

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6746156号
(P6746156)

(45) 発行日 令和2年8月26日(2020.8.26)

(24) 登録日 令和2年8月7日(2020.8.7)

(51) Int.Cl.

G06F 21/64 (2013.01)
G06F 21/12 (2013.01)

F 1

G06F 21/64
G06F 21/12

請求項の数 18 (全 19 頁)

(21) 出願番号 特願2019-520118 (P2019-520118)
 (86) (22) 出願日 平成29年9月26日 (2017.9.26)
 (65) 公表番号 特表2019-535087 (P2019-535087A)
 (43) 公表日 令和1年12月5日 (2019.12.5)
 (86) 國際出願番号 PCT/CN2017/103403
 (87) 國際公開番号 WO2018/068633
 (87) 國際公開日 平成30年4月19日 (2018.4.19)
 審査請求日 令和1年5月27日 (2019.5.27)
 (31) 優先権主張番号 201610887188.5
 (32) 優先日 平成28年10月11日 (2016.10.11)
 (33) 優先権主張国・地域又は機関
中国(CN)

早期審査対象出願

(73) 特許権者 510330264
アリババ・グループ・ホールディング・リミテッド
ALIBABA GROUP HOLDING LIMITED
英國領, ケイマン諸島, グランド・ケイマン, ジョージ・タウン, ワン・キャピタル・プレイス, フォース・フロア, ピー・オーネ, ボックス 847
(74) 代理人 100188558
弁理士 飯田 雅人
(74) 代理人 100205785
弁理士 ▲高▼橋 史生

最終頁に続く

(54) 【発明の名称】ソフトウェアリパッケージング防止方法および装置

(57) 【特許請求の範囲】

【請求項1】

リパッケージング対策のためのコンピュータ実装方法であって、
 インストールパッケージ内のターゲットファイルを実施するステップと、
 前記インストールパッケージ内の安全な動的ライブラリをロードするステップと、
 前記ターゲットファイル内のコードに基づいて、前記ターゲットファイル内に埋め込まれたデジタル透かし情報および前記安全な動的ライブラリ内に記憶された検証情報を取得するステップと、

前記デジタル透かし情報および前記検証情報に基づいて、前記インストールパッケージがリパッケージされたインストールパッケージであるかどうかを判定するステップであつて、

前記デジタル透かし情報が命令であるときに、
動作結果を取得するために前記命令に基づいて動作を実行するステップと、
前記動作結果が前記検証情報内の動作結果と一致するかどうかを判定するステップと、
前記命令が前記検証情報内の命令と一致するかどうかを判定するステップと
を備える、ステップと
を含む、コンピュータ実装方法。

【請求項2】

前記ターゲットファイルが、所定のコードがコンパイルされた後に取得されたターゲッ

10

20

トファイルであり、

前記デジタル透かし情報が、前記ターゲットファイルの終端に埋め込まれる、
請求項1に記載のコンピュータ実装方法。

【請求項3】

前記デジタル透かし情報を取得するステップが、

前記ターゲットファイルのオリジナルの長さに基づいて前記ターゲットファイル内の前記デジタル透かし情報の開始アドレスを決定するステップであって、前記オリジナルの長さが、前記デジタル透かし情報が埋め込まれる前の前記ターゲットファイルの長さである、ステップと、

前記開始アドレスに基づいて前記ターゲットファイルから前記デジタル透かし情報を取得するステップと
10

を含む、請求項2に記載のコンピュータ実装方法。

【請求項4】

前記デジタル透かし情報および前記検証情報に基づいて、前記インストールパッケージが前記リパッケージされたインストールパッケージであるかどうかを判定するステップが、

、
前記デジタル透かし情報が前記検証情報と一致するかどうかを判定するステップと、
前記デジタル透かし情報が前記検証情報と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定するステップと、
20

前記デジタル透かし情報が前記検証情報と一致しないと判定したことに応答して、前記インストールパッケージがリパッケージされたインストールパッケージであると判定するステップと

をさらに含む、請求項1に記載のコンピュータ実装方法。

【請求項5】

前記デジタル透かし情報および前記検証情報に基づいて前記インストールパッケージが前記リパッケージされたインストールパッケージであるかどうかを判定するステップが、

前記デジタル透かし情報が命令であるときに、

動作結果を取得するために前記命令に基づいて動作を実行するステップと、

前記動作結果が前記検証情報と一致するかどうかを判定するステップと、
30

前記動作結果が前記検証情報と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定するステップと、

前記動作結果が前記検証情報と一致しないと判定したことに応答して、前記インストールパッケージがリパッケージされたインストールパッケージであると判定するステップと

をさらに含む、請求項1に記載のコンピュータ実装方法。

【請求項6】

前記動作結果が前記検証情報内の前記動作結果と一致し、かつ前記命令が前記検証情報内の前記命令と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定するステップと、
40

前記動作結果が前記検証情報と一致しないと判定したことに応答して、前記インストールパッケージが前記リパッケージされたインストールパッケージであると判定するステップと

をさらに含む、請求項1に記載のコンピュータ実装方法。

【請求項7】

コンピュータ可読記憶媒体であって、

インストールパッケージ内のターゲットファイルを実施することと、

前記インストールパッケージ内の安全な動的ライブラリをロードすることと、

前記ターゲットファイル内のコードに基づいて、前記ターゲットファイル内に埋め込まれたデジタル透かし情報および前記安全な動的ライブラリ内に記憶された検証情報を取得
50

することと、

前記デジタル透かし情報および前記検証情報に基づいて、前記インストールパッケージがリパッケージされたインストールパッケージであるかどうかを判定することであって、

前記デジタル透かし情報が命令であるときに、

動作結果を取得するために前記命令に基づいて動作を実行することと、

前記動作結果が前記検証情報内の動作結果と一致するかどうかを判定することと、

前記命令が前記検証情報内の命令と一致するかどうかを判定することと

を備える、判定することと

を備える、リパッケージング対策のための動作を実行するためにコンピュータシステムによって実行可能な1つまたは複数の命令を記憶する、コンピュータ可読記憶媒体。

10

【請求項8】

前記ターゲットファイルが、所定のコードがコンパイルされた後に取得されたターゲットファイルであり、

前記デジタル透かし情報が、前記ターゲットファイルの終端に埋め込まれる、

請求項7に記載のコンピュータ可読記憶媒体。

【請求項9】

前記デジタル透かし情報を取得することが、

前記ターゲットファイルのオリジナルの長さに基づいて前記ターゲットファイル内の前記デジタル透かし情報の開始アドレスを決定することであって、前記オリジナルの長さが、前記デジタル透かし情報が埋め込まれる前の前記ターゲットファイルの長さである、決定することであってことと、

前記開始アドレスに基づいて前記ターゲットファイルから前記デジタル透かし情報を取得することと

を含む、請求項7に記載のコンピュータ可読記憶媒体。

20

【請求項10】

前記デジタル透かし情報および前記検証情報に基づいて、前記インストールパッケージが前記リパッケージされたインストールパッケージであるかどうかを判定することが、

前記デジタル透かし情報が前記検証情報と一致するかどうかを判定することと、

前記デジタル透かし情報が前記検証情報と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定することと、

前記デジタル透かし情報が前記検証情報と一致しないと判定したことに応答して、前記インストールパッケージがリパッケージされたインストールパッケージであると判定することと

をさらに含む、請求項7に記載のコンピュータ可読記憶媒体。

30

【請求項11】

前記デジタル透かし情報および前記検証情報に基づいて前記インストールパッケージが前記リパッケージされたインストールパッケージであるかどうかを判定することが、

前記デジタル透かし情報が命令であるときに、

動作結果を取得するために前記命令に基づいて動作を実行することと、

前記動作結果が前記検証情報と一致するかどうかを判定することと、

前記動作結果が前記検証情報と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定することと、

前記動作結果が前記検証情報と一致しないと判定したことに応答して、前記インストールパッケージがリパッケージされたインストールパッケージであると判定することと

をさらに含む、請求項7に記載のコンピュータ可読記憶媒体。

40

【請求項12】

前記動作結果が前記検証情報内の前記動作結果と一致し、かつ前記命令が前記検証情報内の前記命令と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定することと、

前記動作結果が前記検証情報と一致しないと判定したことに応答して、前記インスト

50

ルパッケージが前記リパッケージされたインストールパッケージであると判定することと
をさらに含む、請求項7に記載のコンピュータ可読記憶媒体。

【請求項13】

リパッケージング対策のためのコンピュータ実装システムであって、
1つまたは複数のコンピュータと、

前記1つまたは複数のコンピュータに相互動作可能に結合される1つまたは複数のコン
ピュータメモリデバイスであって、1つまたは複数の命令を記憶する有形の機械可読記憶
媒体を有し、前記1つまたは複数のコンピュータによって実施されたとき、

インストールパッケージ内のターゲットファイルを実施することと、

前記インストールパッケージ内の安全な動的ライブラリをロードすることと、

前記ターゲットファイル内のコードに基づいて、前記ターゲットファイル内に埋め込
まれたデジタル透かし情報および前記安全な動的ライブラリ内に記憶された検証情報を取
得することと、

前記デジタル透かし情報および前記検証情報に基づいて、前記インストールパッケー
ジがリパッケージされたインストールパッケージであるかどうかを判定することであって

—

前記デジタル透かし情報が命令であるときに、

動作結果を取得するために前記命令に基づいて動作を実行することと、

前記動作結果が前記検証情報内の動作結果と一致するかどうかを判定することと

—

前記命令が前記検証情報内の命令と一致するかどうかを判定することと
を備える、判定することと

を備える1つまたは複数の動作を実行する、コンピュータ実装システム。

【請求項14】

前記ターゲットファイルが、所定のコードがコンパイルされた後に取得されたターゲッ
トファイルであり、

前記デジタル透かし情報が、前記ターゲットファイルの終端に埋め込まれる、

請求項13に記載のコンピュータ実装システム。

【請求項15】

前記デジタル透かし情報を取得することが、

前記ターゲットファイルのオリジナルの長さに基づいて前記ターゲットファイル内の前
記デジタル透かし情報の開始アドレスを決定することであって、前記オリジナルの長さが
、前記デジタル透かし情報が埋め込まれる前の前記ターゲットファイルの長さである、決
定することであってことと、

前記開始アドレスに基づいて前記ターゲットファイルから前記デジタル透かし情報を取
得することと

を含む、請求項13に記載のコンピュータ実装システム。

【請求項16】

前記デジタル透かし情報および前記検証情報に基づいて、前記インストールパッケージ
が前記リパッケージされたインストールパッケージであるかどうかを判定することが、

前記デジタル透かし情報が前記検証情報と一致するかどうかを判定することと、

前記デジタル透かし情報が前記検証情報と一致すると判定したことに応答して、前記イ
ンストールパッケージがオリジナルのインストールパッケージであると判定することと、

前記デジタル透かし情報が前記検証情報と一致しないと判定したことに応答して、前記
インストールパッケージがリパッケージされたインストールパッケージであると判定する
ことと

をさらに含む、請求項13に記載のコンピュータ実装システム。

【請求項17】

前記デジタル透かし情報および前記検証情報に基づいて前記インストールパッケージが
前記リパッケージされたインストールパッケージであるかどうかを判定することが、

10

20

30

40

50

前記デジタル透かし情報が命令であるときに、

動作結果を取得するために前記命令に基づいて動作を実行することと、

前記動作結果が前記検証情報と一致するかどうかを判定することと、

前記動作結果が前記検証情報と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定することと、

前記動作結果が前記検証情報と一致しないと判定したことに応答して、前記インストールパッケージがリパッケージされたインストールパッケージであると判定することとをさらに含む、請求項13に記載のコンピュータ実装システム。

【請求項18】

前記動作結果が前記検証情報内の前記動作結果と一致し、かつ前記命令が前記検証情報内の前記命令と一致すると判定したことに応答して、前記インストールパッケージがオリジナルのインストールパッケージであると判定することと、

前記動作結果が前記検証情報と一致しないと判定したことに応答して、前記インストールパッケージが前記リパッケージされたインストールパッケージであると判定することとをさらに含む、請求項13に記載のコンピュータ実装システム。

【発明の詳細な説明】

【技術分野】

【0001】

本出願は、コンピュータ技術の分野に関し、詳細には、リパッケージング対策のための方法および装置に関する。 10

【背景技術】

【0002】

現在、多くの悪意のあるアプリケーションソフトが、オリジナルのインストールパッケージ内に広告を埋め込み、マルウェアを自動的にダウンロードしてルートプログラムなどを実施するために、リパッケージングを介してオリジナルのインストールパッケージを改変している。 20

【0003】

リパッケージングプロセスを図1に示す。最初に、オリジナルのインストールパッケージは、オリジナルのインストールパッケージのソースコードを取得するために逆コンパイルされる。次いで、ユーザは、ソースコードを改変して、たとえば別のコードを追加し、追加されたコードは、広告であってもよいか、またはマルウェアなどを自動的にダウンロードするためのプログラムであってもよい。最終的に、改変されたファイルは、リパッケージされたインストールパッケージを取得するためにリパッケージされる。 30

【0004】

ソースコードは、前のリパッケージングプロセスにおいて改変されるので、リパッケージされたインストールパッケージの自己署名(self-signature)は、もはやオリジナルのインストールパッケージの自己署名ではない。したがって、既存の技術において、リパッケージング対策のための方法は、インストールパッケージがリパッケージされているかどうかを判断するために、インストールパッケージの自己署名を検査することであってもよい。特定の検証プロセスを図2に示す。 40

【0005】

特定のインストールパッケージがインストールされると、インストールパッケージ内のターゲットファイルが実施され、インストールパッケージ内の安全な動的ライブラリがロードされる。一般に、情報セキュリティを確保するために、いくつかの検証情報が、安全な動的ライブラリ内に記憶される。検証情報は、インストールパッケージに対応するオリジナルのインストールパッケージの自己署名を含む。オペレーティングシステムは、インストールパッケージの自己署名を検証するためのインターフェースを提供するので、インストールパッケージの自己署名をターゲットファイルから取得し、オリジナルのインストールパッケージの自己署名を安全な動的ライブラリから取得した後、オペレーティングシステムは、インストールパッケージの自己署名がオリジナルのインストールパッケージの 50

自己署名と一致するかどうかを検証する。それらが一致する場合、オペレーティングシステムは、インストールパッケージはオリジナルのインストールパッケージであると判定する。それらが一致しない場合、オペレーティングシステムは、インストールパッケージはリパッケージされたインストールパッケージであると判定する。

【0006】

既存の技術では、リパッケージング対策のための別の方法は、下記であってもよい。オペレーティングシステムが、今回インストールされたインストールパッケージのハッシュ(ハッシュ値)を計算してハッシュ値を検査し、すなわちそのハッシュ値をオリジナルのインストールパッケージのハッシュ値と比較し、それらが一致する場合、インストールパッケージはオリジナルのインストールパッケージであると判定し、それらが一致しない場合、インストールパッケージはリパッケージされたインストールパッケージであると判定する。10

【0007】

既存の技術におけるリパッケージング対策のための上記2つの方法は、リパッケージングを効果的に防止するために、完全に信頼できるオペレーティングシステム内で使用されなければならない。しかしながら、多くの既存のオペレーティングシステムはオープンソースであるので、ユーザはオリジナルのオペレーティングシステムを改変することができ、それによりオペレーティングシステムは、インストールパッケージの自己署名を検査せず、またはインストールパッケージのハッシュ値を検査しない。そのため、ユーザによってダウンロードされたインストールパッケージがリパッケージされたインストールパッケージであるか否かにかかわらず、オペレーティングシステムは、デフォルトで、インストールパッケージはオリジナルのインストールパッケージであるとみなす。20

【0008】

加えて、リパッケージング対策のための第2の方法では、インストールされるべきインストールパッケージが大きいメモリを占有することがある。そのため、オペレーティングシステムがインストールパッケージのハッシュ値を計算するとき、オペレーティングシステムの検証効率が影響を受ける。

【発明の概要】

【課題を解決するための手段】

【0009】

本出願は、システムがインストールパッケージの自己署名を検査するときに、インストールパッケージがリパッケージされているかどうかを効率的に検証することができないことがあるという既存の問題を緩和するため、およびインストールパッケージがリパッケージされているかどうかを検査するためにシステムがインストールパッケージのハッシュ値を計算するときに、インストールパッケージが大きいメモリを使用しているので検証効率が低いことがあるという既存の問題を緩和するために、リパッケージング対策のための方法および装置を提供する。30

【0010】

本出願はリパッケージング対策のための方法を提供し、本方法は、インストールパッケージ内のターゲットファイルを実施してインストールパッケージ内の安全な動的ライブラリをロードするステップと、ターゲットファイル内のコードに基づいて、ターゲットファイル内に埋め込まれたデジタル透かし情報および安全な動的ライブラリ内に記憶された検証情報を取得するステップと、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証するステップとを実行するステップとを含む。40

【0011】

同様に、本出願はリパッケージング対策のための装置をさらに提供し、装置は実施ユニットと実行ユニットとを含み、実施ユニットは、インストールパッケージ内のターゲットファイルを実施してインストールパッケージ内の安全な動的ライブラリをロードするように構成され、実行ユニットは、ターゲットファイル内のコードに基づいて、ターゲットフ50

ファイル内に埋め込まれたデジタル透かし情報および安全な動的ライブラリ内に記憶された検証情報を取得することと、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証することを実行するように構成される。

【0012】

本出願で提供されるリパッケージング対策のための方法では、インストールパッケージは検証コードを含む。インストールパッケージがインストールされると、オペレーティングシステムは、ターゲットファイル内にデジタル透かし情報をコードに基づいて取得し、インストールパッケージ内の安全な動的ライブラリ内の検証情報を取得して、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証する。本出願で提供されるリパッケージング対策のための方法を使用することによって、以下の有益な効果が達成される。
10

【0013】

1. 本出願では、インストールパッケージは、インストールパッケージがリパッケージされているかどうかを検証するために使用されるコードを含む。したがって、オペレーティングシステムがどれほど改変されようとも、インストールパッケージを検証するプロセスを回避することはできない。上記によって、インストールパッケージがリパッケージされているかどうかを検証するためにインストールパッケージの自己署名を検査するとき、インストールパッケージの自己署名を検証するプロセスをシステムが省略することができるのでインストールパッケージがリパッケージされているかどうかを効率的に検証することができないという既存の問題が緩和される。
20

【0014】

2. 本出願では、インストールパッケージがリパッケージされているかどうかが、デジタル透かし情報および検証情報に基づいて検証される。したがって、インストールパッケージのハッシュ値を計算することによってインストールパッケージがリパッケージされているかどうかを検証する既存の技術における方法と比較して、本出願は、インストールパッケージがリパッケージされているかどうかの検証においてより高い効率を有する。

【0015】

本明細書で説明する添付の図面は、本出願に対してより深い理解を与え、本出願の一部を構成するために使用される。本出願および説明の概略的実装形態は本出願を説明するために使用され、本出願を不適切に制限するものではない。
30

【図面の簡単な説明】

【0016】

【図1】既存の技術における、リパッケージング方法を示す概略的フローチャートである。

【図2】既存の技術における、リパッケージング対策のための方法を示す概略的フローチャートである。

【図3】本出願の実装形態による、リパッケージング対策のための方法を示す概略的フローチャートである。

【図4】本出願の実装形態による、デジタル透かし情報をターゲットファイル内に埋め込む概略的フローチャートである。
40

【図5】本出願の実装形態による、リパッケージング対策のための別の方法を示す概略的フローチャートである。

【図6】本出願の実装形態による、デジタル透かし情報を識別するための方法を示す概略的フローチャートである。

【図7】本出願の実装形態による、リパッケージング対策のためのさらに別の方法を示す概略的フローチャートである。

【図8】本出願の実装形態による、リパッケージング対策のためのさらに別の方法を示す概略的フローチャートである。

【図9】本出願の実装形態による、リパッケージング対策のための装置を示す概略的構造
50

図である。

【発明を実施するための形態】

【0017】

本出願の目的、技術的解決策、および利点をより明確にするために、以下において本出願の特定の実装形態および対応する添付の図面を参照しながら本出願の技術的解決策を明確にかつ包括的に説明する。明らかに、説明する実装形態は、本出願の実装形態の全部ではなく一部にすぎない。本出願における実装形態に基づいて創造のための労力を伴わずに当業者によって取得されるすべての他の実装形態は、本出願の保護範囲の中に入る。

【0018】

以下において、添付の図面を参照しながら本出願の実装形態内で提供される技術的解決策を詳細に説明する。

【0019】

本出願は、システムがインストールパッケージの自己署名を検査するときに、インストールパッケージがリパッケージされているかどうかを効率的に検証することができないことがあるという既存の問題を緩和するため、およびインストールパッケージがリパッケージされているかどうかを検査するためにシステムがインストールパッケージのハッシュ値を計算するときに、インストールパッケージが大きいメモリを占有するので検証効率が低いことがあるという既存の問題を緩和するために、リパッケージング対策のための方法を提供する。本方法のプロセスが図3に示され、以下のステップを含む。

【0020】

ステップ301:インストールパッケージ内のターゲットファイルを実施し、インストールパッケージ内の安全な動的ライブラリをロードする。

【0021】

本ステップでは、ユーザが特定のインストールパッケージをダウンロードしてインストールパッケージをインストールすると、オペレーティングシステムはインストールパッケージ内のターゲットファイルを実施し、インストールパッケージ内の安全な動的ライブラリを同時にロードする。ターゲットファイルは所定のコードに基づいてコンパイルされたターゲットファイルであり、デジタル透かし情報はコンパイルされたターゲットファイル内に埋め込まれる。デジタル透かし情報は、文字列、命令などであってもよい。安全な動的ライブラリは、インストールパッケージについての検証情報を記憶し、検証情報は、インストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証するために使用される。

【0022】

デジタル透かし情報をターゲットファイル内に埋め込む前述の方法では、デジタル透かし情報は、ターゲットファイルの終端に埋め込まれてもよく、またはデジタル透かし情報は、ターゲットファイルの別のロケーションに埋め込まれてもよい。

【0023】

オペレーティングシステムは、ANDROID(登録商標)システムであってもよく、ターゲットファイルはdexファイルであってもよい。dexファイルは、ANDROIDシステム内の実行ファイルであり、Java(登録商標)コードを使用することによって記述される。安全な動的ライブラリは、共有オブジェクト(so)ライブラリであってもよく、c/c++を使用することによって記述される。一般に、soライブラリは、いくつかのセキュリティ情報、たとえば検証情報を記憶する。実行ファイルがロードまたは実施されると、soライブラリは、オペレーティングシステムによってロードされる。または、オペレーティングシステムはWINDOWS(登録商標)であってもよく、ターゲットファイルはexeファイルであってもよく、安全な動的ライブラリはdllライブラリなどである。オペレーティングシステム、ターゲットファイル、および安全な動的ライブラリは、本明細書では限定されない。

【0024】

図4に示すように、ターゲットファイルがdexファイルである場合、デジタル透かし情報がターゲットファイル内に埋め込まれているとき、ターゲットファイルのデータの完全性

10

20

30

40

50

および正確さを確保するために、ターゲットファイルヘッダのチェックサム、署名、ファイルサイズなどの対応する値が、デジタル透かし情報に基づいて再計算される。

【0025】

インストールパッケージがリパッケージされたインストールパッケージである場合、ユーザがオリジナルのインストールパッケージをリパッケージするとき、ユーザはオリジナルのインストールパッケージ内のオリジナルファイル内のコードを改変することができるとしても、オリジナルのインストールパッケージの安全な動的ライブラリ内の検証情報は容易には改変され得ないことに留意することは価値がある。したがって、インストールパッケージの安全な動的ライブラリは、依然としてオリジナルのインストールパッケージの検証情報を記憶している。デジタル透かし情報もまたオリジナルのインストールパッケージ内に埋め込まれていると仮定すると、安全な動的ライブラリ内に記憶された検証情報は、オリジナルファイル内に埋め込まれたデジタル透かし情報であってもよい。1つの理由を図5に示す。

【0026】

オリジナルのインストールパッケージがリパッケージされているとき、オリジナルのインストールパッケージに対応するソースコードファイルを取得するために、オリジナルのインストールパッケージが最初に逆コンパイルされる必要がある。デジタル透かし情報は、所定のコードに基づいてオリジナルファイル内にコンパイルされた後にオリジナルファイルに埋め込まれるので、オリジナルのインストールパッケージが逆コンパイルされているときにデジタル透かし情報Aは失われる。次いで、ユーザは、ソースコードファイルを改変することができる。図5に示すように、別のコードが追加されてもよい。加えて、ユーザは、リパッケージされたインストールパッケージを構築するために、デジタル透かし情報Bをソースコードファイル内にさらに埋め込む。最後に、ソースコードファイルがリパッケージされ、リパッケージされたインストールパッケージの安全な動的ライブラリは、依然として検証情報Aを記憶している。したがって、デジタル透かし情報Aは、オリジナルのインストールパッケージの識別子として使用することができ、インストールパッケージがリパッケージされているかどうかを検証するために使用される。

【0027】

本出願の実装形態を明確に説明するために、本出願の実装形態内のステップは、一例として、デジタル透かし情報をターゲットファイルの終端に埋め込むことによって、以下で説明される。

【0028】

ステップ302:ターゲットファイル内のコードに基づいて以下のステップ、すなわちターゲットファイル内に埋め込まれたデジタル透かし情報および安全な動的ライブラリ内に記憶された検証情報を取得するステップ、ならびにデジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証するステップを実行する。

【0029】

本ステップでは、オペレーティングシステムは、図6に示す方法、すなわちターゲットファイルのオリジナルの長さに基づいてターゲットファイル内のデジタル透かし情報の開始アドレスを決定するステップと、開始アドレスに基づいてターゲットファイルからデジタル透かし情報を取得するステップとを使用することによって、ターゲットファイル内のコードに基づいてターゲットファイル内のデジタル透かし情報を取得する。ターゲットファイルのオリジナルの長さは、本明細書では、デジタル透かし情報が埋め込まれる前のターゲットファイルの長さである。

【0030】

デジタル透かし情報を取得し、安全な動的ライブラリから検証情報を取得した後、オペレーティングシステムは、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証する。検証方法は、Table 1(表1)

10

20

30

40

50

【表1】

Table 1

デジタル透かし情報のタイプ	検証方法
固定長の文字列	文字列が検証情報と一致するかどうかを直接検証する
	透かし値が検証情報と一致するかどうかを検証する
ランダム長の文字列	最初にランダムな文字列の長さを計算し、次いで文字列が検証情報と一致するかどうかを検証する
命令	命令が検証情報と一致するかどうかを検証する
	動作結果が検証情報と一致するかどうかを検証する
	動作結果が検証情報内の動作結果と一致するかどうかを検証し、命令が検証情報内の命令と一致するかどうかを検証する
...	

10

20

に示すように、デジタル透かし情報のタイプに基づいて決定されてもよい。

【0031】

デジタル透かし情報が文字列であるとき、デジタル透かし情報が検証情報と一致するかどうかが検証される。デジタル透かし情報が検証情報と一致する場合、インストールパッケージはオリジナルのインストールパッケージであると判定される。デジタル透かし情報が検証情報と一致しない場合、インストールパッケージはリパッケージされたインストールパッケージであると判定される。

30

【0032】

実際の応用例では、デジタル透かし情報が文字列である場合、文字列は固定長の文字列であってもよく、またはランダム長の文字列であってもよい。デジタル透かし情報が固定長の文字列であるとき、文字列が安全な動的ライブラリ内の検証情報と一致するかどうかは、直接検証されるか、または透かし値が安全な動的ライブラリ内の検証情報と一致するかどうかを判定するために、現在ダウンロードされたインストールパッケージに対応する透かし値が計算される。図7は、dexファイル内の透かし値を計算することによって、インストールパッケージがリパッケージされているかどうかを検証するための方法を示す。詳細は以下のとおりである。

【0033】

dexファイルはdexヘッダとdexボディとを含む。dexヘッダはデータサイズとデータoffとを含み、dexボディはデータを含み、データサイズはデータのサイズを示し、データoffはデータのオフセットを示す。デジタル透かし情報はオリジナルのdexファイル内に埋め込まれたと仮定する。オリジナルのdexファイルがリパッケージされているとき、dexボディ内のデータ部分に対応するコードが改変され、コードが改変された後、オリジナルのdexファイル内に埋め込まれたデジタル透かし情報が失われるばかりでなく、データのサイズも変更される。したがって、dexヘッダ内のデータサイズに対応する値が変化する。オリジナルのdexファイルをリパッケージすることも、以下のとおりであってもよい。dexボディ内のデータ以外の部分に対応するコードが改変され、コードが改変された後、オリジナルのdexファイル内に埋め込まれたデジタル透かし情報もまた失われ、dexボディ内のデ

40

50

ータのロケーションがオフセットされる(図7に示す)。そのため、dexヘッダ内のデータoffに対応する値が変化する。

【0034】

オリジナルのインストールパッケージに対応する透かし値を計算するための方法は、「データサイズ+データoff」である、すなわち透かし値は、データサイズに対応する値とデータoffに対応する値との和である、と仮定する。ユーザがオリジナルのインストールパッケージをリパッケージするとき、データoffまたはデータサイズに対応する値が変化し、リパッケージされたインストールパッケージに対応する透かし値も変化する。この時点において、オペレーティングシステムは、リパッケージされたインストールパッケージに対応する透かし値が安全な動的ライブラリ内の検証情報と一致するかどうかを検証することによって、インストールパッケージがリパッケージされているかどうかを判定することができる。本明細書の透かし値を計算するための方法は、一例にすぎない。実際の応用例では、透かし値を計算するための方法は、実際の状況に基づいて決定することができ、たとえば、「データサイズ-データoff」、「データサイズ/データoff」などであってもよい。10

【0035】

前の方針が、現在ダウンロードされたインストールパッケージがリパッケージされているかどうかを検証するために使用されるとき、オリジナルのインストールパッケージ内のデジタル透かし情報がリパッケージされたインストールパッケージ内に埋め込まれている場合でも、インストールパッケージに対応する透かし値は変化する。したがって、オペレーティングシステムは、インストールパッケージの透かし値に基づいて、インストールパッケージがリパッケージされているかどうかを正確に判断することができる。20

【0036】

デジタル透かし情報がランダム長の文字列であるとき、文字列の長さを最初に計算することができ、次いで文字列の長さが安全な動的ライブラリ内の検証情報の長さと一致するかどうかが検証され、文字列の長さが安全な動的ライブラリ内の検証情報の長さと一致しない場合、インストールパッケージはリパッケージされたインストールパッケージであることが直ちに決定され、または文字列の長さが安全な動的ライブラリ内の検証情報の長さと一致する場合、文字列が安全な動的ライブラリ内の検証情報と一致するかどうかが検証され、文字列が同じく安全な動的ライブラリ内の検証情報と一致する場合、インストールパッケージはオリジナルのインストールパッケージであると判定され、または文字列が安全な動的ライブラリ内の検証情報と一致しない場合、インストールパッケージはリパッケージされたインストールパッケージであると判定される。30

【0037】

デジタル透かし情報が命令であるとき、インストールパッケージを検証するために複数の方法が存在する。例として、3つの検証方法を以下で説明する。

【0038】

第1の方法は、デジタル透かし情報が文字列であるときにインストールパッケージを検証するための前の方法と同様である。オペレーティングシステムは、命令が安全な動的ライブラリ内の検証情報と一致するかどうかを検証し、命令が安全な動的ライブラリ内の検証情報と一致する場合、オペレーティングシステムは、インストールパッケージはオリジナルのインストールパッケージであると判定し、命令が安全な動的ライブラリ内の検証情報と一致しない場合、オペレーティングシステムは、インストールパッケージはリパッケージされたインストールパッケージであると判定する。40

【0039】

第2の方法では、オペレーティングシステムは、動作結果を取得するために命令に基づいて対応する動作を実行し、次いで動作結果が検証情報と一致するかどうかを検証し、動作結果が検証情報と一致する場合、オペレーティングシステムは、インストールパッケージはオリジナルのインストールパッケージであると判定し、動作結果が検証情報と一致しない場合、オペレーティングシステムは、インストールパッケージはリパッケージされたインストールパッケージであると判定する。50

インストールパッケージであると判定する。

【0040】

たとえば、命令は、「XXアドレスにおけるコンテンツが安全な動的ライブラリ内の検証情報と一致するかどうかの照会」であってもよい。オペレーティングシステムは、命令に基づいてターゲットファイル内のXXアドレスからのアドレスに対応するコンテンツ(動作結果)を照会し、コンテンツが安全な動的ライブラリ内の検証情報と一致するかどうかを検証する。コンテンツが安全な動的ライブラリ内の検証情報と一致する場合、オペレーティングシステムは、インストールパッケージはオリジナルのインストールパッケージであると判定する。コンテンツが安全な動的ライブラリ内の検証情報と一致しない場合、オペレーティングシステムは、インストールパッケージはリパッケージされたインストールパッケージであると判定する。10

【0041】

第3の方法では、オペレーティングシステムは、動作結果を取得するために命令に基づいて対応する動作を実行し、動作結果が検証情報内の動作結果と一致するかどうかを検証し、かつ命令が検証情報内の命令と一致するかどうかを検証する。2つの検証の結果が両方とも「一致」である場合、オペレーティングシステムは、インストールパッケージはオリジナルのインストールパッケージであると判定する。2つの検証のいずれかの結果が「不一致」であるか、または2つの検証の結果が両方とも「不一致」である場合、オペレーティングシステムは、インストールパッケージはリパッケージされたインストールパッケージであると判定する。20

【0042】

検証方法を明確に説明するために、本明細書で簡単な例を提供する。安全な動的ライブラリ内の検証情報内の動作結果が“2”であると仮定すると、動作命令は“1+1”であり、ターゲットファイル内のデジタル透かし情報に対応する命令は“3-1”であり、オペレーティングシステムが命令に基づいて動作を実行した後に取得された動作結果も“2”である。オペレーティングシステムが、動作結果が検証情報内の動作結果と一致するかどうかを検証するのみである場合、オペレーティングシステムは、誤ってインストールパッケージはオリジナルのインストールパッケージであるものとみなすが、オペレーティングシステムがさらに、命令が検証情報内の命令と一致するかどうかを検証する場合、オペレーティングシステムは、正確にインストールパッケージはリパッケージされているかどうかを判定する。30

【0043】

ターゲットファイル内のデジタル透かし情報に対応する命令は、単純な「リターン」命令であってもよく、またはオペレーティングシステムと相互作用するために使用される別の命令であってもよい。たとえば、命令は、「ターゲットファイルの自己署名が安全な動的ライブラリ内の検証情報と一致するかどうかをオペレーティングシステムに検証させる」であってもよい。別の例として、命令は、「デジタル透かし情報が埋め込まれる前のターゲットファイルの長さが安全な動的ライブラリ内に記憶された検証情報と一致するかどうかをオペレーティングシステムに検証させる」であってもよい。命令は、ユーザのニーズに基づいてセットされてもよい。40

【0044】

加えて、インストールパッケージがリパッケージされたインストールパッケージである場合、以下の場合があってもよい。デジタル透かし情報が、インストールパッケージに対応するオリジナルファイル内に埋め込まれる。オリジナルのインストールパッケージをリパッケージするプロセスにおいて、オリジナルのインストールパッケージ内のデジタル透かし情報が失われることが前のコンテンツから理解されてもよい。ユーザが、オリジナルファイルのコードを改変した後にデジタル透かし情報を追加しない場合、オペレーティングシステムは、インストールパッケージ内のコードに基づいて、インストールパッケージのターゲットファイルがデジタル透かし情報を含むかどうかを照会することができる。

【0045】

50

インストールパッケージを検証するための前の方法は一例にすぎない。実際の応用例では、複数のタイプのデジタル透かし情報が存在し、複数の対応する検証方法も存在し、それらは本明細書では限定されない。

【0046】

本出願で提供されるリパッケージング対策のための方法では、インストールパッケージは検証コードを含む。インストールパッケージがインストールされるとき、図8に示すように、オペレーティングシステムは、コードに基づいてインストールパッケージ内の安全な動的ライブラリ内の検証情報を取得し、ターゲットファイル内に埋め込まれたデジタル透かし情報を取得し、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証する。本出願で提供されるリパッケージング対策のための方法を使用することによって、以下の有益な効果が達成される。10

【0047】

1. 本出願では、インストールパッケージは、インストールパッケージがリパッケージされているかどうかを検証するために使用されるコードを含む。したがって、オペレーティングシステムがどれほど改変されようとも、インストールパッケージを検証するプロセスを回避することはできない。上記によって、インストールパッケージがリパッケージされているかどうかを検証するためにインストールパッケージの自己署名を検査するときに、インストールパッケージの自己署名を検証するプロセスをシステムが省略することがあるためインストールパッケージがリパッケージされているかどうかを効率に検証することができないという既存の問題が緩和される。20

【0048】

2. 本出願では、インストールパッケージがリパッケージされているかどうかが、デジタル透かし情報および検証情報に基づいて検証される。したがって、インストールパッケージのハッシュ値を計算することによってインストールパッケージがリパッケージされているかどうかを検証する既存の技術における方法と比較して、本出願は、インストールパッケージがリパッケージされているかどうかの検証においてより高い効率を有する。

【0049】

同様に、本出願は、システムがインストールパッケージの自己署名を検査するときに、インストールパッケージがリパッケージされているかどうかを効率的に検証することができないことがあるという既存の問題を緩和するため、およびインストールパッケージがリパッケージされているかどうかを検査するためにシステムがインストールパッケージのハッシュ値を計算するときに、インストールパッケージが大きいメモリを占有するので検証効率が低いことがあるという既存の問題を緩和するために、リパッケージング対策のための装置をさらに提供する。装置の構造が図9に示され、以下のユニット、すなわち実施ユニット901と実行ユニット902とを含む。30

【0050】

実施ユニット901は、インストールパッケージ内のターゲットファイルを実施し、インストールパッケージ内の安全な動的ライブラリをロードするように構成される。

【0051】

実行ユニット902は、ターゲットファイル内のコードに基づいて、ターゲットファイル内に埋め込まれたデジタル透かし情報および安全な動的ライブラリ内に記憶された検証情報を取得することと、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証することとを実行するように構成される。40

【0052】

本装置の実装形態の作業フローは以下のとおりである。最初に、実施ユニット901は、インストールパッケージ内のターゲットファイルを実施してインストールパッケージ内の安全な動的ライブラリをロードし、次いで実行ユニット902は、ターゲットファイル内のコードに基づいて、ターゲットファイル内に埋め込まれたデジタル透かし情報および安全

な動的ライブラリ内に記憶された検証情報を取得することと、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証することとを実行する。

【 0 0 5 3 】

本装置の実装形態では、リパッケージング対策が多くの方法を使用することによって実装される。たとえば、第1の実装形態では、ターゲットファイルは所定のコードがコンパイルされた後に取得されたターゲットファイルであり、デジタル透かし情報はコンパイルされたターゲットファイルの終端に埋め込まれる。

【 0 0 5 4 】

第2の実装形態では、ターゲットファイル内に埋め込まれたデジタル透かし情報を取得することは、ターゲットファイルのオリジナルの長さに基づいてターゲットファイル内のデジタル透かし情報の開始アドレスを決定することであって、オリジナルの長さはデジタル透かし情報が埋め込まれる前のターゲットファイルの長さである、決定することと、開始アドレスに基づいてターゲットファイルからデジタル透かし情報を取得することとを含む。10

【 0 0 5 5 】

第3の実装形態では、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証することは、デジタル透かし情報が検証情報と一致するかどうかを検証することと、デジタル透かし情報が検証情報と一致する場合にインストールパッケージはオリジナルのインストールパッケージであると判定することか、またはデジタル透かし情報が検証情報と一致しない場合にインストールパッケージはリパッケージされたインストールパッケージであると判定することとを含む。20

【 0 0 5 6 】

第4の実装形態では、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証することは、デジタル透かし情報が命令であるときに動作結果を取得するために命令に基づいて動作を実行することと、動作結果が検証情報と一致するかどうかを検証することと、動作結果が検証情報と一致する場合にインストールパッケージはオリジナルのインストールパッケージであると判定することか、または動作結果が検証情報と一致しない場合にインストールパッケージはリパッケージされたインストールパッケージであると判定することとを含む。30

【 0 0 5 7 】

第5の実装形態では、デジタル透かし情報および検証情報に基づいてインストールパッケージがリパッケージされたインストールパッケージであるかどうかを検証することは、デジタル透かし情報が命令であるときに動作結果を取得するために命令に基づいて動作を実行することと、動作結果が検証情報内の動作結果と一致するかどうかを検証することと、命令が検証情報内の命令と一致するかどうかを検証することと、動作結果が検証情報内の動作結果と一致しつつ命令が検証情報内の命令と一致する場合にインストールパッケージはオリジナルのインストールパッケージであると判定することか、またはそうではない場合にインストールパッケージはリパッケージされたインストールパッケージであると判定することとを含む。40

【 0 0 5 8 】

本装置の実装形態において達成される有益な効果は、前の方の実装形態において達成される有益な効果と同じかまたは同様である。繰り返しを避けて簡明にするために、本明細書では詳細を省略する。

【 0 0 5 9 】

本発明の実装形態が方法、システム、またはコンピュータプログラム製品として提供されてもよいことは、当業者には理解されよう。それゆえ、本発明は、ハードウェアのみの実装形態、ソフトウェアのみの実装形態、またはソフトウェアとハードウェアの組合せの50

実装形態の形を取ることができる。加えて、本発明は、コンピュータ使用可能プログラムコードを含む1つまたは複数のコンピュータ使用可能記憶媒体(限定はしないが、ディスクメモリ、CD-ROM、光学メモリなどを含む)上で実装されるコンピュータプログラム製品の形を取ることができる。

【0060】

本発明は、本発明の実装形態による方法、デバイス(システム)、およびコンピュータプログラム製品のフローチャートおよび/またはブロック図を参照して説明される。コンピュータプログラム命令は、フローチャートおよび/またはブロック図における各プロセスおよび/または各ブロック、ならびにフローチャートおよび/またはブロック図におけるプロセスおよび/またはブロックの組合せを実装するために使用されてもよいことを理解されたい。これらのコンピュータプログラム命令は、汎用コンピュータ、専用コンピュータ、埋め込みプロセッサ、またはマシンを生成するための任意の他のプログラマブルデータ処理デバイスのプロセッサに提供することができ、それによりコンピュータまたは任意の他のプログラマブルデータ処理デバイスのプロセッサによって実行される命令は、フローチャート内の1つもしくは複数のプロセス内、および/またはブロック図の1つもしくは複数のブロック内に特定の機能を実装するためのデバイスを生成する。10

【0061】

コンピュータまたは任意の他のプログラマブルデータ処理デバイスに対して、特定の方法を使用することによって働くように命令することができる、これらのコンピュータプログラム命令をコンピュータ可読メモリ内に記憶することができ、それによりコンピュータ可読メモリ内に記憶された命令は、命令デバイスを含む製造品を生成する。命令デバイスは、フローチャート内の1つもしくは複数のプロセス内、および/またはブロック図の1つもしくは複数のブロック内で特定の機能を実施する。20

【0062】

これらのコンピュータプログラム命令は、コンピュータまたは別のプログラマブルデータ処理デバイスにロードすることができ、それにより一連の動作およびステップがコンピュータまたは別のプログラマブルデバイス上で実行され、それによりコンピュータ実装処理が生成される。それゆえ、コンピュータまたは別のプログラマブルデバイス上で実行される命令は、フローチャート内の1つもしくは複数のプロセス内、および/またはブロック図の1つもしくは複数のブロック内で、特定の機能を実施するためのステップを提供する30。

【0063】

典型的な構成では、コンピューティングデバイスは、1つまたは複数の中央処理装置(CPU)と、1つまたは複数の入力/出力インターフェースと、1つまたは複数のネットワークインターフェースと、1つまたは複数のメモリとを含む。

【0064】

メモリは、コンピュータ可読媒体内の揮発性メモリ、ランダムアクセスメモリ(RAM)、および/またはリードオンリーメモリ(ROM)もしくはフラッシュメモリ(フラッシュRAM)などの不揮発性メモリなどを含むことができる。メモリは、コンピュータ可読媒体の一例である。40

【0065】

コンピュータ可読媒体は、任意の方法または技術を使用することによって情報を記憶することができる、永久的および非永久的、取り外し可能および取り外し不可能な媒体を含む。情報は、コンピュータ可読命令、データ構造、プログラムのモジュール、または他のデータであってもよい。コンピュータ記憶媒体の例は、限定はしないが、相ランダム変化メモリ(PRAM:phase-random change memory)、スタティックランダムアクセスメモリ(SRAM)、ダイナミックランダムアクセスメモリ(DRAM)、別のタイプのランダムアクセスメモリ(RAM)、リードオンリーメモリ(ROM)、電気的消去可能プログラマブルリードオンリーメモリ(EEPROM)、フラッシュメモリもしくは別のメモリ技術、コンパクトディスクドライブ(Compact Disc Drive)、CD-ROM、デジタル多用途ディスク(DVD)もしくは別の光学式記憶、磁気カセット

ットテープ、磁気テープ/ディスク記憶、または別の磁気記憶デバイスもしくは任意の他の非伝送媒体を含み、コンピューティングデバイスによってアクセスされてもよい情報を記憶するために使用することができる。本明細書で定義されるように、コンピュータ可読媒体は、変調されたデータ信号およびキャリアなどの一時的コンピュータ可読媒体(一時的媒体)を含まない。

【0066】

「含む」、「包含する」という用語、またはそれらの任意の他の変形は、非排他的の包含をカバーすることを意図しており、それにより要素のリストを含むプロセス、方法、製品、またはデバイスは、それらの要素を含むばかりでなく、明示的に列挙されていない他の要素をも含むか、またはそのようなプロセス、方法、製品、もしくはデバイスに固有の要素をさらに含むことにさらに留意することは価値がある。「～を含む」に先行する要素は、その要素を含むプロセス、方法、製品、またはデバイス内に付加的な同等の要素の存在を排除することではなく、より制約することもない。

10

【0067】

本明細書の実装形態が方法、システム、またはコンピュータプログラム製品として提供されてもよいことは、当業者には理解されよう。それゆえ、本出願は、ハードウェアのみの実装形態、ソフトウェアのみの実装形態、またはソフトウェアとハードウェアの組合せの実装形態の形を取ることができる。加えて、本出願は、コンピュータ使用可能プログラムコードを含む1つまたは複数のコンピュータ使用可能記憶媒体(限定はしないが、ディスクメモリ、CD-ROM、光学メモリなどを含む)上で実装されるコンピュータプログラム製品の形を取ることができる。

20

【0068】

前の説明は、本出願の実装形態にすぎず、本出願を制限することを意図していない。当業者にとって、本出願における様々な変更が存在してもよい。本出願の趣旨および原理内で作成される任意の変更形態、等価な置換、改善などは、本出願の特許請求の範囲内に入るものとする。

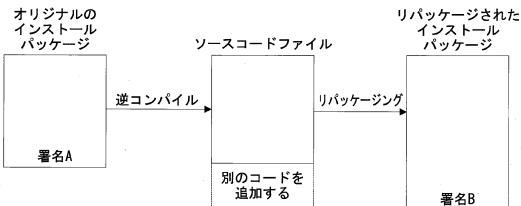
【符号の説明】

【0069】

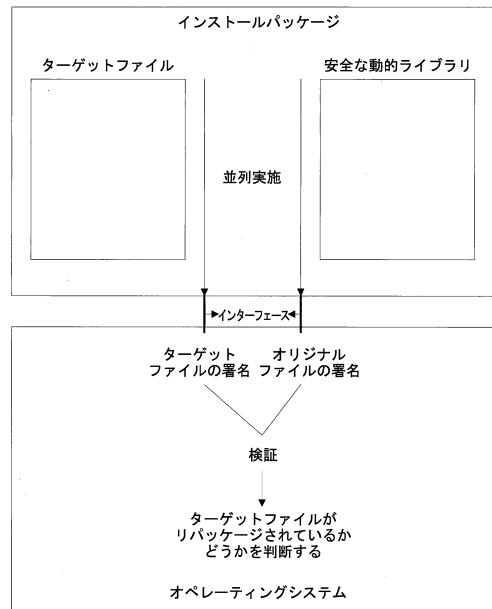
- 901 実施ユニット
- 902 実行ユニット

30

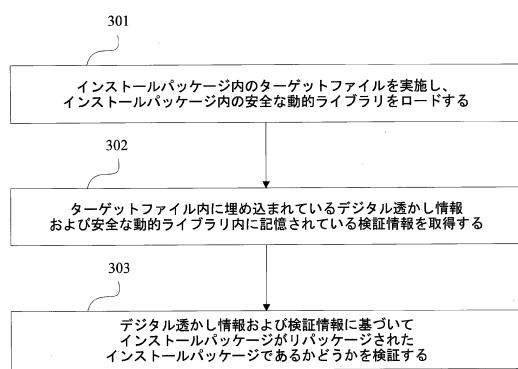
【図1】



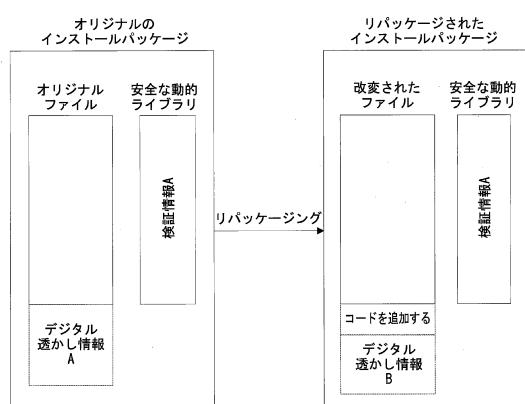
【図2】



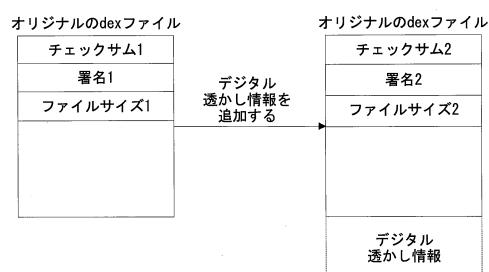
【図3】



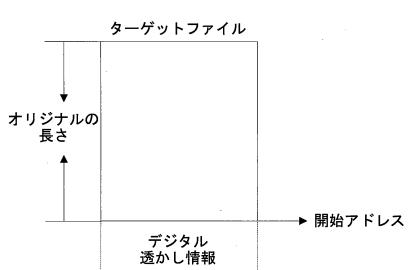
【図5】



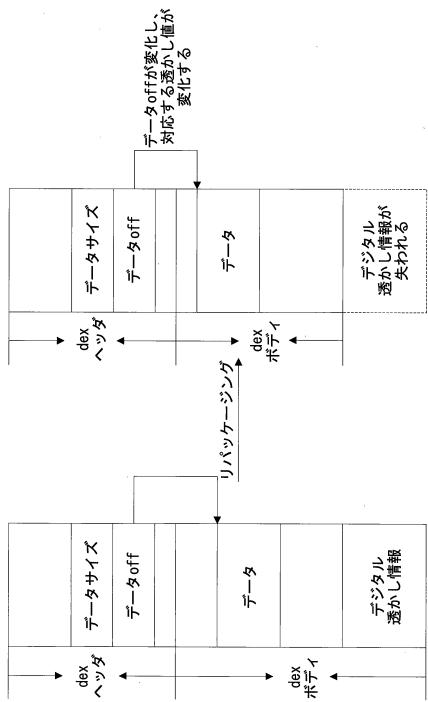
【図4】



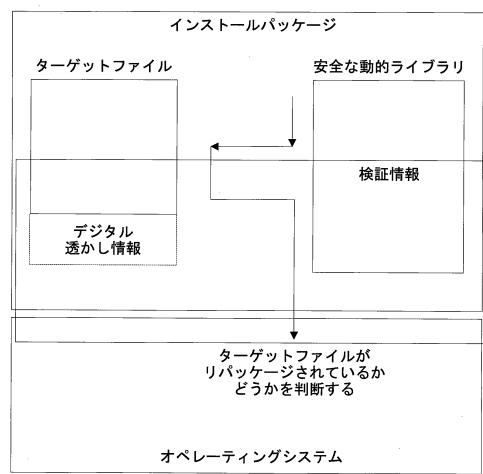
【図6】



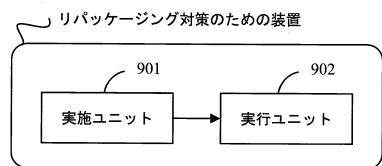
【図7】



【図8】



【図9】



フロントページの続き

(72)発明者 陳 耀光

中華人民共和国 311121 浙江省杭州市余杭区文一西路969号 3号楼5楼阿里巴巴集团
法務部

(72)発明者 王 加水

中華人民共和国 311121 浙江省杭州市余杭区文一西路969号 3号楼5楼阿里巴巴集团
法務部

審査官 宮司 卓佳

(56)参考文献 国際公開第2008/010508 (WO, A1)

特開平10-124490 (JP, A)

特開2013-228814 (JP, A)

米国特許出願公開第2004/0172544 (US, A1)

(58)調査した分野(Int.Cl., DB名)

G06F21/00 - 21/88