(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5743227号
(P5743227)
(45)発行日　平成27年7月1日(2015.7.1)　(24)登録日　平成27年5月15日(2015.5.15)

(51)Int.Cl.　　F I
H04L 9/32 (2006.01)　　H04L 9/00 675D
H04L 9/14 (2006.01)　　H04L 9/00 641
G06F 21/64 (2013.01)　　G06F 21/64 350

請求項の数 95　(全 43 頁)

(21)出願番号　特願2012-508568 (P2012-508568)
(86)(22)出願日　平成22年4月26日 (2010.4.26)
(65)公表番号　特表2012-524954 (P2012-524954A)
(43)公表日　平成24年10月18日 (2012.10.18)
(86)国際出願番号　PCT/US2010/032428
(87)国際公開番号　WO2010/126837
(87)国際公開日　平成22年11月4日 (2010.11.4)
審査請求日　平成23年10月26日 (2011.10.26)
(31)優先権主張番号　12/430,553
(32)優先日　平成21年4月27日 (2009.4.27)
(33)優先権主張国　米国(US)

前置審査

(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　イヴァン・ヒュー・マクリーン
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・ドライヴ・５７７５

審査官　金木　陽一

最終頁に続く

(54)【発明の名称】コードおよびデータ署名を改善するための方法および装置

(57)【特許請求の範囲】
【請求項１】
配信後にソフトウェアを検証し、署名するための方法であって、
第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された前記ソフトウェアに対する第２の検証方法および第２のシグナチャに更新するための要求をクライアントコンピュータから、受信するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、
前記クライアントコンピュータから、検証されるべき前記ソフトウェアの識別子を受信するステップと、
前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するステップと、
前記クライアントコンピュータによって与えられた、前記ソフトウェアに関係する前記情報に基づいて、前記ソフトウェアが信用できるかどうかを判断するステップと、
前記ソフトウェアを検証する際に前記クライアントコンピュータが使用するための、前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを前記クライアントコンピュータに送信するステップと
を含む、方法。

【請求項２】
前記クライアントコンピュータのユーザに関係する情報を前記クライアントコンピュータに要求するステップと、
前記クライアントコンピュータのユーザに関係する情報とユーザに関係する登録された情報を比較することで、前記ソフトウェアが信用できるかどうかを判断するステップとをさらに含む、請求項1に記載の方法。
【請求項３】
前記ソフトウェアの前記受信された識別子が前記ソフトウェアの一部分を含む、請求項1に記載の方法。
【請求項４】
前記ソフトウェアの前記受信された識別子が、前記ソフトウェアとともに与えられた前記第１のシグナチャを含む、請求項1に記載の方法。
【請求項５】
前記ソフトウェアの前記受信された識別子が、前記ソフトウェアとともに与えられた前記第１のシグナチャの一部分を含む、請求項1に記載の方法。
【請求項６】
前記ソフトウェアに関係する前記要求された情報が、前記ソフトウェアの一部分を含む、請求項1に記載の方法。
【請求項７】
前記ソフトウェアに関係する前記要求された情報が、前記ソフトウェアに対して実行されたハッシュ関数の結果を含む、請求項1に記載の方法。
【請求項８】
前記ソフトウェアに関係する前記要求された情報が、前記ソフトウェアによってアクセスされるクライアントコンピュータリソースの識別情報を含む、請求項1に記載の方法。
【請求項９】
前記第２の検証方法は、前記ソフトウェアにハッシュ関数が適用される前に、前記ソフトウェアにアペンドされるかまたはプリペンドされる定数を含み、前記第２のシグナチャが、前記ソフトウェアに前記第２の検証方法を適用することから生じるシグナチャである、請求項1に記載の方法。
【請求項１０】
前記ソフトウェアの前記受信された識別子を使用してデータファイルにアクセスするステップであって、前記データファイルが、前記ソフトウェアを検証することに関係する情報を含む、アクセスするステップをさらに含み、
前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するステップが、前記アクセスされたデータファイル中で識別される情報を要求するステップを含み、
前記ソフトウェアが信用できるかどうかを判断するステップが、前記アクセスされたデータファイル中で識別されるステップを実行するステップを含み、
前記第２の検証方法と前記第２のシグナチャとが、前記アクセスされたデータファイルから取得される、請求項1に記載の方法。
【請求項１１】
前記アクセスされたデータファイルが、対策データベースからアクセスされ、前記対策データベースは、特定のアプリケーションに対する既知の脅威に打ち勝つために、適切な問合せステップと、検証方法と、シグナチャ値とを保持しているデータレコードを用いて更新される、請求項10に記載の方法。
【請求項１２】
配信後にソフトウェアを検証し、署名するための方法であって、
第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された前記ソフトウェアに対する第２の検証方法および第２のシグナチャに更新することを署名サーバに要求するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで

生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップと、
前記署名サーバから受信された、前記ソフトウェアに関係する情報についての要求に応答するステップと、
前記署名サーバから、前記ソフトウェアを検証する際に使用するための、前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを受信するステップと、
前記受信された第２の検証方法と第２のシグナチャとを記憶するステップと、
前記ソフトウェアを検証するために前記第２の検証方法と前記第２のシグナチャとを使用するステップと
を含む、方法。
【請求項１３】
前記ソフトウェアに関係する情報についての要求に応答するステップが、前記ソフトウェアの一部分を前記署名サーバに送信するステップを含む、請求項12に記載の方法。
【請求項１４】
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップが、ソフトウェア識別子を与えるステップを含む、請求項12に記載の方法。
【請求項１５】
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップが、前記ソフトウェアの一部分を与えるステップを含む、請求項12に記載の方法。
【請求項１６】
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップが、前記ソフトウェアとともに含まれる前記第１のシグナチャを与えるステップを含む、請求項12に記載の方法。
【請求項１７】
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップが、前記ソフトウェアとともに含まれる前記第１のシグナチャの一部分を与えるステップを含む、請求項12に記載の方法。
【請求項１８】
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップが、前記署名サーバから受信されたソフトウェア識別子についての要求に応答して実行され、与えられた前記ソフトウェア識別子が、前記受信された要求に応答する、請求項12に記載の方法。
【請求項１９】
前記ソフトウェアに関係する情報についての要求に応答するステップが、前記ソフトウェアに対してハッシュ関数を実行するステップと、結果を前記署名サーバに送信するステップとを含む、請求項12に記載の方法。
【請求項２０】
前記ソフトウェアに関係する情報についての要求に応答するステップが、前記ソフトウェアによってアクセスされるリソースを前記署名サーバに対して識別するステップを含む、請求項12に記載の方法。
【請求項２１】
前記受信された第２の検証方法と第２のシグナチャとが、前記ソフトウェアに関連する第2のシグナチャファイルに記憶される、請求項12に記載の方法。
【請求項２２】
前記ソフトウェアに関連する第2のシグナチャがあるかどうかを判断するステップと、
前記ソフトウェアに関連する第2のシグナチャがない場合、前記署名サーバへの接続を確立するステップと
をさらに含む、請求項12に記載の方法。

【請求項２３】
プロセッサと、
前記プロセッサに結合されたネットワークインターフェース回路であって、前記プロセッサがネットワークを介して通信することを可能にするように構成された、ネットワークインターフェース回路と、
前記プロセッサに結合されたメモリと
を含むサーバであって、
前記プロセッサが、
第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新するための要求を、前記ネットワークインターフェース回路を介してクライアントコンピュータから受信するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、
前記ネットワークインターフェース回路を介して前記クライアントコンピュータから、検証されるべき前記ソフトウェアの識別子を受信するステップと、
前記ネットワークインターフェース回路を介して前記クライアントコンピュータへの、検証されるべき前記ソフトウェアに関係する情報についての要求を送信するステップと、
前記クライアントコンピュータによって与えられた前記ソフトウェアに関係する情報に基づいて、検証されるべき前記ソフトウェアが信用できるかどうかを判断するステップと、
前記ソフトウェアを検証する際に前記クライアントコンピュータが使用するための、前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを、前記ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップと
を含むステップを実行するための実行可能命令で構成された、サーバ。
【請求項２４】
前記プロセッサが、前記クライアントコンピュータのユーザに関係する情報を前記クライアントコンピュータに要求するステップと、
前記クライアントコンピュータのユーザに関係する情報とユーザに関係する登録された情報を比較することで、前記ソフトウェアが信用できるかどうかを判断するステップとをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。
【請求項２５】
前記プロセッサが、検証されるべき前記ソフトウェアの識別子についての要求を、前記ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。
【請求項２６】
前記プロセッサが、ソフトウェア識別子として戻されるべき前記ソフトウェアの特定の部分を識別する、検証されるべき前記ソフトウェアの識別子についての要求を、前記ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。
【請求項２７】
前記プロセッサが、前記ソフトウェアの前記識別子として使用されるべき、検証されるべき前記ソフトウェアの前記第１のシグナチャについての要求を、前記ネットワークイン

ターフェース回路を介して前記クライアントコンピュータに送信するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。

【請求項２８】

前記プロセッサが、前記ソフトウェアの識別子として使用されるべき前記ソフトウェアの前記第１のシグナチャの特定の部分を識別する、検証されるべき前記ソフトウェアの前記識別子についての要求を、前記ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。

【請求項２９】

前記プロセッサが、検証されるべき前記ソフトウェアの一部分についての要求を、前記ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。

【請求項３０】

前記プロセッサが、検証されるべき前記ソフトウェアに対してハッシュ関数を実行し、前記サーバに結果を戻したいという要求を、前記ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。

【請求項３１】

前記プロセッサが、検証されるべき前記ソフトウェアによってアクセスされるクライアントコンピュータリソースを識別したいという要求を、前記ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項23に記載のサーバ。

【請求項３２】

前記第２の検証方法は、前記ソフトウェアにハッシュ関数が適用される前に、前記ソフトウェアにアペンドされるかまたはプリペンドされる定数を含み、前記第２のシグナチャが、検証されるべき前記ソフトウェアに前記第２の検証方法を適用することから生じるシグナチャである、請求項23に記載のサーバ。

【請求項３３】

前記プロセッサが、検証されるべき前記ソフトウェアの前記受信された識別子を使用してデータファイルにアクセスするステップをさらに含むステップを実行するための実行可能命令で構成され、前記データファイルが、前記ソフトウェアを検証することに関係する情報を含み、

前記クライアントコンピュータに対する、検証されるべき前記ソフトウェアに関係する情報についての要求が、前記アクセスされたデータファイル中で識別される情報についての要求を含み、

検証されるべき前記ソフトウェアが信用できるかどうかを判断するステップが、前記アクセスされたデータファイル中で識別されるステップを実行するステップを含み、

前記第２の検証方法と前記第２のシグナチャとが、前記アクセスされたデータファイルから取得される、請求項23に記載のサーバ。

【請求項３４】

前記メモリが対策データベースを記憶しており、前記アクセスされたデータファイルが前記対策データベースからアクセスされ、前記対策データベースは、特定のアプリケーションに対する既知の脅威に打ち勝つために、適切な問合せステップと、検証方法と、シグナチャ値とを保持しているデータレコードを用いて更新される、請求項33に記載のサーバ。

【請求項３５】

プロセッサと、

前記プロセッサに結合されたネットワークインターフェース回路であって、前記プロセッサがネットワークを介して通信することを可能にするように構成された、ネットワーク

インターフェース回路と、
前記プロセッサに結合されたメモリと
を含むコンピュータであって、
前記プロセッサが、
第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新するための要求を署名サーバに前記ネットワークインターフェース回路を介して送信するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、
前記ネットワークインターフェース回路を介して、検証されるべき前記ソフトウェアの識別子を署名サーバに送信するステップと、
前記ネットワークインターフェース回路を介して前記署名サーバから受信された、検証されるべき前記ソフトウェアに関係する情報についての要求に応答するステップと、
前記署名サーバから前記ネットワークインターフェース回路を介して、前記ソフトウェアを検証する際に使用するための、検証されるべき前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを受信するステップと、
前記受信された第２の検証方法と第２のシグナチャとを前記メモリに記憶するステップと、
前記ソフトウェアを検証するために前記第２の検証方法と前記第２のシグナチャとを使用するステップと
を含むステップを実行するための実行可能命令で構成された、コンピュータ。
【請求項３６】
前記プロセッサは、検証されるべき前記ソフトウェアに関係する情報についての要求に応答する前記ステップが、前記ネットワークインターフェース回路を介して前記署名サーバに前記ソフトウェアの一部分を送信するステップを含むような実行可能命令で構成された、請求項35に記載のコンピュータ。
【請求項３７】
前記プロセッサは、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、ソフトウェア識別子を送信するステップを含むような実行可能命令で構成された、請求項35に記載のコンピュータ。
【請求項３８】
前記プロセッサは、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記ソフトウェアの一部分を送信するステップを含むような実行可能命令で構成された、請求項35に記載のコンピュータ。
【請求項３９】
前記プロセッサは、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記ソフトウェアとともに含まれる前記第１のシグナチャを送信するステップを含むような実行可能命令で構成された、請求項35に記載のコンピュータ。
【請求項４０】
前記プロセッサは、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記ソフトウェアとともに含まれる前記第１のシグナチャの一部分を送信するステップを含むような実行可能命令で構成された、請求項35に記載のコンピュータ。
【請求項４１】
前記プロセッサは、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記署名サーバから受信されたソフトウェア識別子についての要求に応答して実行され、前記署名サーバに送信された前記ソフトウェア識別子が、前記受信

された要求に応答するような実行可能命令で構成された、請求項35に記載のコンピュータ。

【請求項４２】

前記プロセッサは、検証されるべき前記ソフトウェアに関係する情報についての要求に応答する前記ステップが、検証されるべき前記ソフトウェアに対してハッシュ関数を実行するステップと、前記ネットワークインターフェース回路を介して結果を前記署名サーバに送信するステップとを含むような実行可能命令で構成された、請求項35に記載のコンピュータ。

【請求項４３】

前記プロセッサは、検証されるべき前記ソフトウェアに関係する情報についての要求に応答する前記ステップが、前記ソフトウェアによってアクセスされるリソースを前記署名サーバに対して識別するステップを含むような実行可能命令で構成された、請求項35に記載のコンピュータ。

【請求項４４】

前記プロセッサは、検証されるべき前記ソフトウェアに関係する情報についての要求に応答する前記ステップが、

前記署名サーバから、前記コンピュータのユーザに関する情報についての要求を受信するステップと、

前記要求された情報について前記ユーザにプロンプトを出すステップと、

前記ユーザから応答を受信するステップと、

ネットワークインターフェース接続を介して前記ユーザの応答を前記署名サーバに送信するステップと

を含むような実行可能命令で構成された、請求項35に記載のコンピュータ。

【請求項４５】

前記プロセッサが、前記受信された第２の検証方法と第２のシグナチャとを、前記メモリに記憶された前記ソフトウェアに関連する第2のシグナチャファイルに記憶するステップをさらに含むステップを実行するための実行可能命令で構成された、請求項35に記載のコンピュータ。

【請求項４６】

前記プロセッサが、

前記メモリに記憶された前記ソフトウェアに関連する前記第2のシグナチャがあるかどうかを判断するステップと、

前記ソフトウェアに関連する第2のシグナチャがない場合、前記ネットワークインターフェース回路を介して前記署名サーバへの接続を確立するステップと

をさらに含むステップを実行するための実行可能命令で構成された、請求項35に記載のコンピュータ。

【請求項４７】

クライアントコンピュータから、第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新するための要求を受信するための手段であって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、手段と、

前記クライアントコンピュータから、検証されるべき前記ソフトウェアの識別子を受信するための手段と、

検証されるべき前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するための手段と、

前記クライアントコンピュータによって与えられた前記ソフトウェアに関係する情報に

基づいて、検証されるべき前記ソフトウェアが信用できるかどうかを判断するための手段と、
前記ソフトウェアを検証する際に前記クライアントコンピュータが使用するための、検証されるべき前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを前記クライアントコンピュータに送信するための手段と
を含む、サーバ。
【請求項４８】
前記クライアントコンピュータのユーザに関係する情報を前記クライアントコンピュータに要求するための手段と、
前記クライアントコンピュータのユーザに関係する情報とユーザに関係する登録された情報を比較することで、前記ソフトウェアが信用できるかどうかを判断するための手段と
をさらに含む、請求項47に記載のサーバ。
【請求項４９】
検証されるべき前記ソフトウェアの識別子についての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するための手段をさらに含む、請求項47に記載のサーバ。
【請求項５０】
ソフトウェア識別子として戻されるべき前記ソフトウェアの特定の部分を識別する、検証されるべき前記ソフトウェアの識別子についての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するための手段をさらに含む、請求項47に記載のサーバ。
【請求項５１】
前記ソフトウェアの前記識別子として使用されるべき、検証されるべき前記ソフトウェアの前記第１のシグナチャについての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するための手段をさらに含む、請求項47に記載のサーバ。
【請求項５２】
前記ソフトウェアの識別子として使用されるべき前記ソフトウェアの前記第１のシグナチャの特定の部分を識別する、検証されるべき前記ソフトウェアの前記識別子についての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するための手段をさらに含む、請求項47に記載のサーバ。
【請求項５３】
前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するための前記手段が、検証されるべき前記ソフトウェアの一部分を要求するための手段を含む、請求項47に記載のサーバ。
【請求項５４】
前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するための前記手段が、前記クライアントコンピュータに、検証されるべき前記ソフトウェアに対してハッシュ関数を実行し、結果を戻すように要求するための手段を含む、請求項47に記載のサーバ。
【請求項５５】
前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するための前記手段が、検証されるべき前記ソフトウェアによってアクセスされるクライアントコンピュータリソースのリストを要求するための手段を含む、請求項47に記載のサーバ。
【請求項５６】
前記第２の検証方法は、前記ソフトウェアにハッシュ関数が適用される前に、前記ソフトウェアにアペンドされるかまたはプリペンドされる定数を含み、前記第２のシグナチャが、検証されるべき前記ソフトウェアに前記第２の検証方法を適用することから生じるシグナチャである、請求項47に記載のサーバ。
【請求項５７】

前記ソフトウェアの前記受信された識別子を使用してデータファイルにアクセスするための手段であって、前記データファイルが、前記ソフトウェアを検証することに関係する情報を含む、アクセスするための手段をさらに含み、
前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するための前記手段が、前記アクセスされたデータファイル中で識別される情報を要求するための手段を含み、
検証されるべき前記ソフトウェアが信用できるかどうかを判断するための前記手段が、前記アクセスされたデータファイル中で識別されるステップを実行するための手段を含み、
前記第２の検証方法と前記第２のシグナチャとを前記クライアントコンピュータに送信するための前記手段が、前記アクセスされたデータファイルから前記第２の検証方法と前記第２のシグナチャとを取得する、請求項47に記載のサーバ。
【請求項５８】
第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新することを署名サーバに要求するための手段であって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、手段と、
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるための手段と、
前記署名サーバから受信された、検証されるべき前記ソフトウェアに関係する情報についての要求に応答するための手段と、
前記署名サーバから、前記ソフトウェアを検証する際に使用するための、検証されるべき前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを受信するための手段と、
前記受信された第２の検証方法と第２のシグナチャとを記憶するための手段と、
前記ソフトウェアを検証するために前記第２の検証方法と前記第２のシグナチャとを使用するための手段と
を含む、コンピュータ。
【請求項５９】
前記ソフトウェアに関係する情報についての要求に応答するための前記手段が、前記ソフトウェアの一部分を前記署名サーバに送信するための手段を含む、請求項58に記載のコンピュータ。
【請求項６０】
前記ソフトウェアの識別子を前記署名サーバに与えるための前記手段が、ソフトウェア識別子を送信する、請求項58に記載のコンピュータ。
【請求項６１】
前記ソフトウェアの識別子を前記署名サーバに与えるための前記手段が、前記ソフトウェアの一部分を前記署名サーバに与えるための手段を含む、請求項58に記載のコンピュータ。
【請求項６２】
前記ソフトウェアの識別子を前記署名サーバに与えるための前記手段が、前記ソフトウェアとともに含まれる前記第１のシグナチャを与えるための手段を含む、請求項58に記載のコンピュータ。
【請求項６３】
前記ソフトウェアの識別子を前記署名サーバに与えるための前記手段が、前記ソフトウェアとともに含まれる前記第１のシグナチャの一部分を与えるための手段を含む、請求項58に記載のコンピュータ。
【請求項６４】

前記ソフトウェアの識別子を前記署名サーバに与えるための前記手段が、
前記署名サーバからソフトウェア識別子についての要求を受信するための手段と、
前記受信された要求に応答するソフトウェア識別子を前記署名サーバに与えるための手段と
を含む、請求項58に記載のコンピュータ。
【請求項６５】
前記ソフトウェアに関係する情報についての要求に応答するための前記手段が、前記ソフトウェアに対してハッシュ関数を実行し、結果を前記署名サーバに送信するための手段を含む、請求項58に記載のコンピュータ。
【請求項６６】
前記ソフトウェアに関係する情報についての要求に応答するための前記手段が、前記ソフトウェアによってアクセスされるリソースを前記署名サーバに対して識別するための手段を含む、請求項58に記載のコンピュータ。
【請求項６７】
前記ソフトウェアに関係する情報についての要求に応答するための前記手段が、
前記署名サーバから、前記コンピュータのユーザに関する情報についての要求を受信するための手段と、
前記要求された情報について前記ユーザにプロンプトを出すための手段と、
前記ユーザから応答を受信するための手段と、
前記ユーザの応答を前記署名サーバに送信するための手段と
を含む、請求項58に記載のコンピュータ。
【請求項６８】
前記受信された第２の検証方法と第２のシグナチャとを、前記ソフトウェアに関連する第2のシグナチャファイルに記憶するための手段をさらに含む、請求項58に記載のコンピュータ。
【請求項６９】
前記ソフトウェアに関連する第2のシグナチャがあるかどうかを判断するための手段と、
前記ソフトウェアに関連する第2のシグナチャがない場合、前記署名サーバへの接続を確立するための手段と
をさらに含む、請求項58に記載のコンピュータ。
【請求項７０】
サーバのプロセッサに、
クライアントコンピュータから、第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新するための要求を受信するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、
前記クライアントコンピュータから、検証されるべき前記ソフトウェアの識別子を受信するステップと、
検証されるべき前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するステップと、
前記クライアントコンピュータによって与えられた前記ソフトウェアに関係する情報に基づいて、検証されるべき前記ソフトウェアが信用できるかどうかを判断するステップと、
前記ソフトウェアを検証する際に前記クライアントコンピュータが使用するための、前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを前記クライアン

トコンピュータに送信するステップと
を含むステップを実行させるように構成されたプロセッサ実行可能命令を記憶した有形記憶媒体。
【請求項７１】
前記有形記憶媒体が、コンピュータのプロセッサに、前記クライアントコンピュータのユーザに関係する情報を前記クライアントコンピュータに要求するステップと、
前記クライアントコンピュータのユーザに関係する情報とユーザに関係する登録された情報を比較することで、前記ソフトウェアが信用できるかどうかを判断するステップとを含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項70に記載の有形記憶媒体。
【請求項７２】
前記有形記憶媒体が、コンピュータのプロセッサに、検証されるべき前記ソフトウェアの識別子についての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップを含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項70に記載の有形記憶媒体。
【請求項７３】
前記有形記憶媒体は、コンピュータのプロセッサに、ソフトウェア識別子として戻されるべき前記ソフトウェアの特定の部分を識別する、検証されるべき前記ソフトウェアの識別子についての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップを含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項70に記載の有形記憶媒体。
【請求項７４】
前記有形記憶媒体は、コンピュータのプロセッサに、前記ソフトウェアの前記識別子として使用されるべき、検証されるべき前記ソフトウェアの前記第１のシグナチャについての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップを含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項70に記載の有形記憶媒体。
【請求項７５】
前記有形記憶媒体は、コンピュータのプロセッサに、前記ソフトウェアの識別子として使用されるべき前記ソフトウェアの前記第１のシグナチャの特定の部分を識別する、検証されるべき前記ソフトウェアの前記識別子についての要求を、ネットワークインターフェース回路を介して前記クライアントコンピュータに送信するステップを含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項70に記載の有形記憶媒体。
【請求項７６】
検証されるべき前記ソフトウェアに関係する前記要求された情報が、前記ソフトウェアの一部分を含む、請求項70に記載の有形記憶媒体。
【請求項７７】
検証されるべき前記ソフトウェアに関係する前記要求された情報が、前記ソフトウェアに対して実行されたハッシュ関数の結果を含む、請求項70に記載の有形記憶媒体。
【請求項７８】
検証されるべき前記ソフトウェアに関係する前記要求された情報が、前記ソフトウェアによってアクセスされるクライアントコンピュータリソースの識別情報を含む、請求項70に記載の有形記憶媒体。
【請求項７９】
前記第２の検証方法は、前記ソフトウェアにハッシュ関数が適用される前に、前記ソフトウェアにアペンドされるかまたはプリペンドされる定数を含み、前記第２のシグナチャが、前記ソフトウェアに前記第２の検証方法を適用することから生じるシグナチャである、請求項70に記載の有形記憶媒体。
【請求項８０】

前記有形記憶媒体は、コンピュータのプロセッサに、検証されるべき前記ソフトウェアの前記受信された識別子を使用してデータファイルにアクセスするステップを含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有し、前記データファイルが、前記ソフトウェアを検証することに関係する情報を含み、

検証されるべき前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するステップが、前記アクセスされたデータファイル中で識別される情報を要求するステップを含み、

検証されるべき前記ソフトウェアが信用できるかどうかを判断するステップが、前記アクセスされたデータファイル中で識別されるステップを実行するステップを含み、

前記第２の検証方法と前記第２のシグナチャとが、前記アクセスされたデータファイルから取得される、請求項70に記載の有形記憶媒体。

【請求項８１】

前記有形記憶媒体が対策データベースを記憶しており、前記アクセスされたデータファイルが前記対策データベースからアクセスされ、前記対策データベースは、特定のアプリケーションに対する既知の脅威に打ち勝つために、適切な問合せステップと、検証方法と、シグナチャ値とを保持しているデータレコードを用いて更新される、請求項80に記載の有形記憶媒体。

【請求項８２】

コンピュータのプロセッサに、

第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新することを署名サーバに要求するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、

検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップと、

前記署名サーバから受信された、検証されるべき前記ソフトウェアに関係する情報についての要求に応答するステップと、

前記署名サーバから、前記ソフトウェアを検証する際に使用するための、検証されるべき前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを受信するステップと、

前記受信された第２の検証方法と第２のシグナチャとを記憶するステップと、

前記ソフトウェアを検証するために前記第２の検証方法と前記第２のシグナチャとを使用するステップと

を含むステップを実行させるように構成されたプロセッサ実行可能命令を記憶した有形記憶媒体。

【請求項８３】

前記有形記憶媒体は、コンピュータのプロセッサに、前記ソフトウェアに関係する情報についての要求に応答するステップが、前記ソフトウェアの一部分を前記署名サーバに送信するステップを含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。

【請求項８４】

前記有形記憶媒体は、コンピュータのプロセッサに、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、ソフトウェア識別子を送信するステップを含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。

【請求項８５】

前記有形記憶媒体は、コンピュータのプロセッサに、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記ソフトウェアの一部分を送信

するステップを含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。
【請求項８６】
前記有形記憶媒体は、コンピュータのプロセッサに、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記ソフトウェアとともに含まれる前記第１のシグナチャを送信するステップを含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。
【請求項８７】
前記有形記憶媒体は、コンピュータのプロセッサに、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記ソフトウェアとともに含まれる前記第１のシグナチャの一部分を送信するステップを含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。
【請求項８８】
前記有形記憶媒体は、コンピュータのプロセッサに、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信する前記ステップが、前記署名サーバから受信されたソフトウェア識別子についての要求に応答して実行され、前記署名サーバに送信された前記ソフトウェア識別子が、前記受信された要求に応答するようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。
【請求項８９】
前記有形記憶媒体は、コンピュータのプロセッサに、前記ソフトウェアに関係する情報についての要求に応答するステップが、前記ソフトウェアに対してハッシュ関数を実行し、結果を前記署名サーバに送信するステップを含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。
【請求項９０】
前記有形記憶媒体は、コンピュータのプロセッサに、前記ソフトウェアに関係する情報についての要求に応答するステップが、前記ソフトウェアによってアクセスされるリソースを前記署名サーバに対して識別するステップを含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。
【請求項９１】
前記有形記憶媒体は、コンピュータのプロセッサに、
前記署名サーバから、前記コンピュータのユーザに関する情報についての要求を受信するステップと、
前記要求された情報について前記ユーザにプロンプトを出すステップと、
前記ユーザから応答を受信するステップと、
ネットワークインターフェース接続を介して前記ユーザの応答を前記署名サーバに送信するステップと
を含むようなさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項82に記載の有形記憶媒体。
【請求項９２】
前記有形記憶媒体は、コンピュータのプロセッサに、前記受信された第２の検証方法と第２のシグナチャとを、前記ソフトウェアに関連する第2のシグナチャファイルに記憶するステップを含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項91に記載の有形記憶媒体。
【請求項９３】
前記有形記憶媒体が、コンピュータのプロセッサに、
検証されるべき前記ソフトウェアに関連する第2のシグナチャがあるかどうかを判断するステップと、
検証されるべき前記ソフトウェアに関連する第2のシグナチャがない場合、前記署名サ

ーバへの接続を確立するステップと
を含むさらなるステップを実行させるように構成されたプロセッサ実行可能命令を有する、請求項92に記載の有形記憶媒体。
【請求項９４】
ソフトウェアを検証し、署名するための方法であって、
クライアントコンピュータにおいて、前記ソフトウェアに関連する第2のシグナチャがあるかどうかを判断するステップと、
前記ソフトウェアに関連する第2のシグナチャがない場合、署名サーバへの接続を確立するステップと、
第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新することを前記署名サーバに要求するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、
検証されるべき前記ソフトウェアの識別子を前記署名サーバに与えるステップと、
前記ソフトウェアに関係する情報を前記クライアントコンピュータに要求するステップと、
前記署名サーバから受信された、前記ソフトウェアに関係する情報についての要求に応答するステップと、
前記署名サーバにおいて、前記クライアントコンピュータから受信された前記ソフトウェアに関係する情報に基づいて、前記ソフトウェアが信用できるかどうかを判断するステップと、
前記ソフトウェアを検証する際に前記クライアントコンピュータが使用するための、前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを前記クライアントコンピュータに送信するステップと、
前記署名サーバから前記第２の検証方法と前記第２のシグナチャとを受信するステップと、
前記受信された第２の検証方法と第２のシグナチャとを第2のシグナチャファイルに記憶するステップと、
前記ソフトウェアを検証するために前記第2のシグナチャファイル中の前記第２の検証方法と前記第２のシグナチャとを使用するステップと
を含む、方法。
【請求項９５】
ネットワークと、
前記ネットワークに結合された少なくとも1つのコンピューティングデバイスと、
前記ネットワークに結合された署名サーバと
を含む、システムであって、
前記少なくとも1つのコンピューティングデバイスが、
デバイスプロセッサと、
前記デバイスプロセッサと前記ネットワークとに結合されたネットワークインターフェース回路であって、前記デバイスプロセッサが前記ネットワークを介して通信することを可能にするように構成された、ネットワークインターフェース回路と、
前記デバイスプロセッサに結合されたメモリと
を含み、
前記デバイスプロセッサは、
前記ネットワークを介して、第１の検証方法および第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべきソフトウェアに対する第２の検証方法および第２のシグナチャに更新するため

の要求を前記署名サーバに送信するステップであって、前記第１のシグナチャおよび前記第２のシグナチャは前記ソフトウェアを署名することで生成され、前記第１の検証方法は前記第１のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法は前記２のシグナチャを用いて前記ソフトウェアを検証し、前記第２の検証方法が前記第１の検証方法と異なる、ステップと、

前記ネットワークを介して、検証されるべき前記ソフトウェアの識別子を前記署名サーバに送信するステップと、

前記ネットワークを介して前記署名サーバから受信された、検証されるべき前記ソフトウェアに関係する情報についての要求に応答するステップと、

前記ネットワークを介して、前記署名サーバから、前記ソフトウェアを検証する際に使用するための、前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを受信するステップと、

前記受信された第２の検証方法と第２のシグナチャとを前記メモリに記憶するステップと、

前記ソフトウェアを検証するために前記第２の検証方法と前記第２のシグナチャとを使用するステップと

を含むステップを実行するためのソフトウェア命令で構成され、

前記署名サーバは、

サーバプロセッサと、

前記サーバプロセッサと前記ネットワークとに結合されたサーバネットワークインターフェース回路であって、前記サーバプロセッサが前記ネットワークを介して通信することを可能にするように構成された、サーバネットワークインターフェース回路と、

前記サーバプロセッサに結合されたサーバメモリと

を含み、

前記サーバプロセッサが、

前記ネットワークを介して前記コンピューティングデバイスから、前記第１の検証方法および前記第１のシグナチャを、前記第１の検証方法および前記第１のシグナチャとともに元の署名サーバから以前配信された検証されるべき前記ソフトウェアに対する前記第２の検証方法および前記第２のシグナチャに更新するための要求を受信するステップと、

前記ネットワークを介して前記コンピューティングデバイスから、検証されるべき前記ソフトウェアの識別子を受信するステップと、

前記ネットワークを介して前記コンピューティングデバイスへの、検証されるべき前記ソフトウェアに関係する情報についての要求を送信するステップと、

前記コンピューティングデバイスによって与えられた前記ソフトウェアに関係する情報に基づいて、検証されるべき前記ソフトウェアが信用できるかどうかを判断するステップと、

前記ソフトウェアを検証する際に前記コンピューティングデバイスが使用するための、前記ソフトウェアに対する前記第２の検証方法と前記第２のシグナチャとを、前記ネットワークを介して前記コンピューティングデバイスに送信するステップと

を含むステップを実行するためのソフトウェア命令で構成された、システム。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、一般にコンピュータセキュリティに関し、より詳細には、コードおよびデータファイルが検証され、署名される方法を改善するための方法および装置に関する。

【背景技術】

【０００２】

コンピュータシステムでは、認証なしに変更されたマルウェアおよびソフトウェアから保護するための様々な機構およびシステムが実装される。最も一般的な方法のうちの1つ

は、コードまたはデータが実行またはアクセスされる前に、検査されるコードまたはデータのデジタルシグナチャを与えることである。セルラーネットワークのセキュリティを保護する必要性と、マルウェア保護ソフトウェアを実装するモバイルデバイスの限られた能力とにより、コードおよびデータにデジタル署名するための方法は、セルラー電話などのモバイルデバイス用に書かれたソフトウェアとともに広く使用されている。

【０００３】

デジタルシグナチャによって完全性保護されるコードまたはデータは、それが販売されるかまたは配信される前に証明局または「署名サーバ」によって「署名」される。署名付きコードまたはデータは、コードまたはデータが、それが「署名」されたときと同じであることを検証するために、検証アルゴリズムとともに使用され得る暗号化値を含んでいるデジタルシグナチャを含む。図1に示すように、典型的な実装形態では、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)は、ステップ2において、シグナチャを取得するためにコードまたはデータをアンパックすることと、ステップ4において、アプリケーションコードおよび/またはデータにわたってハッシュ関数を実行する(あるいは何らかの他の検証アルゴリズムを実行する)ことと、ステップ6において、シグナチャ内に含まれているハッシュ値を取得するためにシグナチャを復号することと、ステップ8において、シグナチャが有効な証明書によって署名されたことを確認することと、ステップ10において、得られたハッシュをデジタルシグナチャ内に含まれているハッシュと比較することとによって、アプリケーションが信用できることを確認する。テスト12において、2つのハッシュ値が等しい場合、そのコードまたはデータは信用される。モバイルデバイスなどのコンピュータは、それらが、デジタル証明書を復号することと、シグナチャが有効な証明書によって署名されていることを確認することとを可能にするために、証明書チェーンをメモリ中に含み得る。シグナチャが有効な証明書によって署名されており、ハッシュ値が等しい(すなわち、テスト12=「はい」)場合、そのコードまたはデータは信用され、クライアントは、そのコードを実行するか、またはそのデータを使用し得る。

【発明の概要】

【発明が解決しようとする課題】

【０００４】

コードおよびデータに署名することは、たいていの場合、効果的なセキュリティシールドを与える。しかしながら、このセキュリティレジームは静的であり、したがって、セキュリティ環境の変化に反応することができない。たとえば、クライアントコア暗号構成要素のいずれかの実装形態中に弱点が発見された場合、あるいはコアハッシュ関数または公開鍵アルゴリズムに対する新しい脆弱性が発見された場合、あるいは論理またはアプリケーションレベルエラーが明らかになった場合、クライアントデバイスは、それ自体をそのような脆弱性から保護することが可能でないことがある。さらに悪いことには、コードおよびデータが完成され、配信されてから数年後、およびクライアントデバイスが商業的に展開されてから数年後に、この状況が起こることがある。

【課題を解決するための手段】

【０００５】

様々な実施形態は、実行可能コンピュータファイルとデータファイルとを検証するために使用される署名および検証プロセスを更新し、改善するための方法およびシステムを提供する。様々な実施形態は、クライアントコンピュータデバイスによってアクセス可能な署名サーバの更新可能な環境を与えることによって、既存のコードおよびデータ署名システムを強化するためのフレキシブルな機構を提供する。この機能は、通常、コード署名システムの破局的な弱点になるであろうことの発見から回復するためのシステムおよび方法を提供する。様々な実施形態は、クライアントコンピュータデバイス上のソフトウェアが信用できるかどうかを判断し、更新された検証方法とシグナチャとをクライアントコンピュータに与えるように構成された署名サーバを含む。一実施形態では、更新された検証方法とシグナチャとは、第2のシグナチャファイル(「シグナチャ2ファイル」と本明細書では呼ぶ)中で与えられ得る。コンピューティングデバイスは、実行のためのアプリケーシ

ョンをアンパックするとき、第2のシグナチャファイルがアプリケーションファイルに関連するかどうかを検査し得る。そうでない場合には、コンピューティングデバイスは、ソフトウェアの第2のシグナチャファイルを要求するために署名サーバに接続し得る。次いで、署名サーバは、コンピューティングデバイスに、署名サーバが、ソフトウェアが信用できるかどうかを判断することを可能にするのに十分な、ソフトウェアに関係する情報を要求し得る。ソフトウェアが信用できると判断された場合、署名サーバは、これ以降ソフトウェアを検証する際に使用するための第2のシグナチャファイルをコンピュータデバイスに送信することができる。第2のシグナチャファイルは、新しいまたは変更された検証方法と新しいシグナチャとを含み得る。

【０００６】

本明細書に組み込まれ、本明細書の一部をなす添付の図面は、本発明の例示的な実施形態を示し、上記の概略的な説明および下記の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】

【０００７】

【図１】署名付きコードおよびデータを検証するための従来の方法のプロセスフロー図である。

【図２】ワイヤードおよびワイヤレスセルラーネットワークのシステムブロック図である。

【図３】一実施方法の概観を示すプロセスフロー図である。

【図４】一実施形態によるアプリケーションデータファイルの図である。

【図５】コンピューティングデバイス中のソフトウェアを検証するための実施方法のプロセスフロー図である。

【図６】コンピューティングデバイス中のソフトウェアを検証するための別の実施方法のプロセスフロー図である。

【図７】署名サーバ中のソフトウェアを検証し、クライアントデバイス上の検証方法およびシグナチャを更新するための一実施方法のプロセスフロー図である。

【図８】署名サーバ中のソフトウェアを検証し、クライアントデバイス上の検証方法およびシグナチャを更新するための別の実施方法のプロセスフロー図である。

【図９】様々な実施形態とともに使用するのに好適な例示的なモバイルデバイスの回路ブロック図である。

【図１０】様々な実施形態とともに使用するのに好適な例示的なパーソナルコンピュータの回路ブロック図である。

【発明を実施するための形態】

【０００８】

様々な実施形態について添付の図面を参照しながら詳細に説明する。可能な場合はいつでも、同じまたは同様の部分を指すために図面全体にわたって同じ参照番号を使用する。特定の例および実装形態になされる言及は、説明のためであり、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。

【０００９】

本明細書中の「例示的」という用語は、「例、事例、または例示として役立つこと」を意味するために使用する。本明細書で「例示的」と記載されたいかなる実装形態も、必ずしも他の実装形態よりも好ましいまたは有利であると解釈すべきではない。

【００１０】

本明細書では、「署名」および「シグナチャ」という用語は、当技術分野でよく知られている方法と様々な実施形態によって可能にされる方法の両方を含む、署名局によって検証されてからソフトウェアコードまたはデータあるいはコードとデータの組合せ(まとめて「ソフトウェア」)が変更されているかどうかをコンピューティングデバイスが判断することを可能にするための方法を指す。そのようなよく知られている方法は、一般に、コード、データ、またはコードとデータを、そのコードおよび/またはデータ中に含まれて

いる特定の情報にほぼ固有である、20バイト値などの大きい数に本質的に圧縮する、ハッシュまたは「フィンガープリント」値を生成するステップを含む。このハッシュまたはフィンガープリント値は、次いで、よく知られているPKI暗号化方式に従ってクライアントデバイスに記憶された公開鍵によって解読され得る秘密鍵を使用して暗号化される。暗号化されたハッシュまたはフィンガープリント値はシグナチャと呼ばれ、シグナチャを生成するプロセスは一般に署名と呼ばれる。いくつかの異なるタイプのハッシュアルゴリズムが使用され、様々な実施形態ではそのようなアルゴリズムのさらなる変更および改良が可能であり、したがって、「署名」および「シグナチャ」という用語の使用は、明細書の範囲または特許請求の範囲を特定の形態の暗号プロセス、ハッシュ関数または検証アルゴリズムに限定するものではない。

【００１１】

様々な実施形態は、(通常「コード」と呼ばれる)署名および検証アプリケーションソフトウェア命令、アプリケーションによって処理または使用されるデータ、ならびにコードとデータの両方を含むアプリケーションファイルに適用され得る。したがって、本明細書では、「ソフトウェア」という用語は、概して、代替的にコードとデータの両方、ならびに結合的にコードとデータを指すために使用される。以下の説明および特許請求の範囲におけるソフトウェアへの言及は、データを除外するもの、または実行可能命令を必要するものと解釈すべきではない。

【００１２】

本明細書で使用する「モバイルデバイス」、「モバイルハンドセット」、「ハンドセット」および「ハンドヘルドデバイス」という用語は、セルラー電話、ワイヤレスモデムをもつ携帯情報端末(PDA)、ワイヤレス電子メール受信機(たとえば、Blackberry(登録商標)およびTreo(登録商標)デバイス)、マルチメディアインターネット対応セルラー電話(たとえば、iPhone(登録商標))、ワイヤレス受話器および同様のパーソナル電子デバイスのうちのいずれか1つまたはすべてを指す。好ましい一実施形態では、モバイルデバイスはセルラーハンドセットデバイス(たとえば、セルフォン)である。ただし、様々な実施形態は、様々なテキストデータ入力方法を実装するコンピューティングデバイス上で実装され得るので、セルラー電話通信機能は不要である。

【００１３】

本明細書で使用する「コンピュータ」および「コンピューティングデバイス」という用語は、たとえば、パーソナルコンピュータ、ラップトップコンピュータ、モバイルデバイス(たとえば、セルラー電話、個人情報端末(PDA)、パームトップコンピュータ、および多機能モバイルデバイス)、メインフレームコンピュータ、サーバ、ならびに統合コンピューティングシステムを含む、存在し得るか、または将来展開されることになる、プログラマブルコンピュータの任意の形態を包含するものである。コンピュータは、一般に、メモリ回路に結合されたソフトウェアプログラマブルプロセッサを含むが、図9～図10に関して以下で説明する構成要素をさらに含み得る。

【００１４】

本明細書で使用する「サーバ」という用語は、ネットワークへのアクセスをもつクライアントサーバアーキテクチャにおいて動作するように構成された様々な市販のコンピュータシステムのいずれかを指す。特に、「サーバ」という用語は、一般に、プロセッサと、メモリ(たとえば、ハードディスクメモリ)と、サーバプロセッサをインターネットなどのネットワークに接続するように構成されたネットワークインターフェース回路とを含むネットワークサーバ、特にインターネットアクセス可能なサーバを指す。サーバは、セキュリティのために専用ハードウェアをも含み得る。

【００１５】

本明細書で使用する「クライアント」という用語は、インターネット接続またはコンピュータプログラムなど、サーバと通信するためのコンピュータプログラムおよび手段を実行することが可能なプロセッサをもつ、モバイルデバイスまたはパーソナルコンピュータなどのコンピューティングデバイスを指すか、あるいは、インターネット接続など、他の

オペレーティングシステム上で実行するコンピュータプログラムと通信するためのリンクを含む、ウェブブラウザまたはオペレーティングシステムなどのコンピュータプログラムを指す。「クライアント」および「サーバ」という用語は、本質的に説明的であり、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。

【００１６】

多くのコンピューティング環境は、妥当性検査されたソフトウェアのみを実行することによってコンピューティングデバイスおよびネットワークを保護する。たとえば、Binary Runtime Environment for Wireless(BREW(登録商標))システムを動作させるセルラー電話などのモバイルデバイスは、各アプリケーションコードを実行する前にそのアプリケーションコードを検証する。これにより、モバイルデバイスとそれらのモバイルデバイスが接続しているセルラーデータネットワークとが、悪意のあるソフトウェアによって攻撃または利用されることを防ぐ。この環境をサポートするために、証明局は、配信より前に、各ソフトウェアアプリケーションがマルウェアおよび利用可能な脆弱性がないことを確認するためにソフトウェアアプリケーションをテストする。アプリケーションは、安全であると判断された場合、証明局によって署名され、検証されたソフトウェアパッケージとして識別される。次いで、モバイルデバイスは、コードを実行するより前に、証明済みソフトウェアをダウンロードし、検証することができる。上記で説明し、図1に示すように、以前のモバイルデバイスは、ステップ2において、コードおよびシグナチャファイルを取得するためにアプリケーションをアンパックすることと、ステップ4において、アンパックされたコードにわたってハッシュアルゴリズムを実行することと、ステップ6において、ハッシュ値を取得するためにシグナチャファイルを解読することと、ステップ8において、シグナチャが有効な証明書で署名されたことを確認することと、ステップ10において、計算されたハッシュ値を、これらのシグナチャを用いて与えられたハッシュ値と比較することと、テスト12において、2つのハッシュ値が等しい場合、ソフトウェアを信用できるとして扱うこととによって、証明済みソフトウェアを検証した。

【００１７】

ソフトウェアを証明するためのこのシステムは、高いセキュリティレベルを提供するが、それでも時間とともに旧式になり攻撃を受けることに対して潜在的に脆弱である。図1に示すセキュリティ方法は静的であり、ソフトウェアアプリケーションに関連するハッシュアルゴリズムおよびシグナチャは、モバイルデバイスおよびソフトウェアの寿命全体にわたって同じままである。時間とともに、暗号セキュリティ方法は、その方法の脆弱性が知られるにつれて「損耗」し得る。ソフトウェアに署名する使用されるために証明書は損なわれ、署名および検証プロセスにおいて使用されるハッシュ関数は攻撃されることがある。したがって、従来のコードおよびデータ署名方法のフレキシビリティのない性質は、時間とともにそれらの方法を攻撃に対して潜在的に脆弱にしている。

【００１８】

これまでは、暗号構成要素またはハッシュ関数に弱点が発見されたとき、検証アルゴリズムおよび関連するシグナチャを変更するためにアプリケーションソフトウェアを更新する必要があったであろう。しかしながら、新しいアプリケーションファイルをダウンロードすることは時間がかかり、大量の帯域幅を消費する。

【００１９】

様々な実施形態は、ソフトウェアが配信され、クライアントデバイスが製造業者を離れた後、署名サーバが、クライアントデバイス中に実装された検証プロセスを更新し、制御することを可能にすることによってコードおよびデータ署名機構の脆弱性を克服するためのシステムおよび方法を提供する。概観では、署名サーバは、ワイヤードまたはワイヤレスネットワークを介してクライアントコンピュータによってアクセスされ得、接続が確立されると、署名サーバは、特定のソフトウェアを検証するためにクライアントコンピュータに情報を要求しおよび/または一連のアクションを命令することができる。クライアントコンピュータは、要求されたアクションを実行し、情報を解釈することを試みることなしに結果をサーバに送信する。受信された情報に基づいて、サーバは、それがソフトウェ

アを信用するかどうかを判断することができる。信用する場合、署名サーバは、コード/データソフトウェアを受け付けるようにクライアントコンピュータに告げることができ、また、随意に、署名サーバが信用でき、クライアントコンピュータが実装するのに信用できる、コンストラクト(すなわち、検証アルゴリズム)を使用して、コード/データのための新しいシグナチャを生成することができる。最後に、クライアントコンピュータは、ソフトウェアの以後の検証のために新しいシグナチャおよび新しい検証アルゴリズムを使用することができる。代替的に、サーバは、ソフトウェアを単に拒否するようにクライアントコンピュータに告げることができる。これは、クライアントコンピュータの脆弱性が非常に悪いために、サーバが、クライアントコンピュータ情報さえ信用することができないか、または必要とされるアクションをクライアントコンピュータがセキュアに実施することを信用することができない場合に起こり得る。

【００２０】

様々な実施形態は、図2に示すものなど、ワイヤードデータネットワークとワイヤレスデータネットワークの両方を含む通信ネットワーク20において実装され得る。通信ネットワーク20は、モバイルハンドセット30およびパーソナルコンピュータ28などのクライアントデバイスが、コードおよび/またはデータを検証し、そのコードおよび/またはデータを検証するための更新されたシグナチャおよび検証方法を受信するために署名サーバ26にアクセスすることを可能にする、インターネット25とセルラーネットワークとを含み得る。

【００２１】

この例示的なネットワーク20では、基地局21は、モバイル交換センター(MSC)22など、ネットワークを動作させるために必要とされる要素を含むセルラーネットワークの一部である。動作中、MSC22は、モバイルハンドセット30が発呼し、呼を受信しているとき、基地局21を介してモバイルハンドセット30との間で呼およびメッセージをルーティングすることが可能である。MSC22はまた、モバイルハンドセット30が呼に関与しているとき、電話地上線トランク(図示せず)への接続を行う。

【００２２】

さらに、MSCは、インターネット25に結合されたサーバゲートウェイ23に結合され得る。サーバゲートウェイ23を通して、モバイルハンドセット30は、インターネットを介して、署名サーバ26、ならびにソフトウェアアプリケーションをダウンロードすることができるコンテンツサーバ27と通信し得る。また、パーソナルコンピュータ28は、インターネットサービスプロバイダによって提供されるものなど、従来のインターネットアクセス方法を使用してインターネットを介して署名サーバ26およびコンテンツサーバ27と通信し得る。そのような通信は、ファイル転送プロトコル(FTP)、ハイパーテキスト転送プロトコル(HTTP)、およびハイパーテキスト転送プロトコルオーバーセキュアソケットレイヤ(HTTP)を使用して送信され得る。通信は、ハイパーテキストマークアップ言語(HTML)と、画像ファイルと、Ｊａｖａ（登録商標）Scriptなどの言語のクライアントサイドスクリプトとを含む、様々なタイプのファイルからなることができる。さらに、そのようなメッセージは、デジタル証明書および署名鍵など、様々なセキュリティ方式に関係するファイルを含み得る。さらに、この例示的なネットワーク20は、この例示的なネットワークにおける署名サーバ26およびコンテンツサーバ27など、ウェブサーバにデジタル証明書ならびに公開鍵および秘密鍵を発行する能力を含む証明局として働くように構成されたサーバである証明局(CA)サーバ24を含む。さらに、CAサーバ24は、CAサーバ24のルート証明書のセットを最新に保つためにセルラーネットワークを介してモバイルハンドセット30と通信し得る。

【００２３】

モバイルデバイス30は、ゲートウェイサーバ23を介してインターネット25に結合する基地局21へのワイヤレスデータネットワークアクセスを介して署名サーバ26と通信し得る。さらに、モバイルデバイス30は、データケーブル29またはローカルエリアワイヤレスネットワークを介して(たとえば、Bluetooth(登録商標)トランシーバを介して)パーソナルコンピュータ28に接続されることによってインターネットサーバと通信し得、パーソナルコンピュータ28は、様々なインターネット接続(たとえば、電話モデム、ケーブルモデム、W

iFi、光ファイバ接続など)を介してインターネットにアクセスする。モバイルデバイス30をパーソナルコンピュータ28に接続することによって、アプリケーションソフトウェアおよびデータがコンテンツサーバ27からモバイルデバイス30にアップロードされ得る。さらに、モバイルデバイス30は、ケーブル29を介してインターネット25に接続されたパーソナルコンピュータ28と通信することによって、ワイヤードインターネット接続を介して署名サーバ26と通信することができる。このようにして、ワイヤレス通信ネットワークよりも高いデータ転送レートを有し得るワイヤードデータリンクを使用して、大量のソフトウェアデータおよびコードをモバイルデバイス30と署名サーバ26との間で交換することができる。

【００２４】

様々な実施形態の概要が、図3に示すプロセスフロー図、および図4に示す例示的なアプリケーションデータファイルに示されている。様々な実施形態では、署名付きソフトウェアを検証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る方法は、署名サーバ26によって生成された第2のシグナチャファイルを含むように変更される。ソフトウェアを検証するための従来の方法の場合のように、ステップ60において、コンピューティングデバイス28、30は、検証されるべきコードをアンパックし、シグナチャファイルにアクセスする。図4に示すように、アプリケーションファイル50は、実行可能コードファイル52と、アプリケーションを実行するために必要とされる関連するデータ54と、シグナチャファイル56と、シグナチャ2(「sig2」)ファイル58とを含み得る。従来の署名付きアプリケーションは、実行可能ファイル52と、データファイル54と、単一のシグナチャファイル56とを含む。追加されたシグナチャ2ファイル58は、新しいまたは変更された検証アルゴリズム58aと新しいシグナチャ58bとを含み得る。新しいまたは変更された検証アルゴリズム58aは、以下でより詳細に説明するように、XMLコマンドなどの実行可能コマンドの形態であるかまたは空であり得る。新しいシグナチャ58bは、新しいまたは変更された検証アルゴリズム58aをアプリケーションコードファイル52および/またはアプリケーションデータ54に適用することによって署名サーバ26によって生成されるシグナチャである。アプリケーションコードファイル52およびアプリケーションデータファイル54は、本明細書では概括的にソフトウェアと呼ぶ。

【００２５】

ステップ60において、ソフトウェアをアンパックし、シグナチャファイルにアクセスすると、テスト62において、コンピューティングデバイス28、30は、シグナチャ2ファイル58が存在するかどうかを判断する。シグナチャ2ファイル58がアプリケーションファイル50中に存在する場合(すなわち、テスト62=「はい」)、ステップ64において、コンピューティングデバイス28、30は、新しいまたは変更された検証アルゴリズム58aを使用して検証値を生成し、その検証値をシグナチャ2値と比較する。シグナチャ2値は、有効な証明書に関連する公開鍵を使用して、シグナチャ2ファイル58のシグナチャ2の一部分58bに記憶されたデータを解読することによって取得される。新しいまたは変更された検証アルゴリズムをアプリケーションソフトウェアに適用することによって生成された値が、シグナチャ2ファイル58から取得されたシグナチャ2値に一致する場合、ソフトウェアは検証され、ステップ66において、コンピューティングデバイス28、30は、コードを実行することまたはデータを使用することに進む。

【００２６】

一方、シグナチャ2ファイル58がアプリケーションファイル50中に存在しない場合(すなわち、テスト62=「いいえ」)、ステップ68において、コンピューティングデバイス28、30は、シグナチャ2ファイルを要求するために署名サーバ26への通信リンクを確立することを試みる。ステップ70において、署名サーバ26は、検証されるべきソフトウェアの識別情報を含む要求を受信し、要求元デバイスにソフトウェアに関する情報を要求することに進む。このプロセスでは、署名サーバ26は、ソフトウェアの様々なサンプルについて要求元デバイス28、30に要請し得、ならびに検証されるべきソフトウェアに対して様々な操作のいずれかを実行するように要求元コンピューティングデバイス28、30に依頼し得る。署名

サーバ26によってコンピューティングデバイス28、30に要求される特定の情報および処理は、その時における既知のまたは潜在的なセキュリティ脅威に基づいて判断され得る。署名サーバ26は、要求した情報を受信し、ステップ72において、その受信された情報を使用して、ソフトウェアが変更されているかあるいは信用できるかを確認する。ソフトウェアが検証されたかまたは信用できると判断された場合、署名サーバ26は、ステップ74において、シグナチャ2ファイルの形態の新しい検証アルゴリズムと新しいシグナチャ(すなわち、シグナチャ2)とを要求元コンピューティングデバイス28、30に与え得る。要求元コンピューティングデバイス28、30は、署名サーバ26からシグナチャ2ファイルを受信し、(アプリケーションコードおよびデータなどをもつ)ファイルを記憶し、次いでステップ64において、そのシグナチャ2ファイル58を使用してソフトウェアを検証することができる。

【００２７】

ステップ74において署名サーバ26によってコンピューティングデバイス28、30に与えられたシグナチャ2ファイル58は、関連するアプリケーションコードおよび/またはデータの以後のすべての検証のために使用され得る。したがって、別のイベントがコンピューティングデバイス28、30に署名サーバ26に連絡するように促すのでなければ、ソフトウェアの以後のすべての検証のためにシグナチャ2ファイル58が使用されることになる。このプロセスは、ソフトウェアディストリビュータが、通常のチャネルを通してそのソフトウェアを販売および配信し、次いで、署名サーバ26を使用して後でソフトウェア検証プロセスおよびシグナチャを更新することを可能にする。いくつかの実施形態では、コンピューティングデバイス28、30は、シグナチャ2ファイルに必要とされる更新があるかどうかを判断するために後で再び署名サーバ26に連絡し、それによって定期的セキュリティ更新が実装されることを可能にし得る。

【００２８】

随意に、シグナチャファイルは存在しないが(すなわち、テスト62=「いいえ」)、シグナチャサーバ26との連絡を確立することができないとコンピューティングデバイス28、30が判断した場合、ステップ64において、コンピューティングデバイス28、30は、アプリケーションファイル50とともに最初に与えられた1次シグナチャファイル56を使用してソフトウェアを検証することに進むことができる。したがって、署名サーバ26が利用可能でないかまたはネットワークアクセスが達成され得ない場合、それにもかかわらず、アプリケーションソフトウェアは、従来技術において検証可能であり得るのと同じ範囲まで検証され得る。そのような検証は、セキュリティ漏洩に対して脆弱になり得るが、それでも、コンピューティングデバイス28、30がシグナチャ2ファイル58を取得するために署名サーバ26との連絡を確立することができるようになる時間まで、ある程度のセキュリティ対策を施す。

【００２９】

図3に示すように、様々な実施形態は、コンピューティングデバイス28、30と署名サーバ26とがそれぞれ検証プロセスの一部分を実行する協働検証環境またはシステムを提供する。概要では、コンピューティングデバイス28、30は、サーバがソフトウェアを信用できることを確認することを可能にするために十分な情報を署名サーバ26に与え、署名サーバ26は、コンピューティングデバイス28、30が、アプリケーションが実行されるたびに署名サーバ26に連絡する必要なしに後でコードおよびデータを検証することを可能にするために、更新された検証ルーチンおよびシグナチャを与える。

【００３０】

コンピューティングデバイス28、30において実行され得る方法ステップの一実施形態を図5に示す。この実施形態では、シグナチャ2ファイル58の存在は、ソフトウェアの処理における基礎ステップとしてテストされる。この実施形態では、コンピューティングデバイス28、30のプロセッサは、ステップ100において、関連するシグナチャファイルにアクセスするためにアプリケーションソフトウェアをアンパックする。いくつかの実装形態では、ステップ100における、シグナチャファイルを取得するためにソフトウェアをアンパックするプロセスは、シグナチャ2ファイルが維持されているメモリの別の部分にアクセス

し得る(すなわち、シグナチャ2ファイルは、図4に示すようにアプリケーションファイル50内にまたはアプリケーションファイル50に隣接して記憶される必要はない)。テスト102において、シグナチャ2ファイルが存在するかどうかを判断するために、アプリケーションに関連するシグナチャファイルを検査する。コンピューティングデバイス28、30が特定のアプリケーションについて署名サーバ26にすでに連絡した後では、シグナチャ2ファイルは通常存在する。シグナチャ2ファイルが存在する場合、ステップ104において、プロセッサは、アンパックされたソフトウェアに対してシグナチャ2ファイル内の検証ルーチンを実行する。以下で説明するように、シグナチャ2ファイル中の検証ルーチンは、標準的なハッシュ関数、変更された標準的なハッシュ関数、または必要に応じて特定のアプリケーションに対して発生する脅威に打ち勝つためのまったく異なる独自の検証アルゴリズムであり得る。プロセッサはまた、ステップ106において、検証値を取得するためにシグナチャ2ファイル内のデジタルシグナチャを解読する。この復号は、署名サーバ26に発行されたデジタル証明書に対応する公開鍵を使用して実行され得る。シグナチャ2ファイルを解読するプロセスでは、プロセッサはまた、ステップ108において、シグナチャが有効な証明書の保持者によって署名されたことを確認し得る。この検証は、シグナチャ2ファイルが信用できる署名局によって受信されたことを確認する。次いでステップ110において、生成された検証値を、シグナチャ2ファイルを解読することによって取得された検証値と比較する。2つの値が等しい場合(すなわち、テスト112=「はい」)、ステップ114において、プロセッサにはソフトウェアが信用できることを通知され、したがって実行が進められる。しかしながら、2つの値が等しくない場合(すなわち、テスト112=「いいえ」)、プロセッサにはソフトウェアが信用できないことが通知され、たいていの場合、アプリケーションコードは実行を阻止されるか、またはデータはメモリから削除される。

【００３１】

アンパックされたソフトウェアを検討することにより、アプリケーションファイル50中に存在するかアプリケーションファイル50に対応するシグナチャ2ファイルがないことが明らかになった場合(すなわち、テスト102=「いいえ」)、ステップ116において、コンピューティングデバイス28、30のプロセッサは署名サーバ26への接続を確立し得る。ネットワークがコンピューティングデバイス28、30に利用可能でないとき、または署名サーバ26が接続要求に応答しないときなど、署名サーバ26への接続が達成されなかった場合(すなわち、テスト118=「いいえ」)、コンピューティングデバイスのプロセッサは、図1を参照しながら上記で説明した従来の検証方法を使用してソフトウェアを検証し得る。

【００３２】

署名サーバ26への接続が確立された場合(すなわち、テスト118=「はい」)、ステップ120において、プロセッサは、シグナチャ2ファイルを要求し、検証情報が要求されている特定のアプリケーションソフトウェアを識別し得る。この情報は、署名サーバ26が、署名サーバのメモリ内で(たとえば、署名サーバ中に維持されている対策データベース内で)対応するデータファイルの位置を特定することを可能にするために十分でなければならない。たとえば、コンピューティングデバイス28、30は、特定のアプリケーションソフトウェアの名前、識別子および/または通し番号を与え得る。一実施形態では、署名サーバ26に与えられるソフトウェア識別情報は、一意の識別子(たとえば、通し番号)、ソフトウェア製品識別子(たとえば、ソフトウェア製品およびバージョン番号)、ソフトウェアとともに与えられた元のデジタルシグナチャ、元のデジタルシグナチャのいくつかの部分またはセグメント(たとえば、署名サーバ26によって要求される特定の部分またはセグメント)、あるいはソフトウェア自体のいくつかの部分またはセグメントのうちのいずれか1つまたはそれらの組合せであり得る。(識別子の上述の形態のうちのいずれかを含む)ソフトウェア識別子を与えるステップは、コンピューティングデバイス28、30によって自律的に達成され得、または接続が確立された後に署名サーバ26から受信された特定のソフトウェア識別子の要求を受信することに応答して達成され得る。コンピューティングデバイス28、30はまた、実行されるべき検証のタイプを署名サーバ26がより良く判断することを可能にするた

めに、モデル識別子を与えることなどによって、コンピューティングデバイス28、30自体の型またはモデルを署名サーバ26に通知し得る。その情報が署名サーバ26に与えられると、コンピューティングデバイス28、30は、署名サーバ26から受信し得る情報の要求を受信し、それに応答するためにスタンバイし得る。アプリケーションソフトウェアとともに与えられた元のシグナチャによってもたらされる検証セキュリティを弱めるかまたは危険にさらす既知の脅威が存在する場合に起こり得るように、署名サーバ26が特定のソフトウェアを確認するように構成されている場合、ステップ122において、署名サーバ26は、ソフトウェアに関する
情報についての1つまたは複数の要求をコンピューティングデバイス28、30に送信する。しかしながら、ソフトウェアに対する既知の脅威が存在しない場合に起こり得るように、特定のアプリケーションソフトウェアについての更新された検証ルーチンまたはシグナチャが署名サーバ26に与えられない場合、ステップ122において、署名サーバ26は、情報を要求するステップをスキップし、単に、1次シグナチャおよび検証方法を使用する命令を含むかまたは標準的な1次検証ハッシュルーチンを使用する1次シグナチャおよび命令を含んでいるシグナチャ2ファイルを与え得る。

【００３３】

コンピューティングデバイス28、30が署名サーバ26からデータについての要求を受信すると、ステップ122において、コンピューティングデバイス28、30のプロセッサは、要求されたステップを実行し、確立されたネットワーク接続を介して要求されたデータを戻す。署名サーバ26は、特定のソフトウェアを検証するために、必要に応じてコンピューティングデバイス28、30の広範囲のデータおよび処理ステップを要求するように構成され得ることが想定される。たとえば、署名サーバ26は、ファイル内の特定の位置(たとえば、ファイルの最初の50バイト、最後の50バイト、および中間部分からの50バイト)において取り出されたソフトウェアのサンプルを要求し得る。代替的に、署名サーバ26は、特定のハッシュ関数を実行し、得られた値を署名サーバ26に与えることなど、ソフトウェアに対して操作を実行するようにコンピューティングデバイス28、30に要求し得る。コンピューティングデバイス28、30に要求される特定の情報は、特定のソフトウェアが信用できるかまたは実行しても安全であると高信頼度で判断するために署名サーバ26が使用することができる情報を与えるように構成されることが想定される。したがって、署名サーバを打ち破ることを試みている当事者が、コンピューティングデバイス28、30によってどのデータが要求され得、またはどの動作が実行され得るかを正確に予想することが不可能になり得るように、署名サーバ26によって要求される特定の情報は要求ごとに異なり得る。さらに、署名サーバによって利用される検証プロセスが予想されるかまたは他の方法で脅かされることが不可能となるように、署名サーバ26によって要求されるデータの使用は機密に維持され得る。したがって、署名サーバ26は、ソフトウェアを検証する方法をさらにマスキングするように、ソフトウェアを確認するために使用するデータよりも多くのデータを要求し得る。

【００３４】

一例として、署名サーバ26は、(a)ソフトウェアコンテンツに値Xをプリペンドし、(b)そのように変更されたソフトウェアコンテンツにハッシュアルゴリズムYを適用し、(c)そのように変更されたソフトウェアコンテンツにハッシュアルゴリズムZを適用し、(d)ハッシュYプロセスとハッシュZプロセスとの結果からハッシュする合成物を作成し、(e)合成ハッシュ値を報告し、(f)ファイルオフセットA、BおよびCにおける値を報告し、(g)ソフトウェアが要求しているランタイム特権を報告し、(h)ソフトウェアが求めているライセンス条件に報告し、(i)ソフトウェアが登録される通知またはMIMEタイプを報告し、(j)クライアントコンピュータが使用している暗号ライブラリのバージョンを報告するという一連のステップを実行するようにコンピューティングデバイス28、30に命令することがあり得る。そのような要求に応答して、ステップ122において、コンピューティングデバイス28、30は、要求されたプロセスを実行し、署名サーバ26に要求されたデータを戻す。次いで、署名サーバ26は、この(または他の)情報および処理結果の一部または全部を使用して

、ソフトウェアが変更されているか、マルウェアを含んでいるか、または他の方法で信用できないかどうかを判断することができる。署名サーバ26はまた、この情報を使用して、コンピューティングデバイス28、30がいくつかの検証プロセスを実行することが可能であるか、または実行するのに信用できるかどうかを判断し得る。

【００３５】

ステップ122における、署名サーバ26が情報を要求し、クライアントコンピューティングデバイス28、30がその要求に応じるプロセスは、自律的または半自律的に達成され得る。一実施形態では、クライアントコンピューティングデバイス28、30は、ユーザの関与なしに、署名サーバ26からの情報についての要求に自動的に応答する。そのような一実施形態では、コンピューティングデバイス28、30と署名サーバ26との間で情報を交換するプロセスは、初めてアプリケーションが起動されたときの余分の遅延のみに気づき得るユーザには見えない。別の実施形態では、コンピューティングデバイス28、30と署名サーバ26との間で情報を交換するプロセスは何らかのユーザ対話を含み得る。たとえば、署名サーバ26は、パッケージング上にまたはダウンロード時に与えられ得るライセンス番号または通し番号など、製品に関する情報を入力するようにユーザに要求し得る。別の例として、署名サーバ26は、ユーザ自身を識別し、ユーザのアドレス、モバイルデバイスの電話番号および電子メールアドレスを入力するようにユーザに要求し得る。ユーザに関するこの追加の情報は、署名サーバ26がユーザを登録し、シグナチャ2ファイルの展開を追跡することを可能にし得る。また、ユーザの電子メールアドレスおよび/またはモバイルデバイスの電話番号の記録を有することは、著しいセキュリティ脅威が識別されたときに必要になり得る、新しいシグナチャ2ファイル更新を取得しなければならないという通知を署名サーバ26がコンピューティングデバイス28、30に送出することを可能にし得る。半自律的な実施形態では、署名サーバ26はまた、検証されるべきソフトウェアのソースに関する質問をユーザに提示し得、そのような情報により、署名サーバ26はソフトウェアが信用できるかどうかを判断することが可能になるであろう。このプロセスの一部として署名サーバ26がユーザ登録情報を収集する実装形態では、署名サーバ26は、1次シグナチャおよび検証方法における既知の脅威または弱点がないときでも、コンピューティングデバイス28、30に情報を要求し得る。いずれの実施形態でも、送信サーバ26は、要求されたデータを取得すること、または要求されたデータを入力するようにユーザに促す表示を生成することをコンピューティングデバイス28
、30に行わせるために必要な命令(たとえば、XMLコード)を与え得る。

【００３６】

コンピューティングデバイス28、30によって与えられた情報を使用して、署名サーバ26は、新しいシグナチャ2ファイル58を準備し、コンピューティングデバイス28、30に送信することができ、コンピューティングデバイス28、30は、ステップ124においてそのシグナチャ2ファイル58を受信する。ソフトウェアが既知の脅威に直面していないか、または配信されるときにソフトウェアとともに与えられた1次シグナチャおよび検証方法が適切であった場合、コンピューティングデバイス28、30に与えられるシグナチャ2ファイル58は、1次シグナチャおよび検証方法を使用するコマンドにすぎない。1次検証方法において使用されるソフトウェアまたはハッシュ関数に対して小さな攻撃がある状況では、シグナチャ2ファイル58は、新しいシグナチャとともに、ハッシュ関数を実行するより前に検証されるべきソフトウェアにランダム値をアペンドまたはプリペンドすることなど、1次検証方法への単純な変更を含み得る。ソフトウェアまたはハッシュ関数に対して大きな攻撃がある状況では、シグナチャ2ファイル58は、新しいシグナチャとともに、異なるハッシュ関数またはまったく新しい検証方法を使用する命令を含み得る。いくつかの状況では、まったく新しいハッシュ関数がダウンロードされる必要があり得る。新しいハッシュ関数の有線または無線ダウンロードを可能にするための十分な帯域幅がある状況では、そのプロセスは直ちに実施され得る。新しいハッシュ関数の無線ダウンロードを可能にするための十分な帯域幅がない状況では、署名サーバ26は、コンピューティングデバイス28、30がダウンロードを完了するための十分な帯域幅をもつネットワークに接続された後の時間に

おいて新しいハッシュ関数をダウンロードすることを可能にする命令をコンピューティングデバイス28、30に与え得る。署名サーバ26は、既知の脅威が現れたとき、そのような脅威に適合されるシグナチャ2ファイル58を与えるための命令で構成されることが想定される。署名サーバ26が、デバイスおよびアプリケーションソフトウェアが展開されたかなり後に検証方法およびシグナチャをコンピューティングデバイス28、30に送信することを可能にするフレキシブルなプロセスを提供することによって、様々な実施形態は、セキュリティ対策が脅威の展開に応答す
ることを可能にする。

【００３７】

署名サーバ26が新しいシグナチャ2ファイル58を送信し、ファイルコンテンツがコンピューティングデバイス28、30によって受信されると、ステップ124において、シグナチャ2ファイル58はメモリに記憶される。一実施形態では、新しいシグナチャ2ファイル58は、図4に示すように対応するアプリケーションファイル50とともに記憶される。代替実施形態では、新しいシグナチャ2ファイル58は、シグナチャ2ファイルを保持するために予約されたメモリの別の部分に記憶される。シグナチャ2ファイル58が受信され、メモリに記憶されると、コンピューティングデバイス28、30のプロセッサは、上記で説明したように、ステップ100～112を実行することによってアプリケーションファイルを検証することに進むことができる。

【００３８】

コンピューティングデバイス28、30において実行され得る処理の代替実施形態を図6に示す。この実施形態では、シグナチャ2ファイル58が存在しないとき、プロセッサは、プロセッサが署名サーバ26と連絡をとる前にソフトウェアが生成されたとき、そのソフトウェアは信用できたことを確認するための1次検証方法を実行する。

【００３９】

図6を参照すると、この実施形態は、図5を参照しながら上記で説明したのと同じステップの多くを含む。実行のためのアプリケーションが選択されると、ステップ100において、シグナチャファイルにアクセスするためにそのソフトウェアをアンパックする。テスト102において、シグナチャ2ファイル58が存在する場合、図5を参照しながら上記で説明したように、ステップ104～114において、対応する検証方法を実施し、適切に検証された場合、ソフトウェアを実行する。しかしながら、シグナチャ2ファイルが存在しない場合(すなわち、テスト102=「いいえ」)、コンピューティングデバイス28、30のプロセッサは、ソフトウェアの配信のときにそのソフトウェアとともに与えられた1次検証手順を実行することに進む。上記で説明したように、これは、ステップ130において、アンパックされたソフトウェアに対して、特定のハッシュ関数などの1次検証方法を実行することを含み得る。ステップ132において、ハッシュ値など、シグナチャ内に含まれている1次検証値を取得するためにシグナチャファイル56を解読する。ステップ134において、シグナチャが有効な証明書によって署名されたかを判断するために、シグナチャを評価する。ステップ136において、計算された検証値を、シグナチャ内に含まれている1次検証値と比較する。テスト140において、2つの検証値が等しくない場合、ソフトウェアは信用できず、ソフトウェアに対するさらなる処理は行わなくてよい。

【００４０】

一方、2つの検証値が等しい場合(すなわち、テスト140=「はい」)、これは、少なくともある時間ではソフトウェアが信用できたことを示し、したがって、コンピューティングデバイス28、30のプロセッサは、ステップ116において、署名サーバ26への接続を確立することを試みる。上記で説明したように、この接続は、インターネットなど、署名サーバ26が常駐するネットワークへのアクセスをもつ任意のワイヤードまたはワイヤレスネットワークにより得る。署名サーバ26への接続が可能でない場合(すなわち、テスト118=「いいえ」)、ソフトウェアは1次検証方法を通して検証されているので、ステップ114においてソフトウェアを実行し得る。署名サーバへの接続が確立された場合(すなわち、テスト118=「はい」)、図5を参照しながら上記でより十分に説明したように、コンピューティン

グデバイス28、30のプロセッサは、ステップ120において、(ソフトウェア識別子を提供することなどによって)アプリケーションを識別し、シグナチャ2ファイルを要求し、ステップ122において、署名サーバの情報要求に応答することに進み、ステップ124において、得られたシグナチャ2ファイルを受信し、記憶する。

【００４１】

署名サーバ26において達成され得る処理ステップの例示的な実施形態を図7に示す。ステップ200において、署名サーバ26は、クライアントコンピューティングデバイス28、30との接続を受け付け、特定のアプリケーションのシグナチャ2ファイルについての要求を受信し得る。この要求の一部として、署名サーバ26には、アプリケーションの名前、識別子または通し番号、ならびに要求元コンピューティングデバイス28、30の型およびモデルが通知され得る。この情報を使用して、ステップ202において、署名サーバ26は、識別されたコンピューティングデバイス28、30内で動作する特定のアプリケーションコードおよび/またはデータを検証することに適した、問合せステップと、検証方法と、対応するシグナチャとを含んでいるデータファイルにアクセスすることができる。検証方法、証明書および個々のアプリケーションに対する脅威が進化するにつれて、署名サーバ26内に維持される対策データベースは、特定のアプリケーションに対する既知の脅威に打ち勝つために、適切な問合せステップと、検証方法と、シグナチャ値とを保持しているデータレコードを用いて更新されることが予期される。このようにして、コンピューティングデバイスが、特定のアプリケーションのそのシグナチャ2ファイルへの更新を要求するときはいつでも、署名サーバ26は、既知の脅威を検出し、その脅威に打ち勝つための最も最新の方法を使用してソフトウェアを検証することが可能である。

【００４２】

サーバメモリから取り出された、対応するデータファイルに記憶された問合せ方法を使用して、ステップ204において、署名サーバ26は、開かれた通信リンクを介してクライアントコンピューティングデバイス28、30との問答を開始する。上述のように、特定のアプリケーションまたはその1次検証方法における既知の脅威または弱点がない場合、コンピューティングデバイス28、30の問合せは実行しなくてもよく、その場合、ステップ204はバイパスされ得る。既知の脅威または弱点がない場合でも、署名サーバ26は、要求が行われたことを記録するために十分な情報をクライアントコンピューティングデバイス28、30に要求し得る。アプリケーションソフトウェアを登録し、ならびに著しい脅威が現れた場合に署名サーバ26がユーザに連絡するために後で使用し得る情報を取得するために、ユーザ識別情報および連絡先についての要求が行われ得る。したがって、様々な実施形態のプロセスは、ライセンスおよび/または被保証人登録プロセスと組み合わせられ得る。

【００４３】

上記で説明したように、クライアントコンピューティングデバイス28、30に要求され得る情報のタイプは、将来現れ得るどんなセキュリティ脅威でも識別し、打ち勝つためのできる限り大きいフレキシビリティを提供するために、基本的に無制限である。たとえば、署名サーバは、ステップ204において、ソフトウェアのサンプルを要求し得、あるいは、ソフトウェアに対して、1つまたは複数のハッシュ関数など、1つまたは複数の関数を実行し、その結果を署名サーバ26に提供するようにクライアントコンピューティングデバイス28、32に要求し得る。別の例として、署名サーバ26はまた、特定のアプリケーションによって生じる潜在的な危険に関する情報を提供することができる、コードがアクセスすることを必要とするリソースに関する情報を要求し得る。別の例として、署名サーバ26は、署名サーバが証明書を独力で検証することを可能にし、ならびにシグナチャを、署名サーバがそのデータレコード内に有するシグナチャと比較するために、ソフトウェアとともに与えられたシグナチャの送信を要求し得る。さらなる例として、署名サーバ26は、マルウェアまたはソフトウェアへの無許可の変更をより良く検出するために、これらの様々なタイプの情報の一部または全部を要求し得る。

【００４４】

クライアントコンピューティングデバイス28、30から受信した情報を使用して、署名サ

ーバ26は、ステップ206において、ソフトウェアが信用できるか、既知の脅威またはマルウェアを含んでいるか、あるいは既知の弱点に対して脆弱であるかを判断するために、サーバメモリに記憶された情報への分析および比較を実行することができる。このステップ206において実行される分析および比較の性質は、特定のアプリケーションに関連する脅威または脆弱性の性質に依存する。特定のアプリケーションソフトウェアに関して行われた判断に応じて、次いでステップ208において、署名サーバ26は、シグナチャ2ファイルをメモリから呼び戻すかまたは他の方法で生成し得る。ステップ206における署名サーバ26によるソフトウェアの分析により、ソフトウェアが信用できないかまたはマルウェアを含んでいることが明らかになった場合、ステップ208において生成される対応するシグナチャ2ファイルは、アプリケーションを終了し、ソフトウェアを実行しないかまたはソフトウェアにアクセスしないための命令であり得る。ステップ206における署名サーバ26によるソフトウェアの分析により、ソフトウェアは信用できるが、既知の脆弱性または脅威があることが明らかになった場合、署名サーバは、ステップ208において、新しい検証方法および対応する新しいシグナチャを含むシグナチャ2ファイルにメモリからアクセスするか、またはそのシグナチャ2ファイルを生成することができる。ステップ206における署名サーバ26によるソフトウェアの分析により、ソフトウェアは信用でき、既知の脆弱性の脅威がないことが明らかになった場合、生成されるシグナチャ2ファイルは、単に、1次検証シグナチャを使用して1次検証方法を実施する命令であり得る。シグナチャ2ファイルは、署名サーバ26中に維持されている対策データベースから呼び戻されるか、またはその時点で署名サーバによって生成され得る。ステップ208においてシグナチャ2ファイルを生成すると、ステップ210にいて、そのファイルをクライアントコンピューティングデバイス28、30に送信する。最後に、プロセスが完了すると、ステップ216において、署名サーバは、コンピューティングデバイスとの接続を終了し得る。

【００４５】

アプリケーションコードおよび/またはデータを検証するために署名サーバ26において実行される分析および比較の性質は機密に保たれ得る。したがって、コンピューティングデバイス28、30への要求は追跡され得るが、プロセスが脅かされることを防ぐために、受信データを分析する際に取られた実際のステップは秘密に保たれ得る。

【００４６】

署名サーバ26上の実装の代替実施形態を図8に示す。この実施形態では、署名サーバ26は、デバイス上に他のアプリケーションソフトウェアが存在するかどうかを問い合わせるためにクライアントコンピューティングデバイス28、30との開かれた接続を利用する。この実施形態では、署名サーバ26は、シグナチャ2ファイルがコンピューティングデバイス28、30に与えられてから現れた脅威に対して脆弱である他のアプリケーションの検証方法およびシグナチャを更新することが可能になる。ユーザは、起動されると、署名サーバ26に連絡するようにコンピューティングデバイスをトリガする新しいアプリケーションを定期的に購入するように予想され得るので、この能力により、署名サーバ26がアプリケーションのセキュリティ改善を定期的に実施することが可能になる。

【００４７】

図8を参照すると、この実施形態は、図7を参照しながら上記で説明したようにステップ200～210を通して進む。ステップ210において、シグナチャ2ファイルがクライアントコンピューティングデバイス28、30に送信されると、ステップ212において、署名サーバ26は、クライアントデバイス上に記憶された他のアプリケーションコードおよび/またはデータのリストを要求し得る。コンピューティングデバイス28、30は、検証されることが可能であるかまたは検証されるべきソフトウェアのリストを維持するように構成され得、その場合、このリストは署名サーバ26に与えられる。別の実装形態では、ソフトウェアアプリケーションのうちのどれが検証更新を必要とするかを署名サーバ26が判断することを可能にするために、クライアントコンピューティングデバイス28、30は、メモリに記憶されたすべてのソフトウェアアプリケーションのリストを単に与え得る。

【００４８】

クライアントコンピューティングデバイス28、30から受信した他のアプリケーションのリストを使用して、署名サーバ26は、テスト214において、それらのアプリケーションのいずれかが新しいシグナチャ2ファイルを必要とするかどうかを判断する。これは、受信リスト中の各アイテムを、署名サーバ26中に維持された脆弱性または対策データベースにおける脆弱アプリケーションのデータベースと比較することによって達成され得る。新しいシグナチャ2ファイルを必要とする他のアプリケーションがないと署名サーバ26が判断した場合(すなわち、テスト214=「いいえ」)、ステップ216において、署名サーバ26は、単にコンピューティングデバイス28、30との接続を終了し、それによって検証更新セッションを終了し得る。しかしながら、クライアントコンピューティングデバイス28、30上に存在する少なくとも1つのアプリケーションが、更新されたシグナチャ2ファイルを必要とすると署名サーバ26が判断した場合(すなわち、テスト214=「はい」)、署名サーバ26は、アプリケーションのうちの1つを選択し、対策データベース内の対応するデータレコードにアクセスし、ステップ202に戻り得る。選択されたアプリケーションについての対策データベースから取り出された情報を使用して、署名サーバ26は、図7を参照しながら上記で説明したように、ステップ204～210の検証およびシグナチャ2ファイル更新処理を進める。署名サーバ26はクライアントコンピューティングデバイス28、30からアプリケーションのリストをすでに取得しているので、ステップ212におけるそのようなリストを要求するステップは繰り返される必要がなく、署名サーバ26は、テスト214において、シグナチャ2ファイル更新を必要とする別のアプリケーションがあるかどうかを判断することに戻ることができる。署名サーバ26は、シグナチャ2ファイル更新を必要するクライアントモバイルデバイス上のアプリケーションがもはや存在しなくなるまでこのループを繰り返すことができ、アプリケーションがもはや存在しなくなったそのポイントにおいて、署名サーバは、ステップ216において、クライアントデバイスとの接続を終了し、それによってアプリケーション評価およびセキュリティ更新セッションを終了し得る。

【００４９】

新しい脅威が既存のソフトウェアコードおよびデータに対して現れるにつれて、脅威の重大度に応じて様々な対策が実装され得る。脅威の性質が、有効なシグナチャ2ファイルを用いたコンピューティングデバイス中に展開されるアプリケーションソフトウェアに問題を生じる可能性が低い場合、それらのアプリケーションをさらに更新する必要がないことがある。たとえば、脅威が特定のゲームアプリケーションに検出された場合、その脅威がコンピューティングデバイス中に達する可能性がある機構はあり得ず、現在展開されているシグナチャ2ファイルが保護を与えるのに十分であるので、その脅威は、コンピューティングデバイス上ですでに展開されたゲームのバージョンに拡大し得ない。そのような場合、新しいアプリケーションが図5または図6を参照しながら上記で説明したようなコンピューティングデバイス上でアクティブにされたときのみ、シグナチャ2ファイルは展開され得る。

【００５０】

脅威の性質が、展開されたアプリケーションに問題を生じる可能性があるが、すべてのコンピューティングデバイスを保護するために即時の行為が必要とされるような重大さがない場合、図8を参照しながら上記で説明したように、コンピューティングデバイスが他のアプリケーションのためにそれらに連絡するときはいつでも、署名サーバ26は、そのような脆弱なアプリケーションを識別し、更新されたシグナチャ2ファイルを与えるように構成され得る。

【００５１】

脅威の性質が、すべてのコンピューティングデバイスについて即時に修正アクションを取る必要があるような即時の問題を生じる可能性がある場合、少なくとも再び特定のアプリケーションを実行する前に、脆弱なアプリケーションを有するすべてのコンピューティングデバイスに、それらに署名サーバ26に連絡するように指示する電子メールまたはSMSメッセージなどの電子メッセージをブロードキャストし得る。上記で説明したように、署名サーバ26は、初期シグナチャ2ファイルを与えるプロセスの一部としてそのようなメッ

セージをブロードキャストすることが可能であるのに十分なユーザ情報を取得し得る。電子メッセージをコンピューティングデバイスに送信するための様々な方法が当技術分野でよく知られており、そのいずれも、そのような警告を与えるために実装され得る。図5および図6を参照しながら上記で説明したように、特定のアプリケーションが厳しい脅威に直面しているという通知を受信すると、コンピューティングデバイスは、それらのアプリケーションからシグナチャ2ファイルを単に削除し得、そのアプリケーションが実行のために選択される次回には、デバイスに署名サーバ26に連絡するようにプロンプトを出すことになる。

【００５２】

様々な実施形態は、ただコードおよびデータのセキュリティ環境を改善し、コード/データ検証方法を改善すること以外に、いくつかの用途を有する。1つには、それらの実施形態はまた、署名なしソフトウェアを妥当性検査するために使用され得る。これを行うために、署名サーバ26は、クライアントコンピュータがコードを実行するかまたはデータを処理するより前にソフトウェアが安全であることを確認するのに十分な情報を、クライアントコンピュータ28、30に要求し、クライアントコンピュータ28、30から受信することができる。この使用は、一例として示される最良のものであり得る。閉じられたクライアント上で署名なし/詳しく調べられていないコードを実行する最も有効な方法のうちの1つは、署名システム自体を攻撃することを考えること、および代わりに合法的に署名されたコードの欠陥を利用し、この欠陥を(たとえば、バッファオーバーランを介して)活用して、悪意のある、または無許可のコードを実行することである。典型的な例では、ゲームコンソールが、すべてのゲームのコードが署名されることを要求し得る。しかしながら、コンソールメーカーが個々のゲームによって扱われるすべてのデータも署名されることを主張することは不可能である。これについてはいくつかの理由があるが、最も明らかな理由は、そのようなデータが一般に静的でなく、頻繁に変化し得ること(たとえば、高ゲームスコアデータファイル)である。ハッカーがこれに乗じることは知られており、特別にクラフティングされた高スコアファイルを作成することによって、ゲームの弱点を利用してコンソール上で悪意のあるコードを実行する。この脆弱性を克服するために、一実施形態では、関連するゲームが起動または複製される前に、(署名なしの)高スコアファイルに関する情報は、妥当性検査のために署名サーバ26にアップロードされ得る。このようにして署名なしデータを妥当性検査することは、脆弱なアプリケーションから危険を除去することができる。

【００５３】

様々な実施形態の別の使用は、署名システムを回避するために変更されたソフトウェアを識別し、特徴づけることである。たとえば、クライアントコンピュータによって署名サーバに送信された情報が予想された情報に一致しない場合、サーバは、コードファイルまたはデータセット全体をアップロードするようにクライアントコンピュータに依頼し得る。このデータが取得されると、サーバは、セキュアな環境の快適さのなかでそれを自由に分析することができる。アップロードされたソフトウェアによって提示された安全または危険を確認した後に、次いで、署名サーバは、上記で説明したように、これ以降使用するための新しいシグナチャおよび検証プロシージャ(すなわち、シグナチャ2ファイル)を与えることができる。

【００５４】

様々な実施形態の別の使用は、ソフトウェアおよびデバイス上の攻撃の発生および拡散を追跡するために、署名サーバが検証プロセス中にいくつかのクライアントから取得されたコードおよびデータ情報を使用することを可能にする。シグナチャ妥当性検査回避が、ある程度の時間期間の間クライアントコンピュータと署名サーバ26の両方をだますのに十分良好である場合でも、脆弱性が判断されると、署名サーバ26は、利用の開始点および成長パターンを判断するためにそのログを追跡し得る。さらに、署名サーバ26が、クライアントコンピュータ28、30に検証のためにソフトウェアファイル全体をダウンロードするように要求する実施形態では、署名サーバは、分析のためにコード/データをキャプチャし

得る。

【００５５】

様々な実施形態は、コンピュータシステム上でコードおよびデータをセキュアにするための知られているシステムに勝るいくつかの利点を与える。実施形態のセキュリティ機構は動的であり、したがって、最悪を仮定し、手に負えないすべてのレガシーシグナチャを拒否する代わりに、単に署名基準を変更することによってクライアントコンピュータ中で最も最近に発見された脆弱性を相殺するように変更され得る。実施形態のセキュリティ機構は、クライアントコンピュータ実装形態が更新されることを必要とせず、したがって、コンピュータおよびモバイルデバイス自体を変更するコストと経費とを節約する。実施形態のセキュリティ機構は、(たいていのモバイルデバイスの場合のように)コード/データ画像全体のコピーを送信することが実現可能でないような、クライアントコンピュータが限られたストレージ機能と送信機能とを有する環境で十分に動作する。実施形態のセキュリティ機構は、署名付きコード/データと署名なしコード/データの両方について動作する。実施形態のセキュリティ機構は、署名システムを回避することにおける試行を追跡し、ならびに試行に関与するコード/データをキャプチャするための機構を設ける。

【００５６】

本明細書で説明する実施形態は、様々なモバイルデバイスのうちのいずれかに実装され得る。一般に、そのようなモバイルデバイスは、図9に示す構成要素を共通に有する。たとえば、モバイルデバイス30は、内部メモリ32とディスプレイ33とに結合されたプロセッサ31を含み得る。さらに、モバイルデバイス30は、プロセッサ31に結合されたワイヤレスデータリンクおよび/またはセルラー電話トランシーバ35に接続される、電磁放射を送信および受信するためのアンテナ34を有する。いくつかの実装形態では、セルラー電話通信のために使用されるトランシーバ35ならびにプロセッサ31およびメモリ32の部分を、ワイヤレスデータリンクを介してデータインターフェースを与えることから、エアインターフェースと総称する。モバイルデバイス30は、一般に、ユーザ入力を受け取るためのキーパッド36または小型キーボードおよびメニュー選択ボタンまたはロッカースイッチ37をも含む。モバイルデバイス30は、FireWireコネクタなど、データケーブルをプロセッサ31、またはUSBメモリデバイス(図示せず)などの外部メモリデバイスに接続するためのコネクタプラグ38をも含み得る。

【００５７】

上記で説明した実施形態は、たとえば、図10に示すパーソナルコンピュータ28など、様々なコンピューティングデバイスのいずれでも実装され得る。そのようなパーソナルコンピュータ28は、一般に、揮発性メモリ282とディスクドライブ283などの大容量不揮発性メモリとに結合されたプロセッサ281を含むコンピュータアセンブリ280を含む。コンピュータアセンブリ280はまた、プロセッサ281に結合されたフロッピー（登録商標）ディスクドライブ284とコンパクトディスク(CD)ドライブ285とを含み得る。一般に、コンピュータ28はまた、キーボード286のようなユーザ入力デバイスと、ディスプレイ287とを含む。コンピュータアセンブリ280はまた、ユニバーサルシリアルバス(USB)ポート(図示せず)など、プロセッサ281に結合された外部メモリデバイスを受けるためのいくつかのコネクタポート、ならびにプロセッサ281をネットワークに結合するためのネットワーク接続回路(図示せず)を含み得る。

【００５８】

様々な態様は、説明した方法のうちの1つまたは複数を実装するように構成されたソフトウェア命令を実行するコンピューティングデバイスプロセッサ31、281によって実装され得る。そのようなプロセッサは、当業者によって諒解され得るように、マイクロプロセッサユニット、マイクロコンピュータユニット、プログラマブル浮動小数点ゲートアレイ(FPGA)、および特定用途向け集積回路(ASIC)であり得る。そのようなソフトウェア命令は、別々のアプリケーションとして、コンピュータのオペレーティングシステムソフトウェアの一部として、オペレーティングシステムによって実装される一連のAPIとして、または実施方法を実装するコンパイルされたソフトウェアとしてメモリ32、282、283に記憶さ

れ得る。さらに、ソフトウェア命令は、ランダムアクセスメモリ32、282、ハードディスクメモリ283、(フロッピー（登録商標）ディスクドライブ284で読取り可能な)フロッピー（登録商標）ディスク、(CDドライブ285で読取り可能な)コンパクトディスク、(EEPROMなどの)読取り専用メモリ、および/あるいは外部メモリチップまたはUSB接続可能な外部メモリ(たとえば、「フラッシュドライブ」)など、コンピュータ30、280に接続されたメモリモジュール(図示せず)を含む、任意の形態の有形プロセッサ可読メモリ上に記憶され得る。代替的に、いくつかのステップまたは方法は、所与の機能に固有の回路によって実行され得る。

【００５９】

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序は、どんな順序でも実行され得る。

【００６０】

本明細書で開示した実施形態に関連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、または両者の組合せとして実装され得ることを当業者なら諒解されよう。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、上記では概してそれらの機能に関して説明した。そのような機能をハードウェアとして実装するか、ソフトウェアとして実装するかは、特定の適用例および全体的なシステムに課される設計制約に依存する。当業者は、説明した機能を特定の適用例ごとに様々な方法で実装し得るが、そのような実装の決定は、本発明の範囲からの逸脱を生じるものと解釈すべきではない。

【００６１】

本明細書で開示した実施形態に関して説明した方法またはアルゴリズムのステップは、直接ハードウェアで実施されるか、プロセッサによって実行されるソフトウェアモジュールで実施されるか、またはその2つの組合せで実施され得る。ソフトウェアモジュールは、RAMメモリ、フラッシュメモリ、ROMメモリ、EPROMメモリ、EEPROMメモリ、レジスタ、ハードディスク、リムーバブルディスク、CD-ROM、または当技術分野で知られている任意の他の形態の記憶媒体のいずれかとすることができるプロセッサ可読メモリ中に常駐し得る。例示的な記憶媒体は、プロセッサが記憶媒体から情報を読み取り、記憶媒体に情報を書き込むことができるように、プロセッサに結合される。代替として、記憶媒体はプロセッサと一体であり得る。プロセッサおよび記憶媒体はASIC中に常駐し得る。ASICはユーザ端末またはモバイルデバイス中に常駐し得る。代替として、プロセッサおよび記憶媒体は、ユーザ端末またはモバイルデバイス中に個別構成要素として常駐し得る。さらに、いくつかの態様では、方法またはアルゴリズムのステップおよび/またはアクションは、コンピュータプログラム製品に組み込まれ得る、機械可読媒体および/または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、あるいはそのセットとして常駐し得る。

【００６２】

様々な実施形態の上記の説明は、当業者が本発明を実施または使用することを可能にするために提供したものである。これらの実施形態に対する様々な変更は、当業者には容易に明らかとなり、本明細書で定義された一般的な原理は、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、本明細書に示した実施形態に限定されるものではなく、代わりに、特許請求の範囲には、本明細書に開示した原理および新規な特徴に一致する最も広い範囲が与えられるべきである。

【符号の説明】

【００６３】

20　ネットワーク

20　通信ネットワーク

21 基地局
22 MSC
22 モバイル交換センター(MSC)
23 ゲートウェイサーバ
23 サーバゲートウェイ
24 CAサーバ
24 証明局(CA)サーバ
25 インターネット
26 シグナチャサーバ
26 署名サーバ
26 送信サーバ
27 コンテンツサーバ
28 コンピューティングデバイス
28 クライアントコンピュータ
28 クライアントコンピューティングデバイス
28 パーソナルコンピュータ
28 要求元コンピューティングデバイス
28 要求元デバイス
29 ケーブル
29 データケーブル
30 コンピューティングデバイス
30 クライアントコンピュータ
30 クライアントコンピューティングデバイス
30 コンピュータ
30 モバイルデバイス
30 モバイルハンドセット
30 要求元コンピューティングデバイス
30 要求元デバイス
31 プロセッサ
32 メモリ
32 ランダムアクセスメモリ
32 内部メモリ
33 ディスプレイ
34 アンテナ
35 セルラー電話トランシーバ
35 トランシーバ
36 キーパッド
37 ロッカースイッチ
38 コネクタプラグ
50 アプリケーションファイル
52 アプリケーションコードファイル
52 実行可能コードファイル
52 実行可能ファイル
54 アプリケーションデータ
54 アプリケーションデータファイル
54 データ
54 データファイル
56 1次シグナチャファイル
56 シグナチャファイル
58 シグナチャ2(「sig2」)ファイル

58　シグナチャ2ファイル
58a　新しいまたは変更された検証アルゴリズム
58b　新しいシグナチャ
280　コンピュータ
280　コンピュータアセンブリ
281　プロセッサ
282　ランダムアクセスメモリ
282　揮発性メモリ
283　ディスクドライブ
283　ハードディスクメモリ
284　フロッピー（登録商標）ディスクドライブ
285　コンパクトディスク(CD)ドライブ
286　キーボード
287　ディスプレイ

【図９】

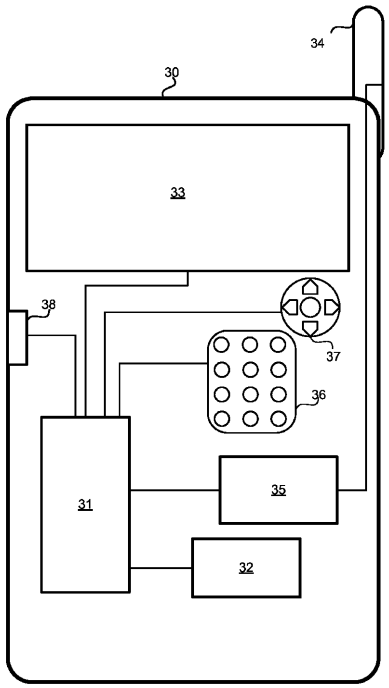


FIG. 9

【図１０】

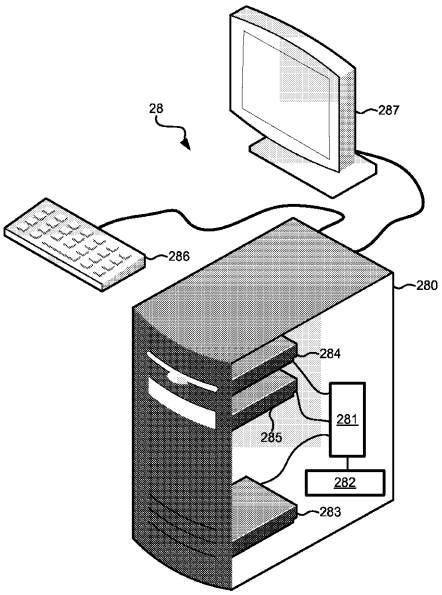
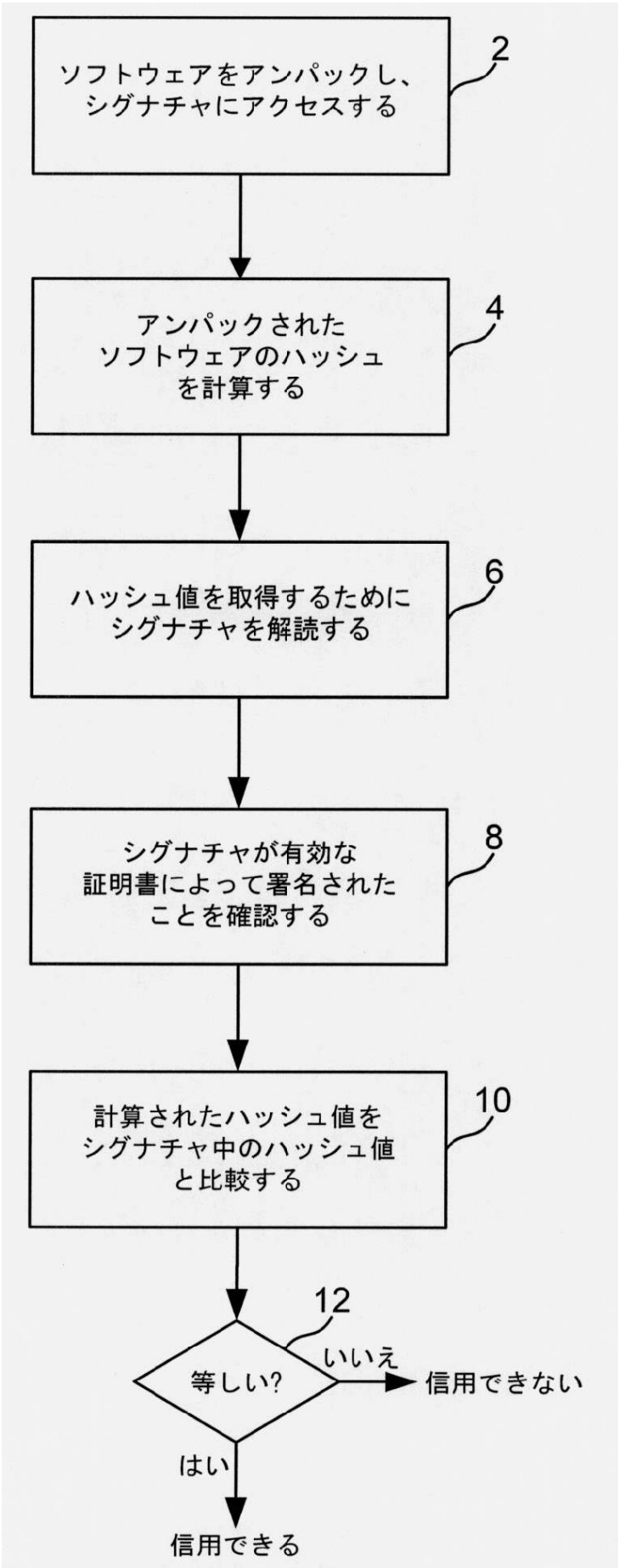
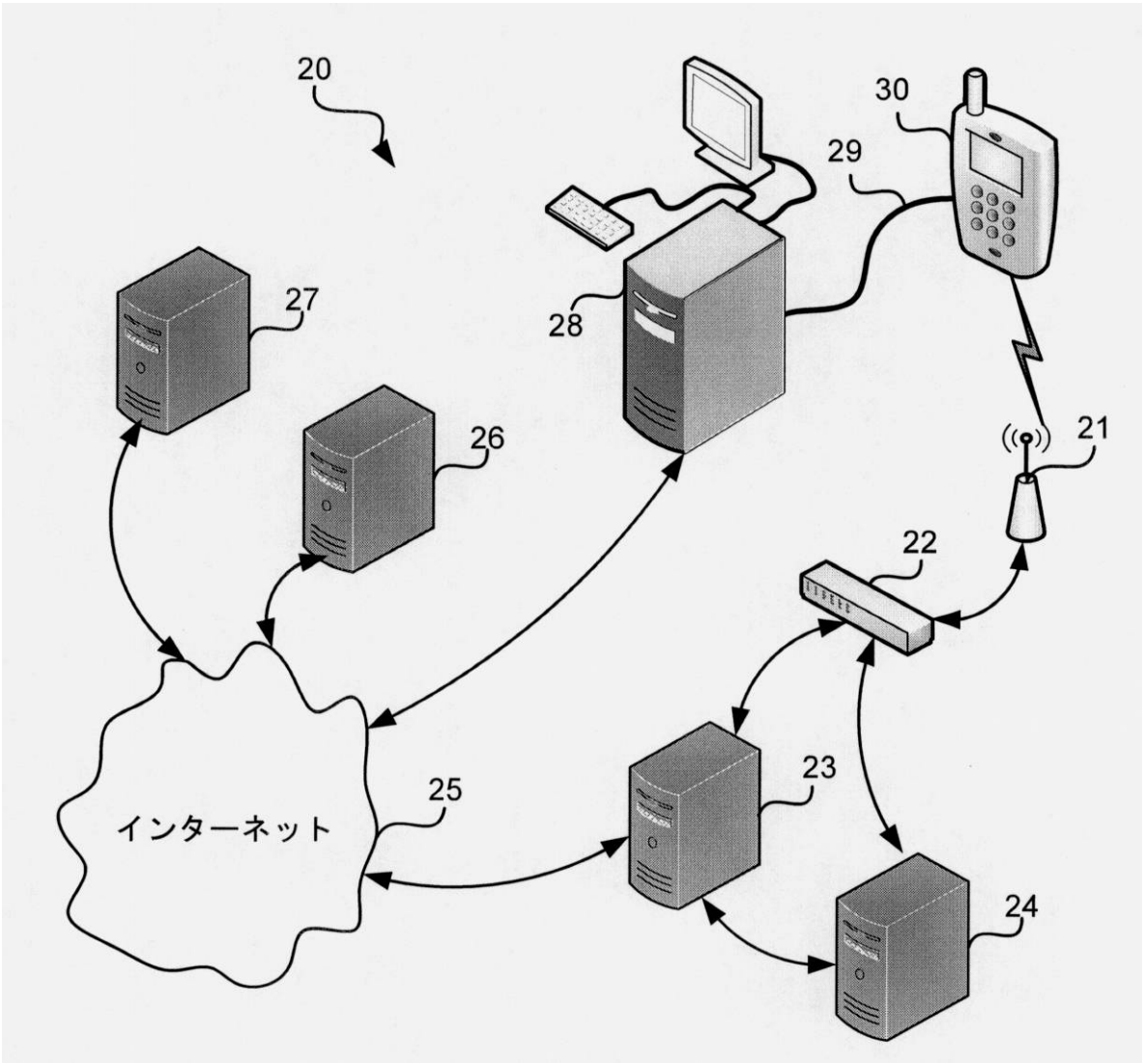


FIG. 10

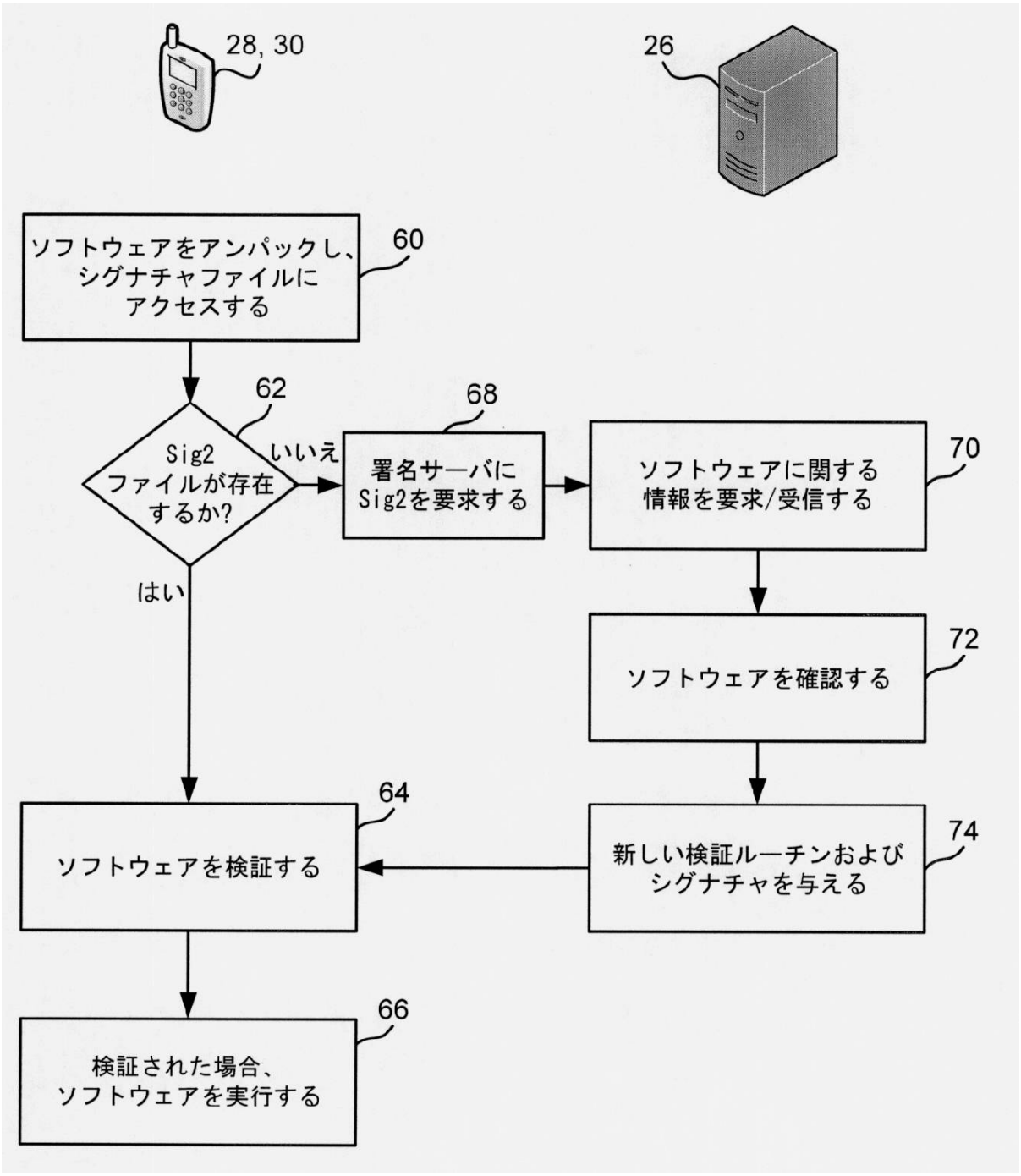
【図１】


ソフトウェアをアンパックし、
シグナチャにアクセスする
2
アンパックされた
ソフトウェアのハッシュ
を計算する
4
ハッシュ値を取得するために
シグナチャを解読する
6
シグナチャが有効な
証明書によって署名された
ことを確認する
8
計算されたハッシュ値を
シグナチャ中のハッシュ値
と比較する
10
12
等しい?
いいえ
信用できない
はい
信用できる

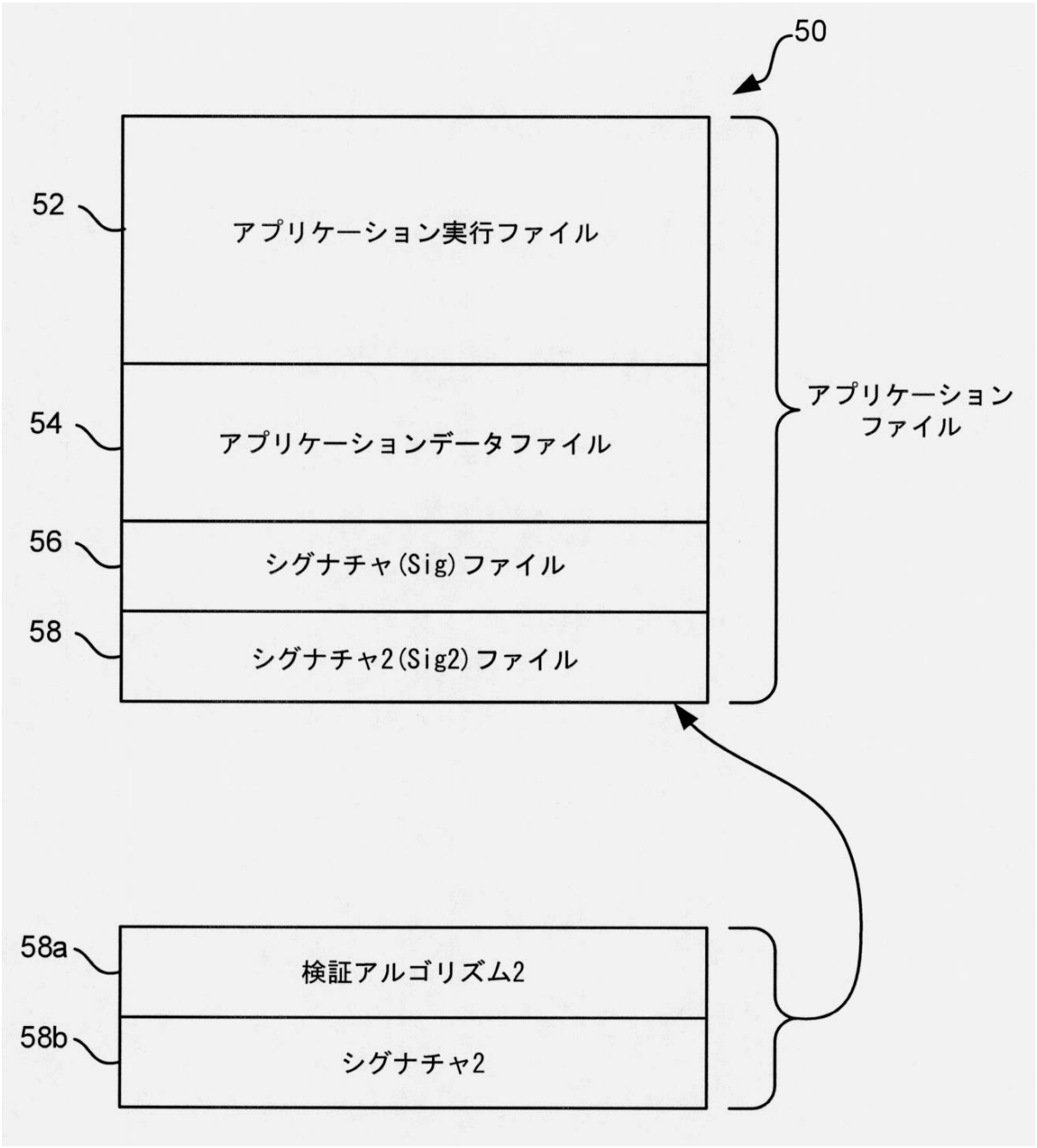
【図２】



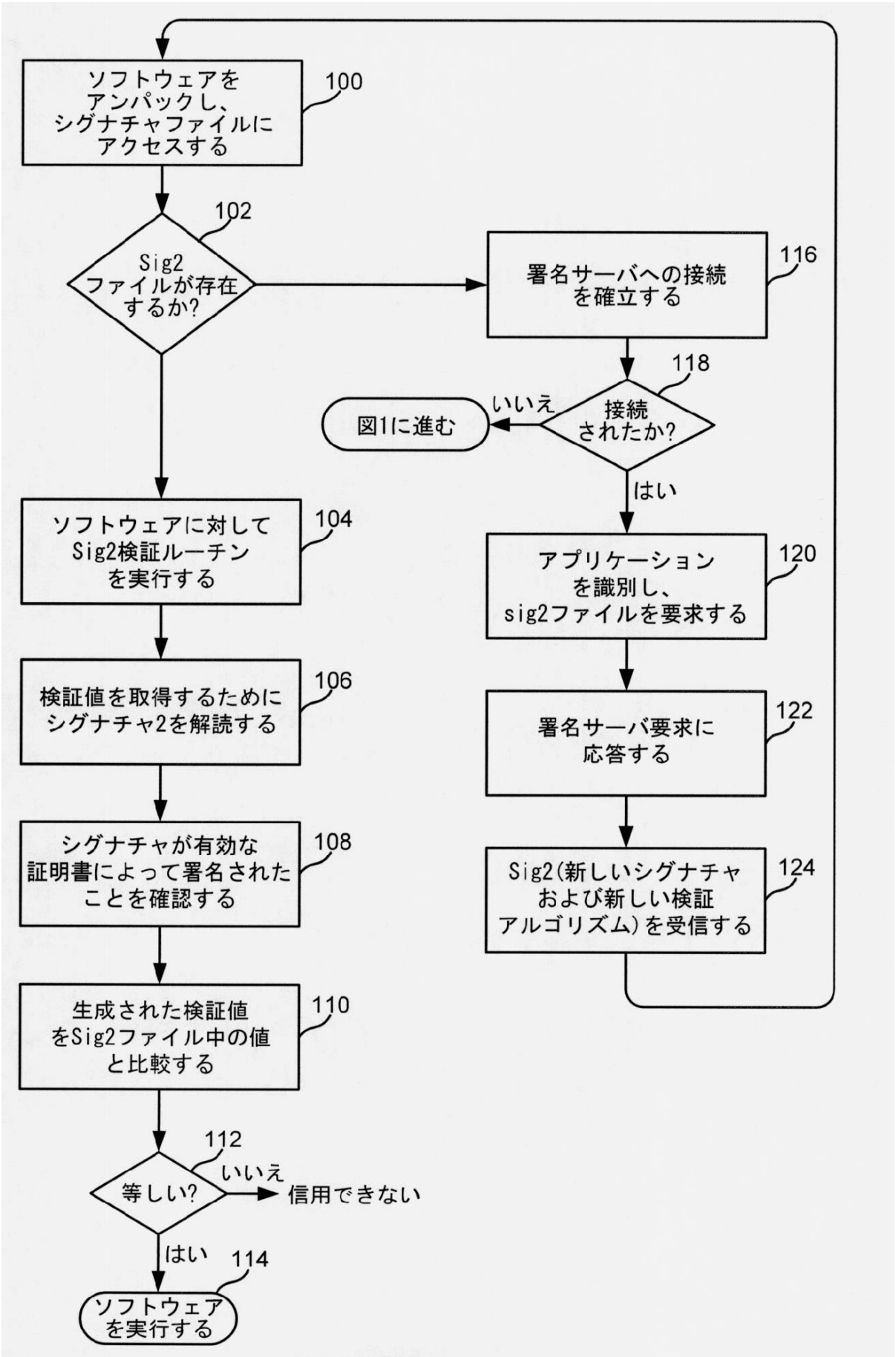
【図３】


28, 30
26
ソフトウェアをアンパックし、シグナチャファイルにアクセスする
60
62
Sig2ファイルが存在するか?
いいえ
はい
68
署名サーバにSig2を要求する
ソフトウェアに関する情報を要求/受信する
70
ソフトウェアを確認する
72
新しい検証ルーチンおよびシグナチャを与える
74
ソフトウェアを検証する
64
検証された場合、ソフトウェアを実行する
66

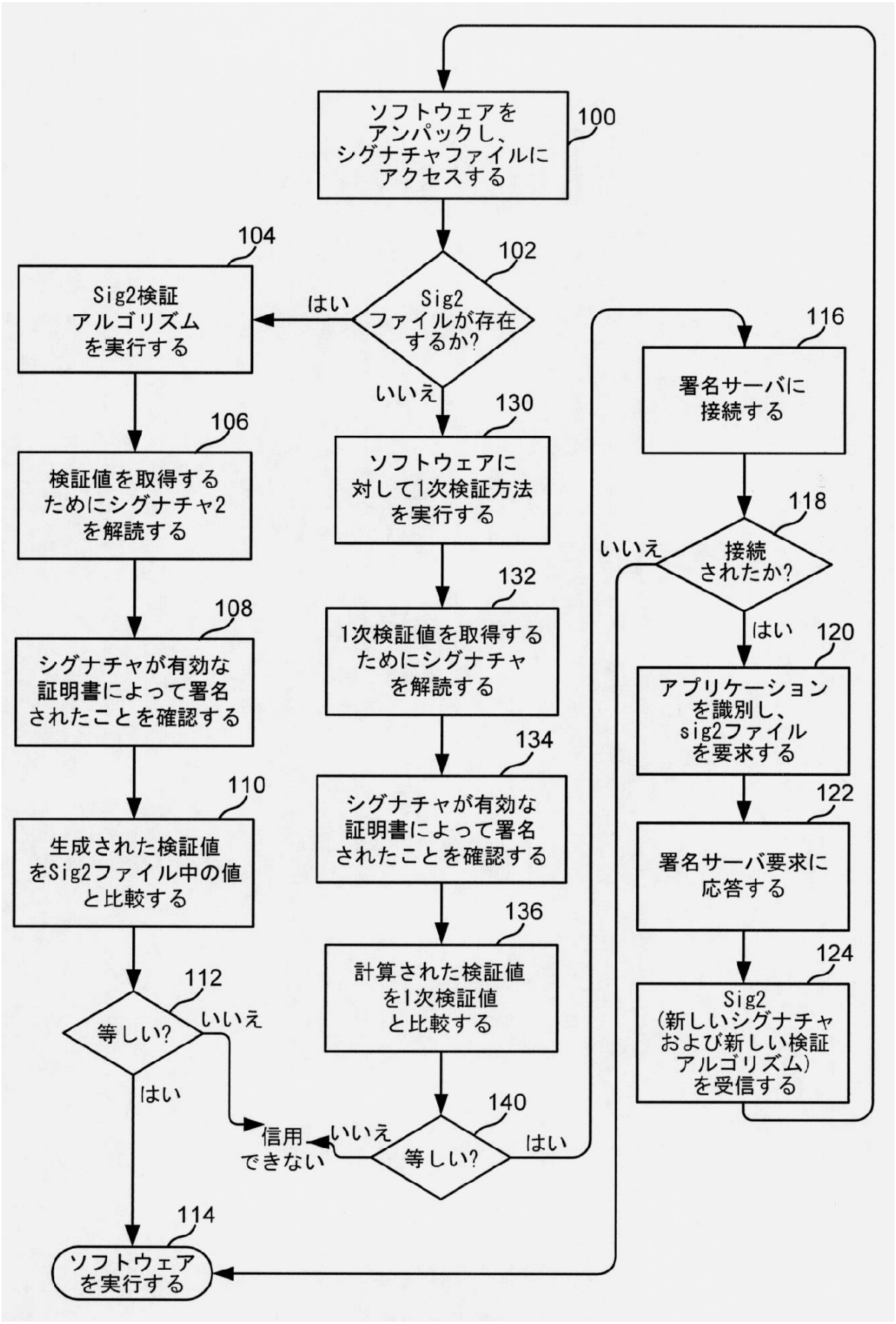
【図４】


50
52
アプリケーション実行ファイル
54
アプリケーションデータファイル
56
シグナチャ(Sig)ファイル
58
シグナチャ2(Sig2)ファイル
アプリケーション
ファイル
58a
検証アルゴリズム2
58b
シグナチャ2

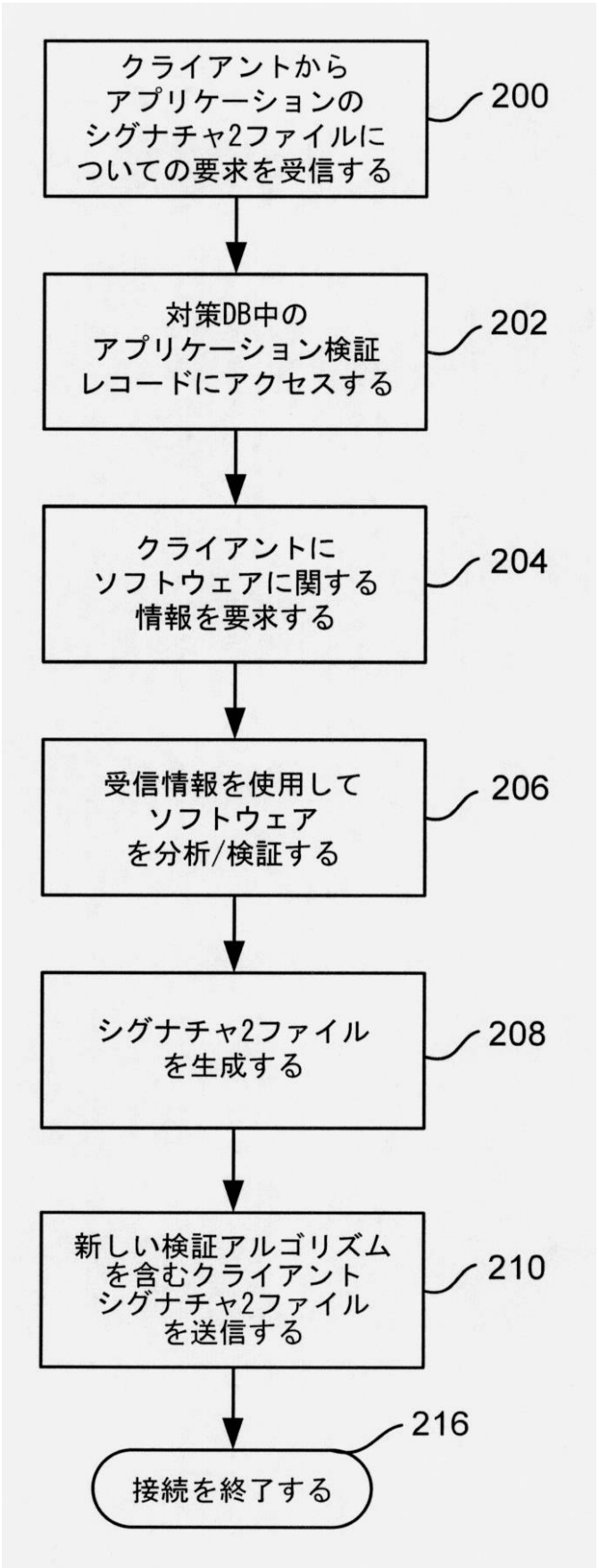
【図５】


ソフトウェアを
アンパックし、
シグナチャファイルに
アクセスする
100
Sig2
ファイルが存在
するか?
102
署名サーバへの接続
を確立する
116
接続
されたか?
118
いいえ
図1に進む
はい
ソフトウェアに対して
Sig2検証ルーチン
を実行する
104
アプリケーション
を識別し、
sig2ファイルを要求する
120
検証値を取得するために
シグナチャ2を解読する
106
署名サーバ要求に
応答する
122
シグナチャが有効な
証明書によって署名された
ことを確認する
108
Sig2(新しいシグナチャ
および新しい検証
アルゴリズム)を受信する
124
生成された検証値
をSig2ファイル中の値
と比較する
110
等しい?
112
いいえ
信用できない
はい
114
ソフトウェア
を実行する

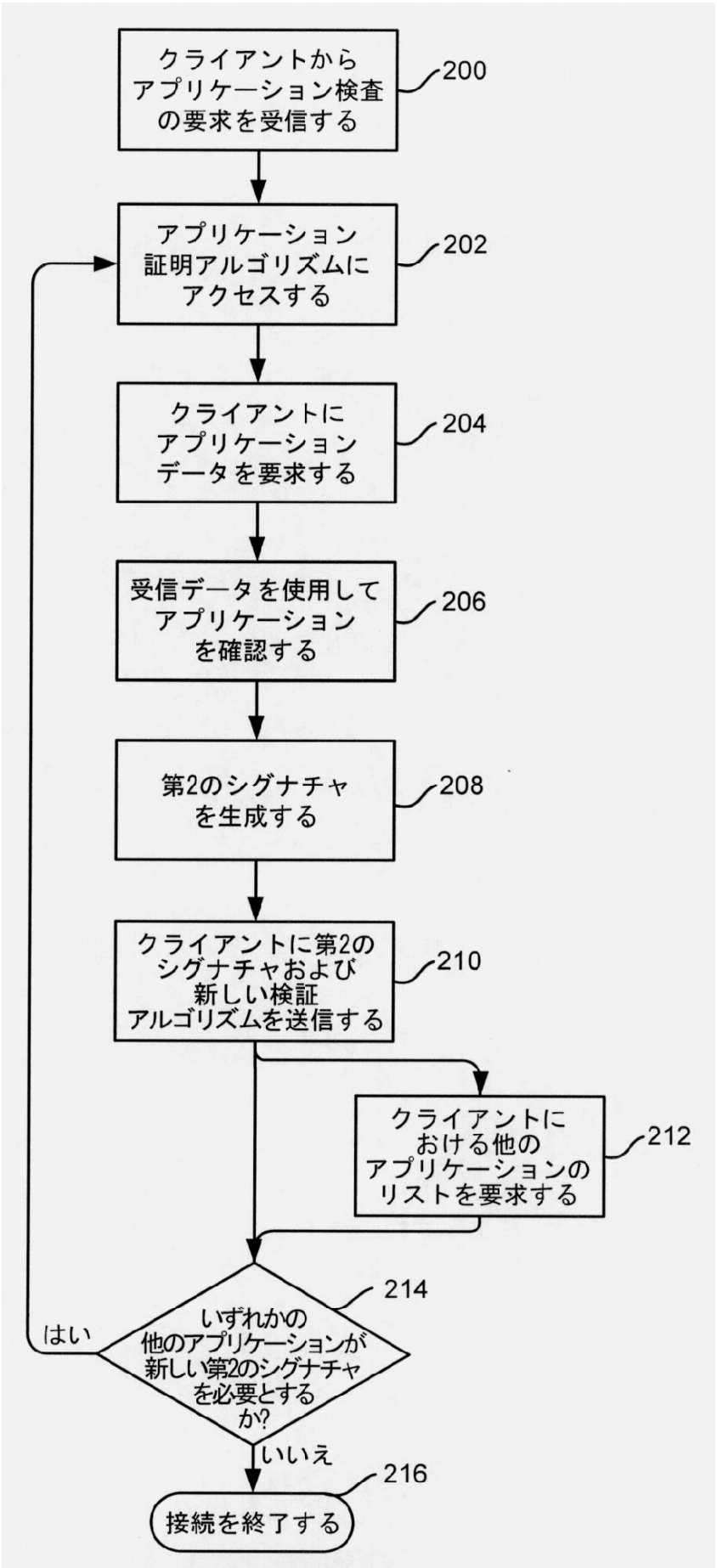
【図６】


ソフトウェアを
アンパックし、
シグナチャファイルに
アクセスする
100
Sig2
ファイルが存在
するか?
102
はい
Sig2検証
アルゴリズム
を実行する
104
いいえ
検証値を取得する
ためにシグナチャ2
を解読する
106
シグナチャが有効な
証明書によって署名
されたことを確認する
108
生成された検証値
をSig2ファイル中の値
と比較する
110
等しい?
112
いいえ
はい
ソフトウェア
を実行する
114
ソフトウェアに
対して1次検証方法
を実行する
130
1次検証値を取得する
ためにシグナチャ
を解読する
132
シグナチャが有効な
証明書によって署名
されたことを確認する
134
計算された検証値
を1次検証値
と比較する
136
等しい?
140
いいえ
はい
信用
できない
署名サーバに
接続する
116
接続
されたか?
118
いいえ
はい
アプリケーション
を識別し、
sig2ファイル
を要求する
120
署名サーバ要求に
応答する
122
Sig2
(新しいシグナチャ
および新しい検証
アルゴリズム)
を受信する
124

【図７】


クライアントから
アプリケーションの
シグナチャ2ファイルに
ついての要求を受信する
200
対策DB中の
アプリケーション検証
レコードにアクセスする
202
クライアントに
ソフトウェアに関する
情報を要求する
204
受信情報を使用して
ソフトウェア
を分析/検証する
206
シグナチャ2ファイル
を生成する
208
新しい検証アルゴリズム
を含むクライアント
シグナチャ2ファイル
を送信する
210
216
接続を終了する

【図８】


クライアントから
アプリケーション検査
の要求を受信する
200
アプリケーション
証明アルゴリズムに
アクセスする
202
クライアントに
アプリケーション
データを要求する
204
受信データを使用して
アプリケーション
を確認する
206
第2のシグナチャ
を生成する
208
クライアントに第2の
シグナチャおよび
新しい検証
アルゴリズムを送信する
210
クライアントに
おける他の
アプリケーションの
リストを要求する
212
いずれかの
他のアプリケーションが
新しい第2のシグナチャ
を必要とする
か?
214
はい
いいえ
接続を終了する
216

フロントページの続き

(56)参考文献 特開２００１－３５０４０５（ＪＰ，Ａ）
特開２００２－２０７４２８（ＪＰ，Ａ）
特開２００５－２０４１２６（ＪＰ，Ａ）
特開２００７－２７９３８（ＪＰ，Ａ）
特開２００７－２６６７９７（ＪＰ，Ａ）
特表２０１１－５０７７４２（ＪＰ，Ａ）
国際公開第２００６／１２８８７６（WO，Ａ１）
佐藤 雅之，大規模タイムスタンプサービスシステムの構築事例，情報処理学会第69回全国大会講演論文集 インタフェース コンピュータと人間社会，２００７年 3月，pp. 4-321--4-322，1H-1

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｈ０４Ｌ ９／３２
Ｇ０６Ｆ ２１／６４
Ｈ０４Ｌ ９／１４