(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2020年12月30日(30.12.2020)

WIPO | PCT

(10) 国際公開番号
WO 2020/261438 A1

(51) 国際特許分類:
G06F 21/53 (2013.01)

(21) 国際出願番号： PCT/JP2019/025414

(22) 国際出願日： 2019年6月26日(26.06.2019)

(25) 国際出願の言語： 日本語

(26) 国際公開の言語： 日本語

(71) 出願人: 日本電気株式会社 (**NEC CORPORATION**) [JP/JP]; 〒1088001 東京都港区芝五丁目７番１号 Tokyo (JP).

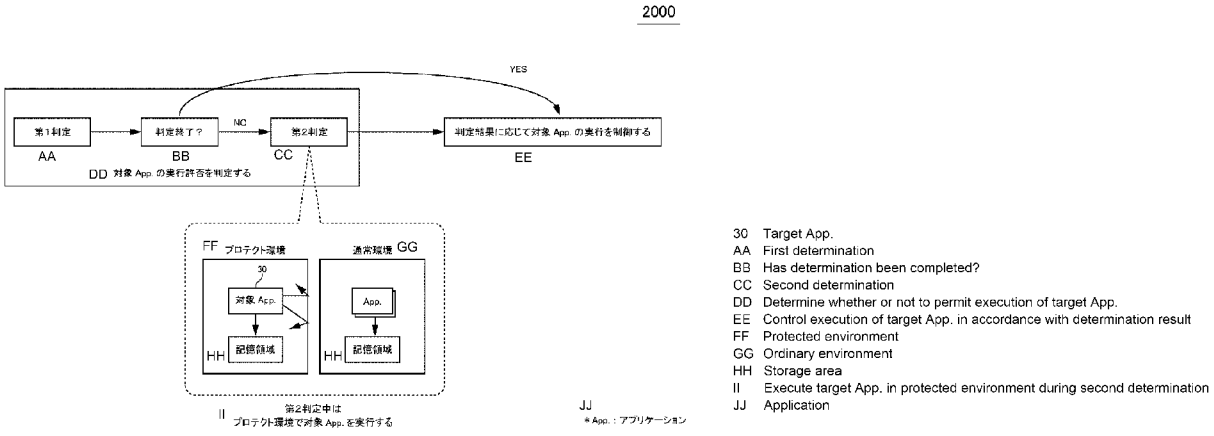
(72) 発明者: 榮 純明 (**SAKAE Yoshiaki**); 〒1088001 東京都港区芝五丁目７番１号 日本電気株式会社内 Tokyo (JP). 磯山 和彦 (**ISOYAMA Kazuhiko**); 〒1088001 東京都港区芝五丁目７番１号 日本電気株式会社内 Tokyo (JP). 小梨 貴史 (**KONASHI Takashi**); 〒1088001 東京都港区芝五丁目７番１号 日本電気株式会社内 Tokyo (JP). 西岡 淳 (**NISHIOKA Jun**); 〒1088001 東京都港区芝五丁目７番１号 日本電気株式会社内 Tokyo (JP).

(74) 代理人: 速水 進治 (**HAYAMI Shinji**); 〒1410031 東京都品川区西五反田７丁目９番２号 ＫＤＸ五反田ビル９階 Tokyo (JP).

(81) 指定国(表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,

(54) **Title:** EXECUTION CONTROL SYSTEM, EXECUTION CONTROL METHOD, AND PROGRAM

(54) 発明の名称： 実行制御システム、実行制御方法、及びプログラム



(57) **Abstract:** An execution control system (2000) determines whether or not to permit execution of a target application (30). The determination includes a first determination and a second determination. The second determination is made if the first determination has not been able to determine whether or not to permit execution of the target application (30). During the second determination after the first determination has been made, the execution control system (2000) executes the target application (30) in a protected environment.

(57) 要約：実行制御システム（２０００）は、対象アプリケーション（３０）の実行許否を判定する。上記判定は、第１判定及び第２判定を含む。第２判定は、第１判定で対象アプリケーション(３０)の実行許否を決定できなかった場合に行われる。実行制御システム（２０００）は、第１判定が終了した後であって第２判定が行われている間、対象アプリケーション（３０）をプロテクト環境で実行する。

WO 2020/261438 A1

［続葉有］

SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) 指定国(表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

添付公開書類:

一 国際調査報告 (条約第21条(3))

明　細　書

発明の名称 ： 実行制御システム、実行制御方法、及びプログラム

技術分野

[0001] 本発明はソフトウエアの動作の制御に関する。

背景技術

[0002] ソフトウエアの動作を制御するシステムが開発されている。例えば特許文献１には、ソフトウエアのインストールをサンドボックス環境で行い、サンドボックス環境内で行われたインストールの間に生成されたアクションに基づいて、そのソフトウエアのインストールが望ましいものであるかどうかを判定する技術が開示されている。

先行技術文献

特許文献

[0003] 特許文献1：特開２０１７－０２１７７３号公報

発明の概要

発明が解決しようとする課題

[0004] 本発明者は、ソフトウエアの動作を制御する新たな技術を見出した。本発明は、上述の課題に鑑みてなされたものであり、その目的の一つは、ソフトウエアの動作を制御する新たな技術を提供することである。

課題を解決するための手段

[0005] 本発明の実行制御システムは、対象ソフトウエアの動作の許否を判定する判定部を有する。判定は、第１判定と、第１判定で対象ソフトウエアの動作の許否を決定できなかった場合に行わる第２判定とを含む。

本発明の実行制御システムは、さらに、第１判定が終わった後であって第２判定が行われている間に、対象ソフトウエアをプロテクト環境で動作させる制御部を有する

[0006] 本発明の実行制御方法は、コンピュータによって実行される。

当該実行制御方法は、対象ソフトウエアの動作の許否を判定する判定ステ

ップを有する。判定は、第１判定と、第１判定で対象ソフトウエアの動作の許否を決定できなかった場合に行わる第２判定とを含む。

当該実行制御方法は、さらに、第１判定が終わった後であって第２判定が行われている間に、対象ソフトウエアをプロテクト環境で動作させる制御ステップを有する。

[0007] 本発明の制御方法はコンピュータによって実行される。制御方法は、１）アプリケーションの異常を検知する処理が行われたアプリケーションについて、そのアプリケーションの導入に関する導入実績情報を取得する取得ステップと、２）取得した導入実績情報を用いて、アプリケーションの評価を行う評価ステップと、を有する。

[0008] 本発明のプログラムは、本発明の実行制御方法の各ステップをコンピュータに実行させる。

発明の効果

[0009] 本発明によれば、ソフトウエアの動作を制御する新たな技術が提供される。

図面の簡単な説明

[0010] 上述した目的、およびその他の目的、特徴および利点は、以下に述べる好適な実施の形態、およびそれに付随する以下の図面によってさらに明らかになる。

[図1]本実施形態の実行制御システムの動作の概要を例示する図である。

[図2]実施形態１の実行制御システムの構成を例示する図である。

[図3]実行制御システムを実現するための計算機を例示する図である。

[図4]実施形態１の実行制御システムによって実行される処理の流れを例示するフローチャートである。

[図5]実行制御システムの装置構成を例示する第１の図である。

[図6]実行制御システムの装置構成を例示する第２の図である。

[図7]導入実績情報をテーブル形式で例示する図である。

[図8]導入基準情報をテーブル形式で例示する図である。

[図9]導入基準情報を管理する構成を例示する図である。

[図10]出力部を有する実行制御システムの機能構成を例示するブロック図である。

発明を実施するための形態

[0011] 以下、本発明の実施の形態について、図面を用いて説明する。尚、すべての図面において、同様な構成要素には同様の符号を付し、適宜説明を省略する。また、特に説明する場合を除き、各ブロック図において、各ブロックは、ハードウエア単位の構成ではなく、機能単位の構成を表している。

[0012] ＜概要＞

図１は、本実施形態の実行制御システム２０００の動作の概要を例示する図である。図１は、実行制御システム２０００の動作についての理解を容易にするための概念的な説明を表す図であり、実行制御システム２０００の動作を具体的に限定するものではない。

[0013] 実行制御システム２０００は、ソフトウエアの動作の許否の判定、及びソフトウエアの動作の制御を行う。以下、本実施形態では、「ソフトウエアの動作」として、「アプリケーションの実行」を扱う。「ソフトウエアの動作」のその他の例である「共有ライブラリのロード」については、後述する変形例で説明する。

[0014] アプリケーションの実行許否の判定、及びアプケーションの実行の制御を行う。ここで、実行制御システム２０００による判定の対象となっているアプリケーションを、対象アプリケーション３０と呼ぶ。例えば、ユーザや他のアプリケーションによって起動されたアプリケーションが、対象アプリケーション３０として扱われる。すなわち、或るアプリケーションが起動されたら、そのアプリケーションの実行が開始される前に、そのアプリケーションの実行許否が実行制御システム２０００によって判定される。ただし、対象アプリケーション３０の実行許否を判定するタイミングは、対象アプリケーション３０が起動されたタイミングに限定されない。

[0015] 例えば実行許否の判定は、対象アプリケーション３０が正常なアプリケー

ションであるかどうかを判定することで行われる。このような判定を行うことで、異常なアプリケーション（例えば、マルウエアである蓋然性が高いアプリケーション）を実行することによって被害が生じることを防ぐことができる。

[0016] 実行制御システム２０００による対象アプリケーション３０の実行許否の判定は、第１判定及び第２判定を含む。第２判定は、第１判定では対象アプリケーション３０の実行許否を決定できなかった場合に実行される。

[0017] 実行制御システム２０００において、対象アプリケーション３０は、少なくとも２種類の実行環境で動作しうる。この実行環境には、プロテクト環境と通常環境が含まれる。プロテクト環境で実行される対象アプリケーション３０の動作は、通常環境で実行される場合よりも制限される。制限の例としては、データの書き込み（記憶領域への書き込みや外部への送信など）が挙げられる。データの書き込みについて制限する場合、例えば、通常環境で実行される対象アプリケーション３０がデータを書き込める記憶領域の少なくとも一部について、プロテクト環境で実行される対象アプリケーション３０はデータを書き込めないようにする。

[0018] 実行制御システム２０００は、第１判定が完了するまでは、対象アプリケーション３０を実行しない。そして、第１判定では対象アプリケーション３０の実行許否を決定できず、第２判定が行われる場合、実行制御システム２０００は、対象アプリケーション３０の実行許否の判定が終わるまでの間、対象アプリケーションをプロテクト環境で動作させる。

[0019] ＜作用効果の一例＞

対象アプリケーション３０のユーザは、対象アプリケーション３０をより早く利用したいと考えられる。そこで、対象アプリケーション３０の実行許否を判定する間、対象アプリケーション３０をプロテクト環境で実行することが考えられる。このようにすることで、実行してよいか否かがまだ定かでない対象アプリケーションを早く実行できるようにしつつ、対象アプリケーションによって他のアプリケーション等がコンプロマイズされることを防ぐ

ことができる。すなわち、アプリケーションのより早い実行開始を望むユーザの要望に応えつつ、そのアプリケーションを実行させることによって問題が生じることを防ぐことができる。

[0020] しかしながら、例えば以下の理由から、実行許否の判定中である対象アプリケーション３０について、必ずしもプロテクト環境で実行することが最良であるとは言えない。プロテクト環境で実行されている対象アプリケーション３０について実行が許可された場合、その後は、対象アプリケーション３０を通常環境で実行する必要がある。そのためには、後述する様に、対象アプリケーション３０の実行環境をプロテクト環境から通常環境に切り替えたり、対象アプリケーション３０の実行を一旦終了して、再度通常環境で対象アプリケーション３０を起動したりする必要がある。これに対し、起動が中断されている状態の対象アプリケーション３０について実行が許可された場合には、中断されている起動を再開すればよい。そのため、対象アプリケーション３０の実行が許可された後に要する処理が比較的シンプルなものとなる。そのため、対象アプリケーション３０の実行許否の判定が完了した後の処理を考慮すると、対象アプリケーション３０の実行許否の判定が短い時間で終わるのであれば、対象アプリケーション３０をプロテクト環境で実行するよりも、対象アプリケーション３０の起動を一時中断しておく方が好ましいと考えられる。

[0021] そこで、実行制御システム２０００では、対象アプリケーション３０の実行許否の判定は、第１判定及び第２判定が含まれており、第１判定では実行許否を決定できず、第２判定が必要となった場合に、対象アプリケーション３０がプロテクト環境で実行される。このようにすることで、対象アプリケーション３０の実行が許可された後に要する処理についても考慮した上で、対象アプリケーション３０のより早い実行開始を望むユーザの要望に応えること、及びその対象アプリケーション３０を実行させることによって問題が生じることを防ぐことができる。

[0022] 以下、本実施形態の実行制御システム２０００についてさらに詳細に説明

する。

[0023] ＜実行制御システム２０００の機能構成の例＞

図２は、実施形態１の実行制御システム２０００の構成を例示する図である。実行制御システム２０００は、判定部２０２０及び制御部２０４０を有する。判定部２０２０は、対象アプリケーション３０の実行許否を判定する。実行許否の判定は、第１判定及び第２判定を含む。制御部２０４０は、第１判定が終わった後、第２判定が行われている間に、対象アプリケーション３０をプロテクト環境で実行する。

[0024] ＜実行制御システム２０００のハードウエア構成＞

実行制御システム２０００の各機能構成部は、各機能構成部を実現するハードウエア（例：ハードワイヤードされた電子回路など）で実現されてもよいし、ハードウエアとソフトウエアとの組み合わせ（例：電子回路とそれを制御するプログラムの組み合わせなど）で実現されてもよい。以下、実行制御システム２０００の各機能構成部がハードウエアとソフトウエアとの組み合わせで実現される場合について、さらに説明する。

[0025] 例えば実行制御システム２０００は、１つの計算機で実現される。図３は、実行制御システム２０００を実現するための計算機１０００を例示する図である。計算機１０００は任意の計算機である。例えば計算機１０００は、Personal Computer（PC）、サーバマシン、タブレット端末、又はスマートフォンなどである。計算機１０００は、実行制御システム２０００を実現するために設計された専用の計算機であってもよいし、汎用の計算機であってもよい。

[0026] 計算機１０００は、バス１０２０、プロセッサ１０４０、メモリ１０６０、ストレージデバイス１０８０、入出力インタフェース１１００、及びネットワークインタフェース１１２０を有する。バス１０２０は、プロセッサ１０４０、メモリ１０６０、ストレージデバイス１０８０、入出力インタフェース１１００、及びネットワークインタフェース１１２０が、相互にデータを送受信するためのデータ伝送路である。ただし、プロセッサ１０４０など

を互いに接続する方法は、バス接続に限定されない。プロセッサ１０４０は、CPU (Central Processing Unit)、GPU (Graphics Processing Unit)、又は FPGA (Field-Programmable Gate Array) などのプロセッサである。メモリ１０６０は、RAM (Random Access Memory) などを用いて実現される主記憶装置である。ストレージデバイス１０８０は、ハードディスクドライブ、SSD (Solid State Drive)、メモリカード、又は ROM (Read Only Memory) などを用いて実現される補助記憶装置である。ただし、ストレージデバイス１０８０は、RAM など、主記憶装置を構成するハードウエアと同様のハードウエアで構成されてもよい。

[0027] 入出力インタフェース１１００は、計算機１０００と入出力デバイスとを接続するためのインタフェースである。ネットワークインタフェース１１２０は、計算機１０００を通信網に接続するためのインタフェースである。この通信網は、例えば LAN (Local Area Network) や WAN (Wide Area Network) である。ネットワークインタフェース１１２０が通信網に接続する方法は、無線接続であってもよいし、有線接続であってもよい。

[0028] ストレージデバイス１０８０は、実行制御システム２０００の機能構成部を実現するプログラムモジュールを記憶している。プロセッサ１０４０は、これら各プログラムモジュールをメモリ１０６０に読み出して実行することで、各プログラムモジュールに対応する機能を実現する。

[0029] 実行制御システム２０００は、２つ以上の計算機で実現されてもよい。この場合における各計算機も、例えば、図３に示したハードウエア構成を持つ。

[0030] ＜処理の流れ＞

図４は、実施形態１の実行制御システム２０００によって実行される処理の流れを例示するフローチャートである。判定部２０２０は、対象アプリケーション３０の実行許否について第１判定を行う（Ｓ１０２）。第１判定で対象アプリケーション３０の実行許否を決定できた場合（Ｓ１０４：完了）、制御部２０４０は、判定結果に基づいて対象アプリケーション３０の実行

を制御する（Ｓ１１０）。一方、第１判定で対象アプリケーション３０の実行許否を決定できなかった場合（Ｓ１０４：未完了）、判定部２０２０は、対象アプリケーション３０の実行許否について第２判定を行う（Ｓ１０６）。また、第２判定が行われている間、制御部２０４０は、対象アプリケーション３０をプロテクト環境で実行する（Ｓ１０８）。

[0031] 第２判定が完了したら、制御部２０４０は、判定結果に基づいて対象アプリケーション３０の実行を制御する（Ｓ１１０）。

[0032] 実行制御システム２０００による処理が開始されるタイミング、例えば、或るアプリケーションがユーザや他のアプリケーションによって起動されたタイミングである。実行制御システム２０００は、起動されたアプリケーションを対象アプリケーション３０として扱って、実行許否の判定を行う。

[0033] ただし、実行制御システム２０００がアプリケーションの実行許否を判定するタイミングは、そのアプリケーションが起動する前であってもよい。例えば、実行制御システム２０００は、新たなアプリケーションが第１装置１０に導入されたら、そのアプリケーションを対象アプリケーション３０として扱って、実行許否の判定を行う。このケースでは、対象アプリケーション３０を起動させた際、既にその対象アプリケーション３０の実行許否の判定が完了していることもありうる。そこで例えば、対象アプリケーション３０が起動されたとき、その対象アプリケーション３０について既に実行許否の判定が完了していれば、実行制御システム２０００は、既に完了している判定の結果に基づいて、対象アプリケーション３０の実行を制御する。また、対象アプリケーション３０が起動されたとき、その対象アプリケーション３０について第１判定が行われていたら、実行制御システム２０００は、第１判定が終わるまで対象アプリケーション３０の起動を中断する。また、対象アプリケーション３０が起動されたとき、その対象アプリケーション３０について第２判定が行われていたら、実行制御システム２０００は、対象アプリケーション３０をプロテクト環境で実行する。

[0034] その他にも例えば、対象アプリケーション３０の実行許否の判定は、定期

的なタイミング（例えば、一日に１回）で行われてもよい。この場合、実行制御システム２０００は、定期的なタイミングで、新たに第１装置１０に導入された各アプリケーション（実行許否の判定がまだ行われていない各アプリケーション）について、実行許否の判定を行う。ただし、このような定期的なタイミングが到来する前に起動された対象アプリケーション３０については、その起動のタイミングで、対象アプリケーション３０の実行許否を判定することが好適である。

[0035] ＜実行制御システム２０００の装置構成の具体例＞

実行制御システム２０００は、様々な装置構成で実現しうる。ここでは、その具体例をいくつか例示する。

[0036] ＜＜構成例１＞＞

図５は、実行制御システム２０００の装置構成を例示する第１の図である。この例において、実行制御システム２０００は、第１装置１０及び第２装置２０で構成される。第１装置１０は、対象アプリケーション３０が実行される装置である。第２装置２０は、対象アプリケーション３０の実行許否を判定する装置である。

[0037] 第１装置１０は、アプリケーションが起動されたことを検知する機能を有する。第１装置１０においてアプリケーションの起動が検知されたら、そのアプリケーションが対象アプリケーション３０として扱われる。第１装置１０は、第２装置２０に対し、対象アプリケーション３０の実行許否を問い合わせるリクエストを送信する。このリクエストには、対象アプリケーション３０の識別情報が含まれている。

[0038] また、第２装置２０には、制御部２０４０が含まれている。制御部２０４０は、対象アプリケーション３０をプロテクト環境で実行する。

[0039] 第２装置２０には、判定部２０２０が設けられている。判定部２０２０は、第１装置１０から上記リクエストを受信し、リクエストに示されている識別情報で特定される対象アプリケーション３０について、実行許否を判定する。判定部２０２０は、第１判定の判定結果を示す通知を第１装置１０に送

信する。この通知には、例えば、「対象アプリケーション３０の識別情報、判定結果」の組み合わせが示されている。

[0040] 上記通知が示す判定結果が許可または不許可である場合、制御部２０４０は、その判定結果に基づいて、対象アプリケーション３０の実行を制御する。これにより、実行制御システム２０００による一連の処理は終了する。

[0041] 一方、上記通知が示す判定結果に、実行許否を決定できなかったこと（すなわち、第２判定が行われること）が示されている場合、制御部２０４０は、対象アプリケーション３０をプロテクト環境で実行する。その後、判定部２０２０は、第２判定の判定結果を示す通知を制御部２０４０に送信する。制御部２０４０は、この通知が示す判定結果に基づいて、対象アプリケーション３０の実行を制御する。

[0042] ＜＜構成例２＞＞

図６は、実行制御システム２０００の装置構成を例示する第２の図である。この例では、判定部２０２０と制御部２０４０の双方が、第１装置１０の中に設けられている。すなわち、対象アプリケーション３０が実行される装置の中で、対象アプリケーション３０の実行許否の判定、及び対象アプリケーション３０の実行の制御が行われる。

[0043] ＜実行許否の判定：Ｓ１０２、Ｓ１０６＞

判定部２０２０は、対象アプリケーション３０の実行許否を判定する（Ｓ１０２、Ｓ１０６）。対象アプリケーション３０の実行許否の判定は、任意の基準を利用して行うことができる。対象アプリケーション３０の実行許否の判定に利用する具体的な基準については後述する。

[0044] 判定部２０２０が行う判定処理は、少なくとも、第１判定及び第２判定という２段階の判定を含む。第２判定は、第１判定で実行許否の決定をできなかった場合（第１判定では対象アプリケーション３０の実行許否の判定を完了できなかった場合）に実行される。そのため、第１判定の結果は、１）対象アプリケーション３０の実行を許可する、２）対象アプリケーション３０の実行を許可しない、及び３）第２判定に進むといういずれかの結果となる

。

[0045] ここで、第１判定の前や、第２判定の後に、さらに別の判定が行われてもよい。すなわち、実行制御システム２０００では、１）判定部２０２０による判定には２以上の複数の段階の判定が含まれ、２）特定の判定（第１判定）が完了するまでは対象アプリケーション３０が実行されず、３）特定の判定が完了しても実行可否を決定できなかったら、次の判定（第２判定）に進むと共に、対象アプリケーション３０がプロテクト環境で実行される。なお、各段階の判定は、１）対象アプリケーション３０の実行を許可する、２）対象アプリケーション３０の実行を許可しない、及び３）次の判定に進むといういずれかの結果となる。

[0046] なお、第２判定よりも後に別の判定が行われる場合、その間も、対象アプリケーション３０はプロテクト環境で実行される。すなわち、第２判定が始まって以降、対象アプリケーション３０の実行許否の判定が完了するまでの間、対象アプリケーション３０はプロテクト環境で実行される。

[0047] 判定部２０２０による多段階の判定は、例えば、段階が進むほど判定に時間を要するものになるように構成される。言い換えれば、所要時間が比較的短い判定を優先的に（より先に）行い、そのような短い時間で終わる判定では対象アプリケーション３０の実行許否を判定することが難しい場合に限って、所要時間が比較的長い判定処理を行うようにする。こうすることで、対象アプリケーション３０の実行許否の判定に要する時間をできる限り短くすることができる。

[0048] 例えば、判定部２０２０による各段階の判定は、それぞれ異なる基準を用いて行われる。例えば、判定の段階が上がるほど、その判定に利用する情報の取得に時間がかかるという構成が考えられる。ここで、第１判定で用いられる基準を示す情報を第１基準情報と呼び、第２判定で用いられる基準を示す情報を第２基準情報と呼ぶ。例えば、第１基準情報は既に記憶装置に格納されている一方で、第２基準情報はその場で生成されるといったケースが考えられる。

[0049] ＜＜判定の再利用について＞＞

一度実行許否の判定が行われた対象アプリケーション３０については、その判定結果を保存しておき、再度の判定は不要とすることが好適である。具体的には、判定部２０２０によって対象アプリケーション３０の実行許否について判定が行われたら、「判定が行われたアプリケーションの識別情報、判定結果」という組み合わせを、所定の記憶装置に格納する。以下、上述の組み合わせで構成される情報を、判定結果情報と呼ぶ。また、判定結果情報が格納される記憶装置を、判定結果情報記憶装置と呼ぶ。

[0050] 判定部２０２０は、対象アプリケーション３０の実行許否の判定をする際、まずは、対象アプリケーション３０の識別情報で、判定結果情報記憶装置に格納されている情報を検索する。対象アプリケーション３０の識別情報を示す判定結果情報が格納されていたら、判定部２０２０は、その判定結果情報に示されている判定結果を利用する。一方、対象アプリケーション３０の識別情報を示す判定結果情報が格納されていなかったら、判定部２０２０は、対象アプリケーション３０について、実行許否の判定を行う。

[0051] ここで、実行許否の判定の基準が更新されるなどの理由により、一度実行許否の判定が行われた対象アプリケーション３０についても、再度判定を行う必要が生じうる。そこで、実行許否の判定の基準が更新されたら、判定部２０２０は、その更新以前に判定結果情報記憶装置に格納された判定結果情報を利用しないようにすることが好適である。そのために、例えば、実行許否の判定の基準が更新されたら、その更新以前に生成された判定結果情報については、判定結果情報記憶装置から削除する。

[0052] ＜プロテクト環境における実行：Ｓ１０８＞

制御部２０４０は、第２判定が行われている間、対象アプリケーション３０をプロテクト環境で実行する（Ｓ１０８）。ここでいうプロテクト環境とは、通常環境と比較して、対象アプリケーション３０の動作の少なくとも一部がより制限されており、対象アプリケーション３０の動作が他のアプリケーションに対して影響を与えにくい環境である。このような環境は、サンド

ボックス環境などとも呼ぶことができる。

[0053] プロテクト環境において対象アプリケーション３０に与える制限には、任意のものを採用できる。例えばプロテクト環境では、対象アプリケーション３０によるデータの読み書きやプロセスの起動などが制限される。例えばデータの書き込みを制限する場合、プロテクト環境で動作する対象アプリケーション３０は、他のアプリケーションからアクセスできない記憶領域に対してデータを書き込むように制御される。例えば、他のアプリケーションと共有されているデータ（ストレージデバイスに格納されているファイル、メモリにマップされたファイル、レジストリ、又は共有メモリ上のデータなど）に対して対象アプリケーション３０が変更を加える場合には、他のアプリケーションからアクセスできない記憶領域にそのデータのコピーを作成し、そのコピーに対して変更が加えられるようにする。このようにすることで、対象アプリケーション３０によって行われたデータの変更が、他のアプリケーションからは認識できなくなる。そのため、対象アプリケーション３０によって書き込まれたデータが他のアプリケーションに悪影響を与えることを防ぐことができる。

[0054] なお、プロテクト環境で動作している対象アプリケーション３０と他のアプリケーション（通常環境で動作しているものであってもよいし、別のプロテクト環境で動作しているアプリケーションであってもよい）が同一のデータに対して書き込みを行うことも考えられる。このような場合、何らかの基準で書き込みの競合を解決する必要がある。例えば、制御部２０４０は、書き込み時点が最も後の書き込みを優先して適用し（有効にし）、その他の書き込みは適用されないようにする（無効にする）。この場合、書き込みが適用されなかったアプリケーションに関し、そのアプリケーションによる書き込みが適用されなかったことの通知を行うことが好適である。

[0055] なお、プロテクト環境で動作している対象アプリケーション３０と他のアプリケーションが、同一のデータの互いに重複しない箇所に対して書き込みを行ったとする。この場合、制御部２０４０は、双方の書き込みを適用する

ようにしてもよい。

[0056] 競合の解決方法の他の例として、制御部２０４０は、対象アプリケーション３０の動作環境がプロテクト環境から通常環境に移行するタイミングで、データへの書き込みに競合があったことをユーザに通知し、どのアプリケーションによる書き込みを適用するのかをユーザに選択させてもよい。この場合、制御部２０４０は、ユーザによって選択されたアプリケーションによる書き込みを適用し、その他のアプリケーションによる書き込みは適用しないようにする。

[0057] その他にも例えば、制御部２０４０は、通常環境で動作しているアプリケーションによる書き込みを優先して適用するようにしてもよい。この場合、プロテクト環境で動作している対象アプリケーション３０によってデータに対する書き込みが行われたら、制御部２０４０は、そのデータのコピーを作成し、そのコピーに対して書き込みを適用する。そして、制御部２０４０は、任意のタイミング（例えば、対象アプリケーション３０がプロテクト環境から通常環境へ移行したタイミング）で、書き込みの競合によりデータのコピーに対して書き込みが行われたこと、及びそのコピーの保存場所などをユーザに通知する。

[0058] その他にも例えば、制御部２０４０は、プロテクト環境で動作している対象アプリケーション３０が或るデータに対して書き込みを行ったら、それ以降、そのデータに対する他のアプリケーションによる書き込みを禁止してもよい。この場合、制御部２０４０は、書き込みの競合により書き込みが禁止されていることをユーザに通知することが好適である。

[0059] データの読み込みを制限する場合、例えば、プロテクト環境で動作する対象アプリケーション３０は、特定の記憶領域に対するリードアクセスが制限される。言い換えれば、リードアクセスできる記憶領域が制限される。例えば特定の領域は、機密情報が記憶されている記憶領域や、OS やミドルウエアが利用するシステム領域などである。このようにすることで、悪意ある対象アプリケーション３０によって機密情報などの重要なデータが盗まれたり、

対象アプリケーション３０によって第１装置１０がコンプロマイズされたりすることを防ぐことができる。ただし、データの書き込みを制限しておけば、たとえ悪意ある対象アプリケーション３０にデータを読まれたとしても、その対象アプリケーション３０による外部へのデータの流出（書き込み）は防ぐことができる。

[0060] なお、データの読み書きが制限されるのは、記憶領域に限定されない。例えば、ネットワークに対するデータの読み書き（外部の装置との通信）が制限されてもよい。これにより、ネットワークを介したデータの漏洩などを防ぐことができる。

[0061] 他のアプリケーションの起動を制限する場合、プロテクト環境で動作する対象アプリケーション３０は、全て又は一部のアプリケーションを起動することができないように制限される。後者の場合、言い換えれば、起動できるアプリケーションが一部のものに限定される。ここで、マルウエアの中には、他のアプリケーション（例えばシェル）を利用して悪意ある動作を行うものもある。対象アプリケーション３０による他のアプリケーションの起動を制限することで、このようなマルウエアによる悪意ある動作を防ぐことができる。

[0062] また、対象アプリケーション３０が他のアプリケーションを起動する場合、当該他のアプリケーションについてもプロテクト環境内で実行されるようにしてもよい。この場合、対象アプリケーション３０と当該他のアプリケーションについては、互いにデータを共有できるようにすることが好適である。

[0063] その他にも例えば、プロテクト環境では、対象アプリケーション３０が利用できる計算機リソースの量が制限されてもよい。計算機リソースとしては、例えば、プロセッサ資源、メモリ資源、ディスク帯域、及びネットワーク帯域などが挙げられる。このように使用できる計算機リソースの量が制限された環境で対象アプリケーション３０を実行することで、例えば、対象アプリケーション３０が過度に計算機リソースを使用することによる他のアプリ

ケーションへの悪影響を防ぐことができる。

[0064] ここで、上述した種々の制御を実現する具体的な方法については、既存の方法を採用することができる。

[0065] ＜実行が許可された場合の制御＞

判定部２０２０によって対象アプリケーション３０の実行が許可された場合、制御部２０４０は、対象アプリケーション３０の実行環境を通常の環境に変更する。例えば、プロテクト環境において、対象アプリケーション３０によるデータの読み書きやアプリケーションの起動、又は対象アプリケーション３０によって使用可能なリソースの量などが、通常の環境の場合よりも制限されていたとする。この場合、対象アプリケーション３０に対する制限が、通常の環境と同様の制限に変更される。

[0066] ここで、プロテクト環境において対象アプリケーション３０によるデータの書き込みが制限される場合、プロテクト環境で実行されていた時に対象アプリケーション３０が書き込んだデータは、通常の環境に移行した後でも利用できることが好適である。そこで例えば、制御部２０４０は、プロテクト環境において対象アプリケーション３０が書き込んだデータを、通常の環境で動作するアプリケーションからもアクセスできる記憶領域に移動又はコピーする。

[0067] 例えば、プロテクト環境で実行されている対象アプリケーション３０が、他のアプリケーションと共有されているデータに対して変更を加えたため、他のアプリケーションからアクセスできない記憶領域にそのデータのコピーを作成し、そのデータに対して変更が加えられたとする。この場合、制御部２０４０は、このコピーに対して加えられた内容を、オリジナルのデータにも反映するようにする。

[0068] ＜実行が許可されなかった場合の制御＞

判定部２０２０によって対象アプリケーション３０の実行が許可されなかった場合、例えば制御部２０４０は、対象アプリケーション３０の実行を終了する。こうすることで、セキュリティ上の脅威である可能性があるアプリ

ケーションなど、実行することが好ましくない対象アプリケーション３０が実行され続けることを防ぐことができる。

[0069] 対象アプリケーション３０の実行を終了する際、制御部２０４０は、プロテクト環境で実行されていた対象アプリケーション３０によって記憶領域に書き込まれたデータを破棄してもよい。なお、制御部２０４０は、対象アプリケーション３０によって書き込まれたデータは、対象アプリケーション３０による活動の記録を表す情報として記録されてもよい。ただしこの場合、記憶領域の最終的な内容だけではなく、対象アプリケーション３０による一連のデータの書き込みの記録を残しておくことが好適である。

[0070] その他にも例えば、制御部２０４０は、判定部２０２０によって対象アプリケーション３０の実行が許可されなかった場合に、対象アプリケーション３０をプロテクト環境内で実行させ続けてもよい。こうすることで、対象アプリケーション３０が他のアプリケーション等に悪影響を及ぼすことを防ぎつつ、ユーザが対象アプリケーション３０の実行を継続することができるようになる。

[0071] ＜実行許否を判定する基準について＞

判定部２０２０は、種々の基準で、対象アプリケーション３０の実行許否を判定する。例えば、対象アプリケーション３０の実行許否の判定には、対象アプリケーション３０の導入に関連する基準を利用することができる。以下、詳細に説明する。

[0072] アプリケーションは、そのアプリケーションを実行する装置に導入される。ここでいう導入とは、装置上で対象アプリケーション３０を実行可能な状態にすることである。例えば対象アプリケーション３０は、第１装置１０に導入されている。

[0073] 対象アプリケーション３０が第１装置１０の外部から取得される場合、第１装置１０に対する対象アプリケーション３０の導入には、対象アプリケーション３０を取得する処理も含まれる。そのため、例えば第１装置１０に対する対象アプリケーション３０の導入は、１）対象アプリケーション３０を

入手する処理、２）入手した対象アプリケーション３０をファイルシステム上に配置する処理、及び３）対象アプリケーション３０に関する設定を行う処理などを含む。

[0074] 対象アプリケーション３０の入手は、例えば、対象アプリケーション３０が提供されているサーバから対象アプリケーション３０をダウンロードしたり、対象アプリケーション３０が記憶されている記憶装置から対象アプリケーション３０を読み出したりする処理である。対象アプリケーション３０をファイルシステム上に配置する処理は、例えば、対象アプリケーション３０の実行ファイルや設定ファイルを、所定のディレクトリに格納する処理である。対象アプリケーション３０に関する設定を行う処理は、例えば、レジストリや設定ファイルなどに対し、対象アプリケーション３０の実行に必要な設定データを書き込む処理である。

[0075] なお、対象アプリケーション３０の実行ファイルを所定のディレクトリに配置する処理や、対象アプリケーション３０に関する設定を行う処理は、対象アプリケーション３０のインストーラを実行することで自動で行われる場合もあれば、対象アプリケーション３０の導入作業を行うユーザによって手動で行われる場合もある。また、対象アプリケーション３０を入手する処理も自動で行われうる。例えば、或るアプリケーションＸが別のアプリケーションＹを必要としている場合に、アプリケーションＸのインストーラがアプリケーションＹの入手を自動で行うようなケースがある。

[0076] 対象アプリケーション３０の導入に関する基準で対象アプリケーション３０の実行の許否を判定する場合、判定部２０２０は、第１装置１０に対する対象アプリケーション３０の導入に関連する情報を取得し、この情報を対象アプリケーション３０の導入に関する基準と比較する。以下、第１装置１０に対する対象アプリケーション３０の導入に関連する情報を導入実績情報と呼ぶ。また、基準情報のうち、第１装置１０に対する対象アプリケーション３０の導入に着目して対象アプリケーション３０の実行許否を判定するために利用されるものを、特に、導入基準情報と呼ぶ。

[0077] 導入実績情報は、対象アプリケーション３０の識別情報に対応づけて、その対象アプリケーション３０の導入に関する情報を示す。対象アプリケーション３０の識別情報は、例えば、対象アプリケーション３０の名称や、対象アプリケーション３０の実行ファイルのパスなどで表される。ただし、対象アプリケーション３０の実行可否の基準が、対象アプリケーション３０が導入されている装置や装置のグループなどによって異なる場合、対象アプリケーション３０の識別情報には、「対象アプリケーション３０が導入されている第１装置１０の識別情報、対象アプリケーション３０の名称など」という組み合わせで表される。

[0078] 導入実績情報に含まれる対象アプリケーション３０の導入に関する情報としては、様々なものを採用できる。例えば導入実績情報は、以下の情報を含みうる。

１）経路情報：対象アプリケーション３０の導入経路に関する情報

２）配置情報：対象アプリケーション３０が配置された場所に関する情報

３）設定情報：対象アプリケーション３０の導入に伴う設定に関する情報

[0079] 以下、上述した種々の情報について、その詳細な内容及びそれらの情報を得る方法について説明する。

[0080] ＜＜１）経路情報について＞＞

経路情報は、対象アプリケーション３０の導入に関わるソフトウエア、ハードウエア、及びサービスなどに関する情報を含む。対象アプリケーション３０の導入に関わるソフトウエアは、例えば、対象アプリケーション３０をダウンロードするために利用されるダウンローダや、対象アプリケーション３０のインストールに利用されるインストーラである。また、対象アプリケーション３０のインストーラ等が圧縮されたファイルを入手する場合、その圧縮ファイルの解凍に利用される解凍ソフトウエアも、対象アプリケーション３０の導入に関わるソフトウエアといえる。対象アプリケーション３０の導入に関わるハードウエアは、例えば、対象アプリケーション３０のインストーラや実行ファイルなどが格納されている記憶装置などである。対象アプ

リケーション３０の導入に関わるサービスは、例えば、対象アプリケーション３０のインストーラなどを提供する Web サイトや、対象アプリケーション３０の提供元と第１装置１０との間に配置されるプロキシなどである。

[0081] 例えば、アプリケーションＸのインストーラＩの圧縮ファイルであるファイルＦがサーバＳで提供されているとする。そして、ダウンローダＤを用いてサーバＳからファイルＦをダウンロードし、ファイルＦを解凍ソフトウエアＢで解凍し、この解凍によって得られたアプリケーションＸのインストーラＩを実行することで、第１装置１０にアプリケーションＸが導入されたとする。この場合、例えばアプリケーションＸについての経路情報は、「サーバＳ、ダウンローダＤ、解凍ソフトウエアＢ、インストーラＩ」という情報を示す。

[0082] 経路情報の生成は、例えば、対象アプリケーション３０の導入に関連しうる種々のイベントの履歴を利用することで実現できる。イベントは、例えば、「主体、客体、内容」の組み合わせで表される。対象アプリケーション３０の導入に関連しうるイベントは、例えば、ファイルのダウンロード、圧縮ファイルの解凍、及びインストーラの実行などである。ここで、これらのイベントの履歴は、記憶装置に記憶させておく。なお、これらのイベントの履歴を記録する技術には既存の技術を利用することができる。例えば、第１装置１０上で実行されたシステムコールがイベントとして記録される。

[0083] 経路情報の生成は、例えば、第１装置１０に常駐させておくエージェントソフトウエアによって行われる。例えば、エージェントソフトウエアは、対象アプリケーション３０の導入に伴って発生しうる特定のイベント（以下、キーイベント）の発生を検知する。例えばキーイベントは、インストーラの実行である。さらにエージェントソフトウエアは、キーイベントの検知に応じて、そのキーイベントに関連する他のイベントを特定していく。例えばキーイベントがインストーラの実行である場合、エージェントソフトウエアは、イベントの履歴の中から、そのインストーラが含まれていた圧縮ファイルの解凍というイベントや、その圧縮ファイルのダウンロードというイベント

を抽出する。

[0084] 上述したイベントの抽出により、「インストーラが含まれる圧縮ファイルのダウンロード->圧縮ファイルの解凍->インストーラの実行」という、対象アプリケーション３０の導入に関連するイベント列を抽出できる。導入経路の情報は、このイベント列から生成することができる。例えば、圧縮ファイルのダウンロードイベントに基づいて、対象アプリケーション３０のインストーラの提供元（ウェブサイトなど）の特定、及びダウンロードに利用されたダウンローダの特定が行える。また、インストーラが含まれる圧縮ファイル解凍というイベントに基づいて、解凍に利用された解凍ソフトウエアを特定することができる。さらに、インストーラの実行というイベントに基づいて、対象アプリケーション３０のインストールに利用されたインストーラを特定することができる。経路情報は、これら特定された種々の情報で構成される。

[0085] なお、キーイベントには、所定の条件に当てはまるイベントを利用できる。例えば、アプリケーションが配置される標準的なディレクトリは OS やミドルウエアごとに予め決まっており、このようなディレクトリに対するファイルの書き込みは、対象アプリケーション３０の導入に関連する蓋然性が高いイベントであると考えられる。そこで例えば、エージェントソフトウエアは、アプリケーションが配置されるべき標準的なディレクトリに対してファイルを書き込むイベントを、キーイベントとして検知する。

[0086] その他にも例えば、アプリケーションの導入は、レジストリや所定の設定ファイル（環境変数が格納されているファイルなど）の更新を伴うことが多い。そこで例えば、エージェントソフトウエアは、レジストリや所定の設定ファイルに対する書き込みのイベントをキーイベントとして検知する。

[0087] その他にも例えば、アプリケーションの導入は、既知のインストーラ（例えば、OS に標準で用意されているインストーラ）を利用して行われることが多い。そこで例えば、エージェントソフトウエアは、このような既知のインストーラの実行を表すイベント（所定のプログラムの実行を表すイベント）

をキーイベントとして検知する。

[0088] なお、キーイベントの検出に用いる所定の条件は、エージェントソフトウエアからアクセス可能な記憶装置に予め記憶させておく。

[0089] ＜＜２）配置情報について＞＞

配置情報は、対象アプリケーション３０に関連するファイル（実行ファイルや設定ファイルなど）が書き込まれた場所（ディレクトリなど）に関する情報などを示す。

[0090] 例えば配置情報の生成は、以下のようにして行われる。まず、前提として、ファイルの書き込みイベントの履歴を記録しておく。そして、前述したエージェントソフトウエアが、このイベントの履歴を利用して、配置情報を生成する。例えばエージェントソフトウエアは、まず、インストーラの実行のイベントを検知する。さらにエージェントソフトウエアは、そのインストーラによって行われたファイルの書き込みイベントを特定する。そして、エージェントソフトウエアは、特定した各イベントにおいてファイルが書き込まれた場所を示す配置情報を生成する。

[0091] ＜＜３）設定情報について＞＞

対象アプリケーション３０によっては、そのインストールに伴い、レジストリや既存の設定ファイルに対して変更が加えられる。設定情報は、このように対象アプリケーション３０の導入に伴って加えられる設定の変更を表す。

[0092] 例えば設定情報は、配置情報と同様に、ファイルの書き込みイベントの履歴を利用して生成される。例えばエージェントソフトウエアは、まず、インストーラの実行のイベントを検知する。さらにエージェントソフトウエアは、そのインストーラによって行われた、レジストリや所定の設定ファイルに対する書き込みイベントを特定する。そして、エージェントソフトウエアは、特定した各イベントについて「イベントにおいて書き込みが行われたファイルの識別情報（パスなど）、そのファイルに対して書き込まれたデータの内容」という組み合わせを示す設定情報を生成する。

[0093] 図７は、導入実績情報をテーブル形式で例示する図である。図７のテーブルを、テーブル２００と呼ぶ。テーブル２００は、識別情報２０２、属性名２０４、及び属性値２０６という２つの列を有する。識別情報２０２は、対象アプリケーション３０の識別情報を表す。属性名２０４は、提供元、ダウンローダ、解凍ソフトウエア、インストーラ、配置情報、及び設定情報などといった情報の種類を表す。属性値２０６は、属性名２０２が示す種類の情報について、その内容を表す。例えば、「識別情報２０２：端末Ｘのアプリケーション Ａ、属性名２０４：ダウンローダ、属性値２０６：ブラウザＸ」という組みを示すレコードは、端末Ｘで実行されているアプリケーション Ａを導入する際、ダウンローダとしてブラウザＸが利用されたことを表す。

[0094] なお、導入実績情報の生成は、必ずしも第１装置１０によって行われる必要はなく、第２装置２０やその他の装置によって行われてもよい。この場合、導入実績情報を生成する装置は、第１装置１０について記録されたイベントの履歴を用いて、第１装置１０に対して導入された各アプリケーションについての導入実績情報を生成する。

[0095] ここで、導入実績情報が生成されるタイミングは、判定部２０２０による判定に利用されるタイミング、又はその前の任意のタイミングである。後者の場合、例えば、第１装置１０に対して新たなアプリケーションが導入されたタイミングで、そのアプリケーションについての導入実績情報が生成される。

[0096] 判定部２０２０が導入実績情報を取得する方法は任意である。例えば導入実績情報が記憶装置に格納される場合、判定部２０２０は、その記憶装置から対象アプリケーション３０についての導入実績情報を取得する。その他にも例えば、判定部２０２０は、前述したエージェントソフトウエアに対して、対象アプリケーション３０の導入実績情報の取得リクエストを送信することで、導入実績情報を取得してもよい。

[0097] 判定部２０２０は、対象アプリケーション３０について導入実績情報を取得し、取得した導入実績情報を導入基準情報と比較することで、対象アプリ

ケーション３０の実行許否を判定する導入基準情報は、ルールやポリシーなどとも呼ぶことができる。

[0098] 例えば導入基準情報は、正常なアプリケーションについての導入経路などを定めた情報である。このような導入基準情報を利用すると、例えば、導入実績情報と導入基準情報との一致度合いが高い場合に、対象アプリケーション３０の正常度が高いと判断することができる。このような導入基準情報を、正常導入基準情報と呼ぶ。

[0099] 例えば正常導入基準情報には、以下の情報が含まれる。

１）正常経路情報：対象アプリケーション３０の正常な導入経路

２）正常配置情報：対象アプリケーション３０の正常な配置場所

３）正常設定情報：対象アプリケーション３０のインストールに伴う正常な設定

[0100] 正常経路情報は、対象アプリケーション３０の導入に関連する正常なソフトウエア、正常なハードウエア、及び正常なサービスなどの情報を表す。例えば、正常経路情報は、対象アプリケーション３０の提供元となる正常なサービスやハードウエア（ウェブサイトや記憶装置など）を表す。さらに例えば、正常経路情報は、正常なインストーラ、正常な解凍ソフトウエア、及び正常なダウンローダなど、アプリケーションの導入に利用されうる正常なソフトウエアを示す。正常導入基準情報は、例えば、アプリケーションごとに定められる。その他にも例えば、正常導入基準情報は、OS 等の実行環境ごとに定められていてもよい。

[0101] また、正常経路情報は、正常な提供元やソフトウエアのセットを表してもよい。例えばこの情報は、「サーバＳ１、ダウンローダＤ１、インストーラＩ１」などといった情報である。

[0102] 正常配置情報は、アプリケーションがインストールされるべき正常な場所（ディレクトリなど）を示す。なお、アプリケーションがインストールされるべき場所は、アプリケーションごとや、OS などの実行環境ごとに定められていてもよい。

[0103] 正常設定情報は、アプリケーションの導入に伴って行われる正常な設定を表す。正常設定情報は、例えば、アプリケーションごとに定められる。例えば、アプリケーションＸが導入された場合にレジストリに所定のレコードＲが追加されることが分かっているとする。この場合、アプリケーションＸについての正常設定情報は、「レジストリに対するレコードＲの追加」を示す。

[0104] 導入基準情報は、異常なアプリケーションについての導入経路などを定めた情報であってもよい。このような導入基準情報を利用すると、例えば、導入実績情報と導入基準情報との一致度合いが高い場合に、対象アプリケーション３０の異常度が高い（正常度が低い）と判断することができる。このような導入基準情報を、異常導入基準情報と呼ぶ。

[0105] 異常導入基準情報には、例えば、以下の情報が含まれうる。

１）異常経路情報：アプリケーションの異常な導入経路

２）異常配置情報：アプリケーションの異常な配置場所

３）異常設定情報：アプリケーションのインストールに伴う異常な設定

[0106] 異常導入基準情報の詳細は、基本的に、正常導入基準情報の説明において「正常」と「異常」を入れ替えることで把握することができる。例えば、正常経路情報がアプリケーションの導入に利用されうる正常なソフトウエアなどを示す一方で、異常経路情報はアプリケーションの導入に利用されうる異常なソフトウエアなどを示す。例えば、マルウエアを拡散させていることで知られている既知の悪意あるウェブサイトがある場合、異常経路情報には、異常なソフトウエアの提供元として、そのウェブサイトの URL などを含めることができる。

[0107] ここで、導入基準情報を正常と異常に二分する代わりに、導入基準情報において、各属性値にその属性値の正常度（又は異常度）を対応づけて示してもよい。例えば、「属性名：インストーラ、属性値：インストーラＩ１、正常度合い：c1」などのような情報を、導入基準情報として利用することができる。

[0108] 図８は、導入基準情報をテーブル形式で例示する図である。このテーブルをテーブル３００と呼ぶ。テーブル３００は、識別情報３０２、属性名３０４、属性値３０６、及び正常度３０８という４つの列を含む。識別情報３０２、属性名３０４、及び属性値３０６については、テーブル２００の識別情報２０２、属性名２０４、及び属性値３０６と同様である。ただし、識別情報２０２にデータが示されていないレコードは、アプリケーションや実行環境に依存しないことを表す。正常度３０８は、対応する属性値の正常度を表す。

[0109] 判定部２０２０は、導入実績情報と導入基準情報とを比較することで、対象アプリケーション３０の実行許否を判定する。例えば判定部２０２０は、導入実績情報と導入基準情報とを比較することで、対象アプリケーション３０の正常度又は異常度を表す評価値を算出する。評価値が対象アプリケーション３０の正常度を表す場合、判定部２０２０は、例えば、評価値が所定の閾値以上であれば対象アプリケーション３０の実行を許可し、評価値が所定の閾値未満であれば対象アプリケーション３０の実行を許可しない。一方、評価値が対象アプリケーション３０の異常度を表す場合、判定部２０２０は、例えば、評価値が所定の閾値以下であれば対象アプリケーション３０の実行を許可し、評価値が所定の閾値より大きければ対象アプリケーション３０の実行を許可しない。

[0110] 対象アプリケーション３０の評価値は、例えば、導入実績情報と導入基準情報との一致度合いに基づいて算出される。ここで、ルールやポリシー（本発明における導入基準情報）と実際の状況（本発明における導入実績情報）との一致度合いを算出する技術自体には、種々の既存の技術を利用することができる。

[0111] 例えば、導入実績情報と導入基準情報との一致度合いは、以下の式（１）などを用いて算出することができる。

[数1]

$$v = \frac{|S|}{|E|} \qquad (1)$$

ここで、v は評価値を表す。E は、導入実績情報に示されている属性値の集合であり、|E| はその集合の要素数を表す。また、S は、導入実績情報と導入基準情報とで互いに一致する属性値の集合であり、|S| はその集合の要素数を表す。

[0112] 導入実績情報と正常導入基準情報とを比較する場合、これらの一致度合いは、対象アプリケーション３０の正常度合いを表す。一方、導入実績情報と異常導入基準情報とを比較する場合、これらの一致度合いは、対象アプリケーション３０の異常度合いを表す。

[0113] また、導入基準情報が各属性についてその正常度を示すとする。この場合、導入実績情報と正常導入基準情報との間で一致する属性値の正常度の積算値や統計値（平均値、中央値、最頻値、最大値、及び最小値など）を、対象アプリケーション３０の正常度を表す評価値として利用することができる。例えば、以下の数式（２）などを用いて評価値を算出することができる。

[数2]

$$v = \frac{\sum_{i \in S} w_i}{\sum_{j \in E} w_j} \qquad (2)$$

ここで、wi は属性値 i に付されている正常度である。

[0114] 一方、導入基準情報が各属性についてその異常度を示すとする。この場合、導入実績情報と異常導入基準情報との間で一致する属性値の異常度の積算値や統計値を、対象アプリケーション３０の異常度を表す評価値として利用することができる。その算出方法は、正常度を表す評価値と同様である。

[0115] なお、判定部２０２０は、導入実績情報と導入基準情報との不一致の度合いを評価に利用してもよい。例えば判定部２０２０は、導入実績情報と正常導入基準情報との一致度合いを表す評価値から、導入実績情報と正常導入基

準情報との不一致度合いを表す評価値を引くことで、対象アプリケーション３０の正常度を表す評価値を算出する。同様に、例えば判定部２０２０は、導入実績情報と異常導入基準情報との一致度合いを表す評価値から、導入実績情報と異常導入基準情報との不一致度合いを表す評価値を引くことで、対象アプリケーション３０の異常度を表す評価値を算出してもよい。

[0116] 前述した導入基準情報を生成する方法は様々である。例えば導入基準情報は、実行制御システム２０００を運用する組織の IT 管理者などによって手動で生成される。その他にも例えば、導入基準情報は、装置によって自動で生成されてもよい。導入基準情報を生成する装置は、第１装置１０であってもよいし、第２装置２０であってもよいし、その他の装置であってもよい。以下、導入基準情報を生成する装置を、基準情報生成装置と呼ぶ。基準情報生成装置は、例えば、第１装置１０や第２装置２０と同様に図３で例示されるハードウエア構成を持つ計算機である。

[0117] 例えば基準情報生成装置は、実行制御システム２０００に含まれる１つ以上の第１装置１０における対象アプリケーション３０の導入の実績に基づいて導入基準情報を生成する。概念的には、１つ以上の第１装置１０それぞれにおけるこれまでのアプリケーションの導入において、より多く利用された導入経路、配置場所、及び設定ほど、それぞれ、正常度が高い導入経路、配置場所、及び設定として扱われる。例えば、各対象アプリケーション３０について、対象アプリケーション３０が導入されたタイミングなどで、導入実績情報を生成しておく。そして、基準情報生成装置は、これまでに生成された導入実績情報を統計処理することで、導入基準情報を生成する。

[0118] 例えば、各属性値の正常度は、これまでに生成された導入実績情報のうち、その属性値を示すものの個数と正の相関を持つように定められる。例えば、正常度は、上記個数を所定の非単調減少関数に入力することで得られる値として定められる。ただし、導入実績情報の個数ではなく、第１装置１０の個数をカウントするようにしてもよい。すなわち、属性値の正常度を、その属性値を示す導入実績情報が生成された第１装置１０の個数と正の相関を持

つように定める。

[0119] 正常度を示す導入基準情報を生成する場合、例えば基準情報生成装置は、前述した方法で正常度が算出された属性値について、その属性値と正常度との組み合わせを含む導入基準情報を生成する。正常導入基準情報を生成する場合、例えば基準情報生成装置は、前述した方法で算出された正常度が所定の閾値以上である属性値を含む正常導入基準情報を生成する。異常導入基準情報を生成する場合、例えば基準情報生成装置は、前述した方法で算出された正常度が所定の閾値以下である属性値を含む正常導入基準情報を生成する。なお、正常導入基準情報の生成に利用する閾値と、異常導入基準情報の生成に利用する閾値は、同じであってもよいし、異なっていてもよい。

[0120] また、基準情報生成装置は、実行制御システム２０００が運用されているグループや外部組織などにおける評判（reputation）に基づいて、各属性値の正常度等を決定してもよい。実行制御システム２０００が運用されているグループにおける評判は、例えば、グループのメンバーに対して行ったアンケートを集計したり、グループ内で運用されている SNS（Social Networking Service）に投稿された情報を収集したりすることで得ることができる。また、外部組織における評判は、例えば、マルウエアなどの悪意あるソフトウエアや悪意あるウェブサイトなどに関する情報を公開しているサイトにアクセスしたりすることで収集することができる。基準情報生成装置は、これらの方法により、導入基準情報に含めうる種々の属性値（アプリケーションの提供元となるサービスやハードウエア、導入に利用されるソフトウエア、アプリケーションの配置場所、アプリケーションの導入によって行われる設定など）について、その評判の情報を収集する。そして、基準情報生成装置は、収集した評判の情報に基づいて、各属性値の正常度や異常度を算出する処理や、各属性値が正常と異常のどちらであるかを判定する処理を行う。そして、基準情報生成装置は、これらの処理結果に基づいて、導入基準情報を生成する。

[0121] また、対象アプリケーション３０が信頼度の高い有名なアプリケーション

である場合、そのアプリケーションの導入経路や配置場所、及びそのアプリケーションの導入に伴って行われる設定についての情報が、信頼できるウェブサイトなど（例えば対象アプリケーション３０の提供元のウェブサイト）などで公開されていることがある。そこで基準情報生成装置は、対象アプリケーション３０の導入について信頼度の高い情報を提供していると考えられるウェブサイト等にアクセスして情報を得ることで、導入基準情報を生成してもよい。

[0122] 判定部２０２０が導入基準情報を取得する方法は様々である。例えば判定部２０２０は、導入基準情報が記憶されている記憶装置から導入基準情報を取得する。その他にも例えば、判定部２０２０は、導入基準情報生成装置から導入基準情報を取得してもよい。

[0123] その他にも例えば、判定部２０２０は、以下で説明する方法で導入基準情報を取得してもよい。図９は、導入基準情報を管理する構成を例示する図である。この例では、判定部２０２０が第２装置２０に設けられていることを前提としている。

[0124] まず、導入基準情報が格納されうる記憶装置として、判定部２０２０からのアクセスに要する時間が比較的短い第１記憶装置７０と、判定部２０２０からのアクセスに要する時間が比較的長い第２記憶装置８０が設けられている。例えば第１記憶装置７０は、判定部２０２０が設けられている装置の内部に設けられている記憶装置や、判定部２０２０が設けられている装置と LAN で接続されている記憶装置である。一方、第２記憶装置８０は、判定部２０２０が設けられている装置と WAN で接続されている記憶装置（例えば、クラウドストレージ）である。

[0125] 導入基準情報は、第１記憶装置７０と第２記憶装置８０の双方に格納されうる。以下、第１記憶装置７０に格納される導入基準情報を第１導入基準情報と呼び、第２記憶装置８０に格納される導入基準情報を第２導入基準情報と呼ぶ。基準情報生成装置の運用開始時における第１導入基準情報は、例えば、IT 管理者によって手動で生成されたものである。また、基準情報生成装

置は、実行制御システム２０００における対象アプリケーション３０の導入の実績に基づいて、第１導入基準情報を更新してもよい。第２導入基準情報は、サーバ９０がインターネット上の情報を収集することによって随時更新されていく。

[0126] 判定部２０２０は、取得した導入実績情報との比較に利用する導入基準情報を取得する際、まずは第１記憶装置７０にアクセスして、第１導入基準情報の取得を試みる。第１導入基準情報の中に、導入実績情報に示されている属性値と一致する属性値が含まれていれば、判定部２０２０は、第１導入基準情報を利用する。一方、導入実績情報に示されている属性値の中に、一致する属性値が第１導入基準情報には存在しないものがあれば、判定部２０２０はサーバ９０にアクセスする。

[0127] 具体的には、判定部２０２０は、属性値を示すリクエストをサーバ９０へ送信する。サーバ９０は、第２記憶装置８０にアクセスし、リクエストに示される属性値が第２導入基準情報に含まれるか否かを判定する。リクエストに示される属性値が第２導入基準情報に含まれている場合、サーバ９０は、その属性値を示す第２導入基準情報のレコードが含まれるレスポンスを、判定部２０２０へ送信する。判定部２０２０は、受信したレコードに含まれる情報を対象アプリケーション３０の実行許否の判定に利用する。また、判定部２０２０は、このようにして取得したレコードを、第１導入基準情報に追加する。こうすることで、次回以降の評価では、同じ情報を第２記憶装置８０ではなく第１記憶装置７０から取得することができるため、情報の取得をより早く行える。一方、リクエストに示される属性値が第２導入基準情報に含まれていない場合、サーバ９０は、所望の情報が第２導入基準情報に含まれない旨を示すレスポンスを判定部２０２０へ送信する。

[0128] 例えば、第１導入基準情報のみを利用した判定を第１判定とし、第２導入基準情報も用いられる判定を第２判定とする。すなわち、第１導入基準情報では情報が不足しており、第２導入基準情報の取得（サーバ９０へのアクセス）が必要になったら、判定部２０２０による判定は、第１判定から第２判

定に進む。そのため、第２装置２０は、第２導入基準情報を取得することになったら、第１装置１０に対し、「第２判定に進む」という通知を送信する。

[0129] また、第２導入基準情報を用いても実行許否を決定できなかった場合、さらに第３判定として、IT 管理者による人手の判定を加えてもよい。例えば、正常度を表す評価値の定義域について２つの閾値 T1 及び T2 を設けておく（T1>T2）。この場合、第２判定において、１）評価値が T1 以上であれば対象アプリケーション３０の実行を許可し、２）評価値が T2 未満であれば対象アプリケーション３０の実行を許可せず、３）評価値が T2 以上かつ T1 未満であれば第３判定を行うようにする。

[0130] 対象アプリケーション３０について第３判定を行う場合、例えば、IT 管理者等の端末（以下、管理者端末）に対し、第３判定が必要な対象アプリケーション３０が存在することを通知する。この通知を受け取った IT 管理者等は、管理者端末に対し、その対象アプリケーション３０の実行を許可するか否かを選択する入力を行う。この入力結果が、判定部２０２０による判定の結果として扱われる。

[0131] 対象アプリケーション３０の実行許否の判定には、対象アプリケーション３０の導入に関する基準以外が利用されてもよい。その他の基準としては、例えば、以下の基準を利用することができる。

１）対象アプリケーション３０の作成者

２）対象アプリケーション３０のシグニチャ（バイナリのハッシュ値など）

３）対象アプリケーション３０自体に関する評判

[0132] 対象アプリケーション３０の作成者が有名な人物や組織である場合、対象アプリケーション３０の正常度は高いと考えられる。また、対象アプリケーション３０のシグニチャが、信頼性が担保されている（例えば、正当な認証局によって認証済みである）アプリケーションについて公開されているシグニチャと一致する場合、対象アプリケーション３０の正常度は高いと考えられる。同様に、第１装置１０に導入された対象アプリケーション３０のシグ

ニチャが、既知にマルウエアのシグニチャと一致する場合、対象アプリケーション３０の正常度は低いと考えられる。さらに、実行制御システム２０００が運用されているグループや外部組織など（例えばインターネット上）における対象アプリケーション３０の評判が高ければ、対象アプリケーション３０の正常度は高いと考えられる。

[0133] このように、対象アプリケーション３０の導入に関する以外の種々の情報も、対象アプリケーション３０の実行許否を判定する行う上で有用なものになりうる。そこで例えば、判定部２０２０は、これら種々の情報をさらに利用して、対象アプリケーション３０の実行許否を判定する。この場合、例えば基準情報に、前述した導入基準情報に加え、対象アプリケーション３０の作成者、シグニチャ、及び評判などに関する基準も加える。例えば、「属性名：作成者、属性値：xyz.inc」などの基準である。また、判定部２０２０は、対象アプリケーション３０について、導入実績情報に加え、対象アプリケーション３０の作成者、シグニチャ、及び評判などに関する情報も取得する。そして、判定部２０２０は、取得した各種の情報を基準情報と比較することで、対象アプリケーション３０の実行許否を判定する。

[0134] ここで、対象アプリケーション３０について取得した作成者、シグニチャ、及び評判などに関する情報と、基準情報に含まれるこれらの情報とを比較する方法は、導入実績情報と基準情報とを比較する方法と同様である。例えば判定部２０２０は、前述した式（１）や（２）に示した評価値の算出式に、対象アプリケーション３０の導入に関連する情報の一致度合いだけでなく、作成者、シグニチャ、及び評判などの一致度合いも含めるようにする。

[0135] なお、基準情報には、必ずしも導入基準情報が含まれる必要はない。すなわち、対象アプリケーション３０の実行許否の判定は、対象アプリケーション３０の作者についての基準など、対象アプリケーション３０の導入に関する基準以外の基準のみを用いて行われてもよい。

[0136] ＜情報の出力＞

実行制御システム２０００の動作に関する種々の情報が対象アプリケーシ

ョン３０のユーザに対して提供されるようにしてもよい。このように対象アプリケーション３０のユーザに対して情報を提供する機能構成部を出力部と呼び、出力部によって出力される情報を出力情報と呼ぶ。図１０は、出力部２０６０を有する実行制御システム２０００の機能構成を例示するブロック図である。出力部２０６０は、第１装置１０と第２装置２０のいずれか一方、又は双方に設けられる。

[0137] 出力部２０６０によって出力された出力情報は、第１装置１０により、第１装置１０のユーザがその内容を把握できる任意の対象で出力される。例えば第１装置１０は、出力情報の内容を表す画面を、第１装置１０に接続されているディスプレイ装置に表示させる。

[0138] 出力情報の内容には、様々なものを採用できる。例えば出力情報には、判定部２０２０による判定の最終結果や途中経過に関する情報が含まれる。例えば最終結果に関する情報には、対象アプリケーション３０の実行が許可されたか否かを表す情報が含まれる。最終結果に関する情報は、例えば、判定部２０２０による判定が終了したタイミングで出力される。

[0139] 対象アプリケーション３０の実行が許可された場合、対象アプリケーション３０を正常に利用できることをユーザが把握できるようにするメッセージなどが出力される。例えば、「対象アプリケーション３０の実行が許可されました。対象アプリケーション３０は正常に利用できます。」などといったメッセージである。

[0140] 一方、対象アプリケーション３０の実行が許可されなかった場合、対象アプリケーション３０を正常に利用できないことをユーザが把握できるようにするメッセージなどが出力される。例えば、「対象アプリケーション３０の実行が許可されませんでした。対象アプリケーション３０を終了します。」などといったメッセージである。

[0141] 判定の途中経過に関する情報としては、例えば、対象アプリケーション３０の実行許否の判定が行われていることをユーザが把握できるようにするメッセージが出力される。例えばこの情報は、判定部２０２０による判定が開

始されたタイミングで出力される。例えば、「対象アプリケーション３０の実行許否を判定しています」などといったメッセージが出力される。

[0142] 判定部２０２０による判定が複数の段階を含む場合、判定の途中経過に関する情報は、例えば、どの段階の判定が行われているのかをユーザが把握できるようにするメッセージなどである。例えば、第１判定では対象アプリケーション３０の実行許否を判断できなかったために第２判定が行われるとする。この場合、「第１判定を完了しました。第２判定を開始します。」や「第２判定実行中」などといったメッセージが出力される。これらのメッセージは、例えば、判定が次の段階に移行するタイミングで出力される。

[0143] なお、対象アプリケーション３０をプロテクト環境で実行する際、その旨を出力情報に含めることが好適である。例えば、「対象アプリケーション３０の実行許否を判定します。対象アプリケーション３０はプロテクト環境で実行されます。」などといったメッセージが出力される。また、対象アプリケーション３０の実行環境をプロテクト環境から通常の環境へ変更する際には、そのことをユーザが把握できるメッセージ等が出力されることが好適である。例えば、「対象アプリケーション３０の実行環境を通常の環境に変更します」などといったメッセージが出力される。

[0144] ＜変形例＞

これまでは、実行制御システム２０００が、アプリケーションについてその実行許否の判定や制御を行うことを説明した。しかしながら、実行制御システム２０００は、アプリケーションに加え、又はこれに代えて、共有ライブラリのロードの許否の判定及び制御を行ってもよい。すなわち、実行制御システム２０００は、アプリケーションの実行許否の判定及びアプリケーションの実行の制御を行う方法と同様の方法で、共有ライブラリのロードの許否の判定及び共有ライブラリーを利用した処理の制御を行う。以下、実行制御システム２０００による判定及び制御の対象となる共有ライブラリを、対象ライブラリと呼ぶ。

[0145] 対象ライブラリのロードの許否の判定は、対象アプリケーション３０の実

行許否の判定と同様に、第１判定と第２判定を含む（前述した様に、３つ以上の判定が含まれてもよい）。実行制御システム２０００は、第１判定が完了するまでは、対象ライブラリのロードを行わない。そして、第１判定では対象ライブラリのロードの許否を決定できず、第２判定が行われる場合、実行制御システム２０００は、対象ライブラリを利用した処理（共有ライブラリで定義されている関数の実行など）がプロテクト環境で行われるように、共有ライブラリをロードする。

[0146] 対象ライブラリを利用した処理がプロテクト環境で実行される場合、その処理には、前述した種々の制限（データの読み書きやプロセスの起動の制限など）が適用される。なお、その制限の具体的な実現方法には、対象アプリケーション３０に対する制限を実現する方法と同様の方法を利用することができる。

[0147] 対象ライブラリは、実行制御システム２０００による判定の結果に応じた扱いに関しても、対象アプリケーション３０と同様とすることができる。すなわち、その処理がプロテクト環境で実行されるようにロードされた対象ライブラリについて、そのロードを許可すると判定された場合、制御部２０４０は、対象ライブラリを利用した処理が通常環境で実行されるようにする（通常環境に移行する）。また、その処理がプロテクト環境で実行されるようにロードされた対象ライブラリについて、そのロードを許可しないと判定された場合、例えば制御部２０４０は、対象ライブラリのアンロードを行う。

[0148] 共有ライブラリのロードの許否の判定の基準には、対象アプリケーション３０の実行の許否の判定の基準と同様のものを利用することができる。例えば一般に、共有ライブラリは、アプリケーションと同様に、何らかの方法で端末に導入される（例えば、インターネット経由でインストールされる）。そのため、共有ライブラリについても、アプリケーションと同様に、その導入経路を把握することができる。そこで例えば、実行制御システム２０００は、対象ライブラリの導入経路を、共有ライブラリの導入に関する基準と比較することにより、対象ライブラリのロードの許否を判定する。その具体的

な方法には、対象アプリケーション３０の導入経路に基づいて対象アプリケーション３０の実行の許否を判定する方法と同様の方法を採用することができる。

[0149] 以上、図面を参照して本発明の実施形態について述べたが、これらは本発明の例示であり、上記以外の様々な構成を採用することもできる。

[0150] 上記の実施形態の一部又は全部は、以下の付記のようにも記載されうるが、以下には限られない。

１． 対象ソフトウエアの動作の許否を判定する判定部を有し、

前記判定は、第１判定と、前記第１判定で前記対象ソフトウエアの動作の許否を決定できなかった場合に行わる第２判定とを含み、

前記第１判定が終わった後であって前記第２判定が行われている間に、前記対象ソフトウエアをプロテクト環境で動作させる制御部を有する、実行制御システム。

２． 前記判定部による判定は、前記対象ソフトウエアの動作が開始された時、及び前記対象ソフトウエアが導入された時の少なくとも一方で開始される、１．に記載の実行制御システム。

３． 前記第２判定に要する時間は前記第１判定に要する時間よりも長い、１．又は２．に記載の実行制御システム。

４． 前記プロテクト環境で動作する前記対象ソフトウエアによって行われるデータの書き込みは、他のソフトウエアからアクセスできない第１記憶領域に対して行われ、

前記対象ソフトウエアの動作が許可された場合、前記制御部は、前記第１記憶領域に書き込まれたデータを、少なくとも１つの他のソフトウエアからアクセス可能な第２記憶領域に書き込む、１．乃至３．いずれか一つに記載の実行制御システム。

５． 前記対象ソフトウエアの動作が許可された場合、前記制御部は、前記対象ソフトウエアの動作環境をプロテクト環境から通常の実行環境に変更する、１．乃至４．いずれか一つに記載の実行制御システム。

６．　前記対象ソフトウエアの動作が許可されなかった場合、前記制御部は、前記対象ソフトウエアの動作を終了する、１．乃至５．いずれか一つに記載の実行制御システム。

７．　コンピュータによって実行される実行制御方法であって、

対象ソフトウエアの動作の許否を判定する判定ステップを有し、

前記判定は、第１判定と、前記第１判定で前記対象ソフトウエアの動作の許否を決定できなかった場合に行わる第２判定とを含み、

前記第１判定が終わった後であって前記第２判定が行われている間に、前記対象ソフトウエアをプロテクト環境で動作させる制御ステップを有する、実行制御方法。

８．　前記判定ステップによる判定は、前記対象ソフトウエアの動作が開始された時、及び前記対象ソフトウエアが導入された時の少なくとも一方で開始される、７．に記載の実行制御方法。

９．　前記第２判定に要する時間は前記第１判定に要する時間よりも長い、７．又は８．に記載の実行制御方法。

１０．　前記プロテクト環境で動作する前記対象ソフトウエアによって行われるデータの書き込みは、他のソフトウエアからアクセスできない第１記憶領域に対して行われ、

前記対象ソフトウエアの動作が許可された場合、前記制御ステップにおいて、前記第１記憶領域に書き込まれたデータを、少なくとも１つの他のソフトウエアからアクセス可能な第２記憶領域に書き込む、７．乃至９．いずれか一つに記載の実行制御方法。

１１．　前記対象ソフトウエアの動作が許可された場合、前記制御ステップにおいて、前記対象ソフトウエアの動作環境をプロテクト環境から通常の実行環境に変更する、７．乃至１０．いずれか一つに記載の実行制御方法。

１２．　前記対象ソフトウエアの動作が許可されなかった場合、前記制御ステップにおいて、前記対象ソフトウエアの動作を終了する、７．乃至１１．いずれか一つに記載の実行制御方法。

１３．　７．乃至１２．いずれか一つに記載の実行制御方法の各ステップをコンピュータに実行させるプログラム。

請求の範囲

[請求項1] 対象ソフトウエアの動作の許否を判定する判定部を有し、

前記判定は、第１判定と、前記第１判定で前記対象ソフトウエアの動作の許否を決定できなかった場合に行わる第２判定とを含み、

前記第１判定が終わった後であって前記第２判定が行われている間に、前記対象ソフトウエアをプロテクト環境で動作させる制御部を有する、実行制御システム。

[請求項2] 前記判定部による判定は、前記対象ソフトウエアの動作が開始された時、及び前記対象ソフトウエアが導入された時の少なくとも一方で開始される、請求項１に記載の実行制御システム。

[請求項3] 前記第２判定に要する時間は前記第１判定に要する時間よりも長い、請求項１又は２に記載の実行制御システム。

[請求項4] 前記プロテクト環境で動作する前記対象ソフトウエアによって行われるデータの書き込みは、他のソフトウエアからアクセスできない第１記憶領域に対して行われ、

前記対象ソフトウエアの動作が許可された場合、前記制御部は、前記第１記憶領域に書き込まれたデータを、少なくとも１つの他のソフトウエアからアクセス可能な第２記憶領域に書き込む、請求項１乃至３いずれか一項に記載の実行制御システム。

[請求項5] 前記対象ソフトウエアの動作が許可された場合、前記制御部は、前記対象ソフトウエアの動作環境をプロテクト環境から通常の実行環境に変更する、請求項１乃至４いずれか一項に記載の実行制御システム。

[請求項6] 前記対象ソフトウエアの動作が許可されなかった場合、前記制御部は、前記対象ソフトウエアの動作を終了する、請求項１乃至５いずれか一項に記載の実行制御システム。

[請求項7] コンピュータによって実行される実行制御方法であって、

対象ソフトウエアの動作の許否を判定する判定ステップを有し、

前記判定は、第１判定と、前記第１判定で前記対象ソフトウエアの動作の許否を決定できなかった場合に行わる第２判定とを含み、

前記第１判定が終わった後であって前記第２判定が行われている間に、前記対象ソフトウエアをプロテクト環境で動作させる制御ステップを有する、実行制御方法。

[請求項8] 前記判定ステップによる判定は、前記対象ソフトウエアの動作が開始された時、及び前記対象ソフトウエアが導入された時の少なくとも一方で開始される、請求項７に記載の実行制御方法。

[請求項9] 前記第２判定に要する時間は前記第１判定に要する時間よりも長い、請求項７又は８に記載の実行制御方法。

[請求項10] 前記プロテクト環境で動作する前記対象ソフトウエアによって行われるデータの書き込みは、他のソフトウエアからアクセスできない第１記憶領域に対して行われ、

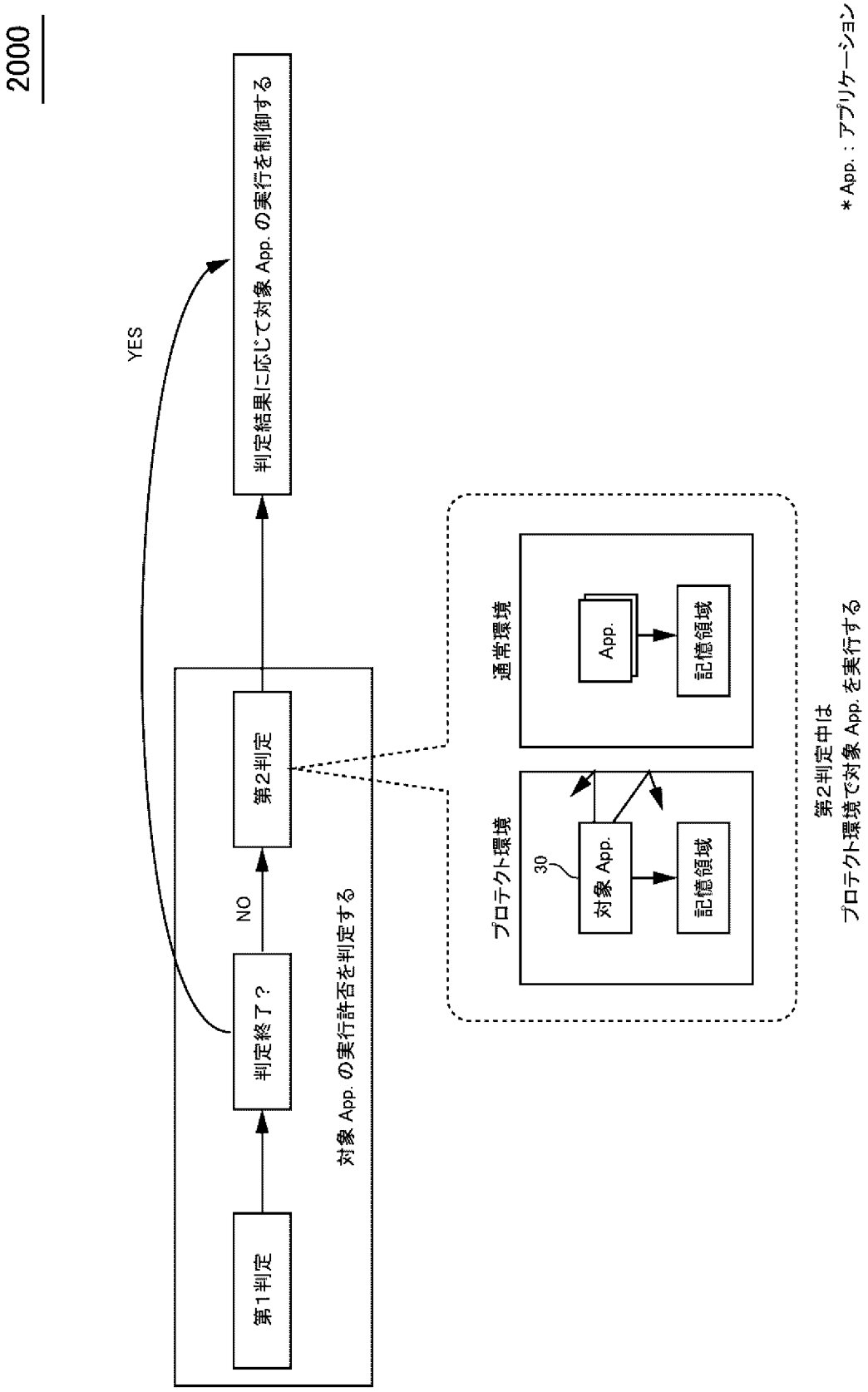
前記対象ソフトウエアの動作が許可された場合、前記制御ステップにおいて、前記第１記憶領域に書き込まれたデータを、少なくとも１つの他のソフトウエアからアクセス可能な第２記憶領域に書き込む、請求項７乃至９いずれか一項に記載の実行制御方法。

[請求項11] 前記対象ソフトウエアの動作が許可された場合、前記制御ステップにおいて、前記対象ソフトウエアの動作環境をプロテクト環境から通常の実行環境に変更する、請求項７乃至１０いずれか一項に記載の実行制御方法。

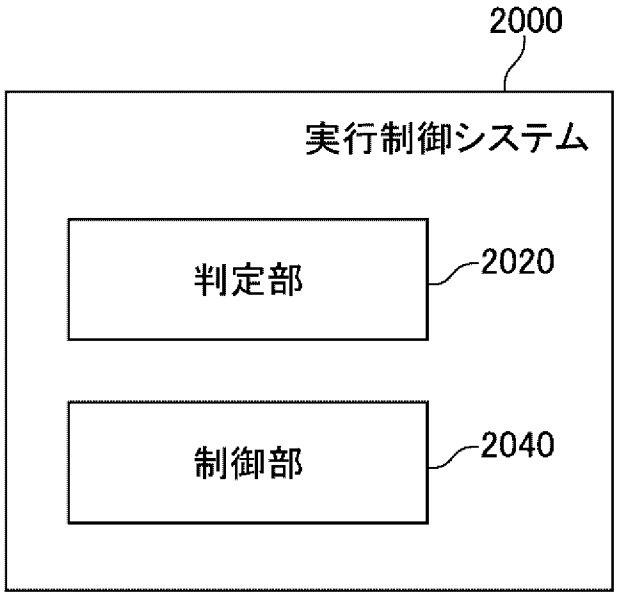
[請求項12] 前記対象ソフトウエアの動作が許可されなかった場合、前記制御ステップにおいて、前記対象ソフトウエアの動作を終了する、請求項７乃至１１いずれか一項に記載の実行制御方法。

[請求項13] 請求項７乃至１２いずれか一項に記載の実行制御方法の各ステップをコンピュータに実行させるプログラム。

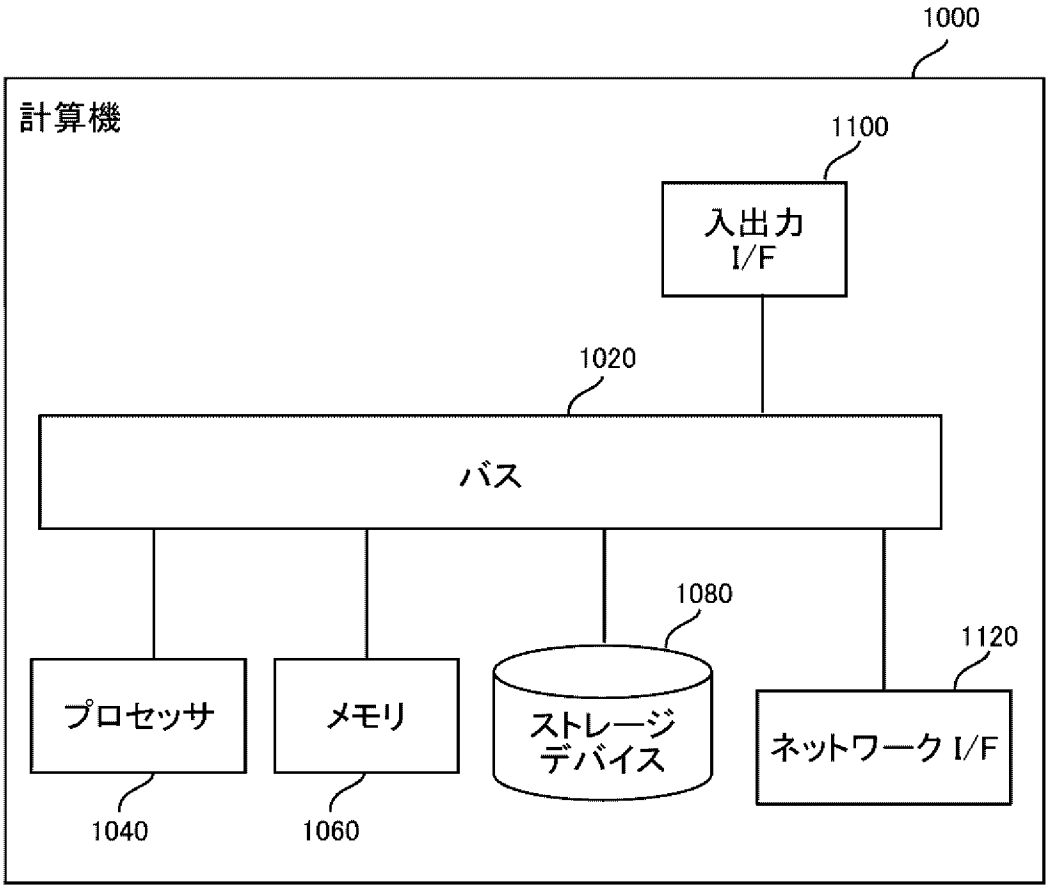
[図1]


2000
判定結果に応じて対象 App. の実行を制御する
YES
第1判定
判定終了？
NO
第2判定
対象 App. の実行許否を判定する
通常環境
App.
記憶領域
プロテクト環境
30
対象 App.
記憶領域
第2判定中は
プロテクト環境で対象 App. を実行する
＊App.：アプリケーション

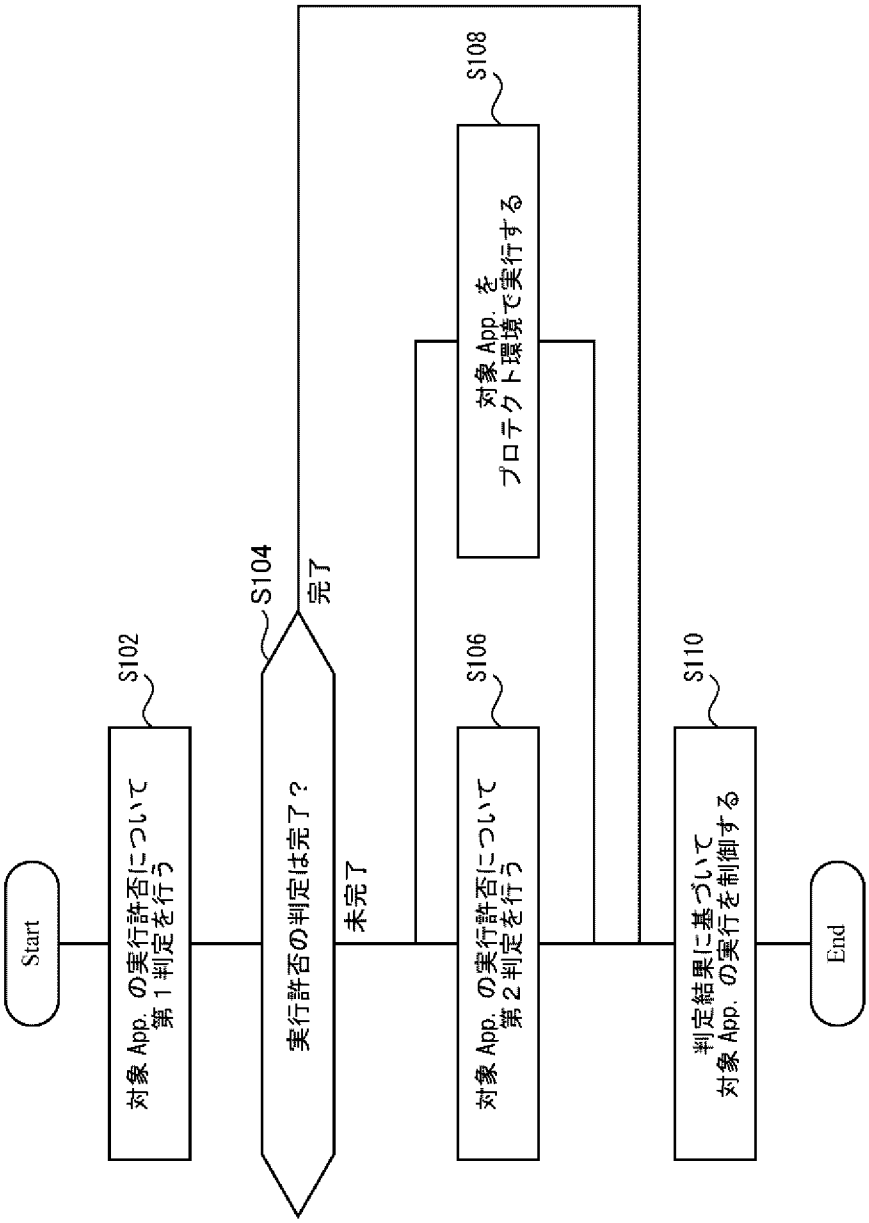
[図2]



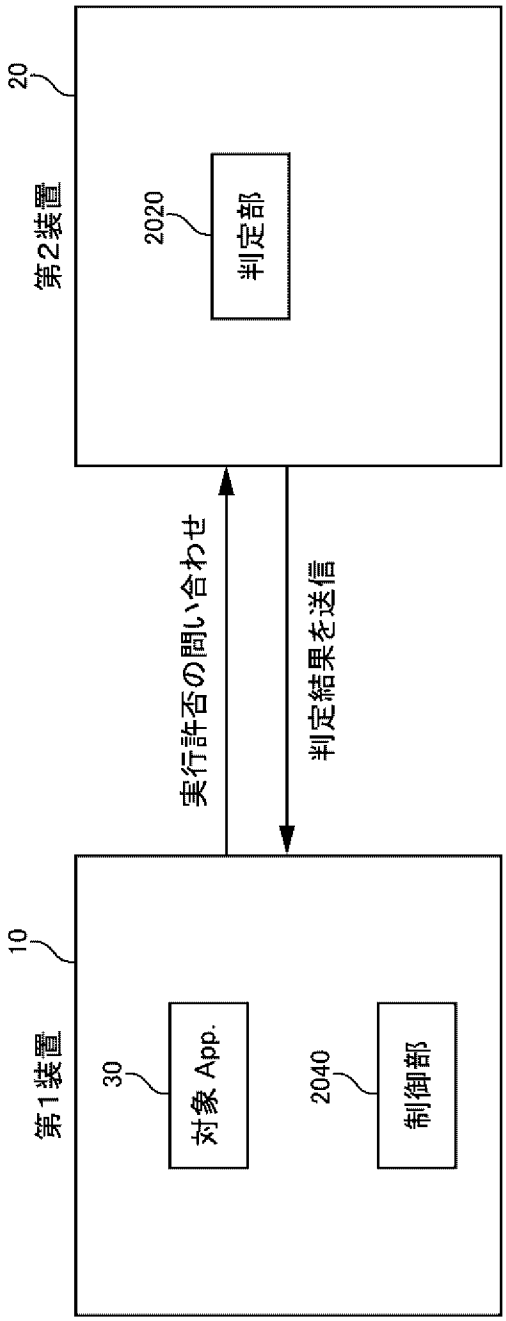
[図3]


1000
計算機
1100
入出力
I/F
1020
バス
1080
1120
プロセッサ
メモリ
ストレージ
デバイス
ネットワーク I/F
1040
1060

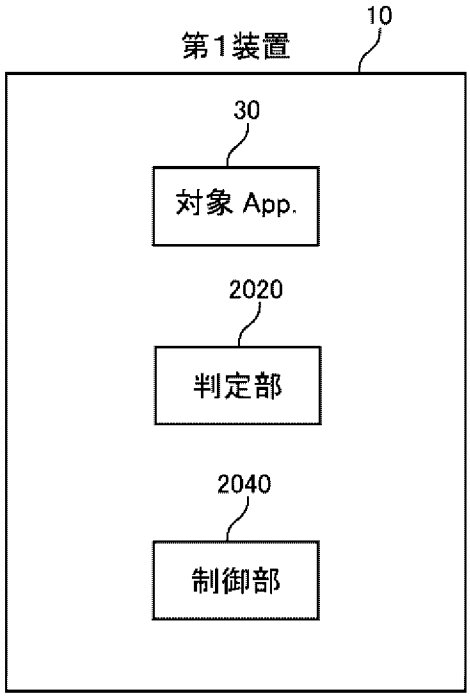
[図4]


Start
対象 App. の実行許否について
第１判定を行う
S102
実行許否の判定は完了？
S104
完了
未完了
対象 App. の実行許否について
第２判定を行う
S106
対象 App. を
プロテクト環境で実行する
S108
判定結果に基づいて
対象 App. の実行を制御する
S110
End

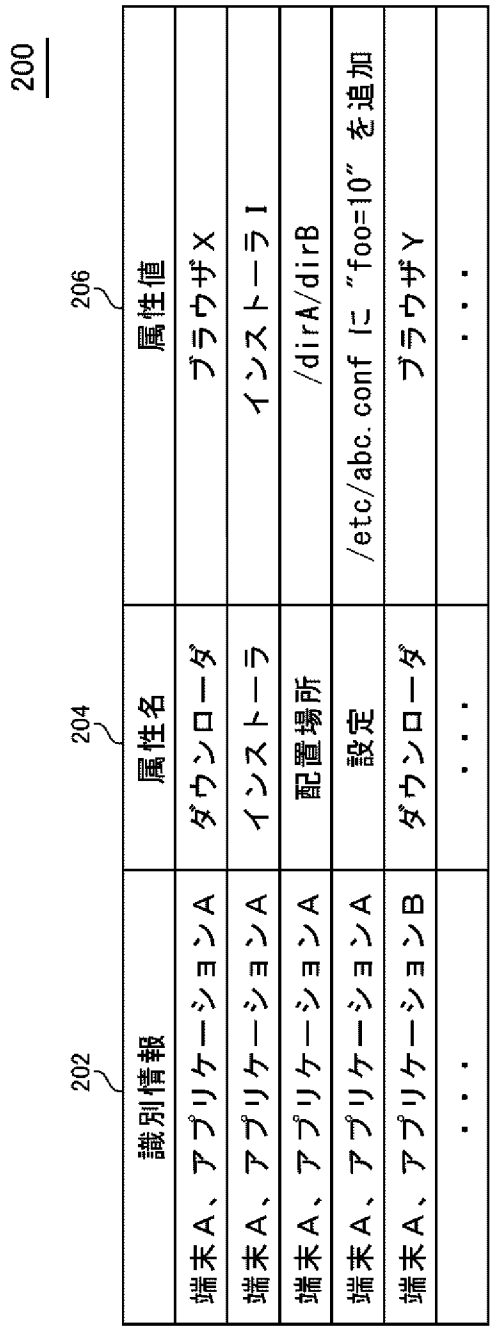
[図5]


20
第2装置
2020
判定部
実行許否の問い合わせ
判定結果を送信
10
第1装置
30
対象 App.
2040
制御部

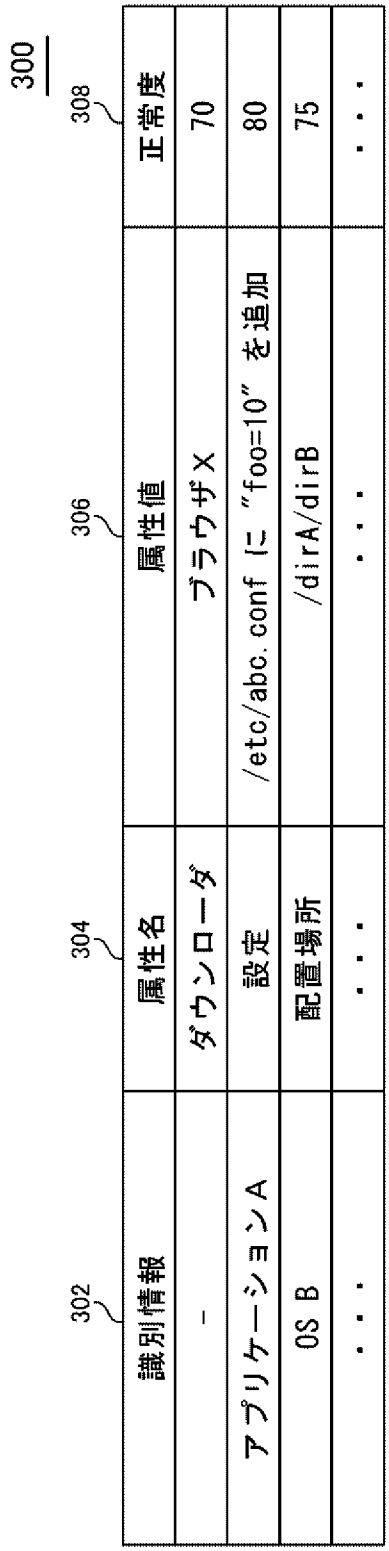
[図6]



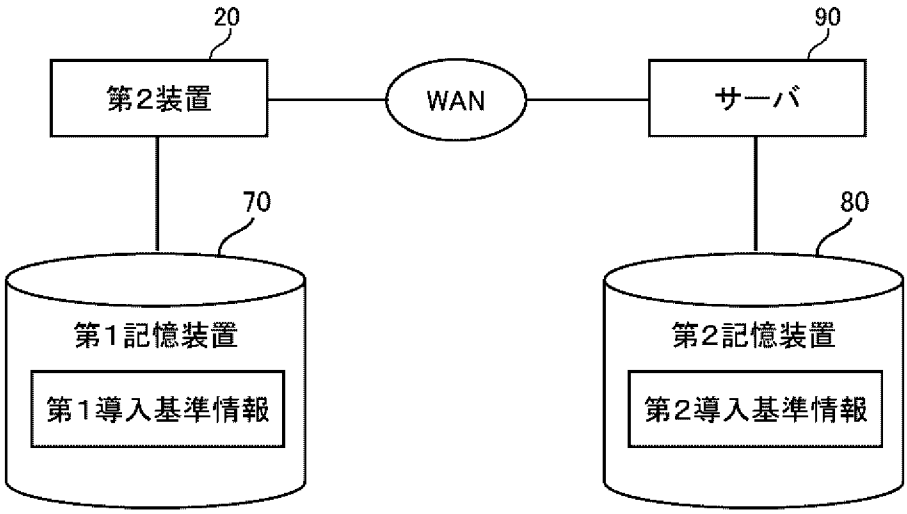
[図7]


200
202
204
206
識別情報
属性名
属性値
端末A、アプリケーションA
ダウンローダ
ブラウザX
端末A、アプリケーションA
インストーラ
インストーラI
端末A、アプリケーションA
配置場所
/dirA/dirB
端末A、アプリケーションA
設定
/etc/abc.conf に "foo=10" を追加
端末A、アプリケーションB
ダウンローダ
ブラウザY
・・・
・・・
・・・

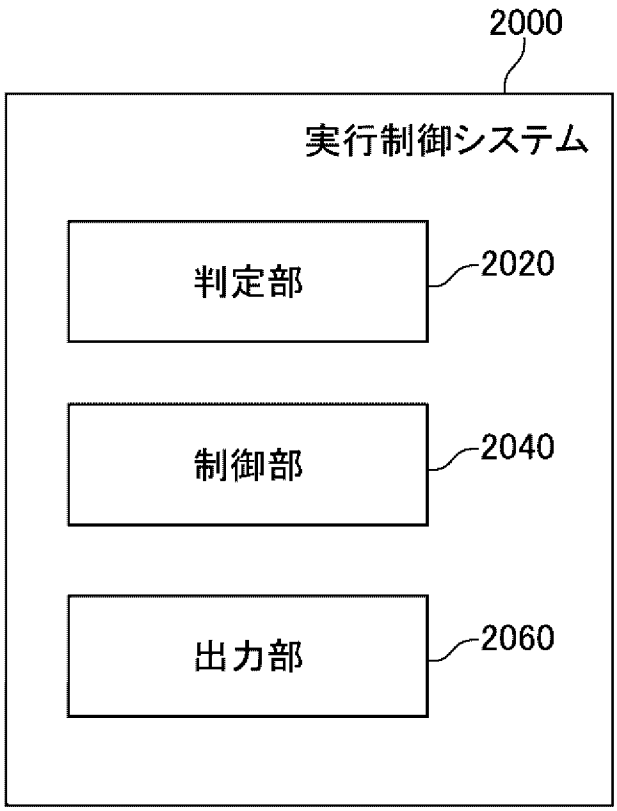
[図8]


300
302
識別情報
304
属性名
306
属性値
308
正常度
-
ダウンローダ
ブラウザX
70
アプリケーションA
設定
/etc/abc.conf に "foo=10" を追加
80
OS B
配置場所
/dirA/dirB
75
・・・
・・・
・・・
・・・

[図9]



[図10]



INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2019/025414

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl. G06F21/53(2013.01)i
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
Int.Cl. G06F21/53

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan	1922-1996
Published unexamined utility model applications of Japan	1971-2019
Registered utility model specifications of Japan	1996-2019
Published registered utility model applications of Japan	1994-2019

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2010-079906 A (SYMANTEC CORPORATION) 08 April 2010 & US 2010/0083376 A1 & EP 2169583 A1	1-13
A	JP 2013-540303 A (LOOKOUT INC.) 31 October 2013 & US 2013/0263260 A1 & CN 103180862 A & KR 10-2013-0129184 A	1-13
A	WO 2007/074565 A1 (NEC CORP.) 05 July 2007 & US 2009/0307781 A1 & EP 1967981 A1 & CN 101346727 A	1-13

☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
09 September 2019 (09.09.2019)	17 September 2019 (17.09.2019)

Name and mailing address of the ISA/	Authorized officer
Japan Patent Office 3-4-3, Kasumigaseki, Chiyoda-ku, Tokyo 100-8915, Japan	Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2019/025414

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2014-021929 A (TOSHIBA CORP.) 03 February 2014 & US 2014/0026198 A1 & WO 2014/017117 A1	1-13
A	US 2008/0120611 A1 (AARON, Jeffrey) 22 May 2008 (Family: none)	1-13

国際調査報告

国際出願番号　ＰＣＴ／ＪＰ２０１９／０２５４１４

Ａ．　発明の属する分野の分類（国際特許分類（ＩＰＣ））

Int.Cl.　G06F21/53(2013.01)i

Ｂ．　調査を行った分野

調査を行った最小限資料（国際特許分類（ＩＰＣ））

Int.Cl.　G06F21/53

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	１９２２－１９９６年
日本国公開実用新案公報	１９７１－２０１９年
日本国実用新案登録公報	１９９６－２０１９年
日本国登録実用新案公報	１９９４－２０１９年

国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）

Ｃ．　関連すると認められる文献

引用文献のカテゴリー＊	引用文献名　及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求項の番号
A	JP 2010-079906 A（シマンテック　コーポレイション）2010.04.08 & US 2010/0083376 A1 & EP 2169583 A1	1-13
A	JP 2013-540303 A（ルックアウト、アイエヌシー．）2013.10.31 & US 2013/0263260 A1 & CN 103180862 A & KR 10-2013-0129184 A	1-13
A	WO 2007/074565 A1（日本電気株式会社）2007.07.05 & US 2009/0307781 A1 & EP 1967981 A1 & CN 101346727 A	1-13

☑　Ｃ欄の続きにも文献が列挙されている。　　☐　パテントファミリーに関する別紙を参照。

＊　引用文献のカテゴリー
「Ａ」特に関連のある文献ではなく、一般的技術水準を示すもの
「Ｅ」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの
「Ｌ」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）
「Ｏ」口頭による開示、使用、展示等に言及する文献
「Ｐ」国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献
「Ｔ」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの
「Ｘ」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの
「Ｙ」特に関連のある文献であって、当該文献と他の１以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの
「＆」同一パテントファミリー文献

国際調査を完了した日	国際調査報告の発送日
０９．０９．２０１９	１７．０９．２０１９

国際調査機関の名称及びあて先
日本国特許庁（ＩＳＡ／ＪＰ）
郵便番号１００－８９１５
東京都千代田区霞が関三丁目４番３号

特許庁審査官（権限のある職員）　５Ｓ　３４５３
田中　啓介
電話番号　０３－３５８１－１１０１　内線　３５４６

Ｃ（続き）．　関連すると認められる文献

引用文献のカテゴリー＊	引用文献名　及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求項の番号
A	JP 2014-021929 A (株式会社東芝) 2014.02.03 & US 2014/0026198 A1 & WO 2014/017117 A1	1-13
A	US 2008/0120611 A1 (AARON, Jeffrey) 2008.05.22(ファミリーなし)	1-13