内のテスト条件を評価した各結果の加重平均を計算することと、加重平均に基づいて、集合的挙動が非良性であるかどうかを決定することとを含むことが可能であるような動作を実行するためのプロセッサ実行可能命令で構成され得る。
【００１４】
さらなる実施形態では、このプロセッサは、複数のソフトウェアアプリケーションの集合的挙動を評価するために分析情報を使用することが、監視された複数のソフトウェアアプリケーションを類別することと、複数のソフトウェアアプリケーションの各カテゴリをプロファイリングすることと、複数のソフトウェアアプリケーションの各カテゴリに関する性能番号ナンバーを生成することとを含むことが可能であるような動作を実行するためのプロセッサ実行可能命令で構成され得る。
【００１５】
さらなる実施形態では、コンピューティングデバイスは、ハードウェアレベルにおけるコンピューティングデバイスメモリの使用およびハードウェアイベントを監視し、収集された挙動情報をプロセッサに送出するように構成された挙動観測器ハードウェアモジュールを含み得る。そのような実施形態では、このプロセッサは、複数のソフトウェアアプリケーションの活動を監視することが、収集された挙動情報を挙動観測器ハードウェアモジュールから受信することを含むような動作を実行するためのプロセッサ実行可能命令で構成され得る。
【００１６】
さらなる実施形態は、上で説明した態様の方法の動作をコンピューティングデバイスプロセッサに実行させるように構成されたプロセッサ実行可能ソフトウェア命令が記憶された非一時的プロセッサ可読記憶媒体を含み得る。さらなる実施形態は、上で説明した態様の方法の動作の機能を実行するための手段を有するコンピューティングデバイスを含み得る。
【００１７】
本明細書に組み込まれ、本明細書の一部を構成している添付の図面は、本発明の例示的な態様を示すものであり、上で与えられた全般的な説明、および下で与えられる詳細な説明とともに、本発明の特徴を説明するのに役立つ。
【図面の簡単な説明】
【００１８】
【図１】様々な実施形態を実装するのに適した、例示的なシステムオンチップのアーキテクチャ図である。
【図２】特定のモバイルデバイスの挙動が良性であるか、または非良性であるかを決定するように構成された、ある実施形態のモバイルデバイス内の例示的な論理構成要素および情報フローを示すブロック図である。
【図３】ある実施形態による、2つ以上のソフトウェアアプリケーションの集合的挙動を評価する方法を示すプロセスフロー図である。
【図４】ある実施形態による、2つ以上のソフトウェアアプリケーション間の関係を決定

する方法を示すプロセスフロー図である。
【図５】ある実施形態による、2つ以上のソフトウェアアプリケーションの集合的挙動が非良性であるかどうかを決定する方法を示すプロセスフロー図である。
【図６】別の実施形態による、2つ以上のソフトウェアアプリケーションの集合的挙動が非良性であるかどうかを決定する方法を示すプロセスフロー図である。
【図７】モバイルデバイスにおいてアプリケーションベースの分類器モデルまたは簡潔な分類器モデルを生成する別の実施形態のモバイルデバイスの方法を示すプロセスフロー図である。
【図８】簡潔な分類器モデルを生成するためにある実施形態のサーバプロセッサによって生成され、デバイスプロセッサによって使用され得る例示的なブーストされた判定株(boosted decision stump)の図である。
【図９】ある実施形態による、動的かつ適応的な観測を実行するように構成された観測器モジュール内の例示的な論理構成要素および情報フローを示すブロック図である。
【図１０】別の実施形態による、観測器デーモンを実装するコンピューティングシステム内の論理構成要素および情報フローを示すブロック図である。
【図１１】モバイルデバイス上で適応的な観測を実行するためのある実施形態の方法を示すプロセスフロー図である。
【図１２】ある実施形態において使用するのに適したモバイルデバイスの構成要素ブロック図である。
【図１３】ある実施形態において使用するのに適したサーバデバイスの構成要素ブロック図である。
【発明を実施するための形態】
【００１９】
添付の図面を参照しながら、様々な実施形態について詳細に説明する。可能な場合はいつでも、同じまた同様の部分を指すために、図面全体を通して同じ参照番号が使用される。特定の例および実装形態へと行われる言及は、説明を目的とし、本発明の範囲または特許請求の範囲を限定するものではない。
【００２０】
概要では、様々な実施形態は、コンピューティングデバイス上で動作する2つ以上のソフトウェアアプリケーションの集合的挙動を評価するために挙動分析技法または機械学習技法を使用する方法、およびその方法を実装するように構成されたコンピューティングデバイスを含む。たとえば、ある実施形態では、コンピューティングデバイスは、デバイス上で動作するソフトウェアアプリケーションの活動を監視し、監視された活動から挙動情報を収集し、収集された挙動情報に基づいて挙動ベクトルを生成し、分析情報を生成するために挙動ベクトルを分類器モデルに適用し、ソフトウェアアプリケーション間の関係を特定するために分析情報を使用し、特定された関係に基づいて、グループとして一緒に評価されるべきソフトウェアアプリケーションを特定し、特定されたソフトウェアアプリケーションの分析結果をアグリゲートし、ソフトウェアアプリケーションの集合的挙動が良性であるか、または非良性であるかを決定するために、アグリゲートされた分析結果を使用するように構成され得る。これらの動作は、デバイスのセキュリティ、性能、または電気消費の特徴に悪影響を有する可能性があり、さもなければ、従来の挙動ベースのセキュリティソリューションによって検出されないことになる、様々な条件または挙動をデバイスが迅速かつ効率的に特定し、それらに応答することを可能にすることによって、コンピューティングデバイスの作用を改善する。
【００２１】
コンピューティングデバイスは、コンピューティングデバイスの性能、電力利用レベル、ネットワーク利用レベル、セキュリティおよび/またはプライバシーを経時的に劣化させることが多い条件、要因、および/または挙動を特定、防止、および/または補正するために挙動分析技法を使用する挙動ベースのセキュリティシステムを備えることができる。たとえば、挙動ベースのセキュリティシステムは、ソフトウェアアプリケーションが良性

であるか、または非良性(たとえば、悪意がある、性能を劣化させるなど)であるかどうかを決定し、特定された問題(たとえば、非良性であると決定される挙動)を補正、修復、是正、隔離、またはさもなければ、解決するための様々な動作を実行するように構成され得る。

【００２２】

そのような挙動ベースのセキュリティシステムは、一般に、コンピューティングデバイスの性能劣化を経時的に防止するために非常に効果的であるが、悪意のあるソフトウェアアプリケーションは、その動作を遮蔽するために協働して結託(colluding)または動作することによってそのようなシステムによる検出を回避する可能性がある。たとえば、ユーザのアドレスブックから情報を盗むとき、第1の悪意のあるソフトウェアアプリケーションは、アドレスブックにアクセスし、情報を符号化し、符号化情報を一般ファイルまたは個別ファイル内に記憶する可能性がある。第2の悪意のあるアプリケーションは、次いで、一般/個別ファイル内に記憶された符号化情報を取り出し、その情報をサーバに送る可能性がある。

【００２３】

一般に、挙動ベースのセキュリティシステムは、この一連の動作(たとえば、アドレスブックデータの読取り、記憶、および送信)はデバイスの通常の動作パターンに整合しないと決定し、この挙動を非良性挙動として分類することが可能であろう。しかしながら、これらの動作は協働して動作する複数のソフトウェアアプリケーションによって実行されるため、既存のソリューションは、これらの動作を同じシーケンスまたは挙動の一部であるとして特定することに失敗することが多い。

【００２４】

個々に、アドレスデータにアクセスする動作、データを符号化する動作、データをファイル内に記憶する動作、およびファイル内に記憶された情報を送信する動作は、必ずしも非良性挙動を示すとは限らない。むしろ、非良性挙動を示すのは、これらの動作の集合的または一連の性能である。それにもかかわらず、既存の挙動ベースのソリューションは、ソフトウェアアプリケーション間の関係を十分に特徴付けない。結果として、既存のソリューションは、単一の挙動の一部として一緒に評価されるべき動作を正確に特定することに失敗する。これらのおよび他の理由で、既存の挙動ベースのセキュリティソリューションは、協働して動作する複数のソフトウェアアプリケーションを含むサイバー攻撃など、ソフトウェアアプリケーションのグループの集合的活動によって引き起こされる挙動および条件を特定し、それらに応答するには十分でない。

【００２５】

既存のシステムのこれらの限界に鑑みて、様々な実施形態は、協働して結託または動作する悪意のあるソフトウェアアプリケーションなど、ソフトウェアアプリケーションのグループの集合的活動によって引き起こされる非良性挙動を知的かつ効率的に特定し、それらに応答するように構成された挙動ベースのセキュリティシステムでコンピューティングデバイスを装備する。

【００２６】

ある実施形態では、挙動ベースのセキュリティシステムは、ソフトウェア間の対話を監視し、ソフトウェアアプリケーション間の関係を特徴付けるベクトルを生成し、アプリケーションが協働して結託または動作しているかどうかを決定するために、挙動ベクトルを分類器モデルに適用するように構成され得る。挙動ベースのセキュリティシステムは、次いで、グループとして一緒に分析されるべきソフトウェアアプリケーションを特定し、特定されたアプリケーションの挙動ベクトルを分類器モデルに適用し、結果として生じる情報をアグリゲートし、アプリケーションの集合的挙動が非良性であるかどうかを決定するために、アグリゲートされた情報を使用することができる。代替的に、挙動ベースのセキュリティシステムは、グループとして一緒に分析されるべきソフトウェアアプリケーションを特定し、特定されたアプリケーションの集合的挙動を特徴付ける挙動ベクトルを生成し、アプリケーションの集合的挙動が非良性であるかどうかを決定するために、生成され

た挙動ベクトルを同じまたは異なる分類器モデルに適用することができる。

【００２７】

様々な実施形態は、そのうちのいくつかが下の実施形態の詳細な説明で説明され、かつ/またはそこから明らかである、いくつかの理由でコンピューティングデバイス(たとえば、モバイルコンピューティングデバイス)の作用を改善する。たとえば、協働して結託または動作するソフトウェアアプリケーションを知的に特定することによって、かつ、特定されたアプリケーションの動作を単一のデバイス挙動の一部として一緒に評価することによって、様々な実施形態は、コンピューティングデバイスが、さもなければ、従来の挙動ベースのセキュリティソリューションによって検出されないことになる性能劣化挙動を特定し、それらに応答することを可能にすることによって、デバイスの作用を改善する。さらに、複数の個々のアプリケーションから収集された挙動情報をアグリゲートし、複数のアプリケーション間の対話を監視することによって、様々な実施形態は、コンピューティングデバイスが、ソフトウェアアプリケーション間の関係をより正確に特徴付け、ソフトウェアアプリケーションのカテゴリをプロファイリングし、ソフトウェアアプリケーションのグループの集合的挙動をより良好に分析し、システムレベルのデバイス動作をより良好に分類することによって、デバイスの作用を改善する。

【００２８】

さらに、様々な実施形態は、コンピューティングデバイスの応答性、性能、または電力の消費の特徴に著しい悪影響またはユーザが知覚できる影響を有さずに、コンピューティングデバイスが、非良性デバイス挙動を迅速かつ効率的に特定し、それらに応答することを可能にする挙動ベースのセキュリティシステムを提供する。したがって、挙動ベースのセキュリティシステムは、モバイルデバイス、および限定されたリソースを有し、バッテリ電力で動作し、性能およびセキュリティが重要な、スマートフォンなど、他のリソース制約のあるコンピューティングデバイスを含め、それらの中で使用するのに非常に適している。

【００２９】

コンピューティングデバイスの機能、機能性、および/または作用に対する追加の改善は、下で提供される実施形態の詳細な説明から明らかになろう。

【００３０】

「性能劣化」という用語は、本明細書では、より長い処理時間、より遅いリアルタイム応答性、より短いバッテリ持続時間、個人データの損失、悪意のある経済活動(たとえば、無許可のプレミアムSMSメッセージを送信すること)、サービス拒否(DoS)、不適切に書かれたまたは設計されたソフトウェアアプリケーション、悪意のあるソフトウェア、マルウェア、ウイルス、断片化されたメモリ、スパイ活動またはボットネット活動のためにモバイルデバイスを乗っ取ることまたは電話を利用することに関する動作などのような、コンピューティングデバイスの多種多様な望ましくない動作および特性を指すために使用される。また、これらの理由のいずれかで性能を劣化させる挙動、活動、および条件は、本明細書では「良性ではない」または「非良性である」と呼ばれる。

【００３１】

「モバイルコンピューティングデバイス」および「モバイルデバイス」という用語は、セルラー電話、スマートフォン、パーソナルまたはモバイルのマルチメディアプレーヤ、携帯情報端末(PDA)、ラップトップコンピュータ、タブレットコンピュータ、スマートブック、ウルトラブック、パームトップコンピュータ、ワイヤレス電子メール受信機、マルチメディアインターネット対応セルラー電話、ワイヤレスゲームコントローラ、および、性能が重要であるメモリとプログラマブルプロセッサとを含み、電力節約の方法が有益であるようなバッテリ電源で動作する同様のパーソナル電子デバイスのうちの、任意の1つまたはすべてを指すように、本明細書では互換的に使用される。様々な実施形態は、限られたリソースを有しバッテリで動作する、スマートフォンなどのモバイルコンピューティングデバイスに対して特に有用であるが、これらの実施形態は一般に、プロセッサを含みアプリケーションプログラムを実行する、任意の電子デバイスにおいて有用である。

【００３２】
一般に、モバイルデバイスの性能および電力効率は経時的に劣化する。近年、アンチウイルス会社(たとえば、McAfee、Symantecなど)は、この劣化を遅くすることを目的とするモバイル用アンチウイルス製品、ファイアウォール製品、および暗号化製品の販売を開始した。しかしながら、これらのソリューションの多くは、モバイルデバイス上で計算集約的なスキャニングエンジンを定期的に実行することに依存しており、そのことは、モバイルデバイスの処理リソースおよびバッテリリソースの多くを消費し、モバイルデバイスを遅くさせるか、もしくはより長期間使えなくさせ、かつ/または別様にユーザ体験を劣化させることがある。加えて、これらのソリューションは通常、知られているウイルスおよびマルウェアの検出に限定され、(たとえば、性能劣化がウイルスまたはマルウェアによって引き起こされないときに)経時的なモバイルデバイスの劣化に複合的に寄与することが多い、複数の複雑な要因および/または相互作用に対処しない。これらのおよび他の理由によって、既存のアンチウイルス製品、ファイアウォール製品、および暗号化製品は、経時的なモバイルデバイスの劣化に寄与し得る多数の要因を特定するための、モバイルデバイスの劣化を防止するための、または経年劣化したモバイルデバイスをその元の条件へと効率的に回復させるための、十分なソリューションを提供しない。

【００３３】
さらに、現代のモバイルデバイスは、高度に構成可能かつ複雑なシステムである。したがって、特定のデバイス挙動が良性であるか非良性である(たとえば、悪意がある、または性能を劣化させる)かを決定するために最も重要である特徴は、各モバイルデバイスにおいて異なり得る。さらに、不十分に書き込まれたか、または設計されたソフトウェアアプリケーション、マルウェア、ウイルス、断片化されたメモリ、バックグラウンド処理などを含む、モバイルコンピューティングデバイスの性能レベルおよび電力利用レベルの経時的な劣化に寄与し得る多種多様な要因が存在する。これらの要因の数、多様性および複雑さゆえに、現代のモバイルコンピューティングデバイスの複雑ではあるがリソースを制約されたシステムの性能および/または電力利用のレベルの劣化に寄与する可能性のある要因のすべてを評価することは、しばしば実現不可能である。したがって、ユーザ、オペレーティングシステム、および/またはアプリケーションプログラム(たとえば、アンチウイルスソフトウェアなど)が、問題の根源を正確かつ効率的に特定することは困難である。結果として、モバイルデバイスのユーザは現在、モバイルデバイスの性能レベルおよび電力利用レベルの経時的な劣化を防止するための、または経年劣化したモバイルデバイスをその元の性能レベルおよび電力利用レベルに回復させるための措置をほとんど有していない。

【００３４】
現在、コンピューティングデバイス上で動作/実行するソフトウェアアプリケーションの挙動をモデル化するための様々なソリューションが存在し、これらのソリューションは、ソフトウェアアプリケーションが、悪意があるか、または良性であるかを決定するために、機械学習技法とともに使用され得る。しかしながら、既存のソリューションは、挙動情報の非常に大きな集積を評価することを必要とし、コンピューティングデバイスのデバイス固有のもしくはアプリケーション固有の特徴を考慮するように挙動モデルを動的に生成せず、挙動モデル内の特徴を知的に優先順位付けず、個別のアプリケーションプログラムもしくはプロセスを評価することに限定され、かつ/またはデバイスにおける計算集約的なプロセスの実行を必要とするので、モバイルデバイスまたはリソースが制約されたデバイス上での使用に適していない。したがって、モバイルコンピューティングデバイスまたはリソースが制限されたコンピューティングデバイスにおいてこれらのソリューションを実装または実行することは、デバイスの応答性、性能、または電力消費の特性に大きな悪影響および/またはユーザが知覚できる影響を及ぼし得る。

【００３５】
これらの問題に鑑みてより良好な性能を提供するために、コンピューティングデバイス(たとえば、モバイルデバイスなど)は、コンピューティングデバイスの応答性、性能、ま

たは電力消費の特徴に著しい悪影響またはユーザが知覚できる影響を及ぼさずに、デバイス内の非良性挙動を知的かつ効率的に特定し、防止し、補正し、またはさもなければそれらに応答するための挙動分析技法を使用するように構成された挙動ベースのセキュリティシステムを装備することができる。

【００３６】

挙動ベースのセキュリティシステムは、観測プロセス、デーモン、モジュール、またはサブシステム(本明細書では「モジュール」と集合的に呼ばれる)、挙動抽出器モジュール、および分析器モジュールを含み得る。観測器モジュールは、コンピューティングデバイスシステム(たとえば、モバイルデバイスシステム)の様々なレベルにおいて、様々なアプリケーションプログラミングインターフェース(API)、レジスタ、カウンタ、または他のデバイス構成要素(本明細書で、集合的に「計測された構成要素」)を、計測または協調させ、計測された構成要素から挙動情報を収集し、収集された挙動情報を(たとえば、メモリ書込み動作、関数呼出しなどを介して)挙動抽出器モジュールに通信するように構成され得る。挙動抽出器モジュールは、各々がデバイスの1つまたは複数の特定のスレッド、プロセス、ソフトウェアアプリケーション、モジュール、または構成要素に関連付けられた観測されたイベント、条件、タスク、活動、および/または挙動(本明細書で、集合的に「挙動」)の多くまたはすべてを表すか、または特徴付ける挙動ベクトルを生成するために、収集された挙動情報を使用することができる。挙動抽出器モジュールは、生成された挙動ベクトルを(たとえば、メモリ書込み、関数呼出しなどを介して)分析器モジュールに通信することができ、分析器モジュールは、ソフトウェアアプリケーションまたはデバイスの挙動が良性であるか、または非良性である(たとえば、悪意がある、不十分に書き込まれた、性能劣化など)かどうかを決定するために、データ、アルゴリズム、および/またはモデルを実行(performing)、実行(executing)、および/または適用することを含み得る挙動分析動作を実行するために挙動ベクトルを使用する。コンピューティングデバイスプロセッサは、次いで、特定された問題(たとえば、非良性であると決定される挙動)を補正、修復、是正、隔離、またはさもなければ解決するための様々な動作を実行することができる。

【００３７】

上記のシステムは、一般に、コンピューティングデバイスの性能レベルおよび電力利用レベルの劣化を経時的に防止するために非常に効果的であるが、高度なサイバー攻撃が増大しており、その悪意のある動作を遮蔽するために2つ以上のソフトウェアアプリケーションを使用することによって、挙動ベースのセキュリティシステムによる検出を避けるか、または回避する可能性がある。たとえば、2つの結託したソフトウェアアプリケーションは、その動作を協調させ、ユーザの個人情報(たとえば、連絡先、クレジットカード番号など)を盗み、挙動ベースのセキュリティシステムによる検出を回避する可能性がある。たとえば、第1の結託しているアプリケーションは、デバイスのメモリの指定された部分内に(または、特定のメモリロケーションにおいて)個人情報を読み取り、書き込む可能性があり、第2の結託しているアプリケーションは、メモリロケーション内に記憶された情報を読み取り、サーバに送信する可能性がある。個々に、これらの動作は悪意のある活動を示さないため、既存の挙動ベースのセキュリティシステムは、この一連の動作を単一の非良性挙動に関連するとして正確に特定することができない場合がある。

【００３８】

一連の動作が単一の挙動に関連することを検出および決定する1つの方法は、データフロー追跡動作を実行することによる。FlowDroidなどのデータフロー追跡ソリューションは、一般に、悪意のあるソフトウェアアプリケーションが検出を回避するのを防止するための効果的なツールである。手短に言えば、データフロー追跡ソリューションは、コンピューティングシステム内のデータ動作(読取り、書込み、データ符号化、データ送信など)の多くまたはすべてを監視し、個々にまたは集合的にデータを不適切に使用するソフトウェアアプリケーションの特定を試みる。しかしながら、データフロー追跡ソリューションは、コンピューティングシステム内のデータフローおよびデータ動作の多くを監視するこ

とを必要とし、かつ/または非常に複雑で電力集約型のプロセスの実行を必要とする。したがって、データフロー追跡ソリューションは、一般に、比較的限定された処理リソース、メモリリソース、およびエネルギーリソースを有する、リソース制約のあるシステムであるモバイルデバイスにおける使用には適さない。さらに、現代のモバイルデバイスは複雑なシステムであり、悪意があるか、またはさもなければ、モバイルデバイスの性能劣化を引き起こすことがある様々なデータフロー、データ動作(読取り、書込み、データ符号化、データ送信など)、プロセス、構成要素、挙動、または要因(または、それらの組合せ)のすべてを評価することは、しばしば実行不可能である。これらのすべての理由で、既存のデータフロー追跡ソリューションは、モバイルコンピューティングデバイスおよびリソース制約のあるコンピューティングデバイスにおける使用には適さない。

【００３９】

これらの問題に鑑みて、様々な実施形態は、データフローを監視すること、またはデバイス内のデータフロー追跡動作を実行することなく、ソフトウェアアプリケーションの選択グループの集合的挙動によって引き起こされる結託攻撃、および他の条件を特定し、分析し、防止し、かつ/またはそれらに応答するようにデバイスプロセッサ(たとえば、モバイルデバイスプロセッサなど)を構成することができる。デバイスプロセッサは、コンピューティングデバイスの応答性、性能、または電力消費の特性に著しい悪影響またはユーザが知覚できる影響を与えることなく、これを達成することができる。したがって、様々な実施形態は、限定されたリソースを含み、性能およびバッテリ寿命が重要であるモバイルコンピューティングデバイスおよびリソース制約のあるコンピューティングデバイスにおいて特に有用である。

【００４０】

様々な実施形態では、デバイスプロセッサ(または、デバイスの挙動ベースのセキュリティシステム)は、2つ以上のソフトウェアアプリケーション間の対話を監視し、監視されるアプリケーション間の関係を特定または特徴付ける関係情報(たとえば、挙動ベクトルなど)を生成し、グループとして一緒に評価されるべきソフトウェアアプリケーションを特定するために関係情報を使用し、特定されたアプリケーションの各々から挙動情報を収集し、特定のアプリケーションの各々から収集された挙動情報を(たとえば、挙動ベクトル内に)アグリゲートし、かつ/または各特定されたアプリケーションを評価した結果を(たとえば、分類器モデルを介して)アグリゲートするように構成され得る。デバイスプロセッサは、次いで、特定されたアプリケーションの集合的挙動を単一のデバイス挙動として評価するために、アグリゲートされた情報を使用することができる。

【００４１】

特定のアプリケーション間の関係および対話の性質を決定することによって、様々な実施形態は、デバイスプロセッサが、2つ以上のアプリケーションが、一緒に動作して、その悪意のある活動を隠しているかどうか、および/またはソフトウェアアプリケーションの小さなまたは専心的なグループの集合的挙動が(たとえば、アプリケーションのうちの1つまたは複数が不十分に設計されていることなどにより)コンピューティングデバイスの性能特性に予想外の悪影響を有するかどうかをより良好に決定することを可能にする。

【００４２】

いくつかの実施形態では、デバイスプロセッサは、監視されたアプリケーションを類別し、アプリケーションの選択グループまたはカテゴリをプロファイリングまたは事前プロファイリングし、かつ/またはアプリケーションのカテゴリに関する性能番号を生成するように構成され得る。性能番号は、エネルギー消費、メモリ使用、帯域幅使用、CPUサイクル、アプリケーション性能に関するユーザ経験、ユーザインターフェース(UI)応答性、および個々のアプリケーション、アプリケーションのグループ、またはアプリケーションのカテゴリの他の同様の測定可能特性など、様々な性能特性を特定、評価、および/または比較する際に使用するのに適した情報を含み得る。挙動分析技法を使用して、アプリケーション(または、アプリケーションのグループまたはカテゴリ)に関する性能番号をプロファイリングまたは生成することによって、様々な実施形態は、デバイスプロセッサが、

コンピューティングデバイスの性能レベルおよび/または電力消費レベルに予測外の影響、不均衡な影響、または悪影響を有するアプリケーションまたはアプリケーションのグループをより良好に特定し、それらに応答することを可能にする。

【００４３】

様々な実施形態では、コンピューティングデバイス(たとえば、モバイルデバイスなど)は、本出願で論じる動作のうちのいずれかまたはすべてを実行するように構成された包括的な挙動監視および分析システムを装備することができる。たとえば、挙動監視および分析システムは、観測器モジュール、挙動抽出器モジュール、および分析器モジュールを含み得る。観測器モジュールは、選択アプリケーション(または、アプリケーションのグループ)間の動作(たとえば、メモリ読取り/書込み動作)、対話、関係、および通信を監視するように構成され得る。これは、メモリの選択部分、選択メモリアドレス、ハードウェア構成要素、ContentResolver APIなど、様々な計測された構成要素を監視することによって達成され得る。これらの計測された構成要素を監視することによって、観測器モジュールは、さもなければ、従来の挙動ベースのセキュリティシステムによって収集されないことになる追加の挙動情報を収集することができる。

【００４４】

挙動抽出器モジュールは、アプリケーション間の関係を特徴付ける挙動ベクトルおよび/または2つ以上のアプリケーションの集合的挙動を表すか、または特徴付ける挙動ベクトルを生成するために、挙動情報(すなわち、観測器モジュールによって収集された情報)を使用するように構成され得る。各挙動ベクトルは、1つまたは複数の「挙動の特徴」を含むか、またはカプセル化する情報構造であり得る。挙動特徴は、コンピューティングデバイス内の観測されたイベント、条件、活動、動作、関係、対話、または挙動のすべてまたは一部を表す抽象的な数またはシンボルであり得る。各挙動の特徴は、可能な値の範囲、それらの値に対して実行され得る動作、値の意味、および他の同様の情報を特定するデータタイプと関連付けられ得る。データタイプは、対応する挙動特徴(または特徴値)がどのように測定され、分析され、重み付けられ、または使用されるべきかを決定するために、コンピューティングデバイスによって使用され得る。

【００４５】

挙動抽出器モジュールは、生成された挙動ベクトルを(たとえば、メモリ書込み動作、関数呼出しなどによって)分析器モジュールに通信することができ、分析器モジュールは、ソフトウェアアプリケーション間の関係の性質(たとえば、2つ以上のソフトウェアアプリケーションが協働して動作しているかなど)を決定し、かつ/またはアプリケーションの集合的挙動が非良性であるかどうかを決定するために、挙動ベクトルを分類器モデルに適用することができる。

【００４６】

分類器モデルは、特定の特徴、要因、データ点、エントリ、API、状態、条件、挙動、ソフトウェアアプリケーション、プロセス、動作、構成要素(本明細書で、集合的に「特徴」)など、またはデバイスの挙動の他の実施形態を迅速かつ効率的にテストまたは評価するためにデバイスプロセッサによって使用され得るデータ、エントリ、判定ノード、判定基準、および/または情報構造を含む挙動モデルであり得る。分類器モデルは、ソフトウェアアプリケーション間の関係および/またはコンピューティングデバイス内で観測されるべき挙動の性質を決定するためにデバイスプロセッサによって使用され得る情報を含むことも可能である。

【００４７】

各分類器モデルは、完全な分類器モデルまたは簡潔な分類器モデルとして類別され得る。完全な分類器モデルは、数千の特徴および数十億のエントリを含み得る大きいトレーニングデータセットに応じて生成される、ロバストなデータモデルであり得る。簡潔な分類器モデルは、特定のコンピューティングデバイスの挙動が良性でないかどうかを決定することに最も関連がある特徴/エントリに対するテストを含むかまたは優先順位付ける、縮小されたデータセットから生成された、より専心的なデータモデルであり得る。ローカル

分類器モデルは、モバイルコンピューティングデバイスにおいて生成される簡潔な分類器モデルであり得る。デバイス固有の分類器モデルは、特定のデバイス内で活動または挙動を分類することに最も関連があると決定される、コンピューティングデバイス固有の特徴/エントリのみを含む/テストする、焦点が絞られたデータモデルを含むローカル分類器モデルであり得る。アプリケーション固有の分類器モデルは、特定のソフトウェアアプリケーション(または、特定のタイプのソフトウェアアプリケーション)が非良性であるかどうかを決定することに最も関連がある特徴/エントリに対するテストを含むかまたは優先順位付ける、専心的なデータモデルを含むローカル分類器モデルであり得る。

【００４８】

マルチアプリケーション分類器モデルは、2つ以上のソフトウェアアプリケーションを評価することに関連する特徴をテストする、アグリゲートされた特徴セットおよび/または判定ノードを含むローカル分類器モデルであり得る。たとえば、マルチアプリケーション分類器モデルは、2つのソフトウェアアプリケーション間の関係を特定または特徴付けることに最も関連がある条件または特徴をテストする判定ノードを含み得る。別の例として、マルチアプリケーション分類器モデルは、2つのソフトウェアアプリケーション(または、特定のタイプのソフトウェアアプリケーション)の集合的挙動が非良性であるかどうかを決定することに最も関連がある条件または特徴をテストする判定ノードを含み得る。

【００４９】

いくつかの実施形態では、デバイスプロセッサは、2つ以上のアプリケーション特定分類器モデルを組み合わせることによって、マルチアプリケーション分類器モデルを生成するように構成され得る。他の実施形態では、デバイスプロセッサは、2つ以上のソフトウェアアプリケーション間の関係、対話、および/または通信を特定することに最も関連があるデバイス特徴を特定し、特定されたデバイス特徴を評価するテスト条件を特定し、特定されたテスト条件を含めるように分類器モデルを生成することによって、マルチアプリケーション分類器モデルを生成することができる。さらなる実施形態では、デバイスプロセッサは、特定されたテスト条件の優先順位、重要性、または成功率を決定し、テスト条件がその優先順位、重要性、または成功率に基づいて順序付けられるように、分類器モデルを生成するように構成され得る。

【００５０】

様々な実施形態では、デバイスプロセッサは、アプリケーション間の関係を決定するため、および/またはアプリケーションの集合的挙動が非良性であるかどうかを決定するために、分類器モデルを生成または使用するように構成され得る。

【００５１】

たとえば、ある実施形態では、デバイスプロセッサは、コンピューティングデバイス上で動作するソフトウェアアプリケーション間の対話を監視し、ソフトウェアアプリケーション間の関係を特徴付けるベクトルを生成し、分析情報を生成するために挙動ベクトルを分類器モデルに適用し、アプリケーションが協働して結託または動作しているかどうかを決定するために分析情報を使用するように構成され得る。デバイスプロセッサは、次いで、グループとして一緒に分析されるべきソフトウェアアプリケーション(たとえば、結託しているアプリケーション)を特定し、特定されたアプリケーションの挙動ベクトルを同じまたは異なる分類器モデル(または、分類器モデルのファミリー)に適用し、結果として生じる分析情報をアグリゲートし、特定されたアプリケーションの集合的挙動が非良性であるかどうかを決定するために、アグリゲートされた分析情報を使用することができる。

【００５２】

別の例として、デバイスプロセッサは、グループとして一緒に分析されるべきソフトウェアアプリケーションを特定し、特定されたアプリケーションの活動を監視し、監視された活動の各々に関する挙動情報を収集し、収集された挙動情報に基づいて、特定されたアプリケーションの集合的挙動を特徴付ける挙動ベクトルを生成し、分析情報を生成するために、生成された挙動ベクトルを分類器モデル(または、分類器モデルのファミリー)に適用し、特定されたアプリケーションの集合的挙動が非良性であるかどうかを決定するため

に分析情報を使用するように構成され得る。

【００５３】

いくつかの実施形態では、デバイスプロセッサはまた、監視されたソフトウェアアプリケーションを類別し、ソフトウェアアプリケーションの各カテゴリをプロファイリングし、かつ/またはソフトウェアアプリケーションの各カテゴリに関する性能番号を生成するために分析情報(すなわち、挙動ベクトルを分類器モデルに適用する結果)を使用するように構成され得る。たとえば、デバイスプロセッサは、ソフトウェアアプリケーションのクラス(たとえば、ゲーム、ソーシャルネットワーキング、金融給など)によって消費される電力の量を計算/推定するために、分析情報を使用することができる。さらに、各特徴の電力消費を事前プロファイリングおよび測定することによって、デバイスプロセッサは、デバイス上で動作するすべての活動またはアプリケーションの電力消費をプロファイリングすることができる。デバイスプロセッサは、バッテリ寿命を予測し、デバイスの著しい量の利用可能なリソースを消費しているアプリケーションの1つまたは複数のクラスを特定し、他の同様の動作を実行するために、そのような情報(たとえば、電力消費の推定)を使用することができる。デバイスプロセッサは、そのような情報をコンピューティングデバイスのユーザに表示すること、またはデバイス挙動をより良好に評価するためにこの情報を使用することができる。

【００５４】

様々な実施形態は、単一のプロセッサシステムおよび多重プロセッサシステム、ならびにシステムオンチップ(SOC)を含む、いくつかの異なるコンピューティングデバイス内で実装され得る。図1は、様々な実施形態を実装するコンピューティングデバイス内で使用され得る例示的なシステムオンチップ(SOC)100アーキテクチャを示すアーキテクチャ図である。SOC100は、デジタル信号プロセッサ(DSP)101、モデムプロセッサ104、グラフィックスプロセッサ106、およびアプリケーションプロセッサ108のような、いくつかの異質のプロセッサを含み得る。SOC100はまた、異種プロセッサ101、104、106、108のうちの1つまたは複数に接続された1つまたは複数のコプロセッサ110(たとえば、ベクトルコプロセッサ)を含んでもよい。各プロセッサ101、104、106、108、110は、1つまたは複数のコアを含み得、各プロセッサ/コアは、他のプロセッサ/コアとは無関係に動作を実行できる。たとえば、SOC100は、第1のタイプのオペレーティングシステム(たとえば、FreeBSD、LINUX、OS Xなど)を実行するプロセッサと、第2のタイプのオペレーティングシステム(たとえば、Microsoft Windows 8)を実行するプロセッサとを含み得る。

【００５５】

SOC100はまた、センサデータ、アナログデジタル変換、ワイヤレスデータ送信を管理し、ゲームおよび映画用の符号化されたオーディオ信号を処理することのような他の特殊な動作を実行するための、アナログ回路およびカスタム回路114を含み得る。SOC100は、電圧調整器、発振器、位相ロックループ、周辺ブリッジ、データコントローラ、メモリコントローラ、システムコントローラ、アクセスポート、タイマ、およびコンピューティングデバイス上で実行するプロセッサおよびクライアントをサポートするために使用される他の同様の構成要素のような、システム構成要素およびリソース116をさらに含み得る。

【００５６】

システム構成要素/リソース116およびカスタム回路114は、カメラ、電子ディスプレイ、ワイヤレス通信デバイス、外部メモリチップなどの周辺デバイスとインターフェースする回路を含み得る。プロセッサ101、104、106、108は、再構成可能な論理ゲートのアレイを含み、かつ/またはバスアーキテクチャ(たとえば、CoreConnect、AMBAなど)を実装し得る、相互接続/バスモジュール124を介して、1つまたは複数のメモリ要素112、システム構成要素およびリソース116、ならびにカスタム回路114に相互接続され得る。通信は、高性能ネットワークオンチップ(NoCs)などの高度な相互接続によって提供され得る。

【００５７】

プロセッサ101、104、106、108、110のうちの1つまたは複数内で実行しているオペレーティングシステムは、ソフトウェアアプリケーションによるメモリの割振りおよび使用を

制御し、協調させ、複数のソフトウェアアプリケーションにわたって物理メモリを区分するように構成され得る。したがって、オペレーティングシステムは、様々なソフトウェアアプリケーションによるメモリの割振りおよび使用を管理し、1つのプロセスによって使用されるメモリが別のプロセスによってすでに使用されているメモリに干渉しないことを確実にする1つまたは複数のメモリ管理システムまたはメモリ管理プロセス(たとえば、ウイルスメモリマネージャなど)を含み得る。

【００５８】

上で論じたソフトウェアベースのメモリ管理システムまたはメモリ管理プロセス(たとえば、OC、VMMなど)に加えて、SOC100は、中央処理装置(CPU)メモリ管理ユニット(MMU)およびシステムMMUのような、1つまたは複数のハードウェアベースのメモリ管理システムを含み得る。CPU MMUおよびシステムMMUは、物理アドレスへの仮想アドレスの転換、キャッシュ制御、バス調停、およびメモリ保護のような、様々なメモリ関連の動作を実行することを担うハードウェア構成要素であり得る。たとえば、CPU MMUは、アドレス転換サービスおよび保護機能性をメインCPU(たとえば、アプリケーションプロセッサ108)に提供することを担ってよく、システムMMUは、アドレス転換サービスおよび保護機能性を他のハードウェア構成要素(たとえば、デジタル信号プロセッサ101、モデムプロセッサ104、グラフィックスプロセッサ106など)に提供することを担ってよい。

【００５９】

SOC100は、ハードウェアレベルにおいて、かつ/またはハードウェアイベント(たとえば、メモリの読取り動作および書込み動作など)に基づいて、ソフトウェアアプリケーションによるMMUおよびメモリ要素112のアクセスまたは使用を監視するように構成されたプログラマブル論理回路(PLC)であり得るハードウェアベースのメモリ監視ユニット113を含んでもよい。ハードウェアベースのメモリ監視ユニット113は、デバイスの他のハードウェア、ならびにソフトウェアベースのメモリ管理システムおよびMMUとは別個であってよく、かつそれらから独立して動作してよい。

【００６０】

様々な実施形態では、ハードウェアベースのメモリ監視ユニット113は、メモリ使用情報を収集するために、ソフトウェアアプリケーションによるMMUおよびメモリ要素112のアクセスおよび使用を監視し、アプリケーション間の関係を特定するため、および/またはソフトウェアアプリケーションによるメモリの使用が疑わしいまたは結託している挙動を示すかどうかを決定するために、収集されたメモリ使用情報を(PLCにプログラムされ得る)メモリ使用パターンと比較するように構成され得る。ハードウェアベースのメモリ監視ユニット113は、次いで、特定された関係および/または疑わしいまたは結託している挙動を観測器モジュールまたは分析器モジュールに(たとえば、プロセッサ101、104、106、108を介して)報告することができる。

【００６１】

SOC100は、クロック118および電圧調整器120のような、SOCの外部のリソースと通信するための入力/出力モジュール(図示せず)をさらに含んでもよい。SOCの外部のリソース(たとえば、クロック118、電圧調整器120)は、内部SOCプロセッサ/コア(たとえば、DSP101、モデムプロセッサ104、グラフィックプロセッサ106、アプリケーションプロセッサ108など)のうちの2つ以上によって共有され得る。

【００６２】

SOC100はまた、スピーカ、ユーザインターフェース要素(たとえば、入力ボタン、タッチスクリーンディスプレイなど)、マイクロフォンアレイ、物理条件(たとえば、位置、方向、動き、方位、振動、圧力など)を監視するためのセンサ、カメラ、コンパス、GPS受信機、通信回路(たとえば、Bluetooth(登録商標)、WLAN、WiFiなど)、および現代の電子デバイスの他のよく知られている構成要素(たとえば、加速度計など)を含む、センサからセンサデータを収集するのに適したハードウェアおよび/またはソフトウェア構成要素を含み得る。

【００６３】

上で論じたSOC100に加えて、様々な実施形態が、単一のプロセッサ、複数のプロセッサ、マルチコアプロセッサ、またはこれらの任意の組合せを含み得る、多種多様なコンピューティングシステムにおいて実装され得る。

【００６４】

図2は、非良性デバイス挙動を特定し、それらに応答するために、挙動分析技法を使用するように構成された挙動ベースのセキュリティシステム200を含むある実施形態のモバイルコンピューティングデバイス102内の例示的な論理構成要素および情報フローを示す。図2に示す例では、コンピューティングデバイスは、挙動観測器モジュール202、挙動抽出器モジュール204、挙動分析器モジュール208、およびアクチュエータモジュール210を含む実行可能命令モジュールで構成されたデバイスプロセッサ(たとえば、モバイルデバイスプロセッサ)を含むモバイルコンピューティングデバイス102である。モジュール202～210の各々は、ソフトウェア、ハードウェア、またはこれらの組合せにおいて実装される、スレッド、プロセス、デーモン、モジュール、サブシステム、または構成要素であり得る。様々な実施形態では、モジュール202～210は、オペレーティングシステムの部分内(たとえば、カーネル内、カーネル空間内、ユーザ空間内など)で、個別のプログラムもしくはアプリケーション内で、専用のハードウェアバッファもしくはプロセッサ内で、またはそれらの任意の組合せにて実装され得る。ある実施形態では、モジュール202～210のうちの1つまたは複数は、モバイルコンピューティングデバイス102の1つまたは複数のプロセッサ上で実行するソフトウェア命令として実装される場合がある。

【００６５】

挙動観測器モジュール202は、デバイスの様々なレベル/モジュールにおいてアプリケーションプログラミングインターフェース(API)、カウンタ、ハードウェアモニタなどを計測し、ある時間期間にわたって様々なレベル/モジュールにおいて活動、条件、動作、およびイベント(たとえば、システムイベント、状態変化など)を監視するように構成され得る。たとえば、挙動観測器モジュール202は、モバイルコンピューティングデバイス102の様々なソフトウェア構成要素およびハードウェア構成要素を監視し、モバイルコンピューティングデバイス102の活動に関連付けられた、監視されたおよび測定可能な構成要素の対話、通信、トランザクション、イベント、または動作に関係する挙動情報を収集するように構成され得る。そのような活動は、ソフトウェアアプリケーションのハードウェア構成要素の使用、動作またはタスクの性能、モバイルコンピューティングデバイス102の処理コアにおけるソフトウェアアプリケーションの実行、プロセスの実行、タスクまたは動作の実行、デバイスの挙動などを含む。

【００６６】

さらなる例として、挙動観測器モジュール202は、ソフトウェアアプリケーションによるデバイスメモリの割振りまたは使用を監視することによって、モバイルコンピューティングデバイス102の活動を監視するように構成され得る。ある実施形態では、これは、コンピューティングデバイスのメモリ管理システム(たとえば、仮想メモリマネージャ、メモリ管理ユニットなど)の動作を監視することによって達成され得る。そのようなシステムは、一般に、1つのプロセスによって使用されるメモリが別のプロセスによってすでに使用されているメモリに干渉しないことを確実にするために、様々なアプリケーションプログラムによってシステムメモリの割振りおよび使用を管理することを担う。したがって、メモリ管理システムの動作を監視することによって、デバイスプロセッサは、2つのプロセスに同じメモリスペースが割り振られているかどうか、2つのプロセスが同じメモリアドレスまたは同じメモリロケーションから情報を読み取り、そこに情報を書き込んでいるかどうか、または2つのプロセスが他の疑わしいメモリ関連の動作を実行しているかどうかなど、2つのアプリケーションが協働して動作しているかどうかを決定する際に使用するのに適した挙動情報を収集することができる。

【００６７】

挙動観測器モジュール202は、監視された活動、条件、動作、またはイベントに関係する挙動情報を収集し、収集された情報をメモリ(たとえば、ログファイルなど)に記憶する

ことができる。挙動観測器モジュール202は、次いで、収集された挙動情報を挙動抽出器モジュール204に(たとえば、メモリ書込み動作、関数呼出しなどを介して)通信することができる。

【００６８】

ある実施形態では、挙動観測器モジュール202は、ハードウェアレベルにおいて、かつ/またはハードウェアイベント(たとえば、メモリ読取り動作およびメモリ書込み動作など)に基づいて、デバイスメモリの割振りまたは使用を監視することによって、モバイルコンピューティングデバイス102の活動を監視するように構成され得る。さらなる実施形態では、挙動観測器モジュール202は、監視機能のより高速なニアリアルタイムの実行のために、ハードウェアモジュール(たとえば、図1を参照して上で説明したメモリ監視ユニット113)内で実装され得る。たとえば、挙動観測器モジュール202は、プログラマブル論理要素が、ハードウェアレベルにおいて、かつ/またはハードウェアイベント(たとえば、メモリ読取り動作およびメモリ書込み動作など)に基づいて、コンピューティングデバイスメモリの割振りまたは使用を監視するように、かつさもなければ、様々な実施形態を実装するように構成されたプログラマブル論理回路(PLC)を含むハードウェアモジュール内で実装され得る。そのようなハードウェアモジュールは、ハードウェアイベント監視の結果を、挙動抽出器モジュール204を実装しているデバイスプロセッサに送出することができる。PLCは、よく知られているPLCプログラミング方法を使用して、いくつかのハードウェアを監視し、本明細書で説明する様々な実施形態のいくつかの動作を実装するように構成され得る。他の回路を使用して、ハードウェアモジュール内の実装形態の方法のいくつかの動作を実装することも可能である。

【００６９】

同様に、モジュール202～210の各々は、実施形態の方法のいくつかの動作を実行するためにPLCプログラミング方法を使用するように構成された1つまたは複数のPLC要素をSoCに含めることによってなど、ハードウェアモジュール内で実装され得る。

【００７０】

挙動抽出器モジュール204は、収集された挙動情報を受信および取り出し、1つまたは複数の挙動ベクトルを生成するためにこの情報を使用するように構成され得る。様々な実施形態では、挙動抽出器モジュール204は、ソフトウェアアプリケーションの観測された挙動、関係、または対話の簡潔な定義を含めるように挙動ベクトルを生成するように構成され得る。たとえば、各挙動ベクトルは、ソフトウェアアプリケーションの集合的挙動を値またはベクトルデータ構造で簡潔に記述することができる。ベクトルデータ構造は、一連の数字を含んでよく、その各々は、コンピューティングデバイスのカメラが使用されているかどうか(たとえば、ゼロであるか、または1であるか)、コンピューティングデバイスからどの程度のネットワークトラフィックが送信されているか、またはコンピューティングデバイスによってどの程度のネットワークトラフィックが生成されているか(たとえば、毎秒20KBなど)、どれだけ多くのインターネットメッセージが通信されているか(たとえば、SMSメッセージの数など)、および/または挙動観測器モジュール202によって収集された任意の他の挙動情報など、デバイスの特徴または挙動を示す。ある実施形態では、挙動抽出器モジュール204は、挙動ベクトルが、コンピューティングデバイス(たとえば、挙動分析器モジュール208)がアプリケーション間の関係を迅速に認識すること、特定すること、または分析することを可能にする識別子として機能するように挙動ベクトルを生成するように構成可能である。

【００７１】

挙動分析器モジュール208は、2つ以上のソフトウェアアプリケーション間の関係の性質を特定するために挙動ベクトルを分類器モデルに適用するように構成され得る。挙動分析器モジュール208は、集合的デバイス挙動(すなわち、デバイス上で動作する2つ以上のソフトウェアアプリケーションの集合的活動)が、経時的にデバイスの劣化に寄与する(または、寄与する可能性が高い)かつ/またはさもなければ、デバイス上に問題を生じさせる非良性挙動であるかどうかを決定するために、挙動ベクトルを分類器モデルに適用するよう

に構成されることも可能である。

【００７２】

挙動分析器モジュール208は、活動または挙動が良性でないことをアクチュエータモジュール210に通知することができる。それに応答して、アクチュエータモジュール210は、特定された問題を修復する、是正する、隔離する、またはさもなければ解決するために、様々なアクションまたは動作を実行することができる。たとえば、アクチュエータモジュール210は、挙動ベクトルを(たとえば、分析器モジュールによって)分類器モデルに適用する結果が、ソフトウェアアプリケーションの集合的挙動が良性でないことを示すとき、ソフトウェアアプリケーションのうちの1つまたは複数を中止または終了するように構成され得る。

【００７３】

様々な実施形態では、挙動観測器モジュール202は、アプリケーションフレームワークまたはランタイムライブラリ、システムコールAPI、ファイルシステムおよびネットワーキングサブシステム動作、デバイス(センサデバイスを含む)状態変化、ならびに他の同様のイベント内のライブラリAPIコールに関する情報を収集することによって、モバイルコンピューティングデバイス102の活動を監視するように構成され得る。さらに、挙動観測器モジュール202は、ファイルシステムの活動を監視することができ、ファイルシステムの活動は、ファイル名、ファイルアクセスのカテゴリ(個人情報または通常のデータファイル)を探索すること、ファイル(たとえば、type exe、zipなど)を作成または削除すること、ファイル読出し/書込み/探索動作、ファイルパーミッションを変更することなどを含み得る。

【００７４】

挙動観測器モジュール202はまた、接続のタイプ、プロトコル、ポート番号、デバイスが接続されるサーバ/クライアント、接続の数、通信の量または頻度などを含み得るデータネットワーク活動を監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。挙動観測器モジュール202は、電話ネットワーク活動を監視することができ、電話ネットワーク活動は、送出、受信、または傍受された通話またはメッセージ(たとえば、SMSなど)のタイプおよび数(たとえば、かけられたプレミアムコールの数)を監視することを含み得る。

【００７５】

挙動観測器モジュール202はまた、フォークの数、メモリアクセス動作、開かれたファイルの数などを監視することを含み得るシステムリソース使用を監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。挙動観測器モジュール202は、ディスプレイがオンかまたはオフか、デバイスがロックされているかまたはロックされていないか、バッテリの残量、カメラの状態など様々な要因を監視することを含み得る、モバイルコンピューティングデバイス102の状態を監視することができる。挙動観測器モジュール202はまた、たとえば、重要なサービス(ブラウザ、契約プロバイダなど)に対する意図、プロセス間通信(IPC)の程度、ポップアップウィンドウなどを監視することによって、プロセス間通信を監視することができる。

【００７６】

挙動観測器モジュール202はまた、カメラ、センサ、電子ディスプレイ、WiFi通信構成要素、データコントローラ、メモリコントローラ、システムコントローラ、アクセスポート、タイマ、周辺デバイス、ワイヤレス通信構成要素、外部メモリチップ、電圧レギュレータ、発振器、フェーズロックループ、周辺ブリッジ、ならびに、プロセッサ、およびモバイルコンピューティングデバイス102上で実行するクライアントをサポートするために使用される他の同様の構成要素を含み得る、1つまたは複数のハードウェア構成要素のドライバの統計データおよび/またはステータスを監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。

【００７７】

挙動観測器モジュール202はまた、モバイルコンピューティングデバイス102および/ま

たはコンピューティングデバイスのサブシステムの状態またはステータスを示す、1つまたは複数のハードウェアカウンタを監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。ハードウェアカウンタは、モバイルコンピューティングデバイス102内で発生するハードウェア関連の活動またはイベントのカウント値または状態を記憶するように構成されたプロセッサ/コアの専用レジスタを含み得る。

【００７８】

挙動観測器モジュール202はまた、ソフトウェアアプリケーションの活動または動作、アプリケーションダウンロードサーバ(たとえば、Apple(登録商標) App Storeサーバ)からのソフトウェアダウンロード、ソフトウェアアプリケーションによって使用されるコンピューティングデバイス情報、呼情報、テキストメッセージング情報(たとえば、SendSMS、BlockSMS、ReadSMSなど)、メディアメッセージング情報(たとえば、ReceiveMMS)、ユーザアカウント情報、位置情報、カメラ情報、加速度計情報、ブラウザ情報、ブラウザベースの通信の内容、音声ベースの通信の内容、短距離無線通信(たとえば、Bluetooth（登録商標）、WiFiなど)、テキストベースの通信の内容、記録されたオーディオファイルの内容、電話帳または連絡先情報、連絡先リストなどを監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。

【００７９】

挙動観測器モジュール202は、ボイスメールを含む通信(VoiceMailComm)、デバイス識別子を含む通信(DeviceIDComm)、ユーザアカウント情報を含む通信(UserAccountComm)、カレンダー情報を含む通信(CalendarComm)、ロケーション情報を含む通信(LocationComm)、記録されたオーディオ情報を含む通信(RecordAudioComm)、加速度計情報を含む通信(AccelerometerComm)などを含む、モバイルコンピューティングデバイス102の送信または通信を監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。

【００８０】

挙動観測器モジュール202は、コンパス情報、コンピューティングデバイスの設定、バッテリ寿命、ジャイロスコープ情報、圧力センサ、磁気センサ、スクリーン活動などの使用とそれらに対する更新/変更とを監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。挙動観測器モジュール202は、ソフトウェアアプリケーションとの間で通信される通知(AppNotifications)、アプリケーション更新などを監視することができる。挙動観測器モジュール202は、第2のソフトウェアアプリケーションのダウンロードおよび/またはインストールを要求している第1のソフトウェアアプリケーションに関する条件またはイベントを監視することができる。挙動観測器モジュール202は、パスワードの入力のような、ユーザ検証に関する条件またはイベントを監視することができる。

【００８１】

挙動観測器モジュール202はまた、アプリケーションレベル、無線レベル、およびセンサレベルを含む、モバイルコンピューティングデバイス102の複数のレベルにおいて条件またはイベントを監視することによって、モバイルコンピューティングデバイス102の活動を監視することもできる。アプリケーションレベルの観測には、顔認識ソフトウェアを介してユーザを観測すること、ソーシャルストリームを観測すること、ユーザによって入力された注釈を観測すること、PassBook(登録商標)、Google(登録商標) Wallet、Paypal(登録商標)、および他の類似のアプリケーションまたはサービスの使用に関するイベントを観測することなどが含まれ得る。アプリケーションレベルの観測には、仮想プライベートネットワーク(VPN)の使用に関するイベント、および同期、音声探索、音声制御(たとえば、1語を発することによる電話のロック/アンロック)、言語翻訳機、計算用のデータのオフローディング、ビデオストリーミング、ユーザ活動なしでのカメラの使用、ユーザ活動なしでのマイクロフォンの使用などに関するイベントを観測することも含まれ得る。

【００８２】

無線レベルの観測には、無線通信リンクを確立するかまたは情報を送信する前のモバイルコンピューティングデバイス102とのユーザの対話、デュアル/マルチ加入者識別モジュール(SIM)カード、インターネット無線、モバイルフォンテザリング、計算のためのデータのオフロード、デバイス状態通信、ゲームコントローラまたはホームコントローラとしての使用、車両通信、コンピューティングデバイス同期などのうちのいずれかまたは複数の存在、実在、または量を決定することが含まれ得る。無線レベルの観測にはまた、測位、ピアツーピア(p2p)通信、同期、車両対車両通信、および/または機械対機械(m2m)のための、無線(WiFi、WiMax、Bluetooth（登録商標）など)の使用を監視することも含まれ得る。無線レベルの観測には、ネットワークトラフィックの使用、統計データ、またはプロファイルを監視することがさらに含まれ得る。

【００８３】

センサレベルの観測には、モバイルコンピューティングデバイス102の使用環境および/または外部環境を判断するために、磁気センサまたは他のセンサを監視することが含まれ得る。たとえば、コンピューティングデバイスプロセッサは、デバイスが(たとえば、ホルスタ内の磁石を検知するように構成された磁石センサを介して)ホルスタ内にあるか、または(たとえば、カメラもしくは光センサによって検出される光の量を介して)ユーザのポケット内にあるかを判断するように構成される場合がある。たとえば、モバイルコンピューティングデバイス102がホルスタに入れられている間に発生する、ユーザによるアクティブな使用(たとえば、写真またはビデオを撮ること、メッセージを送ること、音声通話を行うこと、音を録音することなど)に関連する活動および機能は、(たとえば、ユーザを追跡またはスパイするために)デバイス上で実行している不正なプロセスのサインである可能性があるので、モバイルコンピューティングデバイス102がホルスタ内にあることを検出することは、疑わしい挙動を認識することに関係する場合がある。

【００８４】

使用環境または外部環境に関連するセンサレベルの観測の他の例には、NFCシグナリングを検出すること、クレジットカードスキャナ、バーコードスキャナ、またはモバイルタグリーダから情報を収集すること、ユニバーサルシリアルバス(USB)電力充電源の存在を検出すること、キーボードまたは補助デバイスがモバイルコンピューティングデバイス102に結合されていることを検出すること、モバイルコンピューティングデバイス102が(たとえば、USBなどを介して)別のコンピューティングデバイスに結合されていることを検出すること、LED、フラッシュ、フラッシュライト、または光源が変更または(たとえば、緊急シグナリングアプリケーションなどを、悪意をもって無効にして)無効化されているかどうかを決定すること、スピーカまたはマイクロフォンがオンにされているかまたは電源投入されていることを検出すること、充電または電力供給イベントを検出すること、モバイルコンピューティングデバイス102がゲームコントローラとして使用されていることを検出することなどが含まれ得る。センサレベルの観測には、医療もしくはヘルスケアのセンサから、またはユーザの体をスキャンすることから情報を収集すること、USB/オーディオジャックに差し込まれた外部センサから情報を収集すること、(たとえば、振動インターフェースなどを介して)触知センサまたは触覚センサから情報を収集すること、モバイルコンピューティングデバイス102の熱状態に関する情報を収集することなども含まれ得る。

【００８５】

監視される要因の数を管理可能レベルまで削減するために、ある実施形態では、挙動観測器モジュール202は、コンピューティングデバイスの劣化に寄与する可能性があるすべての要因の小さいサブセットである、挙動または要因の初期セットを監視/観測することによって、粗い観測を実施するように構成され得る。ある実施形態では、挙動観測器モジュール202は、挙動および/または要因の初期セットをサーバおよび/またはクラウドサービスもしくはネットワーク内の構成要素から受信することができる。ある実施形態では、挙動/要因の初期セットは、機械学習分類器モデル内で指定され得る。

【００８６】

各分類器モデルは、コンピューティングデバイスの挙動の特定の特徴または実施形態を評価するために、コンピューティングデバイスプロセッサによって使用され得る、データおよび/または情報構造(たとえば、特徴ベクトル、挙動ベクトル、構成要素リストなど)を含む挙動モデルであり得る。各分類器モデルはまた、コンピューティングデバイス内のいくつかの特徴、要因、データ点、エントリ、API、状態、条件、挙動、アプリケーション、プロセス、動作、構成要素など(本明細書では総称して「特徴」)を監視するための判定基準を含み得る。分類器モデルは、コンピューティングデバイスにプレインストールされてよく、ネットワークサーバからダウンロードもしくは受信されてよく、コンピューティングデバイス内で生成されてよく、またはそれらの任意の組合せであってよい。分類器モデルは、クラウドソーシングソリューション、挙動モデル化技法、機械学習アルゴリズムなどを使用することによって、生成され得る。

【００８７】

各分類器モデルは、完全な分類器モデルまたは簡潔な分類器モデルとして類別され得る。完全な分類器モデルは、数千の特徴および数十億のエントリを含み得る大きいトレーニングデータセットに応じて生成される、ロバストなデータモデルであり得る。簡潔な分類器モデルは、特定の活動が進行中のクリティカル活動であるかどうか、および/または特定のモバイルデバイスの挙動が良性ではないかどうかを決定することに最も関連がある特徴/エントリのみを含む/テストする、縮小されたデータセットから生成された、より専心的なデータモデルであり得る。一例として、デバイスプロセッサは、ネットワークサーバから完全な分類器モデルを受信し、完全な分類器に基づいて、コンピューティングデバイス内で学習分類器モデルを生成し、デバイスの挙動を良性または非良性(すなわち、悪意がある、性能を劣化させるなど)のいずれかであるとして分類するために、ローカルに生成された学習分類器モデルを使用するように構成され得る。

【００８８】

ローカルに生成される簡潔な分類器モデルは、コンピューティングデバイスにおいて生成される簡潔な分類器モデルである。すなわち、現代のコンピューティングデバイス(たとえば、モバイルデバイスなど)は非常に構成可能かつ複雑なシステムであるため、特定のデバイス挙動が非良性(たとえば、悪意がある、または性能劣化である)かどうかを決定するために最も重要な特性は各デバイスにおいて異なり得る。さらに、特徴の異なる組合せは、特定の挙動が非良性であるかどうかをそのデバイスが迅速かつ効率的に決定するために、各デバイスにおける監視および/または分析を必要とすることがある。しかし、監視および分析を必要とする特徴の的確な組合せ、ならびに各特徴または特徴の組合せの相対的な優先順位または重要性は、しばしば、それにおいて挙動が監視または分析されるべきである特定のデバイスから取得された情報を使用してのみ決定され得る。これらのおよび他の理由で、様々な実施形態は、モデルが使用されるコンピューティングデバイス内に分類器モデルを生成することができる。これらのローカル分類器モデルは、そのデバイスプロセッサが、その特定のデバイス上の挙動が非良性である(デバイスの性能の劣化に寄与している)かを決定する際に最も重要である特定の特徴を正確に特定することを可能にする。ローカル分類器モデルはまた、デバイスプロセッサが、その特定のデバイス内の挙動を分類するためのその相対的な重要性に従ってテストまたは評価される特徴を優先順位付けすることを可能にする。

【００８９】

デバイス固有の分類器モデルは、特定のコンピューティングデバイス内で活動または挙動を分類することに最も関連があると決定される、コンピューティングデバイス固有の特徴/エントリのみを含む/テストする専心的なデータモデルを含む分類器モデルである。アプリケーション固有の分類器モデルは、特定のソフトウェアアプリケーションを評価することに最も関連がある特徴/エントリのみを含む/テストする専心的なデータモデルを含む分類器モデルである。コンピューティングデバイスにおいてアプリケーション固有の分類器モデルをローカルに動的に生成することによって、様々な実施形態は、デバイスプロセッサが、特定のソフトウェアアプリケーションの動作がそのデバイスの望ましくない挙動

または性能を劣化させる挙動に寄与しているかどうかを決定するのに最も重要な少数の特徴に対する監視動作および分析動作に専念することを可能にする。

【００９０】

マルチアプリケーション分類器モデルは、2つ以上の特定のソフトウェアアプリケーション(または、特定のタイプのソフトウェアアプリケーション)の集合的挙動が非良性であるかどうかを決定することに最も関連がある特徴/エントリに対するテストを含みまたは優先順位付ける、専心的なデータモデルを含むローカル分類器モデルであり得る。マルチアプリケーション分類器モデルは、特徴のアグリゲートされたセットをテスト/評価するアグリゲートされた特徴セットおよび/または判定ノードを含み得る。デバイスプロセッサは、コンピューティングデバイス上で動作する2つ以上のソフトウェアアプリケーション間の関係、対話、および/または通信を特定することに最も関連があるデバイス特徴を特定し、特定されたデバイス特徴のうちの1つを評価するテスト条件を特定し、特定されたテスト条件の優先順位、重要性、または成功率を決定し、その重要性または成功率に従って、特定されたテスト条件を優先順位付けまたは順序付け、その決定された優先順位、重要性、または成功率に従ってそれらが順序付けられるように、特定されたテスト条件を含めるように分類器モデルを生成することによって、マルチアプリケーション分類器モデルを生成するように構成され得る。デバイスプロセッサは、2つ以上のアプリケーション特定分類器モデルを組み合わせることによって、マルチアプリケーション分類器モデルを生成するように構成され得る。

【００９１】

様々な実施形態では、デバイスプロセッサは、2つ以上のアプリケーションが協働して結託または動作している、またはアプリケーションがグループとして一緒に分析されるべきであると決定することに応答して、マルチアプリケーション分類器モデルを生成するように構成され得る。デバイスプロセッサは、アプリケーションの各特定されたグループまたはクラスに関してマルチアプリケーション分類器モデルを生成するように構成され得る。しかしながら、すべてのグループを分析することは、デバイスの限定された、かなりのリソース量を消費する可能性がある。したがって、ある実施形態では、デバイスプロセッサは、(たとえば、他のアプリケーションとのその対話などに基づいて)アプリケーションが結託挙動に関与している確率を決定し、結託挙動の高い確率があるソフトウェアアプリケーションを含むグループだけに関して分類器モデルを知的に生成するように構成され得る。

【００９２】

挙動分析器モジュール208は、監視された活動(または、挙動)が良性であるか、または非良性であるかを決定するために、挙動抽出器モジュール204によって生成された挙動ベクトルを分類器モデルに適用するように構成され得る。ある実施形態では、その挙動分析動作の結果が良性または非良性のいずれかとして挙動を分類するために十分な情報を提供しないとき、挙動分析器モジュール208は「疑いがある」として挙動を分類することができる。

【００９３】

挙動分析器モジュール208は、結託しているソフトウェアアプリケーションを特定すること、いくつかのアプリケーションをグループとして評価するべきであると決定することに応答して、かつ/または監視された活動または挙動が疑わしいと決定することに応答して、挙動観測器モジュール202に通知するように構成され得る。それに応答して、挙動観測器モジュール202は、その観測の粒度(すなわち、コンピューティングデバイスの特徴が監視される際の詳細度のレベル)を調節し、かつ/または、挙動分析器モジュール208から受信された情報(たとえば、リアルタイム分析動作の結果)に基づいて監視されるアプリケーション/要因/挙動を変更し、新たな情報または追加の挙動情報を生成または収集し、さらなる分析/分類のために新たな/追加の情報を挙動分析器モジュール208に送ることができる。

【００９４】

挙動観測器モジュール202と挙動分析器モジュール208との間のそのようなフィードバック通信により、集合的挙動が良性または非良性として分類されるまで、疑わしいかもしくは性能を劣化させる挙動の根源が特定されるまで、処理もしくはバッテリの消費しきい値に到達するまで、または、観測の細分性のさらなる変化、調整、または向上から、疑わしいかもしくは性能を劣化させる挙動の根源が特定され得ないとデバイスプロセッサが決定するまで、モバイルコンピューティングデバイス102が観測の細分性を繰り返し向上させる(すなわち、より微細にまたはより詳細に観測する)こと、または観測される特徴/挙動を変更することが可能になる。そのようなフィードバック通信により、モバイルコンピューティングデバイス102が、コンピューティングデバイスの過剰な量の処理リソース、メモリリソース、またはエネルギーリソースを消費することなく、挙動ベクトルおよび分類器モデルを調整または修正することも可能になる。

【００９５】

挙動観測器モジュール202および挙動分析器モジュール208は、限られた粗い観測結果から疑わしい挙動を特定するため、挙動を動的に決定してより詳細に観測するため、および観測のために必要な詳細さのレベルを動的に決定するために、コンピューティングシステムの挙動のリアルタイムの挙動分析を個別にまたは集合的に提供することができる。これは、デバイス上で大量のプロセッサリソース、メモリリソース、またはバッテリリソースの必要なく、モバイルコンピューティングデバイス102が問題を効率的に特定し、問題を防止することを可能にする。

【００９６】

様々な実施形態では、モバイルコンピューティングデバイス102のデバイスプロセッサは、綿密な監視を必要とするクリティカルデータリソースを特定し、そのクリティカルデータリソースにアクセスするとき、ソフトウェアアプリケーションによって行われるAPI呼出しを(たとえば、挙動観測器モジュール202を介して)監視し、API呼出しのパターンを2つ以上のソフトウェアアプリケーションによる非良性挙動を示すと特定し、API呼出しの特定されたパターンとリソース使用とに基づいて、挙動ベクトルを生成し、(たとえば、挙動分析器モジュール208を介して)挙動分析動作を実行するために挙動ベクトルを使用し、挙動分析動作に基づいて、ソフトウェアアプリケーションのうちの1つまたは複数が非良性であるかどうかを決定するように構成され得る。

【００９７】

ある実施形態では、デバイスプロセッサは、コンピューティングデバイス上で動作するソフトウェアアプリケーションによって最も頻繁に使用されるAPIを特定し、特定されたホットAPIの使用に関する情報をデバイスのメモリ内のAPIログ内に記憶し、非良性挙動を特定するために、APIログ内に記憶された情報に基づいて挙動分析を実行するように構成され得る。

【００９８】

様々な実施形態では、モバイルコンピューティングデバイス102は、活動または挙動が非良性であるかどうかを決定することに最も関連のある特徴、要因、およびデータ点を知的にかつ効率的に特定するために、ネットワークサーバとともに動作するように構成され得る。たとえば、デバイスプロセッサは、ネットワークサーバから完全な分類器モデルを受信し、コンピューティングデバイスまたはデバイス上で動作するソフトウェアアプリケーションの特徴および機能性に固有である簡潔な分類器モデル(すなわち、データ/挙動モデル)を生成するために、受信された完全な分類器モデルを使用するように構成され得る。デバイスプロセッサは、様々なレベルの複雑さ(または「簡潔さ」)の簡潔な分類器モデルのファミリーを生成するために、完全な分類器モデルを使用することができる。簡潔な分類器モデルの最も簡潔なファミリー(すなわち、最も少数のテスト条件に基づく簡潔な分類器モデル)は、モデルが良性または非良性のいずれかとして類別することができない(したがって、モデルによって疑わしいものとして類別される)挙動に遭遇するまで、ルーチン的に適用されてよく、遭遇した時点で、よりロバストな(すなわち、より簡潔ではない)簡潔な分類器モデルが、挙動を類別する試みにおいて適用され得る。生成された簡潔

な分類器モデルのファミリー内の一層ロバストな簡潔な分類器モデルの適用は、挙動の最終的な分類が達成されるまで適用され得る。このようにして、デバイスプロセッサは、挙動を最終的に分類するためにロバストな分類器モデルが必要とされる状況に、最も完全だがリソース集約的な簡潔な分類器モデルの使用を制限することによって、効率と精度との間でバランスをとることができる。

【００９９】

様々な実施形態では、デバイスプロセッサは、完全な分類器モデル内に含まれた有限状態機械表示/表現をブーストされた判定株に変換することによって、簡潔な分類器モデルを生成するように構成され得る。デバイスプロセッサは、デバイス固有の特徴、条件、または構成に基づいて、ブーストされた判定株の完全セットを剪定するかまたは選別して、完全な分類器モデル内に含まれたブーストされた判定株のサブセットを含む分類器モデルを生成することができる。デバイスプロセッサは、次いで、コンピューティングデバイス挙動を知的に監視、分析、および/または分類するために、簡潔な分類器モデルを使用することができる。

【０１００】

ブーストされた判定株は、厳密に1つのノード(および、したがって、1つのテスト質問またはテスト条件)と重み値とを有し、したがって、データ/挙動の二項分類における使用に十分に適した1レベル判定木である。すなわち、挙動ベクトルをブーストされた判定株に適用することで、2値の回答(たとえば、YesまたはNo)がもたらされる。たとえば、ブーストされた判定株によってテストされる質問/条件が「ショートメッセージサービス(SMS)送信の頻度が毎分x回未満であるか」である場合、「3」という値をブーストされた判定株に適用することで、(「3回未満」のSMS送信に対して)「yes」の回答または(「3回以上」のSMS送信に対して)「no」の回答のいずれかがもたらされる。

【０１０１】

ブーストされた判定株は、非常に簡単かつ根本的である(したがって、著しい処理リソースを必要としない)ので効率的である。ブーストされた判定株はまた非常に並列化可能であり、したがって、(たとえば、コンピューティングデバイス内の複数のコアまたはプロセッサによって)多くの株が並列に/同時に適用またはテストされ得る。

【０１０２】

ある実施形態では、デバイスプロセッサは、完全な分類器モデルに含まれる分類器基準のサブセット、ならびに、コンピューティングデバイスの構成、機能性、および接続される/含まれるハードウェアに関連のある特徴に対応する分類器基準のみを含む簡潔な分類器モデルを生成するように構成され得る。デバイスプロセッサは、デバイスに存在する、または関連のある特徴および機能のみを監視するために、この簡潔な分類器モデルを使用することができる。次いで、デバイスプロセッサは、コンピューティングデバイスの現在の状態および構成に基づいて、様々な特徴および対応する分類器基準を含むように、または削除するように、簡潔な分類器モデルを定期的に変更または再生成することができる。

【０１０３】

一例として、デバイスプロセッサは、挙動モデル(たとえば、分類器)の完全な特徴セットに関連付けられた判定株を含む、大きいブーストされた判定株の分類器モデルを受信し、コンピューティングデバイスの現在の構成、機能性、動作状態、および/または接続される/含まれるハードウェアに関連のある特徴のみを大きい分類器モデルから選択すること、ならびに、選択された特徴に対応するブーストされた判定株のサブセットを簡潔な分類器モデル内に含めることによって、大きい分類器モデルから1つまたは複数の簡潔な分類器モデルを導出するように構成され得る。この実施形態では、コンピューティングデバイスに関連のある特徴に対応する分類器基準は、選択された特徴の少なくとも1つをテストする大きい分類器モデルに含まれるブーストされた判定株であり得る。次いで、デバイスプロセッサは、コンピューティングデバイスの現在の状態および構成に基づいて様々な特徴を含むように、または削除するように、ブーストされた判定株の簡潔な分類器モデルを定期的に変更または再生成することができ、その結果、簡潔な分類器モデルは、アプリ

ケーション固有またはデバイス固有の特徴のブーストされた判定株を含め続ける。

【０１０４】

さらに、デバイスプロセッサはまた、特定のソフトウェアアプリケーション(Google(登録商標) walletおよびeTrade(登録商標))に、および/または特定のタイプのソフトウェアアプリケーション(たとえば、ゲーム、ナビゲーション、金融、ニュース、生産性など)に関連のある条件または特徴を特定するアプリケーション固有の分類器モデルを動的に生成することもできる。これらの分類器モデルは、完全な分類器モデル内に含まれる判定ノードの(または、受信された完全な分類器モデルから生成されたより簡潔な分類器モデル内に含まれる判定ノードの)低減された、およびより専心的なサブセットを含むように生成され得る。これらの分類モデルを組み合わせて、マルチアプリケーション分類器モデルを生成することができる。

【０１０５】

様々な実施形態では、デバイスプロセッサは、システム内の各ソフトウェアアプリケーションのための、および/またはシステム内のソフトウェアアプリケーションの各タイプのための、アプリケーションベースの分類器モデルを生成するように構成され得る。デバイスプロセッサはまた、危険性が高い、または乱用されやすいソフトウェアアプリケーションおよび/またはアプリケーションタイプ(たとえば、金融アプリケーション、ポイントオブセールアプリケーション、生体センサアプリケーションなど)を動的に特定し、危険性が高い、または乱用されやすいものとして特定されるソフトウェアアプリケーションおよび/またはアプリケーションタイプのみのためのアプリケーションベースの分類器モデルを生成するように構成され得る。様々な実施形態では、デバイスプロセッサは、動的に、反応的に、前もって、および/または新しいアプリケーションがインストールもしくは更新されるたびに、アプリケーションベースの分類器モデルを生成するように構成され得る。

【０１０６】

一般に、各ソフトウェアアプリケーションは、コンピューティングデバイス上でいくつかのタスクまたは活動を実行する。いくつかのタスク/活動がコンピューティングデバイスにおいて実行される特定の実行状態は、挙動または活動が追加のまたはより綿密な精査、監視、および/または分析に値するか否かの強い指標であり得る。したがって、様々な実施形態では、デバイスプロセッサは、それにおいていくつかのタスク/活動が実行される実際の実行状態を特定する情報を使用して、その挙動監視および分析動作に専心し、活動がクリティカル活動であるか、および/または活動が非良性であるかどうかをより良く決定するように構成され得る。

【０１０７】

様々な実施形態では、デバイスプロセッサは、ソフトウェアアプリケーションによって実行された活動/タスクを、それにおいてそれらの活動/タスクが実行された実行状態に関連付けるように構成され得る。たとえば、デバイスプロセッサは、実行状態が関連のあるソフトウェアの特徴、活動、または動作(たとえば、ロケーションアクセス、SMS読取り動作、センサアクセスなど)をリストするサブベクトルまたはデータ構造において、計測された構成要素を監視することから収集された挙動情報を含む挙動ベクトルを生成するように構成され得る。ある実施形態では、このサブベクトル/データ構造は、それにおいて各特徴/活動/動作が観測された実行状態を特定する影の特徴値サブベクトル/データ構造に関連して記憶され得る。一例として、デバイスプロセッサは、その値が、ソフトウェアアプリケーションがバックグラウンド状態において動作中であったときに、ソフトウェアアプリケーションがロケーション情報にアクセスした数または割合を特定する、「location_background」データフィールドを含む挙動ベクトルを生成し得る。これによって、デバイスプロセッサが、コンピューティングデバイスの他の観測された/監視された活動とは無関係に、および/またはそれと並列に、この実行状態情報を分析することが可能になる。このようにして挙動ベクトルを生成することで、システムが情報(たとえば、頻度または割合)を経時的に統合することも可能になる。

【０１０８】
様々な実施形態では、デバイスプロセッサは、監視された活動に関してクエリに対する回答を生成するために、機械学習分類器における判定ノードに入力され得る情報を含むために挙動ベクトルを生成するように構成され得る。
【０１０９】
様々な実施形態では、デバイスプロセッサは、実行情報を含めるように挙動ベクトルを生成するように構成され得る。実行情報は、挙動の一部(たとえば、バックグラウンドプロセスによって3秒間に5回使用されたカメラ、フォアグラウンドプロセスによって3秒間に3回使用されたカメラなど)として、または無関係の特徴の一部として、挙動ベクトル内に含まれ得る。ある実施形態では、実行状態情報は、影の特徴値サブベクトルまたはデータ構造として、挙動ベクトル内に含まれ得る。ある実施形態では、挙動ベクトルは、実行状態が関連のある特徴、活動、タスクに関連して、影の特徴値サブベクトル/データ構造を記憶し得る。
【０１１０】
図3は、ある実施形態による、2つ以上のソフトウェアアプリケーションの集合的挙動を評価するための挙動分析技法を使用する方法300を示す。方法300は、モバイルコンピューティングデバイスまたはリソース制約のあるコンピューティングデバイスの処理コア内で実行することができる。
【０１１１】
ブロック302で、処理コアは、デバイス上で動作するソフトウェアアプリケーションの活動を監視することができる。ブロック304で、処理コアは、監視された活動から挙動情報を収集することができる。ブロック306で、処理コアは、収集された挙動情報に基づいて挙動ベクトルを生成することができる。ブロック308で、処理コアは、分析情報を生成するために挙動ベクトルを分類器モデル(または分類器モデルのファミリー)に適用することができる。ブロック310で、処理コアは、ソフトウェアアプリケーション間の関係を特定するために分析情報を使用することができる。ブロック312で、処理コアは、特定された関係に基づいて、グループとして一緒に評価されるべきソフトウェアアプリケーションを特定することができる。ブロック314で、処理コアは、特定されたソフトウェアアプリケーションの挙動情報および/または分析結果をアグリゲートすることができる。ブロック316で、処理コアは、ソフトウェアアプリケーションの集合的挙動が良性であるか、または非良性であるかを決定するために、アグリゲートされた分析結果を使用することができる。
【０１１２】
図4は、ある実施形態による、ソフトウェアアプリケーション間の関係を決定するための挙動分析技法を使用する方法400を示す。方法400は、モバイルコンピューティングデバイスまたはリソース制約のあるコンピューティングデバイスの処理コア内で実行することができる。ブロック402で、処理コアは、コンピュータデバイス上で動作するソフトウェアアプリケーション間の対話を監視することができる。ブロック404で、処理コアは、ソフトウェアアプリケーション間の関係を特徴付ける挙動ベクトルを生成することができる。ブロック406で、処理コアは、分析情報を生成するために挙動ベクトルを分類器モデル(または分類器モデルのファミリー)に適用することができる。ブロック408で、処理コアは、アプリケーションが協働して結託または動作しているかどうかなど、アプリケーション間の関係の性質を決定するために分析情報を使用することができる。
【０１１３】
図5は、ある実施形態による、特定されたアプリケーションの集合的挙動が非良性であるかどうかを決定するための挙動分析技法を使用する方法500を示す。方法500は、モバイルコンピューティングデバイスまたはリソース制約のあるコンピューティングデバイスの処理コア内で実行することができる。ブロック502で、処理コアは、グループとして一緒に分析されるべきソフトウェアアプリケーション(たとえば、結託しているアプリケーション)を特定することができる。ブロック504で、処理コアは、特定されたアプリケーショ

ンの挙動ベクトルを分類器モデル(または、分類器モデルのファミリー)に適用することができる。ブロック506で、処理コアは、挙動ベクトルの各アプリケーションによって生成された分析情報を分類器モデルにアグリゲートすることができる。ブロック508で、処理コアは、特定されたアプリケーションの集合的挙動が非良性であるかどうかを決定するために、アグリゲートされた分析情報を使用することができる。

【０１１４】

図6は、別の実施形態による、特定されたアプリケーションの集合的挙動が非良性であるかどうかを決定するための挙動分析技法を使用する方法600を示す。方法600は、モバイルコンピューティングデバイスまたはリソース制約のあるコンピューティングデバイスの処理コア内で実行することができる。ブロック602で、処理コアは、グループとして一緒に分析されるべきソフトウェアアプリケーションを特定することができる。ブロック604で、処理コアは、特定されたアプリケーションの活動を監視することができる。ブロック606で、処理コアは、監視された活動の各々に関する挙動動情報を収集することができる。ブロック608で、処理コアは、収集された挙動情報に基づいて、特定されたアプリケーションの集合的挙動を特徴付ける挙動ベクトルを生成することができる。ブロック610で、処理コアは、分析情報を生成するために、生成された挙動ベクトルを分類器モデル(または、分類器モデルのファミリー)に適用することができる。ブロック612で、処理コアは、特定されたアプリケーションの集合的挙動が非良性であるかどうかを決定するために、分析情報を使用することができる。

【０１１５】

図7は、コンピューティングデバイスの挙動を分類するための簡潔な分類器モデルのファミリーを使用する実施形態の方法700を示す。方法700は、モバイルコンピューティングデバイスまたはリソース制約のあるコンピューティングデバイスの処理コアによって実行することができる。

【０１１６】

ブロック702で、処理コアは、観測を実行して、コンピューティングデバイスシステムの様々なレベルにおいて計測された様々な構成要素から挙動情報を収集することができる。ある実施形態では、これは、図2を参照して上で論じた挙動観測器モジュール202を介して達成され得る。ブロック704で、処理コアは、観測結果、収集された挙動情報、および/またはコンピューティングデバイスの挙動を特徴付ける、挙動ベクトルを生成することができる。また、ブロック704で、処理コアは、簡潔な分類器モデル、または複雑さ(または「簡潔さ」)のレベルが様々である簡潔な分類器モデルのファミリーを生成するために、ネットワークサーバから受信された完全な分類器モデルを使用することができる。これを達成するために、処理コアは、完全な分類器モデル内に含まれるブーストされた判定株のファミリーを選別して、減らされた数のブーストされた判定株を含む、かつ/または限られた数のテスト条件を評価する、簡潔な分類器モデルを生成することができる。

【０１１７】

ブロック706で、処理コアは、コンピューティングデバイスによってまだ評価または適用されていない簡潔な分類器モデルのファミリー内で最も簡潔な分類器(すなわち、最少の数の異なるコンピューティングデバイスの状態、特徴、挙動、または条件に基づくモデル)を選択することができる。ある実施形態では、これは、処理コアが分類器モデルの順位付けられたリスト内の最初の分類器モデルを選択することによって達成され得る。

【０１１８】

ブロック708で、処理コアは、収集された挙動情報または挙動ベクトルを選択された簡潔な分類器モデル内の各々のブーストされた判定株に適用することができる。ブーストされた判定株は2値の判定であり、簡潔な分類器モデルは同じテスト条件に基づく多くの2値の判定を選択することによって生成されるので、簡潔な分類器モデル内のブーストされた判定株に挙動ベクトルを適用する処理は、並列動作で実行され得る。代替的に、ブロック530で適用される挙動ベクトルは、簡潔な分類器モデル内に含まれる限られた数のテスト条件パラメータをちょうど含むように切り捨てられてよく、またはフィルタリングされて

よく、それによってモデルを適用する際の計算量はさらに低減される。

【０１１９】

ブロック710で、処理コアは、収集された挙動情報を簡潔な分類器モデル内の各々のブーストされた判定株に適用した結果の加重平均を計算または決定することができる。ブロック712で、処理コアは、計算された加重平均としきい値とを比較することができる。決定ブロック714で、処理コアは、この比較の結果、および/または選択された簡潔な分類器モデルを適用することによって生成された結果が疑わしいかどうかを決定することができる。たとえば、処理コアは、これらの結果が、高い信頼度で悪意があるまたは良性のいずれかであるとして挙動を分類するために使用され得るかどうかを決定することができ、そうでない場合、挙動を疑わしいものとして扱う。

【０１２０】

結果が疑わしいと処理コアが決定する(たとえば、決定ブロック714=「Yes」の)場合、処理コアは、ブロック706～712における動作を繰り返して、挙動が高い信頼度で悪意があるまたは良性であるとして分類されるまで、より多くのデバイスの条件、特徴、挙動、または条件を評価するより強い(すなわち、より簡潔ではない)分類器モデルを選択および適用することができる。挙動が高い信頼度で悪意があるまたは良性のいずれかであるとして分類され得ると決定することなどによって、結果が疑わしくない(たとえば、決定ブロック714=「No」)と処理コアが決定する場合、ブロック716において、処理コアは、良性であるまたは潜在的に悪意があるとしてモバイルデバイスの挙動を分類するために、ブロック712において生成された比較の結果を使用することができる。

【０１２１】

ある代替的な実施形態の方法では、上で説明した動作は、簡潔な分類器モデル内にまだ存在しないブーストされた判定株を順次選択し、選択された判定株と同じコンピューティングデバイスの状態、特徴、挙動または条件に依存する(および、したがって、1つの決定結果に基づいて適用され得る)すべての他のブーストされた判定株を特定し、同じコンピューティングデバイスの状態、特徴、挙動または条件に依存する選択されたすべての特定された他のブーストされた判定株を簡潔な分類器モデルに含め、テスト条件の決定された数に等しい回数、プロセスを繰り返すことによって、達成され得る。選択されたブーストされた判定株と同じテスト条件に依存するすべてのブーストされた判定株が毎回簡潔な分類器モデルに追加されるので、このプロセスが実行される回数を限定することは、簡潔な分類器モデルに含まれるテスト条件の数を限定することになる。

【０１２２】

図8は、様々な実施形態に従って使用するのに適した、ブーストされた判定木/分類器を生成するのに適した例示的なブースト方法800を示す。ブロック802で、プロセッサは、判定木/分類器を生成および/または実行し、判定木/分類器の実行によってトレーニングサンプルを収集し、トレーニングサンプルに基づいて新しい分類器モデル(h1(x))を生成することができる。トレーニングサンプルは、コンピューティングデバイスの挙動、ソフトウェアアプリケーション、またはコンピューティングデバイスにおけるプロセスの、以前の観測または分析から収集された情報を含み得る。トレーニングサンプルおよび/または新しい分類器モデル(h1(x))は、以前の分類器に含まれる質問またはテスト条件のタイプに基づいて、および/または、挙動分析器モジュール208の分類器モデル内の以前のデータ/挙動モデルまたは分類器の実行/適用から収集された正確さまたは性能の特性に基づいて生成され得る。ブロック804で、プロセッサは、第2の新しい木/分類器(h2(x))を生成するために、生成された判定木/分類器(h1(x))によって誤分類されたエントリの重みをブーストする(または増大させる)ことができる。ある実施形態では、トレーニングサンプルおよび/または新しい分類器モデル(h2(x))は、分類器の以前の実行または使用(h1(x))の誤り率に基づいて生成され得る。ある実施形態では、トレーニングサンプルおよび/または新しい分類器モデル(h2(x))は、分類器の以前の実行または使用においてデータ点の誤り率または誤分類に寄与したと判定される属性に基づいて生成され得る。

【０１２３】

ある実施形態では、誤分類されたエントリは、それらの相対的な正確さまたは有効性に基づいて重み付けられ得る。ブロック806で、プロセッサは、第3の新しい木/分類器(h3(x))を生成するために、生成された第2の木/分類器(h2(x))によって誤分類されたエントリの重みをブーストする(または増大させる)ことができる。ブロック808で、ブロック804～806の動作が、数「t」の新しい木/分類器($h_t(x)$)を生成するために繰り返され得る。

【０１２４】

第1の判定木/分類器(h1(x))によって誤分類されたエントリの重みをブーストし、または増大させることによって、第2の木/分類器(h2(x))は、第1の判定木/分類器(h1(x))によって誤分類されたエントリをより正確に分類することができるが、第1の判定木/分類器(h1(x))によって正しく分類されたエントリのいくつかを誤分類することもある。同様に、第3の木/分類器(h3(x))は、第2の判定木/分類器(h2(x))によって誤分類されたエントリをより正確に分類することができるが、第2の判定木/分類器(h2(x))によって正しく分類されたエントリのいくつかを誤分類することがある。すなわち、木/分類器のファミリーh1(x)～$h_t(x)$を生成することは、全体として収束するシステムをもたらさないことがあり、並列に実行され得るいくつかの判定木/分類器をもたらす。

【０１２５】

図9は、ある実施形態による、動的で適応的な観測を実行するように構成されたコンピューティングシステムの挙動観測器モジュール202における例示的な論理構成要素および情報フローを示す。挙動観測器モジュール202は、適応フィルタモジュール902、スロットルモジュール904、観測器モードモジュール906、高レベル挙動検出モジュール908、挙動ベクトル生成器910、およびセキュアバッファ912を含み得る。高レベル挙動検出モジュール908は、空間相関モジュール914と時間相関モジュール916とを含み得る。

【０１２６】

観測器モードモジュール906は、分析器ユニット(たとえば、図2を参照して上で説明した挙動分析器モジュール208)および/またはアプリケーションAPIを含み得る様々なソースから制御情報を受信することができる。観測器モードモジュール906は、様々な観測器モードに関する制御情報を適応フィルタモジュール902および高レベル挙動検出モジュール908に送ることができる。

【０１２７】

適応フィルタモジュール902は、複数のソースからデータ/情報を受信し、受信された情報をインテリジェントにフィルタリングし、受信された情報から選択された情報のより小さなサブセットを生成することができる。このフィルタは、分析器モジュールから、またはAPIを通じて通信するより高レベルのプロセスから受信された、情報または制御に基づいて適合され得る。フィルタリングされた情報は、スロットルモジュール904に送られてよく、スロットルモジュール904は、高レベル挙動検出モジュール908が要求または情報であふれないことまたは過負荷にならないことを確実にするために、フィルタから流れる情報の量を制御することを担い得る。

【０１２８】

高レベル挙動検出モジュール908は、スロットルモジュール904からのデータ/情報と、観測器モードモジュール906からの制御情報と、コンピューティングデバイスの他のコンポーネントからの状況情報とを受信することができる。高レベル挙動検出モジュール908は、受信された情報を使用して空間相関および時間相関を実行し、高レベルの挙動を検出または特定することができ、このことは、デバイスに準最適なレベルで実行させ得る。空間相関および時間相関の結果は、挙動ベクトル生成器910へ送られてよく、挙動ベクトル生成器910は、相関情報を受信し、特定のプロセス、アプリケーション、またはサブシステムの挙動を記述する挙動ベクトルを生成することができる。ある実施形態では、挙動ベクトル生成器910は、特定のプロセス、アプリケーション、またはサブシステムの各々の高レベルの挙動が挙動ベクトルの要素であるように、挙動ベクトルを生成することができる。ある実施形態では、生成された挙動ベクトルは、セキュアバッファ912内に記憶され得る。高レベル挙動検出の例は、特定のイベントの存在、別のイベントの量または頻度、

複数のイベント間の関係、イベントが発生する順序、いくつかのイベントの発生の時間差などの検出を含み得る。

【０１２９】

様々な実施形態では、挙動オブザーバモジュール202は、適応観測を実行し、観測の粒度を制御することができる。すなわち、挙動観測器モジュール202は、観測されるべき関連する挙動を動的に特定し、特定された挙動が観測されるべき詳細さのレベルを動的に決定することができる。このようにして、挙動観測器モジュール202は、様々なレベル(たとえば、複数の粗いレベルおよび微細なレベル)においてシステムがコンピューティングデバイスの挙動を監視することを可能にする。挙動観測器モジュール202は、観測されているものにシステムが適応することを可能にし得る。挙動観測器モジュール202は、多様なソースから取得され得る情報の専心的なサブセットに基づいて観測されている要因/挙動をシステムが動的に変更することを可能にし得る。

【０１３０】

上で論じたように、挙動観測器モジュール202は、適応的観測技法を実行し、様々なソースから受信された情報に基づいて観測の粒度を制御することができる。たとえば、高レベル挙動検出モジュール908は、スロットルモジュール904、観測器モードモジュール906からの情報と、コンピューティングデバイスの他の構成要素(たとえば、センサ)から受信された状況情報とを受信することができる。ある例として、時間相関を実行する高レベル挙動検出モジュール908は、カメラが使用されたことと、コンピューティングデバイスが写真をサーバへアップロードしようとしていることとを検出し得る。高レベル挙動検出モジュール908はまた、空間相関を実行して、デバイスがホルスタに入れられておりユーザのベルトに取り付けられていた間に、コンピューティングデバイス上のアプリケーションが写真を撮ったかどうかを決定することができる。高レベル挙動検出モジュール908は、この検出された高レベルの挙動(たとえば、ホルスタに入れられたままでカメラを使用すること)が受け入れられるかまたは一般的な挙動であるかどうかを決定することができ、このことは、コンピューティングデバイスの現在の挙動を過去の挙動と比較すること、および/または、複数のデバイスから収集された情報(たとえば、クラウドソーシングサーバから受信された情報)にアクセスすることによって、達成され得る。ホルスタに入れられている間に写真を撮り、それらをサーバにアップロードすることは、(ホルスタに入れられている状況で観測される通常の挙動から決定され得るように)異常な挙動であるので、この状況では、高レベル挙動検出モジュール908は、これを、潜在的に脅威をもたらす挙動として認識し、適切な対応(たとえば、カメラを遮断する、警報を鳴らすなど)を開始することができる。

【０１３１】

ある実施形態では、挙動観測器モジュール202は、複数の部分に実装され得る。

【０１３２】

図10は、ある実施形態の観測器デーモンを実装するコンピューティングシステム1000における論理構成要素および情報フローをより詳細に示す。図10に示す例では、コンピューティングシステム1000は、ユーザ空間に、挙動検出器1002モジュール、データベースエンジン1004モジュール、および挙動分析器モジュール208を含み、カーネル空間に、リングバッファ1014、フィルタルール1016モジュール、スロットリングルール1018モジュール、およびセキュアバッファ1020を含む。コンピューティングシステム1000は、ユーザ空間に、挙動検出器1002およびデータベースエンジン1004を含み、カーネル空間に、セキュアバッファマネージャ1006、ルールマネージャ1008、およびシステムヘルスモニタ1010を含む、観測器デーモンをさらに含み得る。

【０１３３】

様々な実施形態は、システム挙動を特徴付けるために、webkit、SDK、NDK、カーネル、ドライバ、およびハードウェアを包含するコンピューティングデバイス上でレイヤにまたがる観測を提供できる。挙動の観測は、リアルタイムで行われ得る。

【０１３４】

観測器モジュールは、適応的観測技法を実行し、観測の粒度を制御することができる。上で論じたように、コンピューティングデバイスの劣化に寄与する可能性のある多数(すなわち、数千)の要因が存在し、デバイスの性能の劣化に寄与し得る異なる要因のすべてを監視/観測することは実現不可能であり得る。これを克服するために、様々な実施形態は、観測されるべきである関連のある挙動を動的に特定し、特定された挙動が観測されるべき詳細さのレベルを動的に決定する。

【０１３５】

図11は、ある実施形態による、動的で適応的な観測を実行するための例示的な方法1100を示す。ブロック1102で、デバイスプロセッサは、コンピューティングデバイスの劣化に寄与し得る多数の要因/挙動のサブセットを監視/観測することによって、粗い観測を実行することができる。ブロック1103で、デバイスプロセッサは、粗い観測に基づいて、粗い観測および/またはコンピューティングデバイスの挙動を特徴付ける挙動ベクトルを生成することができる。ブロック1104で、デバイスプロセッサは、コンピューティングデバイスの劣化に潜在的に寄与し得る、粗い観測と関連付けられるサブシステム、プロセス、および/またはアプリケーションを特定することができる。これは、たとえば、複数のソースから受信された情報とコンピューティングデバイスのセンサから受信された状況的情報とを比較することによって達成され得る。ブロック1106で、デバイスプロセッサは、粗い観測に基づいて挙動分析動作を実行することができる。ある実施形態では、ブロック1103および1104の一部として、デバイスプロセッサは、図2～図10を参照して上で論じた動作の1つまたは複数を実行することができる。

【０１３６】

決定ブロック1108で、デバイスプロセッサは、疑わしい挙動または潜在的な問題が挙動分析の結果に基づいて特定され補正され得るかどうかを決定することができる。疑わしい挙動または潜在的な問題が挙動分析の結果に基づいて特定され補正され得るとデバイスプロセッサが決定する(すなわち、決定ブロック1108=「Yes」)とき、ブロック1118で、プロセッサは、挙動を補正するためのプロセスを開始し、ブロック1102に戻って追加の粗い観測を実行することができる。

【０１３７】

疑わしい挙動または潜在的な問題が挙動分析の結果に基づいて特定および/または補正され得ないとデバイスプロセッサが決定する(すなわち、決定ブロック1108=「No」)とき、決定ブロック1109で、デバイスプロセッサは、問題の可能性があるかどうかを決定できる。ある実施形態では、デバイスプロセッサは、コンピューティングデバイスが潜在的な問題に遭遇することおよび/または疑わしい挙動に関与することの確率を計算し、計算された確率が所定のしきい値よりも大きいかどうかを決定することによって、問題の可能性があるかどうかを決定することができる。計算された確率が所定のしきい値よりも大きくない、および/または、疑わしい挙動または潜在的問題が存在するおよび/または検出可能である可能性はないとデバイスプロセッサが決定する(すなわち、決定ブロック1109=「No」)とき、プロセッサは、ブロック1102に戻って追加の粗い観測を実行することができる。

【０１３８】

疑わしい挙動または潜在的問題が存在するおよび/または検出可能である可能性があるとデバイスプロセッサが決定する(すなわち、決定ブロック1109=「Yes」)とき、ブロック1110で、デバイスプロセッサは、特定されたサブシステム、プロセス、またはアプリケーションに対して、より深いログ取得/観測または最終的なログ取得を実行することができる。ブロック1112で、デバイスプロセッサは、特定されたサブシステム、プロセス、またはアプリケーションに対して、より深くより詳細な観測を実行することができる。ブロック1114で、デバイスプロセッサは、より深くより詳細な観測に基づいて、さらなるおよび/またはより深い挙動分析を実行することができる。決定ブロック1108で、デバイスプロセッサは、より深い挙動分析の結果に基づいて、疑わしい挙動または潜在的な問題が特定され補正され得るかどうかを再び決定することができる。より深い挙動分析の結果に基づ

いて疑わしい挙動または潜在的な問題が特定され補正され得ないとデバイスプロセッサが決定する(すなわち、決定ブロック1108=「No」)とき、詳細さのレベルが問題を特定するのに十分に微細になるまで、または、詳細さを追加しても問題が特定され得ないかまたは問題が存在しないと決定されるまで、プロセッサはブロック1110～1114の動作を繰り返すことができる。

【０１３９】

より深い挙動分析の結果に基づいて疑わしい挙動または潜在的な問題が特定され補正され得るとデバイスプロセッサが決定する(すなわち、決定ブロック1108=「Yes」)とき、ブロック1118で、デバイスプロセッサは、問題/挙動を補正するための動作を実行することができ、プロセッサはブロック1102に戻って追加の動作を実行することができる。

【０１４０】

ある実施形態では、方法1100のブロック1102～1118の一部として、デバイスプロセッサは、限られた粗い観測値から疑わしい挙動を特定するため、より詳細に観測するために挙動を動的に決定するため、および観測のために必要な詳細さの的確なレベルを動的に決定するために、システムの挙動のリアルタイム挙動分析を実行することができる。これは、デバイス上で大量のプロセッサリソース、メモリリソース、またはバッテリリソースを使用する必要なく、デバイスプロセッサが問題を効率的に特定し、問題が発生するのを防止することを可能にする。

【０１４１】

様々な実施形態は、ソフトウェアアプリケーションの選択グループの集合的挙動を監視および分析するために(許可、ポリシー、またはルールベースの手法ではなく)挙動分析および/または機械学習技法を使用することによって、既存のソリューションを改善する。現代のコンピューティングデバイスは高度に構成可能で複雑なシステムであり、ソフトウェアアプリケーションが結託しているかどうかを決定するために最も重要である要因が各デバイスにおいて異なり得るので、挙動分析または機械学習技法の使用は重要である。さらに、デバイスの特徴/要因の異なる組合せは、ソフトウェアアプリケーションが結託しているかどうかをそのデバイスが決定するために、各デバイスにおける分析を必要とし得る。さらに、しばしば監視および分析を必要とする特徴/要因の的確な組合せは、活動が進行中であるとき、活動が実行される特定のコンピューティングデバイスから得られる情報を使用して決定することしかできない。これらのまたは他の理由で、既存のソリューションは、挙動が進行中の間、コンピュータデバイスのかなりの処理リソース量、メモリリソース量、または電力リソース量を消費せずに、コンピューティングデバイス内の複数のソフトウェアソリューションの集合的挙動、またはソフトウェアアプリケーション間の関係をリアルタイムで監視し、検出し、特徴付けるために十分ではない。

【０１４２】

様々な実施形態(限定はしないが、図1～図11を参照して上で論じた実施形態を含む)は、様々なコンピューティングデバイス上で実装される場合があり、その一例がスマートフォンの形で図12に示される。スマートフォン1200は、内部メモリ1204と、ディスプレイ1212と、スピーカ1214とに結合されたプロセッサ1202を含み得る。加えて、スマートフォン1200は、プロセッサ1202に結合されたワイヤレスデータリンクおよび/または携帯電話トランシーバ1208に接続され得る、電磁放射を送り、受信するためのアンテナを含み得る。スマートフォン1200は通常、ユーザ入力を受信するためのメニュー選択ボタンまたはロッカースイッチ1220も含む。

【０１４３】

典型的なスマートフォン1200はまた、マイクロフォンから受信された音をワイヤレス送信に適したデータパケットにデジタル化し、受信された音のデータパケットを復号し、スピーカに供給されて音を発生させるアナログ信号を生成する、音声符号化/復号(コーデック)回路1206を含む。また、プロセッサ1202、ワイヤレストランシーバ1208、およびコーデック1206のうちの1つまたは複数は、デジタル信号プロセッサ(DSP)回路(個別に図示せず)を含んでよい。ある実施形態では、プロセッサ1202は、図1に示したSOC100など、シス

テムオンチップ(SPC)内に含まれてよい。ある実施形態では、プロセッサ1202は、図1に示したアプリケーションプロセッサ108であり得る。ある実施形態では、プロセッサ1202は処理コア(たとえば、IPコア、CPUコアなど)であり得る。

【０１４４】

実施形態の方法の部分は、実施形態の方法を実行している間にデバイスプロセッサによってアクセスされ得る正常な動作挙動のデータベースを維持することなどの、サーバにおいて発生する処理のいくつかによって、クライアントサーバアーキテクチャ内で行われてよい。そのような実施形態は、図13に示すサーバ1300のような、様々な市販のサーバデバイスのいずれかにおいて実装され得る。そのようなサーバ1300は通常、揮発性メモリ1302と、ディスクドライブ1303などの大容量の不揮発性メモリとに結合されたプロセッサ1301を含む。サーバ1300はまた、プロセッサ1301に結合されたフロッピーディスクドライブ、コンパクトディスク(CD)ドライブまたはDVDディスクドライブ1304を含み得る。サーバ1300はまた、他のブロードキャストシステムコンピュータおよびサーバに結合されたローカルエリアネットワークのような、ネットワーク1305とのデータ接続を確立するための、プロセッサ1301に結合されたネットワークアクセスポート1306を含み得る。

【０１４５】

プロセッサ1202、1301は、以下で説明する様々な実施形態の機能を含む、種々の機能を実行するようにソフトウェア命令(アプリケーション)によって構成され得る任意のプログラマブルマイクロプロセッサ、マイクロコンピュータ、または1つまたは複数の多重プロセッサチップであり得る。いくつかのモバイルデバイスでは、1つのプロセッサをワイヤレス通信機能専用にし、1つのプロセッサを他のアプリケーションの実行専用にするというように、複数のプロセッサ1202が設けられ得る。通常、ソフトウェアアプリケーションは、それらがアクセスされ、プロセッサ1202、1301にロードされる前に、内部メモリ1204、1302、1303内に記憶され得る。プロセッサ1202、1301は、アプリケーションソフトウェア命令を記憶するのに十分な内部メモリを含み得る。

【０１４６】

本出願で使用する「構成要素」、「モジュール」などの用語は、限定はされないが、特定の動作もしくは機能を実行するように構成された、ハードウェア、ファームウェア、ハードウェアとソフトウェアの組合せ、ソフトウェア、または実行中のソフトウェアなどの、コンピュータ関連のエンティティを含むものとする。たとえば、構成要素は、プロセッサ上で実行するプロセス、プロセッサ、オブジェクト、実行ファイル、実行スレッド、プログラム、および/またはコンピュータであってもよいが、それらに限定されない。例として、コンピューティングデバイス上で実行するアプリケーションとコンピューティングデバイスの両方は、構成要素と呼ばれる場合がある。1つまたは複数の構成要素が、プロセスおよび/または実行スレッド内に存在することがあり、1つの構成要素が、1つのプロセッサもしくはコアに局在することがあり、かつ/または2つ以上のプロセッサもしくはコアに分散されることがある。加えて、これらの構成要素は、様々な命令および/またはデータ構造が記憶された様々な非一時的コンピュータ可読媒体から実行することができる。構成要素は、ローカルプロセスおよび/またはリモートプロセス、関数呼出しまたはプロシージャ呼出し、電子信号、データパケット、メモリ読出し/書込み、ならびに他の知られているネットワーク、コンピュータ、プロセッサ、および/またはプロセス関連の通信方法によって通信することができる。

【０１４７】

様々な実施形態の動作を実行するためのプログラマブルプロセッサ上での実行のためのコンピュータプログラムコードまたは「プログラムコード」は、C、C++、C#、Smalltalk、Java（登録商標）、JavaScript（登録商標）、Visual Basic、構造化照会言語(たとえば、Transact-SQL)、Perlなどの高水準プログラミング言語または様々な他のプログラミング言語で記述され得る。本出願で使用する場合、コンピュータ可読記憶媒体上に記憶されたプログラムコードまたはプログラムは、そのフォーマットがプロセッサによって理解可能である(オブジェクトコードなどの)機械語コードを指し得る。

【０１４８】
多くのモバイルコンピューティングデバイスのオペレーティングシステムのカーネルは、(非特権コードが実行する場合)ユーザ空間内に編成され、(特権コードが実行する場合)カーネル空間内に編成される。この分離は、カーネル空間の一部であるコードが一般公有使用許諾(GPL)で許諾される必要がある一方で、ユーザ空間において実行するコードがGPL許諾されなくてもよい、Android(登録商標)および他のGPL環境において特に重要である。本明細書で論じた様々なソフトウェア構成要素/モジュールは、明示的に別段の記述がない場合、カーネル空間またはユーザ空間のいずれかに実装され得ることを理解されたい。
【０１４９】
上記の方法の説明およびプロセスフロー図は、単に説明のための例として提供したものであり、様々な実施形態のステップを提示された順序で実行しなければならないことを要求または暗示するものではない。当業者なら諒解するように、上記の実施形態におけるステップの順番は、任意の順番で実行可能である。「その後」、「次いで」、「次」などの語は、ステップの順番を限定するものではなく、これらの語は単に、読者に本方法の説明を案内するために使用される。さらに、たとえば冠詞「a」、「an」、または「the」を使用する、請求項の要素に対する単数形でのいかなる参照も、要素を単数形に限定すると解釈されるべきではない。
【０１５０】
本明細書で開示する実施形態に関して説明した様々な例示的な論理ブロック、モジュール、回路、およびアルゴリズムステップは、電子ハードウェア、コンピュータソフトウェア、またはその両方の組合せとして実装される場合がある。ハードウェアとソフトウェアのこの互換性を明確に示すために、様々な例示的な構成要素、ブロック、モジュール、回路、およびステップを、一般にそれらの機能性に関して上述した。そのような機能性がハードウェアとして実装されるか、ソフトウェアとして実装されるかは、特定の用途およびシステム全体に課せられる設計制約によって決まる。当業者は、説明した機能性を特定のアプリケーションごとに様々な方法で実装し得るが、そのような実装決定は本発明の範囲からの逸脱を引き起こすものと解釈されるべきではない。
【０１５１】
本明細書で開示した実施形態に関して説明した様々な例示的論理、論理ブロック、モジュール、および回路を実装するために使用されるハードウェアは、汎用プロセッサ、デジタル信号プロセッサ(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)もしくは他のプログラマブル論理デバイス、個別ゲートもしくはトランジスタ論理、個別ハードウェア構成要素、または本明細書で説明した機能を実行するように設計されたそれらの任意の組合せを用いて実装もしくは実行され得る。汎用プロセッサはマルチプロセッサであり得るが、代替的に、プロセッサはいかなる従来のプロセッサ、コントローラ、マイクロコントローラ、またはステートマシンであってもよい。プロセッサは、コンピューティングデバイスの組合せ、たとえば、DSPとマルチプロセッサとの組合せ、複数のマルチプロセッサ、DSPコアと連携する1つもしくは複数のマルチプロセッサ、または任意の他のそのような構成としても実装され得る。代替的に、いくつかのステップまたは方法は、所与の機能に特有の回路によって実行され得る。
【０１５２】
1つまたは複数の例示的な実施形態では、説明した機能は、ハードウェア、ソフトウェア、ファームウェア、またはそれらの任意の組合せにおいて実装されてもよい。ソフトウェアで実装される場合、機能は、非一時的コンピュータ可読記憶媒体または非一時的プロセッサ可読記憶媒体上に1つまたは複数のプロセッサ実行可能命令またはコードとして記憶され得る。本明細書で開示した方法またはアルゴリズムのステップは、非一時的なコンピュータ可読媒体またはプロセッサ可読記憶媒体上に存在し得るプロセッサ実行可能ソフトウェアモジュールにおいて実施され得る。非一時的コンピュータ可読記憶媒体または非一時的プロセッサ可読記憶媒体は、コンピュータまたはプロセッサによってアクセスされる場合がある任意の記憶媒体であってもよい。限定ではなく例として、そのような非一時

的コンピュータ可読記憶媒体または非一時的プロセッサ可読記憶媒体は、RAM、ROM、EEPROM、FLASHメモリ、CD-ROMもしくは他の光ディスクストレージ、磁気ディスクストレージもしくは他の磁気ストレージデバイス、または命令もしくはデータ構造の形態で所望のプログラムコードを記憶するために使用され得、コンピュータによってアクセスされ得る任意の他の媒体を含み得る。ディスク(disk)およびディスク(disc)は、本明細書で使用するとき、コンパクトディスク(disc)(CD)、レーザーディスク（登録商標）(disc)、光ディスク(disc)、デジタル多用途ディスク(disc)(DVD)、フロッピーディスク(disk)、およびブルーレイディスク(disc)を含み、ディスク(disk)は通常、データを磁気的に再生し、ディスク(disc)は、レーザーを用いてデータを光学的に再生する。上記の組合せも、非一時的コンピュータ可読記憶媒体およびプロセッサ可読記憶媒体の範囲内に含まれる。加えて、方法またはアルゴリズムの動作は、コンピュータプログラム製品に組み込まれる場合がある、非一時的プロセッサ可読媒体および/または非一時的コンピュータ可読媒体上のコードおよび/または命令の1つまたは任意の組合せまたはセットとして存在する場合がある。

【０１５３】

開示した実施形態の前述の説明は、いかなる当業者も本発明を作成または使用することができるように与えたものである。これらの実施形態に対する様々な修正は当業者には容易に明らかとなり、本明細書で定義した一般原理は、本発明の趣旨または範囲から逸脱することなく、他の実施形態に適用され得る。したがって、本発明は、本明細書に示す実施形態に限定されるものではなく、以下の特許請求の範囲、ならびに本明細書で開示する原理および新規の特徴と一致する最も広い範囲を与えられるべきである。

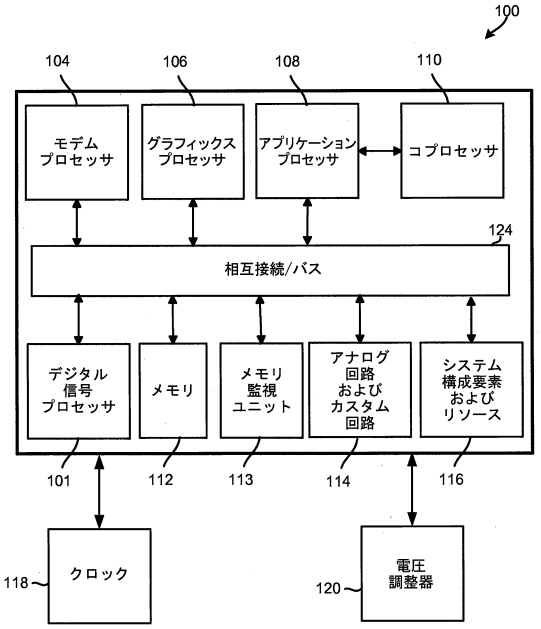
【符号の説明】

【０１５４】

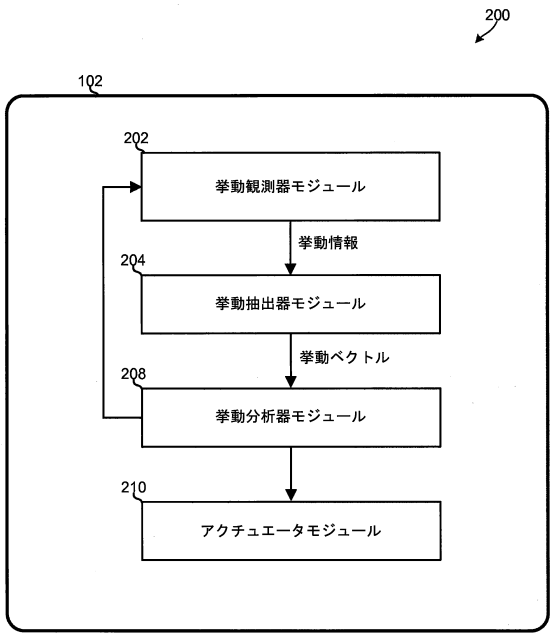
100　システムオンチップ(SOC)
101　デジタル信号プロセッサ(DSP)、異種プロセッサ、プロセッサ、デジタル信号プロセッサ
104　モデムプロセッサ、異種プロセッサ、プロセッサ
106　グラフィックスプロセッサ、異種プロセッサ、プロセッサ
108　アプリケーションプロセッサ、異種プロセッサ、プロセッサ
110　コプロセッサ、プロセッサ
112　メモリ要素
113　ハードウェアベースのメモリ監視ユニット
114　アナログ回路およびカスタム回路
116　システム構成要素/リソース、リソース
118　クロック
120　電圧調整器
124　相互接続/バスモジュール
200　挙動ベースのセキュリティシステム
202　挙動観測器モジュール、モジュール
204　挙動抽出器モジュール、モジュール
208　挙動分析器モジュール、モジュール
210　アクチュエータモジュール、モジュール
300　方法
400　方法
500　方法
600　方法
700　方法
800　ブースト方法
902　適応フィルタモジュール
904　スロットルモジュール
906　観測器モードモジュール

908 高レベル挙動検出モジュール
910 挙動ベクトル生成器
912 セキュアバッファ
914 空間相関モジュール
916 時間相関モジュール
1000 コンピューティングシステム
1002 挙動検出器
1004 データベースエンジン
1006 セキュアバッファマネージャ
1008 ルールマネージャ
1010 システムヘルスモニタ
1014 リングバッファ
1016 フィルタルール
1018 スロットリングルール
1020 セキュアバッファ
1100 方法
1200 スマートフォン
1202 プロセッサ
1204 内部メモリ
1206 音声符号化/復号(コーデック)回路、コーデック
1208 ワイヤレスデータリンクおよび/または携帯電話トランシーバ、ワイヤレストランシーバ
1212 ディスプレイ
1214 スピーカ
1220 メニュー選択ボタンまたはロッカースイッチ
1300 サーバ
1301 プロセッサ
1302 揮発性メモリ、内部メモリ
1303 ディスクドライブ、内部メモリ
1304 フロッピーディスクドライブ、コンパクトディスク(CD)またはDVDディスクドライブ
1305 ネットワーク
1306 ネットワークアクセスポート

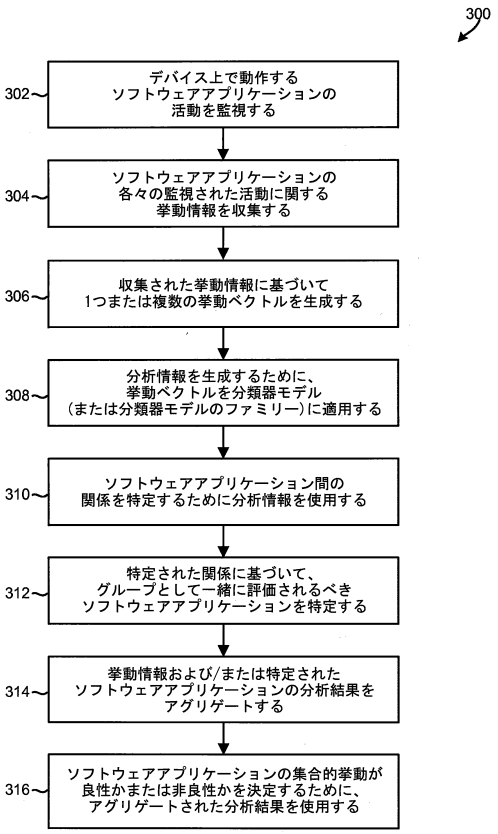
【図１】


100
104
106
108
110
モデム プロセッサ
グラフィックス プロセッサ
アプリケーション プロセッサ
コプロセッサ
124
相互接続/バス
デジタル 信号 プロセッサ
メモリ
メモリ 監視 ユニット
アナログ 回路 および カスタム 回路
システム 構成要素 および リソース
101
112
113
114
116
118
クロック
120
電圧 調整器

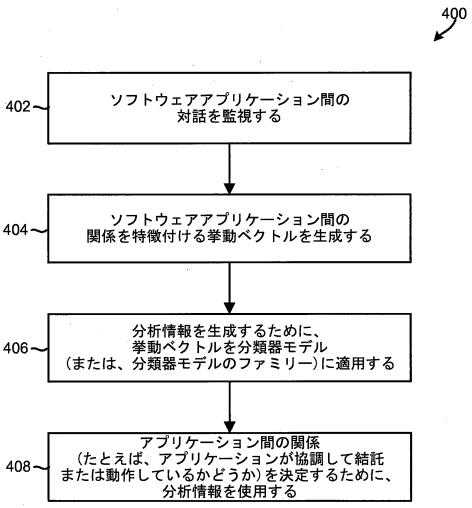
【図２】


200
102
202
挙動観測器モジュール
挙動情報
204
挙動抽出器モジュール
挙動ベクトル
208
挙動分析器モジュール
210
アクチュエータモジュール

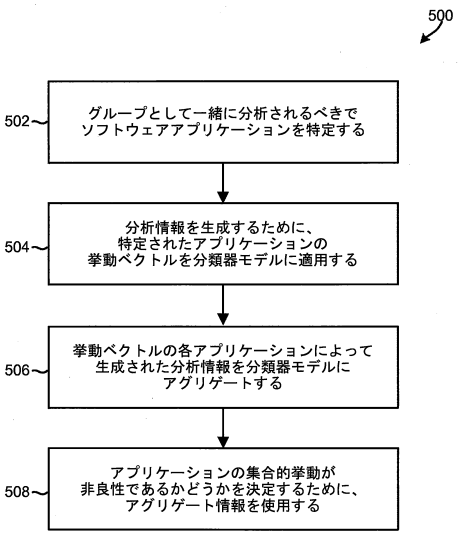
【図３】


300
302
デバイス上で動作する ソフトウェアアプリケーションの 活動を監視する
304
ソフトウェアアプリケーションの 各々の監視された活動に関する 挙動情報を収集する
306
収集された挙動情報に基づいて 1つまたは複数の挙動ベクトルを生成する
308
分析情報を生成するために、 挙動ベクトルを分類器モデル （または分類器モデルのファミリー）に適用する
310
ソフトウェアアプリケーション間の 関係を特定するために分析情報を使用する
312
特定された関係に基づいて、 グループとして一緒に評価されるべき ソフトウェアアプリケーションを特定する
314
挙動情報および/または特定された ソフトウェアアプリケーションの分析結果を アグリゲートする
316
ソフトウェアアプリケーションの集合的挙動が 良性かまたは非良性かを決定するために、 アグリゲートされた分析結果を使用する

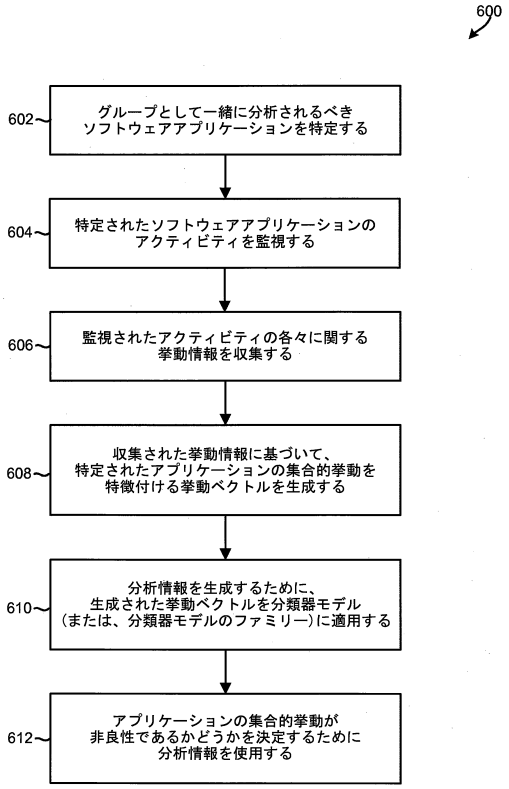
【図４】


400
402
ソフトウェアアプリケーション間の 対話を監視する
404
ソフトウェアアプリケーション間の 関係を特徴付ける挙動ベクトルを生成する
406
分析情報を生成するために、 挙動ベクトルを分類器モデル （または、分類器モデルのファミリー）に適用する
408
アプリケーション間の関係 （たとえば、アプリケーションが協調して結託 または動作しているかどうか）を決定するために、 分析情報を使用する

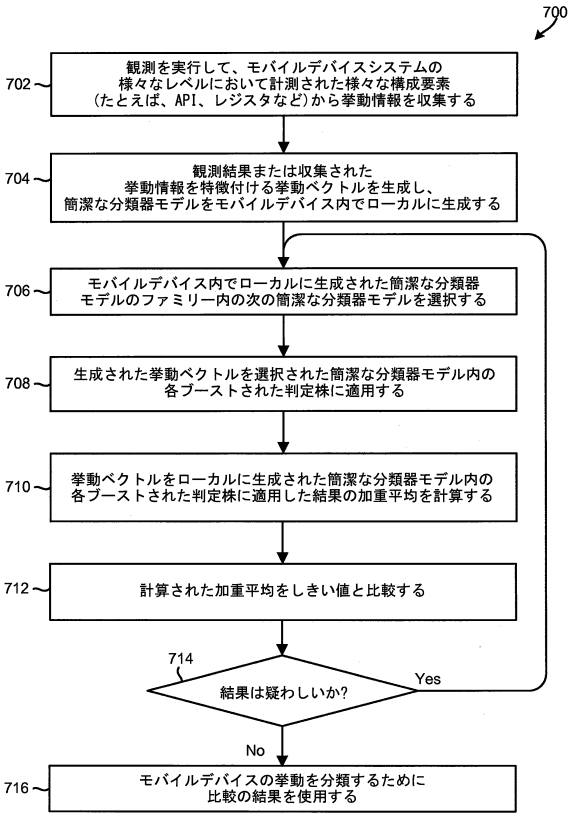
【図５】


500
502
グループとして一緒に分析されるべきで
ソフトウェアアプリケーションを特定する
504
分析情報を生成するために、
特定されたアプリケーションの
挙動ベクトルを分類器モデルに適用する
506
挙動ベクトルの各アプリケーションによって
生成された分析情報を分類器モデルに
アグリゲートする
508
アプリケーションの集合的挙動が
非良性であるかどうかを決定するために、
アグリゲート情報を使用する

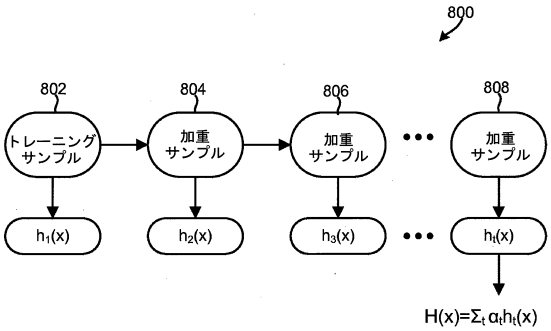
【図６】


600
602
グループとして一緒に分析されるべき
ソフトウェアアプリケーションを特定する
604
特定されたソフトウェアアプリケーションの
アクティビティを監視する
606
監視されたアクティビティの各々に関する
挙動情報を収集する
608
収集された挙動情報に基づいて、
特定されたアプリケーションの集合的挙動を
特徴付ける挙動ベクトルを生成する
610
分析情報を生成するために、
生成された挙動ベクトルを分類器モデル
(または、分類器モデルのファミリー)に適用する
612
アプリケーションの集合的挙動が
非良性であるかどうかを決定するために
分析情報を使用する

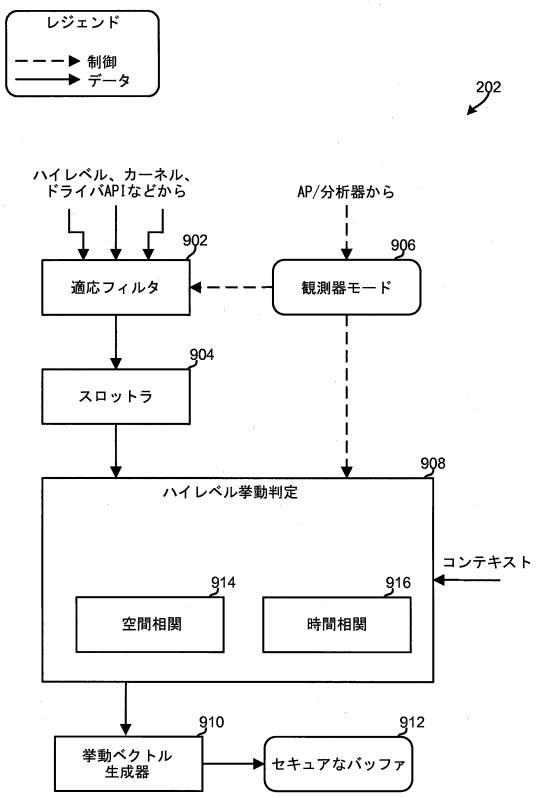
【図７】


700
702
観測を実行して、モバイルデバイスシステムの
様々なレベルにおいて計測された様々な構成要素
(たとえば、API、レジスタなど)から挙動情報を収集する
704
観測結果または収集された
挙動情報を特徴付ける挙動ベクトルを生成し、
簡潔な分類器モデルをモバイルデバイス内でローカルに生成する
706
モバイルデバイス内でローカルに生成された簡潔な分類器
モデルのファミリー内の次の簡潔な分類器モデルを選択する
708
生成された挙動ベクトルを選択された簡潔な分類器モデル内の
各ブーストされた判定株に適用する
710
挙動ベクトルをローカルに生成された簡潔な分類器モデル内の
各ブーストされた判定株に適用した結果の加重平均を計算する
712
計算された加重平均をしきい値と比較する
714
結果は疑わしいか?
Yes
No
716
モバイルデバイスの挙動を分類するために
比較の結果を使用する

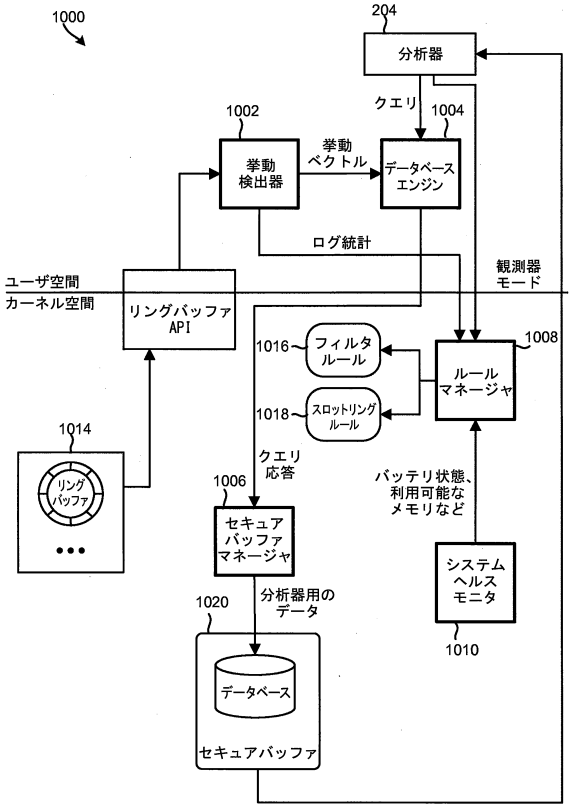
【図８】


800
802
804
806
808
トレーニング
サンプル
加重
サンプル
加重
サンプル
加重
サンプル
h1(x)
h2(x)
h3(x)
ht(x)
H(x)=Σt αtht(x)

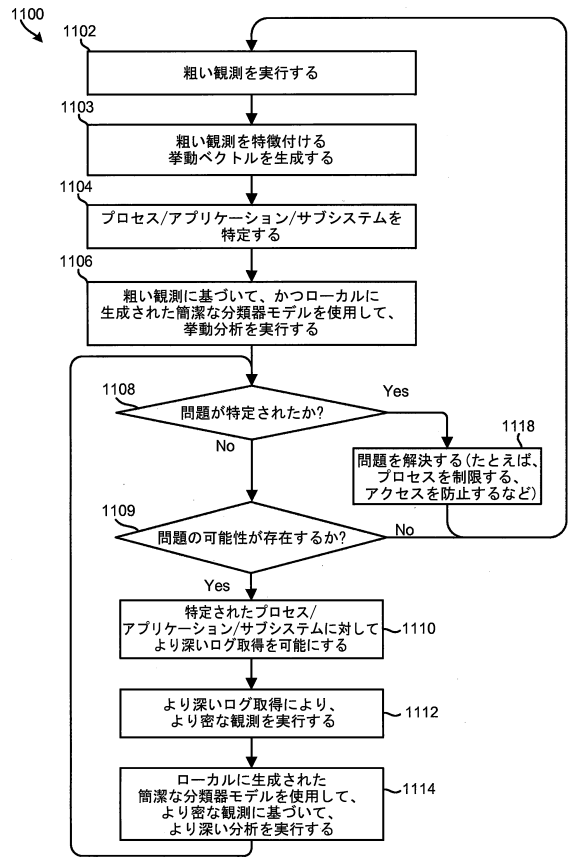
【図９】



【図１０】



【図１１】



【図１２】

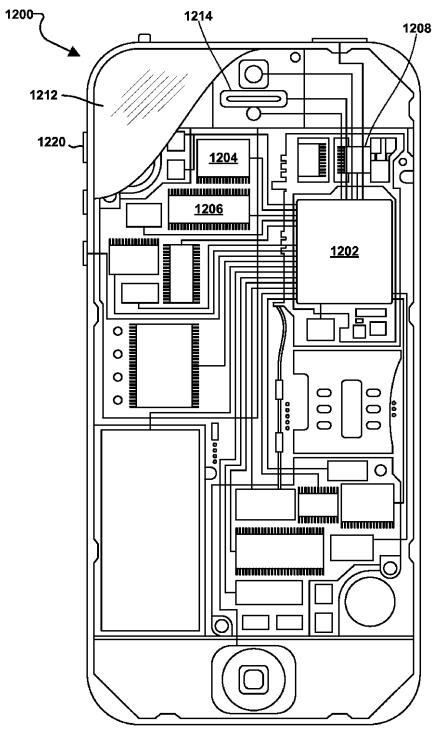


FIG. 12

【図１３】

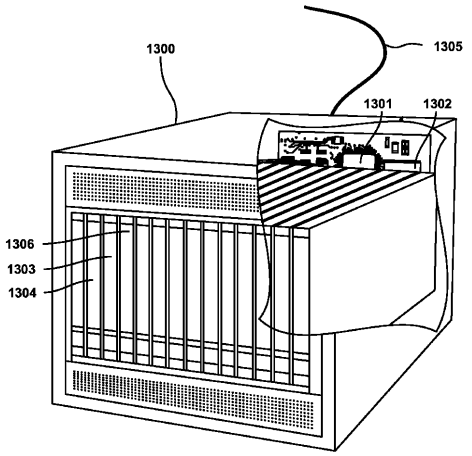


FIG. 13

フロントページの続き

(51)Int.Cl. FI
Ｇ０６Ｆ 11/30 １４０Ｅ

(72)発明者 イン・チェン
アメリカ合衆国・カリフォルニア・９２１２１－１７１４・サン・ディエゴ・モアハウス・ドライヴ・５７７５

審査官 金沢 史明

(56)参考文献 国際公開第２０１２／０４６４０６（WO，Ａ１）
米国特許出願公開第２０１３／０２４７１８７（ＵＳ，Ａ１）
国際公開第２０１３／１７３００１（WO，Ａ１）
特開２００７－０４８３１５（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ ２１／５６
Ｇ０６Ｆ ９／４８
Ｇ０６Ｆ １１／０７
Ｇ０６Ｆ １１／３０
Ｇ０６Ｆ １１／３４