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

特許第6165469号
(P6165469)

(45)発行日　平成29年7月19日(2017.7.19)　(24)登録日　平成29年6月30日(2017.6.30)

(51)Int.Cl.		F I		
G06F 21/12	(2013.01)	G06F	21/12	310
G06F 21/55	(2013.01)	G06F	21/55	320
G06F 11/34	(2006.01)	G06F	11/34	176

請求項の数 11　(全 16 頁)

(21)出願番号　特願2013-41248(P2013-41248)
(22)出願日　平成25年3月1日(2013.3.1)
(65)公開番号　特開2014-170327(P2014-170327A)
(43)公開日　平成26年9月18日(2014.9.18)
審査請求日　平成28年2月29日(2016.2.29)

前置審査

(73)特許権者　000104652
キヤノン電子株式会社
埼玉県秩父市下影森１２４８番地
(74)代理人　100076428
弁理士　大塚　康徳
(74)代理人　100115071
弁理士　大塚　康弘
(74)代理人　100112508
弁理士　高柳　司郎
(74)代理人　100116894
弁理士　木村　秀二
(74)代理人　100130409
弁理士　下山　治
(74)代理人　100134175
弁理士　永川　行光

最終頁に続く

(54)【発明の名称】情報処理装置およびその制御方法、並びに、情報処理システム

(57)【特許請求の範囲】
【請求項１】
プロセスの動きを検知する検知手段と、
前記検知手段によってプロセスの動きが検知されると、プロセスの動きに関する第一のリストに基づいて前記プロセスの動きを許可または禁止する第一の制御手段と、
前記第一の制御手段にて許可または禁止されたプロセスのログ情報を出力する出力手段と、
前記出力手段によって出力されたログ情報に対応するプロセスの情報を第二のリストに保持する保持手段と、
前記第二のリストに保持されたプロセスの情報に基づき、前記第一の制御手段にて許可または禁止されたプロセスのログ情報が既に出力されたログ情報に一致するか否かを判定する判定手段と、
前記判定手段による判定結果に基づき、前記出力手段によるログ情報の出力を制御する第二の制御手段と
を有し、
前記第二の制御手段は、前記プロセスのログ情報が既に出力されたログ情報に一致しない場合は、前記ログ情報を出力し、かつ、前記プロセスの情報に基づき前記第二のリストに保持されたプロセスの情報を更新し、　前記第二のリストに保持されたプロセスの情報は、所定時間経過後に削除されることを特徴とする情報処理装置。
【請求項２】

前記第二の制御手段は、前記プロセスのログ情報が既に出力されたログ情報に一致する場合は前記ログ情報の出力を抑制する請求項1に記載された情報処理装置。

【請求項３】

前記プロセスの動きにはプログラムの起動、プログラムによるネットワークアクセス要求のいずれか、または組み合わせが含まれる請求項1または請求項2に記載された情報処理装置。

【請求項４】

プロセスの動きを検知する検知手段と、

前記検知手段によってプロセスの動きが検知されると、プロセスの動きに関する第一のリストに基づいて前記プロセスの動きを許可または禁止する第一の制御手段と、

前記第一の制御手段にて許可または禁止されたプロセスのログ情報を出力する出力手段と、

前記出力手段によって出力されたログ情報に対応するプロセスの情報を第二のリストに保持する保持手段と、

前記第二のリストに保持されたプロセスの情報に基づき、前記第一の制御手段にて許可または禁止されたプロセスのログ情報が既に出力されたログ情報に一致するか否かを判定する判定手段と、

前記判定手段による判定結果に基づき、前記出力手段によるログ情報の出力を制御する第二の制御手段と

を有し、

前記第二の制御手段は、前記プロセスのログ情報が既に出力されたログ情報に一致しない場合は、前記ログ情報を出力し、かつ、前記プロセスの情報に基づき前記第二のリストに保持されたプロセスの情報を更新し、

前記出力手段にてログ情報が出力されたときにメッセージが表示されることを特徴とする情報処理装置。

【請求項５】

プロセスの動きを検知する検知手段、および、前記検知手段によってプロセスの動きが検知されると、プロセスの動きに関する第一のリストに基づいて前記プロセスの動きを許可または禁止する第一の制御手段と、前記第一の制御手段にて許可または禁止されたプロセスのログ情報を出力する出力手段を有する情報処理装置の制御方法であって、

保持手段が、前記出力手段によって出力されたログ情報に対応するプロセスの情報を第二のリストに保持し、

判定手段が、前記第二のリストに保持されたプロセスの情報に基づき、前記第一の制御手段にて許可または禁止されたプロセスのログ情報が既に出力されたログ情報に一致するか否かを判定し、

第二の制御手段が、前記判定手段による判定結果に基づき、前記出力手段によるログ情報の出力を制御し、前記プロセスのログ情報が既に出力されたログ情報に一致しない場合は、前記ログ情報を出力し、かつ、前記プロセスの情報に基づき前記第二のリストに保持されたプロセスの情報を更新し、

前記第二のリストに保持されたプロセスの情報は、所定時間経過後に削除されることを特徴とする制御方法。

【請求項６】

プロセスの動きを検知する検知手段、および、前記検知手段によってプロセスの動きが検知されると、プロセスの動きに関する第一のリストに基づいて前記プロセスの動きを許可または禁止する第一の制御手段と、前記第一の制御手段にて許可または禁止されたプロセスのログ情報を出力する出力手段を有する情報処理装置の制御方法であって、

保持手段が、前記出力手段によって出力されたログ情報に対応するプロセスの情報を第二のリストに保持し、

判定手段が、前記第二のリストに保持されたプロセスの情報に基づき、前記第一の制御手段にて許可または禁止されたプロセスのログ情報が既に出力されたログ情報に一致する

か否かを判定し、
第二の制御手段が、前記判定手段による判定結果に基づき、前記出力手段によるログ情報の出力を制御し、前記プロセスのログ情報が既に出力されたログ情報に一致しない場合は、前記ログ情報を出力し、かつ、前記プロセスの情報に基づき前記第二のリストに保持されたプロセスの情報を更新し、
前記出力手段にてログ情報が出力されたときにメッセージが表示されることを特徴とする制御方法。
【請求項７】
請求項1から請求項4の何れか一項に記載された情報処理装置と、
前記情報処理装置からログ情報を取得し、前記情報処理装置に対応させて前記ログ情報を管理するサーバ装置を有する情報処理システム。
【請求項８】
前記サーバ装置は、
インストーラを作成する作成手段を備え、
前記作成手段が作成したインストーラに関する情報を、前記プロセスの動きに関する第一のリストに登録することを特徴とする請求項7に記載された情報処理システム。
【請求項９】
前記サーバ装置は、
前記プロセスの動きに関する第一のリストを前記情報処理装置へ送信する手段を備え、
前記プロセスの動きに関する第一のリストは、前記情報処理装置から取得したログ情報に基づいて更新されることを特徴とする請求項7または請求項8に記載された情報処理装置システム。
【請求項１０】
コンピュータを請求項1から請求項4の何れか一項に記載された情報処理装置の各手段として機能させるためのプログラム。
【請求項１１】
コンピュータを請求項7から請求項9の何れか一項に記載されたサーバ装置の各手段として機能させるためのプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、プロセスのログ情報を取得する情報処理に関する。
【背景技術】
【０００２】
近年、企業に対するサイバー攻撃の新しい形態として、企業内の特定の社員が使用または所有するコンピュータを標的にし、そのコンピュータにマルウェアを感染させ、企業内の情報を盗み出す、標的型攻撃という手法が増えている。
【０００３】
従来のアンチウィルスソフトなどは、ブラックリスト方式によるウィルス定義ファイルを用いる。しかし、コンピュータウィルスを含むマルウェアの種類は日に万単位で増加している。そのため、ウィルス定義ソフトの更新は、マルウェアの急増に追いつかない状況にあり、従来のアンチウィルスソフトによる標的型攻撃への対処は難しい状況にある。
【０００４】
そのため、所定のプログラムの起動や通信のみを許可し、それ以外のプログラムの起動や通信を制限する、所謂ホワイトリスト型のプロセス制御、通信制御を用いる標的型攻撃への対処が存在する（例えば、特許文献1）。
【０００５】
通常、ホワイトリスト型の制御方式の運用を開始する際、情報処理装置の通常使用に支障を来さないよう、事前に情報処理装置の利用実態を調査して、適切な制御設定を施す必要がある。この事前調査の期間を「トレーニング期間」と呼ぶ。トレーニング期間におけ

る利用実態の調査には、情報処理装置の操作履歴が利用されることが多い。
【０００６】
通常操作履歴は、通常、リアルタイムに発生するすべてのイベントを網羅的に記録することを目的に設計されるているため、必然的に、記録される履歴の件数が多い特性がある。この特性は、フォレンジック(forensic)のような監査目的には非常に有用ではある。しかし、トレーニング期間における利用実態の調査、および、ホワイトリストの選定という目的には、いささか冗長であり、情報処理装置を管理する管理者には操作履歴の処理作業が負担となる。
【０００７】
そこで、不要なログ情報を特定する処理を自動的に行う要求があり、その一例として、特許文献2が開示するシステムが存在する。
【０００８】
しかし、特許文献2が開示する自動化処理は、どのようなログ情報（操作履歴）が不要かを予め設定しておく必要がある。そもそも情報処理装置に対してユーザがどのような操作を行うかは全く予想が付かないため、そのような自動化処理によれば、不要なログ情報が削除されない、あるいは、必要なログ情報が削除される、といった事態も想定される。
【０００９】
また、例えば、ネットワークアクセスが許可されない（禁止されている）場合も、通常、CPUはネットワークへのアクセスを繰り返す（リトライする）。従って、リトライした分のログ情報も記録され、ログ情報が膨大になる可能性がある。
【先行技術文献】
【特許文献】
【００１０】
【特許文献１】特開2009-259160号公報
【特許文献２】特開2007-317130号公報
【発明の概要】
【発明が解決しようとする課題】
【００１１】
本発明は、ログ情報の情報量を軽減することを目的とする。
【課題を解決するための手段】
【００１２】
本発明は、前記の目的を達成する一手段として、以下の構成を備える。
【００１３】
本発明にかかる情報処理装置は、
プロセスの動きを検知する検知手段と、
前記検知手段によってプロセスの動きが検知されると、プロセスの動きに関する第一のリストに基づいて前記プロセスの動きを許可または禁止する第一の制御手段と、
前記第一の制御手段にて許可または禁止されたプロセスのログ情報を出力する出力手段と、
前記出力手段によって出力されたログ情報に対応するプロセスの情報を第二のリストに保持する保持手段と、
前記第二のリストに保持されたプロセスの情報に基づき、前記第一の制御手段にて許可または禁止されたプロセスのログ情報が既に出力されたログ情報に一致するか否かを判定する判定手段と、
前記判定手段による判定結果に基づき、前記出力手段によるログ情報の出力を制御する第二の制御手段と
を有し、
前記第二の制御手段は、前記プロセスのログ情報が既に出力されたログ情報に一致しない場合は、前記ログ情報を出力し、かつ、前記プロセスの情報に基づき前記第二のリストに保持されたプロセスの情報を更新し、前記第二のリストに保持されたプロセスの情報

は、所定時間経過後に削除されることを特徴とする。
また他の側面によれば本発明に係る情報処理装置は、
プロセスの動きを検知する検知手段と、
前記検知手段によってプロセスの動きが検知されると、プロセスの動きに関する第一のリストに基づいて前記プロセスの動きを許可または禁止する第一の制御手段と、
前記第一の制御手段にて許可または禁止されたプロセスのログ情報を出力する出力手段と、
前記出力手段によって出力されたログ情報に対応するプロセスの情報を第二のリストに保持する保持手段と、
前記第二のリストに保持されたプロセスの情報に基づき、前記第一の制御手段にて許可または禁止されたプロセスのログ情報が既に出力されたログ情報に一致するか否かを判定する判定手段と、
前記判定手段による判定結果に基づき、前記出力手段によるログ情報の出力を制御する第二の制御手段と
を有し、
前記第二の制御手段は、前記プロセスのログ情報が既に出力されたログ情報に一致しない場合は、前記ログ情報を出力し、かつ、前記プロセスの情報に基づき前記第二のリストに保持されたプロセスの情報を更新し、
前記出力手段にてログ情報が出力されたときにメッセージが表示されることを特徴とする。

【発明の効果】
【００１４】
本発明によれば、ログ情報の情報量を軽減することができる。
【図面の簡単な説明】
【００１５】
【図１】ホワイトリスト制御システムの構成例を示すブロック図。
【図２】サーバが存在しないホワイトリスト制御システムの構成例を示すブロック図。
【図３】ホワイトリストの一例を示す図。
【図４】サーバに存在するホワイトリストマスタの一例を示す図。
【図５】ホワイトリスト型のネットワークアクセス制御におけるログ情報の出力を説明するフローチャート。
【図６】ホワイトリスト型のプログラム起動制御におけるログ情報の出力を説明するフローチャート。
【図７】プロセス情報が格納されたリストの一例を示す図。
【図８】管理コンソールから制御プログラムのインストーラを作成する例を説明するフローチャート。
【発明を実施するための形態】
【００１６】
以下、本発明にかかる実施例の情報処理システムの情報処理を図面を参照して詳細に説明する。
【００１７】
［システム構成］
図1のブロック図によりホワイトリスト制御システムの構成例を示す。ホワイトリスト制御システムは、情報処理装置と、情報処理装置を管理するサーバ装置を含む。
【００１８】
図1において、クライアントコンピュータ（以下、クライアント）10は、ホワイトリスト制御システムにおける情報処理装置である。クライアント10は、例えば、企業、学校、行政機関または家庭などに設置されたパーソナルコンピュータ(PC)や、個人が使用または所有するタブレット端末やスマートフォンなどのコンピュータ機器である。
【００１９】

サーバコンピュータ（以下、サーバ）20は、ホワイトリスト制御システムにおける情報処理装置を管理するサーバ装置である。サーバ20は、複数のクライアント10からホワイトリスト120の情報を取得してデータベース化したり、定期的にクライアント10にホワイトリストデータを送信してホワイトリスト120の更新を行う。

【００２０】

ネットワーク300は、インターネットやイントラネットなどのコンピュータネットワークである。クライアント10は、ネットワーク300を介して、サーバ20や、図示しないウェブサーバやFTPサーバなどと接続する。

【００２１】

なお、簡潔化のため図1にはクライアント10とサーバ20を一台ずつ示すが、実際には、ホワイトリスト制御システムに複数のクライアントや複数のサーバが存在することができる。

【００２２】

●クライアント

クライアント10において、演算装置10Cはマイクロプロセッサ(CPU)である。演算装置10Cは、メモリ10EのROMに格納されたBIOSなどのブートプログラムに従い記憶装置10Bに格納されたオペレーティングシステム(OS)を起動し、さらにOSに従い各種の常駐プログラム（例えば制御プログラム113など）を起動する。その際、演算装置10Cは、メモリ10EのRAMをワークエリアとして使用する。また、OSは例えばWindows（登録商標）、Mac OS（登録商標）、Linux（登録商標）、iOS（商標）、Android（商標）などである。

【００２３】

記憶装置10Bは、ハードディスクドライブ(HDD)やソリッドステートドライブ(SSD)などであり、OSのほかにクライアント10上で稼働する各種のプログラム100やデータ101を格納する。記憶装置10Bが格納する各種プログラム100には識別プログラム110、登録プログラム111、検知プログラム112、制御プログラム113、ファイル検索ツール114などが含まれる。また、記憶装置10Bが格納する各種データ101にはホワイトリスト120、昇格基準ルール130、ブラックリスト140、操作履歴150などが含まれる。

【００２４】

なお、プログラム100は、識別プログラム110など各種機能ごとのプログラムを複数備えてもよいし、各種機能を備えた一つのプログラムでもよい。

【００２５】

I/Oデバイス10Aは、ポインティングデバイス（マウスなど）やキーボードに接続するための入出力インタフェイス(I/F)、または、タッチパネルを組み込んだディスプレイなどである。なお、キーボードはソフトウェアキーボードでもよい。また、I/Oデバイス10Aは、入力された操作者の音声を音声認識機能によって認識し、認識した音声を演算装置10Cへ伝達する、マイク等を含む音声式入力部でもよい。また、I/Oデバイス10Aは、情報を表示するためのユーザインタフェイス(UI)としても機能する。

【００２６】

ネットワークI/F 10Dは、ネットワーク300とのインタフェイスであり、他のコンピュータと通信するための通信回路である。演算装置10Cは、ネットワークI/F 10Dを介して、例えばホワイトリスト120の一部データなどの情報をサーバ20から受信し、また、各種情報をサーバ20に送信する。

【００２７】

●サーバ

サーバ20において、演算装置20Cはマイクロプロセッサ(CPU)である。演算装置20Cは、メモリ20EのROMに格納されたBIOSなどのブートプログラムに従い記憶装置20Bに格納されたOSを起動する。さらに、演算装置20Cは、記憶装置20Bから管理コンソール210をメモリ20EのRAMにロードする。そして、複数のクライアント10から情報（例えば、ホワイトリスト120の情報など）を取得してデータベース化したり、逆に、クライアント10に対して情報を送信してホワイトリスト120の更新などを行う。

【００２８】
記憶装置20Bは、HDDやSSDなどであり、OSのほかにサーバ20上で稼働する管理コンソール210を含む各種のプログラム200やデータ201を格納する。詳細は後述するが、記憶装置10Bが格納する各種データ201にはホワイトリストマスタ220、昇格基準ルール230、ブラックリストマスタ240、操作履歴250などが含まれる。
【００２９】
I/Oデバイス20Aは、ポインティングデバイス（マウスなど）、キーボード、モニタに接続するためのインタフェイス(I/F)であり、モニタは情報を表示するためのUIとして機能する。ネットワークI/F 20Dは、ネットワーク300とのインタフェイスであり、クライアント10など他のコンピュータと通信するための通信回路である。
【００３０】
演算装置20Cは、ネットワークI/F 20Dを介して、複数のクライアント10からホワイトリスト120やブラックリスト140に関する情報を受信し、受信した情報に基づき、ホワイトリストマスタ220やブラックリストマスタ250を管理する。
【００３１】
ホワイトリスト制御システムにおいて、サーバ20は必須の構成ではない。図2のブロック図によりサーバ20が存在しないホワイトリスト制御システムの構成例を示す。図2の構成においては、クライアント10とサーバ20の通信は不要になるため、ネットワーク300およびネットワークI/F 10Dもオプションである。
【００３２】
また、ホワイトリスト制御システムはシンクライアント（例えば、ターミナルサービスなど）を利用した構成としてもよい。シンクライアントは、クライアントがサーバにリモート接続し、サーバ上に生成された仮想デスクトップ環境を利用してサーバ上でアプリケーションプログラムを実行できるようにするシステムである。
【００３３】
［プログラムおよびデータ］
識別プログラム110は、演算装置10Cによって実行され、別途起動されるプログラムからファイル名（プログラム名）、ハッシュ値、バージョン情報、ファイルサイズ、ファイルパス、ディジタル署名などの情報（以下、プログラム情報）を取得する。また、プログラム情報には、当該プログラムが格納されているPC名も含まれる。また、識別プログラム110は、取得したプログラム情報に基づき当該プログラムを識別する識別機能を有し、取得したプログラム情報と後述する昇格基準ルール130を照会して、当該プログラムが昇格基準を満たすか否かを判定する。
【００３４】
ホワイトリスト120は、プログラムの起動や通信を許可するプログラムに関する情報をリスト化したものである。ホワイトリスト120を構成する情報には、識別プログラム110によって取得されたプログラム情報が用いられる。
【００３５】
図3によりホワイトリスト120の一例を示す。ホワイトリスト120は、各プログラムについて、プログラム名（実行ファイル名）、ハッシュ値、バージョン情報、ファイルサイズ、接続先IPアドレス、接続先ポート番号、昇格権限フラグの七種類の情報を保持する。なお、図3は一例であり、ホワイトリスト120として保持する情報の種類と数は図3に限定されるものではない。
【００３６】
ホワイトリストマスタ220は、複数のホワイトリスト120をリスト化したものである。図4によりサーバ20に存在するホワイトリストマスタ220の一例を示す。ホワイトリストマスタ220は、複数のクライアントのホワイトリスト120に関連する情報をクライアントの名称や符号に対応付けて保持する。図4の例では、PC003に対応するホワイトリスト003として、複数のプログラムのプロセス名、ハッシュ値、接続先IPアドレス、接続先ポート番号、登録日時、最終起動日時が保持された例を示す。なお、図4は一例であり、ホワイトリス

トマスタ220として保持する情報の種類と数は図4に限定されるものではない。

【００３７】

検知プログラム112は、演算装置10Cによって実行され、プロセスの動きを監視する監視機能と、プロセスの動きを検知する検知機能を有する。プロセスの動きには、プログラムの起動、起動されたプログラムによる別のプログラムの生成、ネットワークアクセス要求などが含まれる。

【００３８】

登録プログラム111は、演算装置10Cによって実行され、検知プログラム112に起動や生成が検知されたプログラムの、識別プログラム110が取得したプログラム情報に基づき、当該プログラムをホワイトリスト120に登録する登録機能を有する。

【００３９】

制御プログラム113は、演算装置10Cによって実行され、検知プログラム112によりクライアント10上でプログラムの起動が検知されると、当該プログラムの起動を許可または禁止（阻止）する起動制御機能を有する。また、検知プログラム112によりプログラムによるネットワークアクセス要求が検知されると、当該プログラムのネットワークアクセスを許可または禁止（阻止）するネットワークアクセス制御機能を有する。詳細は後述するが、さらに、検知プログラム112により検知されたプロセスのログ情報が既に出力されたログ情報に一致するか否かを判定する判定機能、検知プログラム112により検知されたプロセスのログ情報の出力を制御するログ出力制御機能を有す。

【００４０】

ネットワークアクセス制御機能に使用される、プログラムごとのネットワークアクセスの許可または禁止を示すホワイトリストに相当する情報（例えば、アクセス制御リスト(ACL)）は、記憶装置10Bの所定領域にテーブルとして格納されている。制御プログラム113は、ACLをメモリ10Eにロードし、ACLに基づきネットワークアクセス制御を行う。

【００４１】

昇格基準ルール130は、プログラムやファイルが、信頼できる発行者によって発行されたものか否かを判断するためのルールである。昇格基準ルール130は、プログラム情報を元に、管理者やユーザなどが定義したルールである。ルールには、例えば、プログラムやファイルに付加されたディジタル署名、ディジタル証明書が正当か否かの検証、ファイルの署名者名が予め記憶された名前か否かの判定、ファイル名が指定条件を満たすか否かの判定などがある。

【００４２】

また、昇格基準ルール130として適用するルールの種類や数は、前記の具体例に限定されるものではない。例えば「ディジタル署名やディジタル証明書が正当であり、かつ、ファイル名に"Setup"または"Update"という文字が含まれている」という複数の組み合わせのルールを適用することもできる。この場合「画像ビューワ(Viewer.exe)の脆弱性を突いてマルウェアを生成させるマルウェア」が動作したとき、画像ビューワのディジタル署名が正当でも、そのファイル名に前記の文字列が含まれない。その結果、画像ビューワの脆弱性を突く攻撃を防ぐ効果が得られる。

【００４３】

クライアント10が有する操作履歴150には、識別プログラム110によって取得されたプログラム情報などがユーザの操作履歴として記録される。例えば、プログラムの起動に対しては、実行ファイル名（プログラム名）、ハッシュ値、バージョン情報、ファイルサイズ、ファイルパス、ディジタル署名、プログラムの起動の許可/禁止（成功/失敗）、などが操作履歴150に記録される。また、ネットワークアクセスの要求に対しては、実行ファイル名（プログラム名）、ハッシュ値、バージョン情報、ファイルサイズ、接続先IPアドレス、接続先ポート番号、ネットワークアクセスの許可／禁止（成功/失敗）などが操作履歴150に記録される。また、プロセスの動きが検知された日時なども操作履歴150に記録される。

【００４４】

制御プログラム113は、定期的（例えば、一時間置きや一日置き）に操作履歴150（データファイル）をサーバ20へ送信する。サーバ20の操作履歴250（データファイル）には、クライアント10からに送信された操作履歴150が、クライアント10の名称や符号に対応付けて記録され、管理される。なお、制御プログラム113はサーバ20へリアルタイムに操作履歴を送信してもよい。

【００４５】

●ログ情報の出力

図5のフローチャートによりホワイトリスト型のネットワークアクセス制御におけるログ情報の出力を説明する。

【００４６】

検知プログラム112は、プログラムによるネットワークアクセス要求を監視し(S201)、プログラムによるネットワークアクセス要求を検知すると識別プログラム110を呼び出す。識別プログラム110は、当該プログラムのプログラム情報を取得する(S202)。制御プログラム113は、取得されたプログラム情報とACLに基づき、当該プログラムのネットワークアクセスの許可または禁止を判定し(S203)、判定結果に基づき当該プログラムのネットワークアクセスを制御する。

【００４７】

制御プログラム113は、当該プログラムのネットワークアクセスが許可されている場合、ネットワークアクセスを許可する(S204)。つまり、当該プログラムが発行したコマンドや出力したデータをネットワークI/F 10Dに転送し、ネットワークI/F 10Dが受信した当該プログラムあてのデータを当該プログラムに転送する。なお、ACLには、プログラムごとに、接続先のIPアドレスやポート番号の制限が設定されている場合がある。その場合、制御プログラム113は、当該制限に従いフィルタリングを行う。

【００４８】

また、制御プログラム113は、当該プログラムのネットワークアクセスが禁止されている場合、ネットワークアクセスを許可しない(S205)。つまり、当該プログラムとネットワークI/F 10Dの間のデータ転送を実行せず、当該プログラムのネットワークアクセス要求に対してエラーメッセージを返す。

【００４９】

制御プログラム113は、上記ネットワークアクセス制御を行った後、ログ情報を出力する。過去に出力されたログ情報（以下、既出力ログ情報）に関する情報が、メモリ10Eに割り当てられたリストに保持されている。

【００５０】

制御プログラム113は、リストを参照して、これから出力しようとするログ情報（以下、新ログ情報）が既出力ログ情報に一致するか否かを判定する(S206)。判定条件としては、プロセス名、ハッシュ値、IPアドレス、ポート番号、禁止/許可など一意に判定することができる情報（以下、プロセス情報）を使用すればよい。もし、新ログ情報が既出力ログ情報に一致する場合、新ログ情報の出力は抑制され、処理はステップS201に戻る。従って、新ログ情報の操作履歴150への書き出しは行われない。

【００５１】

一方、新ログ情報に一致する既出力ログ情報がないと判定した場合、制御プログラム113は、プロセス情報に基づきリストを更新し(S207)、新ログ情報をメモリ10Eの所定領域に保持する(S208)。

【００５２】

図5には、ステップS208に続いて、ログ情報が操作履歴150に書き出される(S209)ように記載するが、実際には、ステップS208の後、処理はステップS201に戻る。そして、タイマ割込などによる所定のタイミングで、メモリ10Eに保持されたすべてのログ情報がまとめて操作履歴150に書き出され、メモリ10Eに保持されたログ情報が削除される(S209)。なお、ログ情報がメモリ10Eに保持される度に、ログ情報を操作履歴150に書き出し、所定のタイミングでメモリ10Eに保持されたログ情報を削除してもよい。

【００５３】
図6のフローチャートによりホワイトリスト型のプログラム起動制御におけるログ情報の出力を説明する。
【００５４】
検知プログラム112は、プログラムの起動を監視し(S301)、プログラムの起動を検知すると識別プログラム110を呼び出す。識別プログラム110は、当該プログラムのプログラム情報を取得する(S302)。制御プログラム113は、取得されたプログラム情報とホワイトリスト120に基づき、当該プログラムの起動の許可または禁止を判定し(S303)、判定結果に基づき当該プログラムの起動を制御する。
【００５５】
制御プログラム113は、当該プログラムの起動が許可されている場合、当該プログラムの起動を許可し(S304)、当該プログラムの起動が禁止されている場合、当該プログラムの起動を阻止する(S305)。
【００５６】
制御プログラム113は、上記プログラム起動制御を行った後、上記リストを参照して、これから出力しようとするログ情報（新ログ情報）が既出力ログ情報に一致するか否かを判定する(S306)。判定条件としては、プロセス名、ハッシュ値、禁止/許可など一意に判定することができるプロセス情報を使用すればよい。もし、新ログ情報が既出力ログ情報に一致する場合、新ログ情報の出力は抑制され、処理はステップS301に戻る。従って、新ログ情報の操作履歴150への書き出しは行われない。
【００５７】
一方、新ログ情報に一致する既出力ログ情報がないと判定した場合、制御プログラム113は、プロセス情報に基づきリストを更新し(S307)、新ログ情報をメモリ10Eの所定領域に保持する(S308)。
【００５８】
図6には、ステップS308に続いて、ログ情報が操作履歴150に書き出される(S309)ように記載するが、実際には、ステップS308の後、処理はステップS301に戻る。そして、タイマ割込などによる所定のタイミングで、メモリ10Eに保持されたすべてのログ情報がまとめて操作履歴150に書き出され、メモリ10Eに保持されたログ情報が削除される(S309)。なお、ログ情報がメモリ10Eに保持される度に、ログ情報を操作履歴150に書き出し、所定のタイミングでメモリ10Eに保持されたログ情報を削除してもよい。
【００５９】
図5、図6に示す処理手順は一例である。例えば、新ログ情報に一致する既出力ログ情報がない場合、リストを更新し、新ログ情報をメモリ10Eに保持することなく、新ログ情報を操作履歴150に書き出して、処理をステップS201（またはS301）に戻してもよい。あるいは、ステップS206（またはS306）の判定前に新ログ情報を操作履歴150に書き出し、その後、判定を行って、新ログ情報に一致する既出力ログ情報がある場合は操作履歴150に書き出した新ログ情報を削除してもよい。
【００６０】
図7によりプロセス情報が格納されたリストの一例を示す。リストには、プロセス名、ハッシュ値、接続先のIPアドレス、接続先のポート番号、接続の許可/禁止、起動の許可/禁止などが記録される。所定時間の経過後、リスト内の全データを削除しても構わないし、削除しなくても構わない。なお、図7は一例であり、リストに保持する情報の種類と数は図7に限定されるものではない。
【００６１】
このように、リストに基づき、ログ情報を出力する（操作履歴150に書き出す）か否かが制御され、ネットワークアクセス制御やプログラム起動制御などに関する重複するログ情報の出力が抑制される。その結果、例えば、ネットワークアクセスが禁止されている場合のリトライによるログ情報を操作履歴150から削減することができる。つまり、重複するログ情報を特定し除去することで、操作履歴150の情報量を軽減することが可能である

。
【００６２】
　また、トレーニング期間中、リストを維持することができる。つまり、制御プログラム110は、クライアント10の再起動時や電源オフ時にリストを記憶装置10Bの所定領域に退避し、クライアント10が起動後、当該リストをメモリ10Eに復帰させる。リストを維持すれば、トレーニング期間中に取得される操作履歴150における、プロセスの起動や通信ログの重複する取得を防いで、操作履歴150の情報量をさらに軽減することができる。
【００６３】
　また、ホワイトリスト120やACLによって許可されていないプロセスの起動や通信があった場合、その操作や通信が禁止されていることを示す警告メッセージは、例えば、そのログ情報が出力された時点でディスプレイに表示される。実施例においては、重複するログ情報は特定され除去されて、警告メッセージが繰り返し表示されることがない。その結果、クライアント10を操作するユーザの作業に与える影響を軽減することができる。
【００６４】
　また、本実施例の手法は、ホワイトリスト利用時に限らず、クライアント10にインストールされているプログラムの資産管理を実施する際など、重複する情報が取得される場合にも利用可能である。
【００６５】
［ホワイトリスト制御処理］
●処理の概要
　検知プログラム112は、グローバルフック（APIフック、フィルタドライバ）などを用いて、クライアント10におけるプログラムの起動を検知し、プログラムの起動を検知すると識別プログラム110を呼び出す。識別プログラム110は、起動されるプログラムのプログラム情報を取得して、当該プログラムが昇格基準ルール130を満たすか否かを検証する。
【００６６】
　検証の結果、当該プログラムが昇格基準ルール130を満たすと判断された場合、制御プログラム113は、当該プログラムの起動を許可し、登録プログラム111を呼び出す。登録プログラム111は、識別プログラム110から当該プログラムのプログラム情報を受け取り、当該プログラムをホワイトリスト120に登録する。
【００６７】
　なお、起動されるプログラムが昇格基準ルール130を満たすとしても、当該プログラムがブラックリスト140に登録されている場合、制御プログラム113は登録プログラム111に当該プログラムの登録を実行させない。つまり、当該プログラムの起動・実行の禁止が維持される。
【００６８】
　次に、登録プログラム111は、ホワイトリスト120に登録したプログラムに「昇格権限」を与える。昇格権限の有無は、例えば、ホワイトリスト120の昇格権限フラグに設定する。あるいは、ホワイトリスト120に対応するテーブルを記憶装置10Bに格納し、当該テーブルの各レコードに昇格権限の有無を登録してもよい。昇格権限は、以下のように定義される権限である。
【００６９】
　昇格権限を有するプログラム（親プログラム）が何らかのプログラム（子プログラム）を生成した場合、子プログラムは無条件にホワイトリスト120に登録される。そして、親プログラムが子プログラムを起動した場合、子プログラムにも昇格権限が与えられる。この昇格権限に関する登録処理を、図1に示す構成が行う場合、以下のような処理になる。
【００７０】
　検知プログラム112は、親プログラムの挙動をグローバルフックなどを用いて監視する。検知プログラム112は、親プログラムによる子プログラムの生成を検知すると、識別プログラム110に子プログラムのプログラム情報を取得させる。識別プログラム110は、取得したプログラム情報を登録プログラム111に渡す。登録プログラム111は、受け取ったプロ

グラム情報に基づきホワイトリスト120に追加するレコードのデータを作成し、作成したデータをホワイトリスト120に追加する。

【００７１】

さらに、検知プログラム112は、親プログラムによる子プログラムの起動を監視する。検知プログラム112は、親プログラムによる子プログラムの起動を検知すると、登録プログラム111を呼び出す。呼び出された登録プログラム111は、子プログラムの昇格権限フラグに「有」を設定する。また、制御プログラム113は、子プログラムの起動を許可する。

【００７２】

昇格権限を利用すれば、例えばソフトウェアのセキュリティパッチなど、子プログラムを生成する挙動を有するプログラムが生成したプログラムを自動的にホワイトリスト120に追加することができる。言い換えれば、生成される子プログラムをホワイトリスト120に登録する、ホワイトリストの更新にかかる作業を軽減することができる。

【００７３】

なお、昇格権限に関する処理は、親プログラムから子プログラムを生成した場合に限らず、親プログラムに対する変更（例えば、リネーム）、または、子プログラムに対する変更の場合でも可能である。

【００７４】

また、昇格基準ルール130に「プログラムに正当なディジタル署名が付加されているかを判定する」を適用する。こうすれば、信頼できる発行元が発行したプログラムであり、マルウェアのような悪意のあるプログラムではないことを操作者の意識を介さずに判断可能である。なお、正当なディジタル署名が付加されているか否かは、例えばWindows（登録商標）アプリケーションプログラミングインタフェース(API)などを用いる方法などがある。

【００７５】

また、昇格基準ルール130を満たさないプログラムと判断された場合、制御プログラム113は一般的なホワイトリスト制御を行う。つまり、識別プログラム110は、当該プログラムが昇格基準ルール130を満たさないと判断すると、当該プログラムがホワイトリスト120に登録されているか否かを判定する。制御プログラム113は、当該プログラムがホワイトリスト120に登録されていると判定されると、昇格権限を有するか否か判定し、当該プログラムの起動を許可する。また、当該プログラムがホワイトリスト120に登録されていないと判定されると当該プログラムの起動をグローバルフックなどを用いて阻止する。

【００７６】

また、識別プログラム110により、起動されるプログラムがブラックリスト140に登録されているか否かを判定する。そして、ブラックリスト140に登録されている場合は当該プログラムの起動を阻止し、ブラックリスト140に登録されていない場合は当該プログラムの起動を許可する方式の採用も可能である。

【００７７】

●インストーラの自動登録

上記では、既にホワイトリストマスタ220（ホワイトリスト120）に登録されているプログラムが昇格基準ルール230を満たす例について説明した。しかし、プログラムに設定情報などを付加してインストーラを動的（実行ファイルに他のファイルを結合する）に作成するシステムにおいては、上記の条件（例えば、ディジタル署名による判断など）を満たすことが困難な場合がある。そのような動的に作成されるインストーラ（またはアップデータ）を自動的にホワイトリストマスタ220に登録する必要がある。

【００７８】

図8のフローチャートにより管理コンソール210から制御プログラム113のインストーラを作成する例を説明する。なお、アップデータの作成の場合も同様の処理になる。

【００７９】

サーバ20の操作者の指示により、管理コンソール210が起動され、制御プログラム113のインストーラが作成される(S801)。管理コンソール210は、インストーラを作成するとと

もに、ホワイトリストマスタ220に当該インストーラの情報が存在するか否かを判定し(S802)、当該インストーラの情報が既に存在する場合はインストーラ作成処理を終了する。

【００８０】

また、ホワイトリストマスタ220に当該インストーラの情報が存在しない場合、管理コンソール210は、当該インストーラの情報をホワイトリストマスタ220に登録する(S803)。そして、定期的（例えば、一時間置きや一日置き）にホワイトリストマスタ220のデータをクライアント10へ送信する(S804)。これにより、クライアント10のホワイトリスト120が更新され、当該クライアントにおける当該インストーラの起動が許可される。

【００８１】

ステップS802以降の処理は、管理コンソール210の起動に特定の起動オプションを与えることで、指定されたプログラム（ファイル）をホワイトリストマスタ220に登録する構成も可能である。特定の起動オプションは、例えば「-register c:\sample\sample.exe」である。

【００８２】

上記では、ホワイトリストマスタ220に未登録のプログラム（インストーラやアップデータをホワイトリストマスタ220に登録する例を説明した。ホワイトリストマスタ220に登録されていないパッケージをホワイトリストマスタ220に登録する場合も上記と同様の処理になる。また、図2に示すサーバ20が存在しない構成の場合、管理コンソール210の処理もクライアント10上で実行されることは言うまでもない。

【００８３】

また、上記ではホワイトリスト制御の一例として、プロセスの起動を説明したが、ネットワークアクセス等の所定の動作について、ホワイトリスト制御することができる。その場合、ホワイトリスト120には、接続先IPアドレス、接続先ポート番号などが含まれる。

【００８４】

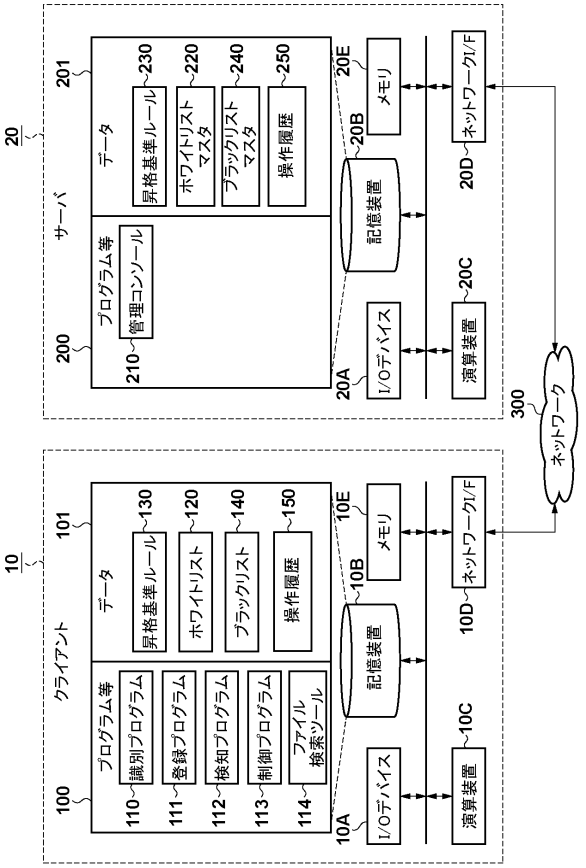
また、プログラムの起動ログを操作履歴150や250に保存し、当該プログラムが所定期間（例えば一月）起動されず、ホワイトリスト120やホワイトリストマスタ220に当該プログラムの情報が含まれる場合、その情報を自動的に削除することもできる。

【００８５】

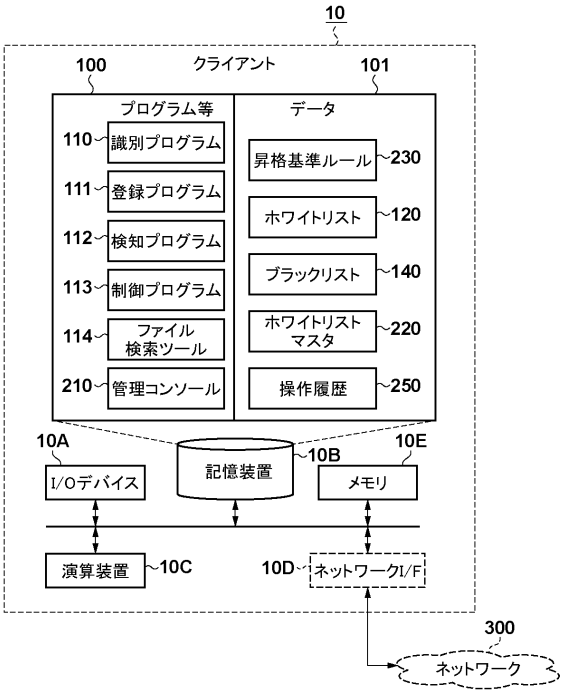
［その他の実施例］

また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステムあるいは装置のコンピュータ（又はCPUやMPU等）がプログラムを読み出して実行する処理である。

【図１】



【図２】



【図３】

120

プログラム名	ハッシュ値	バージョン	ファイルサイズ	昇格権限	接続先IP	接続先ポート
lexplore.exe	72AE6B5FDA794D2	1.0.0.1	56.3KB	無	xxx.111.111.111	80
svchost.exe	1A8C6D902A1200B	2.1	1.43MB	無	xxx.111.111.112	80
dwn.exe	A8C6D902A1200B9	10.7.1	321.8KB	有		
xcel.exe	F41B4C7361164DD0	4.0	214KB	無	xxx.111.111.114	8080
inword.exe	5BC866A3F1C29B8	1.10.2	2.1MB	無	xxx.111.111.115	80
⋮	⋮	⋮	⋮	⋮	⋮	⋮

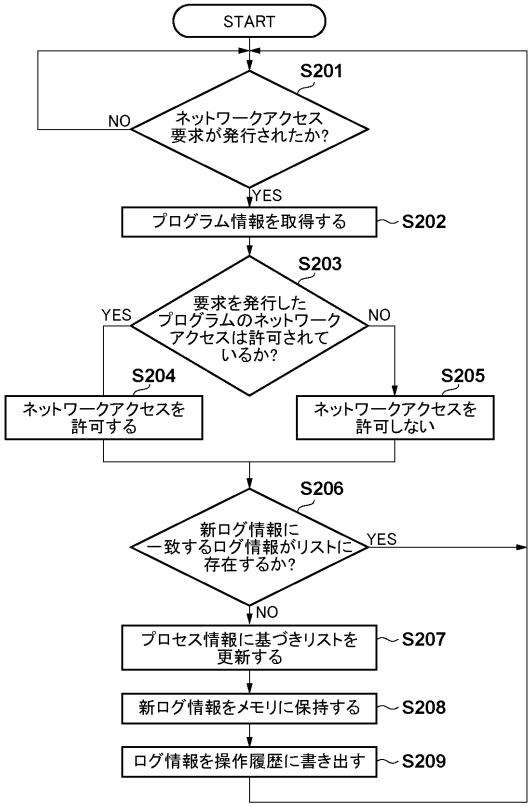
【図４】

220

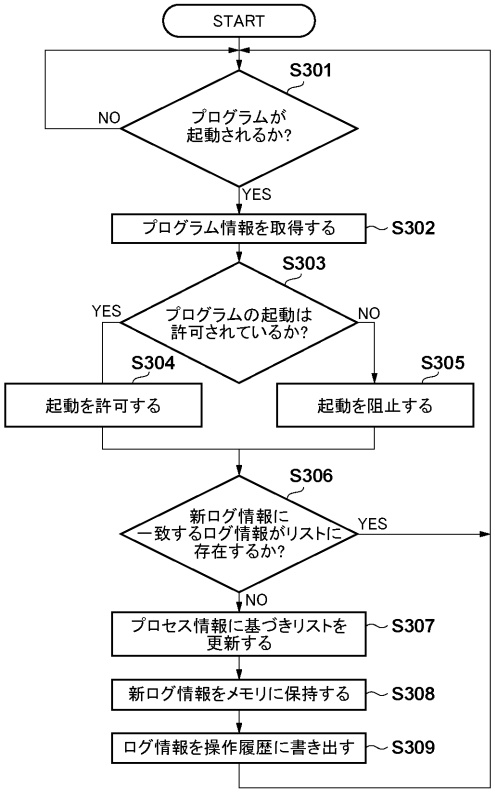
PC001	ホワイトリスト-001
PC002	ホワイトリスト-002
PC003	ホワイトリスト-003
⋮	⋮

プロセス名	ハッシュ値	接続先IP	接続先ポート	登録日時	最終起動日時
lexplore.exe	72AE6B5FDA794D2	xxx.111.111.111	80	2012/3/4 10:00	2012/3/4 11:00
svchost.exe	1A8C6D902A1200B	xxx.111.111.112	80	2012/3/5 03:00	2012/3/5 10:00
dwn.exe	A8C6D902A1200B9			2012/3/13 10:00	2012/3/14 10:00
xcel.exe	F41B4C7361164DD0	xxx.111.111.114	8080	2012/3/21 11:00	2012/3/26 10:00
inword.exe	5BC866A3F1C29B8	xxx.111.111.115	80	2012/3/24 20:00	2012/4/1 10:00

【図５】



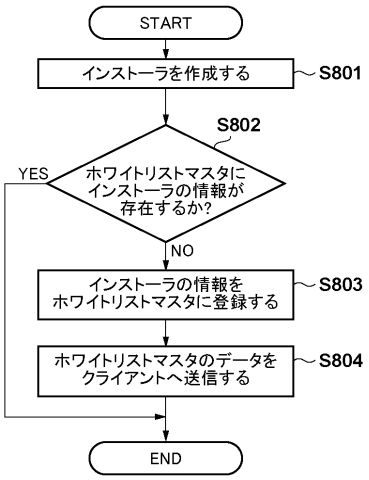
【図６】



【図７】

プロセス名	ハッシュ値	接続先IP	接続先ポート	起動	接続
lexplore.exe	72AE6B5FDA794D2	xxx.111.111.111	80		許可
svchost.exe	1A8C6D902A1200B	xxx.111.111.112	80		禁止
dwn.exe	A8C6D902A1200B9			禁止	
xcel.exe	F41B4C736164DD0	xxx.111.111.114	8080		禁止
inword.exe	5BC866A3F1C29B8	xxx.111.111.115	80		禁止

【図８】



フロントページの続き

(72)発明者　関口　あずさ
埼玉県秩父市下影森１２４８番地　キヤノン電子株式会社内
(72)発明者　佐藤　智規
埼玉県秩父市下影森１２４８番地　キヤノン電子株式会社内

審査官　伏本　正典

(56)参考文献　特開２０１１－１４１８０６（ＪＰ，Ａ）
特開２００７－２８００９６（ＪＰ，Ａ）
特開２０００－２９３２１９（ＪＰ，Ａ）
特開２００９－２５９１６０（ＪＰ，Ａ）
特開平０９－１１４７０８（ＪＰ，Ａ）
特開平０４－１５７５９９（ＪＰ，Ａ）
特開２００２－１５７１２２（ＪＰ，Ａ）
特開２０１０－００９１８６（ＪＰ，Ａ）
特開２００３－０５７０７６（ＪＰ，Ａ）
特開２０１０－０７２９８４（ＪＰ，Ａ）
米国特許出願公開第２００９／００８３８５２（ＵＳ，Ａ１）
坂上　行男，セキュリティ対策ソフトウェア　ＷｈｉｔｅＳｈｉｅｌｄ，明電時報，株式会社明電舎，２００９年１０月２６日，Ｎｏ．４，１０～１３頁

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　２１／１２
Ｇ０６Ｆ　２１／５５
Ｇ０６Ｆ　１１／３４