(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第4693245号
(P4693245)

(45)発行日　平成23年6月1日(2011.6.1)　(24)登録日　平成23年3月4日(2011.3.4)

(51)Int.Cl.		F I		
G06F 21/22	(2006.01)	G06F	9/06	660J
G06F 21/24	(2006.01)	G06F	12/14	560C
G06F 9/30	(2006.01)	G06F	9/30	380
G06K 19/073	(2006.01)	G06K	19/00	P

請求項の数 14　(全 6 頁)

(21)出願番号　特願2000-614120 (P2000-614120)
(86)(22)出願日　平成12年4月19日 (2000.4.19)
(65)公表番号　特表2002-543492 (P2002-543492A)
(43)公表日　平成14年12月17日 (2002.12.17)
(86)国際出願番号　PCT/EP2000/003530
(87)国際公開番号　WO2000/065442
(87)国際公開日　平成12年11月2日 (2000.11.2)
審査請求日　平成19年4月19日 (2007.4.19)
(31)優先権主張番号　199 18 620.0
(32)優先日　平成11年4月23日 (1999.4.23)
(33)優先権主張国　ドイツ(DE)

(73)特許権者　596007511
ギーゼッケ　ウント　デブリエント　ゲーエムベーハー
Ｇｉｅｓｅｃｋｅ　＆　Ｄｅｖｒｉｅｎｔ　ＧｍｂＨ
ドイツ連邦共和国　Ｄ－８１６７７　ミュンヘン　プリンツレーゲンテンシュトラッセ　１５９
(74)代理人　100073184
弁理士　柳田　征史
(74)代理人　100090468
弁理士　佐久間　剛

最終頁に続く

(54)【発明の名称】外部からの不正操作に対するコンピュータコアの保護

(57)【特許請求の範囲】

【請求項１】

中央処理装置(ＣＰＵ)をもつコンピュータを外部からの不正操作に対して保護するための方法において：

最終チェックサムが、前記ＣＰＵによる命令の処理終了時に発生する前記ＣＰＵのレジスタの内容に基づく数学的組合せにより生成されて、格納され；

開始チェックサムが、前記ＣＰＵによる次の命令の処理開始前に発生する前記レジスタの内容に基づいて生成され；

前記開始チェックサムが前記最終チェックサムと一致しない場合にエラーメッセージが出されることを特徴とする方法。

【請求項２】

前記命令のローディングに際して、前記命令を実行するために必要なクロックサイクル数を計数し、前記計数された必要なクロックサイクル数があらかじめ定められたクロックサイクル数より多いかまたは少ない場合にエラー信号を出力するために、カウンタをスタートさせることを特徴とする請求項１記載の方法。

【請求項３】

前記エラー信号がクロック信号の供給を中断させるか、あるいは前記クロック信号供給の停止を生じさせることを特徴とする請求項２記載の方法。

【請求項４】

ある命令を実行するために必要な前記クロックサイクル数が論理回路により前記命令の

オペレーションコード(ｏｐｃｏｄｅ)から得られることを特徴とする請求項１から３いずれか１項記載の方法。
【請求項５】
前記数学的組合せが前記レジスタ内容の排他的論理和演算により生じることを特徴とする請求項１から４いずれか１項記載の方法。
【請求項６】
前記方法の開始がランダムまたは所定のイベントにより起動されることを特徴とする請求項１から５いずれか１項記載の方法。
【請求項７】
前記方法が時間に基づく態様で起動されることを特徴とする請求項６記載の方法。
【請求項８】
前記方法が前記ＣＰＵの１つまたはそれ以上のレジスタの内容があらかじめ定められたパターンに一致するときに起動されることを特徴とする請求項６記載の方法。
【請求項９】
前記方法が、状況ごとに、あらかじめ定められた数の命令の処理後に起動されることを特徴とする請求項６記載の方法。
【請求項１０】
請求項１から６いずれか１項記載の方法を実行するためのコンピュータのための中央処理装置(ＣＰＵ)において：
－　チェックサムを生成するため、論理素子によって前記ＣＰＵのいくつかのレジスタを組み合わせたもの；
－　前記論理素子により生成された第１のチェックサムを格納するためのチェックサムメモリ；
－　前記論理素子により生成された第２のチェックサムを前記メモリに格納された前記第１のチェックサムと比較するためのコンパレータ；及び
－　前記チェックサムメモリへの前記第１のチェックサムの前記格納を制御するため及び前記コンパレータを制御するための制御デバイス；
を備えることを特徴とする中央処理装置。
【請求項１１】
命令実行に必要とされるクロックサイクル数を計数するためのカウンタを備えることを特徴とする請求項１０記載の中央処理装置。
【請求項１２】
命令実行に必要なクロックサイクル数を前記命令のオペレーションコードから決定するための論理回路を備えることを特徴とする請求項１０または１１記載の中央処理装置。
【請求項１３】
請求項１０から１２いずれか１項記載の中央処理装置を備えることを特徴とするコンピュータ。
【請求項１４】
請求項１０から１２いずれか１項記載の中央処理装置を備えることを特徴とするスマートカード。
【発明の詳細な説明】
【０００１】
発明の属する技術分野
本発明は外部からの不正操作に対するコンピュータの保護に関し、特に、コンピュータコアすなわち中央処理装置(ＣＰＵ)に存在するデータの保護に関する。
【０００２】
発明の背景
例えばバス暗号化、メモリ暗号化等により、コンピュータのメモリ領域を不正操作から保護することが知られている。例えば、ドイツ国特許第３７　０９　５２４　Ｃ２号はプログラムメモリの格納セル内容をチェックするための検査ルーチンを開示している。プログラ

ムの実行開始時または実行中に格納セル内容全体にわたるチェックサムを生成し、これを前もってプログラムメモリに格納されたチェックサムと比較することにより、本来の格納セル内容の変化も動作中にのみおこる変化も検出することができ、変化があればエラーメッセージが出される。

【０００３】

発明の概要

本発明の課題は外部からの不正操作に対してコンピュータをより有効に保護する方法を提出することである。

【０００４】

上記課題は、本発明にしたがい、方法、その方法を実行するための中央処理装置、及び特許請求の範囲の独立請求項の特徴にしたがうような中央処理装置をもつコンピュータ及びスマートカードにより解決される。本発明の有用な実施形態は従属請求項に提示される。

【０００５】

本発明は、コンピュータコア、すなわちコンピュータの中央処理装置(ＣＰＵ)にあるデータは非暗号化形態にあり、したがって容易に不正操作できるから、コンピュータコアにあるデータを外部からの不正操作に対して保護することにより、コンピュータの保全性を高めるという着想を出発点としている。

【０００６】

そのような処置を実現するために、命令がＣＰＵにより処理された後に、ＣＰＵのいくつかのレジスタの内容から数学的組み合わせにより、例えば排他的論理和演算(ＸＯＲ演算)によりチェックサムが決定されて、最終チェックサムとしてメモリに格納される。次の命令がＣＰＵにより処理される前にチェックサムが再び生成される。すなわち開始チェックサムである。開始チェックサムを、一致しなければならない、最終チェックサムと比較することにより、ＣＰＵのレジスタ内容が最終命令処理後に不正操作されたか否かを確認することができる。レジスタ内容として、８０５１タイプのプロセッサにある、アキュムレータ(ａｃｃｕ)，Ｂ－アキュムレータ，データポインタ(ＤＰＴＲ，ＤＰＬ，ＤＰＨ)，レジスタバンクのレジスタ(Ｒ０～Ｒ７)，プログラムステータスワード(ＰＳＷ)，スタックポインタ(ＳＰ)，特殊ファンクションレジスタ(ＳＰＲ)等のような、非ゼロ状態と想定することができるＣＰＵ領域の内容を用いることができよう。

【０００７】

保全性をさらに高めるため、さらに、命令ローディング時に、その命令を実行するために必要なクロックサイクル数を計数するためのカウンタをスタートさせることができる。カウンタは、ハードウエアにより構成されることが好ましい。ロジックが、命令オペレーションコード(ｏｐｃｏｄｅ)から実行に必要なクロックサイクル数を得て、カウンタ値に変換する。カウンタは次いで実行される命令と平行して作動する。実行される命令が定められたクロックサイクル数内で実行されるか否かがチェックされる。命令があらかじめ定められた時間内に実行されなかった場合には、命令をさらに実行することができないように、例えば、クロックの供給が停止される。あるいはリセットを起動し、よって中央処理装置をリセットすることができる。命令の実行終了が早すぎる場合、すなわち命令カウンタの限界値に達していない内に新しいオペレーションコードが既に認識されている場合にも、同じ処置をとることができる。

【０００８】

保全対応レジスタの論理的組合せはハードウエアまたはソフトウエアにより実現できる。２つの連続する命令間でのチェックサム生成は、例えばランダムまたは所定のイベントに基づいて、あるいは定常的に実行することができる。

【０００９】

発明の詳細な説明

以下で図面を参照して本発明をさらに詳細に説明する。

【００１０】

図１は８０５１プロセッサ、すなわち８ビットプロセッサの構造を示す。データは、既知

の暗号化法におけるバスまたはメモリ暗号化により不正操作から保護されるが、コンピュータのコア、すなわち中央処理装置(ＣＰＵ)においては、データは非暗号化形態で存在する。ここで本発明の方法は、ＣＰＵの１つまたはそれ以上のレジスタが不正操作されているか否かを決定する。

【００１１】

図２は、不正操作され得るようなＣＰＵの保全対応領域、すなわち、スタックポインタ(ＳＰ)，アキュムレータ(ＡＣ)，Ｂ－アキュムレータ(ＢＡＣ)，レジスタ(Ｒ０～Ｒ７)，内蔵ＲＡＭの最下位及び最上位領域に対するデータポインタＤＰＬ及びＤＰＨを例として示す。上記のレジスタがチェックサムを生成するために論理的に組み合わされる。図２においては、状況ごとに、２つの８ビットレジスタが排他的論理和(ＸＯＲ)ゲートにより組み合わされる。すなわち、レジスタＲ０とＲ２のＸＯＲをとることにより新しい８ビットパターンが得られ、この８ビットパターンとレジスタＲ１とＲ７のＸＯＲをとることで得られる８ビットパターンとでさらにＸＯＲがとられる。得られる８ビットパターンのＸＯＲをさらにとることにより、チェックサムとしてはたらき、図２で“開始チェックサム”と指定される、８ビットパターンが最終的に得られる。演算にかかる時間及びリソース等に関して特に有利であるＸＯＲをとる代わりに、チェックサムを生成するために別の実施形態を選ぶことも当然可能である。

【００１２】

レジスタの組合せが論理素子によるハードウエアで実行される場合は、レジスタ内容が変わると直ちにチェックサムが変わる。すなわち、ＣＰＵで処理される命令の実行中に、チェックサムは何度も変わることになろう。しかし、本方法を実行するための重要なチェックサムは、命令実行後のチェックサム及び次の命令の実行前のチェックサムだけであり、これはこれらの２つのチェックサム(ある命令の最終チェックサム及びその次の命令の開始チェックサム)がコンパレータで比較されるからである。

【００１３】

比較は以下のようにして実行される。第１の命令の実行終了時に発生するチェックサムが最終チェックサムとしてＣＰＵのメモリに格納される。第１の命令の実行後から次の第２の命令のＣＰＵへのローディングまでの間にＣＰＵの不正操作が行われたか否かを確認するために、上述したように第２の命令のローディングと平行して開始チェックサムが生成される。第１ステップ，ａ)において、開始チェックサムが先に実行された第１の命令からメモリに格納された最終チェックサムとコンパレータにより比較される。ＣＰＵに不正操作が行われていない場合には、開始チェックサムと最終チェックサムとが一致し、比較結果の値はゼロである。コンパレータは信号を出力し、この信号にしたがって、この時点で利用できるチェックサムが第２の命令の実行後の第２ステップ，ｂ)において新しい最終チェックサムとしてメモリに格納される。すなわち、この場合には第２の命令の実行は中断されない。しかし、開始チェックサムと最終チェックサムとの比較により非ゼロの値が得られた場合には、ＣＰＵの不正操作が行われたと推定しなければならない。したがって、コンパレータの出力信号は第２ステップ，ｂ)に進ませず、図２に示される例では命令処理をアボートさせるエラーメッセージであるｃ)を出させる。例えば、プロセッサを停止させることができ、保全センサが起動され、あるいはスマートカードの場合には、スマートカードが端末により用いられずにおかれる。

【００１４】

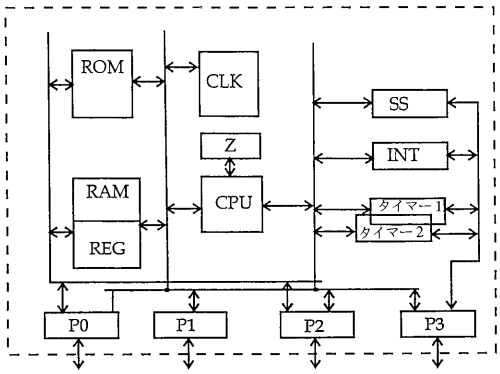
上述の保全機構はまた、一方ではある命令の実行終了時に決定され、また一方では次の命令の実行開始時に決定されて、比較されるチェックサムにより、ソフトウエアで精確に実現できる。例えば、対応プログラムをプロセッサのＲＯＭまたはＥＰＲＯＭに格納し、最終チェックサムをプロセッサのビットアドレス指定可能なＲＡＭに格納できる。

【００１５】

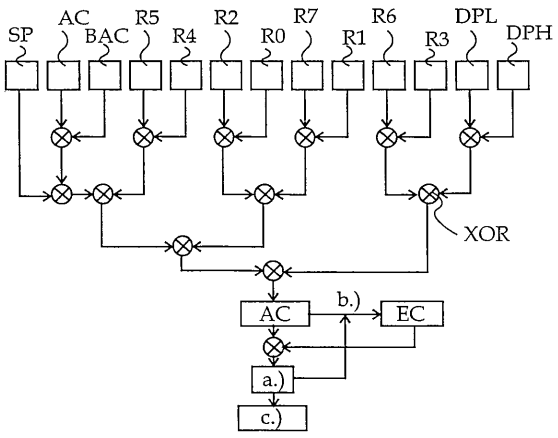
上述の方法はそれぞれの命令実行の前に必ず行われる必要はない。本発明の一実施形態においては、ランダムまたは所定のイベントに基づいて本方法が実行される。第１の実施形態にしたがえば、本方法は時間に基づく態様で起動させることができる。

【００１６】
別の実施形態にしたがえば、ＣＰＵの１つまたはそれ以上のレジスタの内容があらかじめ定められたパターンと一致することにより、本方法を起動させることができる。
【００１７】
本発明のまた別の実施形態においては、状況ごとに、あらかじめ定められた数の命令の処理後に本方法が起動される。
【００１８】
好ましい実施形態は、実行後にチェックサムが最終チェックサムとしてメモリに格納された命令と実行開始時に開始チェックサムが生成される次の命令との間に、比較的長い、定められた時間がある場合にのみ、本方法が起動される実施形態である。これにより、貴重なコンピュータ能力を多くの命令ごとにプログラムを実行することで浪費することが回避される。ＣＰＵの不正操作が、特にスマートカードにおいて、プログラム実行中には行われず、スマートカードがスマートカード端末から取り出されるときに行われるとすれば、最後に述べた手段によりＣＰＵの不正操作を、たとえ不正操作が行われたとしても、確実に検出できる。
【図面の簡単な説明】
【図１】　８０５１プロセッサを例としてマイクロコントローラの構造を示す
【図２】　中央処理装置のいくつかの領域を組み合わせるためのロジックを示す

【図１】



【図２】



フロントページの続き

(72)発明者　バルディッシュヴァイラー，ミヒャエル
ドイツ連邦共和国　Ｄ－８１９２９　ミュンヘン　フリードリッヒ－エッカルトシュトラーセ　６０

審査官　深沢　正志

(56)参考文献　特開平０６－０８３７１９（ＪＰ，Ａ）
特許第３１２４２７８（ＪＰ，Ｂ２）

(58)調査した分野(Int.Cl.，ＤＢ名)
G06F　21/00 - 21/24
G06F　 9/30