(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6530786号
(P6530786)
(45)発行日　令和1年6月12日(2019.6.12)　(24)登録日　令和1年5月24日(2019.5.24)

(51)Int.Cl.　　F I
G06F 16/906 (2019.01)　G06F 16/906
G06F 21/12 (2013.01)　G06F 21/12　380
G06F 21/55 (2013.01)　G06F 21/55

請求項の数 12　外国語出願　(全 21 頁)

(21)出願番号　特願2017-126120(P2017-126120)
(22)出願日　平成29年6月28日(2017.6.28)
(65)公開番号　特開2018-63694(P2018-63694A)
(43)公開日　平成30年4月19日(2018.4.19)
審査請求日　平成29年10月2日(2017.10.2)
(31)優先権主張番号　2016139468
(32)優先日　平成28年10月10日(2016.10.10)
(33)優先権主張国　ロシア(RU)
(31)優先権主張番号　15/475,885
(32)優先日　平成29年3月31日(2017.3.31)
(33)優先権主張国　米国(US)

(73)特許権者　515348585
エーオー　カスペルスキー　ラボ
ＡＯ　Ｋａｓｐｅｒｓｋｙ　Ｌａｂ
ロシア国、１２５２１２　モスクワ、レニングラドスコ　ショス　３９エー／３
(74)代理人　110001139
ＳＫ特許業務法人
(74)代理人　100130328
弁理士　奥野　彰彦
(74)代理人　100130672
弁理士　伊藤　寛之
(72)発明者　オレグ　ブイ．　キュプリーヴ
ロシア　１２５２１２　モスクワ，　レニングラドスコ　ショス　３９エー／３，
エーオー　カスペルスキー　ラボ
最終頁に続く

(54)【発明の名称】Ｗｅｂページの悪意のある要素を検出するシステム及び方法

(57)【特許請求の範囲】
【請求項１】
Webページの悪意のある要素を検出する方法であって、
プロセッサがWebページの要素に関するデータを取得するステップと、
前記プロセッサが前記Webページの要素を特徴付ける少なくとも１つのN次元ベクトルを生成するステップと、
前記プロセッサが既知の悪意のあるWebページの要素の統計モデルをデータベース上で検索するステップと、
前記プロセッサが前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタとを、前記N次元ベクトルと前記統計モデルの全てのクラスタの中心との距離を測定することによって比較するステップと、
前記比較の結果に基づいて、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタのN次元ベクトルであって前記クラスタの中心から最も離れたベクトルとの間のN次元空間における近接度が第１の選択された閾値以下である場合に、前記プロセッサが前記Webページの少なくとも１つの悪意のある要素を識別するステップと、
を備え、
前記プロセッサが前記Webページの要素に関するデータを取得するステップは、前記Webページに埋め込まれたスクリプトからデータを取得することを含み、前記スクリプトは、少なくとも前記Webページの少なくとも１つの要素に関するデータを収集するためにユーザデバイスで実行されるよう構成される、方法。

【請求項２】
前記プロセッサが前記取得されたデータのハッシュを計算するステップと、
前記プロセッサが前記Webページの既知の悪意のある要素の少なくとも１つのハッシュを取得するステップと、
前記プロセッサが前記計算されたハッシュを前記少なくとも１つのハッシュと比較して前記少なくとも１つの要素が悪意のあるものであるかどうかを判定するステップと、
をさらに備える、請求項１に記載の方法。
【請求項３】
前記比較の結果に基づき、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタの中心との間のN次元空間における距離が前記クラスタの半径以下である場合にも、前記プロセッサが前記Webページの少なくとも１つの悪意のある要素を識別する、請求項１に記載の方法。
【請求項４】
前記比較の結果に基づき、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタの中心との間のN次元空間における近接度が第２の選択された閾値以下である場合にも、前記プロセッサが前記Webページの少なくとも１つの悪意のある要素を識別する、請求項１に記載の方法。
【請求項５】
Webページの悪意のある要素を検出するシステムであって、
少なくとも１つのプロセッサを備え、
前記プロセッサは、
Webページの要素に関するデータを取得し、
前記Webページの要素を特徴付ける少なくとも１つのN次元ベクトルを生成し、
既知の悪意のあるWebページの要素の統計モデルを検索し、
前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタとを、前記N次元ベクトルと前記統計モデルの全てのクラスタの中心との距離を測定することによって前記プロセッサにより比較し、
前記比較の結果に基づいて、前記少なくとも１つのN次元ベクトルと前記統計モデルの前記クラスタのN次元ベクトルであって前記クラスタの中心から最も離れたベクトルとの間のN次元空間における近接度が第１の選択された閾値以下である場合に、前記Webページの少なくとも１つの悪意のある要素を識別するよう構成され、
前記Webページの要素に関するデータを取得することは、前記Webページに埋め込まれたスクリプトからデータを取得することを含み、前記スクリプトは、少なくとも前記Webページの少なくとも１つの要素に関するデータを収集するためにユーザデバイスで実行されるよう構成される、システム。
【請求項６】
前記プロセッサはさらに、
前記取得されたデータのハッシュをプロセッサにより計算し、
前記Webページの既知の悪意のある要素の少なくとも１つのハッシュを取得し、
前記計算されたハッシュを前記少なくとも１つのハッシュと比較して前記少なくとも１つの要素が悪意のあるものであるかどうかを判定するよう構成される、請求項５に記載のシステム。
【請求項７】
前記比較の結果に基づき、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタの中心との間のN次元空間における距離が前記クラスタの半径以下である場合にも、前記Webページの少なくとも１つの悪意のある要素を識別する、請求項５に記載のシステム。
【請求項８】
前記比較の結果に基づき、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタの中心との間のN次元空間における近接度が第２の選択された閾値以下である場合

にも、前記Webページの少なくとも１つの悪意のある要素を識別する、請求項５に記載のシステム。

【請求項９】
Webページの悪意のある要素を検出するためのコンピュータ実行可能命令が格納された非一時的コンピュータ可読媒体であって、
Webページの要素に関するデータを取得させる命令と、
前記Webページの要素を特徴付ける少なくとも１つのN次元ベクトルを生成させる命令と、
既知の悪意のあるWebページの要素の統計モデルを検索させる命令と、
前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタとを、前記要素のN次元ベクトルと前記統計モデルの全てのクラスタの中心との距離を測定することで比較させる命令と、
前記比較の結果に基づいて、前記少なくとも１つのN次元ベクトルと前記統計モデルの前記クラスタのN次元ベクトルであって前記クラスタの中心から最も離れたベクトルとの間のN次元空間における近接度が第１の選択された閾値以下である場合に、前記Webページの少なくとも１つの悪意のある要素を識別させる命令と、
を備え、
前記Webページの要素に関するデータを取得させる命令は、前記Webページに埋め込まれたスクリプトからデータを取得させることを含み、前記スクリプトは、少なくとも前記Webページの少なくとも１つの要素に関するデータを収集するためにユーザデバイスで実行されるよう構成される、非一時的コンピュータ可読媒体。
【請求項１０】
前記取得されたデータのハッシュを計算させる命令と、
前記Webページの既知の悪意のある要素の少なくとも１つのハッシュを取得させる命令と、
前記計算されたハッシュを前記少なくとも１つのハッシュと比較して前記少なくとも１つの要素が悪意のあるものであるかどうかを判定させる命令と、
をさらに備える、請求項９に記載の非一時的コンピュータ可読媒体。
【請求項１１】
前記比較の結果に基づき、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタの中心との間のN次元空間における距離が前記クラスタの半径以下である場合にも、前記Webページの少なくとも１つの悪意のある要素を識別させる、請求項９に記載の非一時的コンピュータ可読媒体。
【請求項１２】
前記比較の結果に基づき、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタの中心との間のN次元空間における近接度が第２の選択された閾値以下である場合にも、前記Webページの少なくとも１つの悪意のある要素を前記プロセッサにより識別させる、請求項９に記載の非一時的コンピュータ可読媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、一般に、サイバーセキュリティの分野に関し、より具体的には、ウェブページの悪意のある要素を検出するシステム及び方法に関する。
【背景技術】
【０００２】
近年、銀行などの金融機関は、銀行業務の一環としてWebバンキングサービス（インターネットバンキング）を積極的に採用している。Webバンキングは、リモートバンキングサービス技術の一般的な用語であり、いつでも、インターネットにアクセス可能な任意のコンピュータから口座にアクセスし、（リモートバンキングサービス技術により）操作す

ることができる。これらの操作を実行するには、典型的にはウェブクライアント（ブラウザ等）が使用される。

【０００３】

このような技術の普及は、自然とハッカー、とりわけリモートサービスシステムのユーザの口座から資金を盗むことに関心を持つ者の興味をかきたてる。Webバンキングのユーザに対する一般的な攻撃の一つは、ユーザに対して表示されるWebページのコンテンツの代わりに悪意のあるソフトウェアを配置することである。悪意のあるソフトウェアは、Webページ内にHTMLコードを埋め込む。この攻撃は、しばしば「マン・イン・ザ・ブラウザ」又は「Webコードインジェクション」攻撃と呼ばれる。この攻撃は、例えばトロイの木馬アプリケーションにより開始される。トロイの木馬は、被害者のブラウザに悪意のある拡張子のファイルをインストールし、再起動したときに当該ファイルをブラウザで実行する。その後、特定のWebサイト（大抵は銀行サイト）にルーティングされるユーザのトラフィックを傍受する。そして、ユーザに表示されるWebページを（ロード中又はオープン中に）改ざんすることで、Webページの特定の要素の外観を変更したり、被害者の認証データを盗み出したり、ユーザが送金した資金を第三者の口座に転送したりするのである。

【０００４】

現存する解決策は、外部コードをWebページに導入する攻撃に対し、ネット上においてユーザの安全な作業を増やすことを目指すものである。しかしながら、これらの解決策は、悪意のあるソフトウェアによってWebページが改ざんされたかどうかを効果的に判断はせず、追加のソフトウェアをインストールしない限り、ユーザ側のWebページのバージョンの異常な要素を特定することもない。同時に、各種のセキュリティクライアント、シンクライアント（ライトエージェント）、その他のアンチウイルス手段等の追加のソフトウェアは、必ずしもユーザ側でインストール可能なわけではなく、最終的には、アンチウイルスアプリケーションの動作に過誤を発生させる。例えば、第１の種類の過誤は、データ転送チャネルを捕捉して転送される全ての情報にアクセスするため、コンピューティングシステムに「マン・イン・ザ・ブラウザ」タイプの攻撃を許してしまうものであり、第２の種類の過誤は、ユーザ側での正当なWebページの変更が異常であると誤って判断されることである。

【発明の概要】

【０００５】

Webページの悪意のある要素を検出するシステム及び方法が開示される。１つの例示的な方法は、テストされたWebページの要素に関するデータを取得するステップと、前記テストされたWebページの要素を特徴付ける少なくとも１つのN次元ベクトルを生成するステップと、既知の悪意のあるWebページの要素の統計モデルを検索するステップと、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタとを、前記N次元ベクトルと前記統計モデルの全てのクラスタの中心との距離を測定することによって比較するステップと、前記比較の結果に基づいて、前記テストされたWebページの少なくとも１つの悪意のある要素を識別するステップとを備える。

１つの例示的な態様では、前記テストされたWebページの要素に関するデータを取得するステップは、前記Webページに埋め込まれたスクリプトからデータを取得することを含み、前記スクリプトは、少なくとも前記Webページの少なくとも１つの要素に関するデータを収集するためにユーザデバイスで実行されるよう構成される。

【０００６】

１つの例示的な態様では、前記方法は、前記取得されたデータのハッシュを計算させる命令と、前記Webページの既知の悪意のある要素の少なくとも１つのハッシュを取得させる命令と、前記計算されたハッシュを前記少なくとも１つのハッシュと比較して前記少なくとも１つの要素が悪意のあるものであるかどうかを判定させる命令と、をさらに備える。

【０００７】

１つの例示的な態様では、前記比較の結果に基づき、前記Webページの少なくとも１つ

の要素の前記少なくとも１つのN次元ベクトルと前記Webページの統計モデルのクラスタの中心との間のN次元空間における距離が前記クラスタの半径以下である場合に、前記テストされたWebページの少なくとも１つの悪意のある要素を識別するステップを備える。

【０００８】

１つの例示的な態様では、前記比較の結果に基づき、前記Webページの少なくとも１つの要素の前記少なくとも１つのN次元ベクトルと前記Webページの統計モデルのクラスタの中心との間のN次元空間における近接度が第１の選択された閾値以下である場合に、前記テストされたWebページの少なくとも１つの悪意のある要素を識別するステップを備える。

【０００９】

１つの例示的な態様では、前記比較の結果に基づき、前記Webページの少なくとも１つの要素の少なくとも１つのN次元ベクトルと前記Webページの前記統計モデルの前記クラスタのN次元ベクトルであって前記クラスタの中心から最も離れたベクトルとの間のN次元空間における近接度が第２の選択された閾値以下である場合に、前記テストされたWebページの少なくとも１つの悪意のある要素を識別するステップを備える。

【００１０】

１つの例示的な態様では、Webページの悪意のある要素を検出するシステムは、少なくとも１つのプロセッサを備え、前記プロセッサは、テストされたWebページの要素に関するデータを取得し、前記テストされたWebページの要素を特徴付ける少なくとも１つのN次元ベクトルを生成し、既知の悪意のあるWebページの要素の統計モデルを検索し、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタとを、前記要素のN次元ベクトルと前記統計モデルの全てのクラスタの中心との距離を測定することで比較し、前記比較の結果に基づいて、前記テストされたWebページの少なくとも１つの悪意のある要素を識別するよう構成される。

【００１１】

１つの例示的な態様では、Webページの悪意のある要素を検出するためのコンピュータ実行可能命令が格納された非一時的コンピュータ可読媒体が、テストされたWebページの要素に関するデータを取得させる命令と、前記テストされたWebページの要素を特徴付ける少なくとも１つのN次元ベクトルを生成させる命令と、既知の悪意のあるWebページの要素の統計モデルを検索させる命令と、前記少なくとも１つのN次元ベクトルと前記統計モデルのクラスタとを、前記要素のN次元ベクトルと前記統計モデルの全てのクラスタの中心との距離を測定することで比較させる命令と、前記比較の結果に基づいて、前記テストされたWebページの少なくとも１つの悪意のある要素を識別させる命令と、を備える。

【００１２】

１つの例示的な態様において、本発明の例示的な態様の上記簡略化した概要は、本発明の基本的な理解を提供する役割を果たす。この概要は、全ての熟考された形態を含んだ広範な概要ではなく、全ての態様の重要な要素を特定することも、本発明の任意の又は全ての態様の範囲を定めることも意図していない。本発明の１つ以上の態様は、前述の目的を達成するため、特に請求の範囲において挙げられている特徴を含んでいる。

【図面の簡単な説明】

【００１３】

添付図面は、本明細書に組み込まれ、本明細書の一部を構成する。添付図面は、本願の１つ以上の例示的な態様を示しており、発明の詳細な説明とともに、それらの態様の原理及び実装を説明する役割を果たす。

【図１】図１は、本発明の態様に係る、Webページの異常且つ悪意のある要素を検出するための例示的なシステムを示す図である。

【図２】図２は、本発明の態様に係るＮ次元空間の例を、統計モデル及びクラスタのメトリックとともに示す図である。

【図３】図３は、本発明の態様に係る、Webページの異常且つ悪意のある要素を検出するための例示的な方法を示す図である。

【図４】図４ａ、図４ｂ、図４ｃは、本発明の態様に係る、統計モデルの視覚化の例を示す図である。
【図５】図５は、Webページの異常且つ悪意のある要素を検出するシステム及び方法の態様を実施可能な汎用コンピュータシステムの一例を示す図である。
【発明を実施するための形態】
【００１４】
以下では、Webページの異常且つ悪意のある要素を検出するためのシステム、方法及びコンピュータプログラム製品についての、発明の例示的な態様を説明する。当業者であれば、以下の記載はあくまでも例示であり、あらゆる制限をも意図するものではないことが理解されよう。本発明の利益を享受する当業者であれば、他の態様も容易に示唆されることであろう。添付の図面では、その詳細を示すための参照符号が付されている。図面と以下の記載において、同一の又はそれに相当する項目については、同一の参照符号を用いるものとする。
【００１５】
以下の用語が、本発明の例示的な態様を説明する際に使用される。
【００１６】
Webページは、Webクライアント（ブラウザ）によって処理されるようWebサーバによって生成されたデータ含み、ハイパーテキストマークアップ言語（HTML、XHTML、XML、WML、VML、PGML、SVG、XBRL等）やスクリプト言語（JScript、JavaScript、ActionScript、Tcl、Lua、Perl、PHP、Python、REBOL、Ruby等）を用いて構成される。
【００１７】
本明細書において、コンテンツはWebページのコンテンツのことを示すこととする。
【００１８】
スクリプトは、スクリプト言語で書かれた実行中のプロシージャを含み、厳密に決定されたWebページを表示する際に到着する要求によって、サーバ側又はクライアント側で実行されてもよい。
【００１９】
インラインスクリプトは、その実行形式のコード（body）がWebページのコンテンツの一部となったスクリプトを含む。例えば、インラインスクリプトはタグ<script> </script>の間に配置される。
【００２０】
タグ（マーカ）は、ハイパーテキストマークアップ言語（HTML）の特別な構成を含む。タグは、山カッコ<name_tag>で囲まれたテキストを構成する。各タグは、タグ及びそれに続くコンテンツをどのように表示するかの特定の命令をブラウザにもたらす。特定の場合において、タグは、タグを明確にし、タグの可能性を拡張してより柔軟な制御を可能にする属性を有する（例えば、コンテナタグの中身）。例えば、<script src = "URL"> ... </ script>などである。src属性は、スクリプトのbodyの場所を示す。
【００２１】
コンテナタグは、開始タグおよび終了タグを有するペアタグを含む。コンテナタグは、テキストとハイパーテキスト言語の他の要素の両方を含む。
【００２２】
Webページの要素（マークアップ言語の要素）は、開始タグ及び終了タグ（ある場合、例えば
タグの場合には、開始タグと終了タグは一致する。）の組み合わせと、タグ間のコンテンツを含むことができる。Webページの要素の全体は、Webページのコンテンツを形成する。要素には、対応するタグの名前によって区別される少なくとも次の種類のものが存在する：ハイパーリンク、テキストブロック、テキストフォーマット、リスト、オブジェクト（例えば、メディアファイル、アプレット、スクリプト、ネイティブコード等）、イメージ、イメージマップ、表、フォーム、文字等。
【００２３】
要素のN次元ベクトルには、順序付けられたn個の実数の組が含まれ、その数にはベクト

ルの座標が含まれる。ベクトルの座標の個数は、ベクトルの次元数として知られている。座標は、N次元空間における、Webページの対応する要素（スクリプトなど）又はWebページの同じ種類の要素グループ（フォームの要素など）の位置を定める（図２は２次元空間の例を示す）。ベクトルは、要素又は要素グループのコンテンツに関する情報を変換することによって得られる。ベクトルは、要素又は要素グループのコンテンツに関する特定の情報を表す。一例では、各座標が要素のコンテンツの特性の１つを表す。例えば、１つの座標がスクリプト内の演算子の数を示し、別の座標は評価演算子の数を示す。数値に、要素のコンテンツの文字列パラメータの辞書式順序、又は異なる要素の文字列パラメータ間のレーベンシュタイン距離を反映させることもできる。例えば図２は、ベクトル、特に、座標（1666,1889）及び（1686,1789）を有する２次元ベクトルの例を示す。

【００２４】

クラスタは、N次元空間において、厳密に定義された要素又は要素グループについてのベクトルの座標の許容値の集合を含有する。１つの例示的な態様によれば、ある選択された要素又は要素グループは、その要素のN次元ベクトルからあるクラスタの中心までの距離がそのクラスタの半径よりも小さい場合に、当該クラスタに割り当てられる。図２は、クラスタ２１０'の一例を示す。一例において、ある要素のN次元ベクトルとあるクラスタの要素のうち最も近いN次元ベクトルとの間の距離（図2の「d'」）の値が許容できる最大距離（距離[d']の閾値）よりも小さいか、又は、ある要素のN次元ベクトルからあるクラスタの中心までの距離（図2の「d」）の値がこのクラスタの半径より小さい場合、その要素は当該クラスタに割り当てられる。例えば、ベクトル（1666,1889）からクラスタの中心までの距離はクラスタの半径よりも小さいので、そのコンテンツがベクトルに表された要素又は要素グループは、当該クラスタに属することになる。一方、ベクトル（1686,1789）からクラスタの中心までの距離はクラスタの半径よりも大きく、最も近いN次元ベクトルまでの距離は閾値よりも大きいので、そのコンテンツがベクトルに表された要素又は要素グループは、当該クラスタに属さない。近接度を評価するための距離の種類には、直線距離、ユークリッド距離、ユークリッド距離の2乗、一般化されたミンコフスキー指数距離、チェビシェフ距離、マンハッタン距離等があるが、これらに限定されるわけではない。

【００２５】

近接度（類似度、類似度係数）は、Webページの要素の類似性を判定するための無次元パラメータを含む。近接度を定めるために用いる方法（基準）には、落合係数、Jaccard係数、Sokal-Sneath係数、Kulczynski係数、対称なDice係数などが含まれるが、これらに限定されない。

【００２６】

クラスタの中心（重心）は、N次元空間における複数のN次元ベクトルの幾何平均位置である。１つのベクトルからなるクラスタの場合は、そのベクトルをクラスタの中心とすることができる。

【００２７】

クラスタの半径（図2の「R」）は、クラスタ内に存在するN次元ベクトルのクラスタ中心からの最大距離である。

【００２８】

クラスタリングには、階層型（凝集型及び分割型）と非階層型を含む様々な既知のアルゴリズム及び手法を使用することができる。

【００２９】

Webページの要素の統計モデル（Webページの要素のモデル）は、１種類の要素又は１種類の要素グループについてのクラスタ２１０の集合を含む。例えば、Webページのスクリプトの統計モデルや、Webページのフォームの統計モデルである。図２に示すように、Webページの要素の統計モデルは２２０と表される。１つのクラスタで構成されるモデルの場合、そのクラスタが要素のモデルとなる。

【００３０】

Webページの統計モデル（Webページのモデル）は、Webページの全ての種類の要素及び／又は要素グループ（異なる種類の要素を備えた要素グループを含む）のクラスタの集合を含む。例えば、認証ページの統計モデルである。言い換えると、Webページの統計モデル２３０は、Webページの要素のモデル２２０の集合を含む。同様に、Webサイトの統計モデルは、Webサイトの全種類の要素及び／又は要素グループのクラスタの集合を含む。つまり、Webサイトの統計モデル（図示せず）は、Webページのモデル２３０の集合を含むことになる。

【００３１】

Webページの悪意のある要素の統計モデルは、同一及び／又は異なる種類の既知の悪意のある要素のクラスタ２１０のグループ、又は、同一及び／又は異なる種類の要素グループを含む。例えば、悪意のあるスクリプトの統計モデル、Webページの悪意のあるフォームの統計モデル、Webページの悪意のあるスクリプト及びフォームの統計モデルである。単一のクラスタから構成されるモデルの場合、そのクラスタは悪意のある要素のモデルとなる。特定の種類のモデルの構築には、既知の悪意のある要素が用いられる。ここで、既知の悪意のある要素は、そのコンテンツがN次元ベクトルに変換された後クラスタ化され、モデルに結合される。それらの形式では、Webページの統計モデル及びWebページの悪意のある要素の統計モデルは互いに異なっていない。つまり、モデルを構築するために用いられる要素が異なっており、1番目のケースでは、モデルは潜在的に危険な要素から構築され、2番目のケースでは、モデルは既知の悪意のある要素から構築される。

【００３２】

Webページの異常要素には、Webページの要素であって、そのベクトルが特定のタイプの要素に対して構築されたいずれのWebページの統計モデルのクラスタにも関係しないもの、又は、統計的有意性が閾値以下のものが含まれる。

【００３３】

要素の統計的有意性は、モデル構築のために取得したWebページの総数又は所定のセクション（評価セクション）においてモデルの構築のために取得したWebページの数に対する、評価対象の要素がWebページのコンテンツ内で見つかった回数の比とする。ここで、セクションの範囲は、例えば要素の観測の開始時間のような特定の時間以降に、モデルの構築のため得られたWebページの数によって決定される。例えば、１００ページを取得し、評価対象の要素が３０回見つかった場合、統計的有意性は３０％である。

【００３４】

クラスタの統計的有意性は、モデル構築のために取得したWebページの総数又は所定のセクションにおいてモデルの構築のために取得したWebページの数に対する、Webページのコンテンツ内で評価され、そのベクトルがクラスタを形成する要素の数の比とする。ここで、セクションの範囲は、例えばクラスタの観測の開始時間のような特定の時間以降に、モデルの構築のために得られたWebページの数によって決定される。

【００３５】

要素又はクラスタの統計的有意性の値を、統計的有意性の閾値とすることができる。この値を超えると、当該要素又はクラスタ（及びクラスタの要素）は統計的に有意であるとみなされる。そして、クラスタの要素の統計的有意性の値が設定された閾値よりも低い場合、要素又はクラスタは異常であるとみなされる。

【００３６】

クラスタの形成には、階層的方法を用いることができ、例えば、凝集法により形成することができる。凝集法では、（距離の）最も近い要素のN次元ベクトルの組が取り出されてクラスタが生成され、又は、（距離の）最も近いクラスタの組が１つのクラスタに併合される。この方法を用いる場合、線形又はユークリッド距離や、一般化された指数関数的なMinkowski距離、Chebyshev距離又はManhattan距離を使用することができる。そして、最小の相互距離を有するベクトルの組は最も近いものとされ、クラスタは、その半径ができるだけ半径の閾値に近づくよう分割される。ここで、最も近づいた半径は、クラスタを分離する次の回における半径の閾値を超えるものであっても良い。別の場合、許容可能な

近接度が残っているクラスタ又はベクトルがなくなるまで、クラスタを分離することができる。この場合、許容可能な近接度は、設定された閾値を超えない値である。中心間の距離が最も短い２つのクラスタが最も近いクラスタである。

【００３７】

あるいは、分割法によりクラスタを形成することもできる。分割法では、相互距離が最大許容距離よりも小さいベクトルの集合によってクラスタが形成される。距離の最大許容度は閾値によって決定され、クラスタは、例えば、その半径が当該半径の閾値と同じかそれ未満になるまで分割される。

【００３８】

図１は、本発明の態様に係る、Webページの異常且つ悪意のある要素を検出するとともにWebページ１００の統計モデル及びWebページの要素の統計モデルを構築するために用いられる、異常且つ悪意のある要素を検出するための例示的なシステムを示す。システムは、Webクライアント１１０がインストールされたユーザデバイス１２０、Webサーバ１３０、コントロールサーバ１５０、データベース１６０を含むことができるが、これらに限定されるわけではない。

【００３９】

１つの例示的な態様では、WebブラウザなどのWebクライアント１１０をユーザのデバイス１２０にインストールすることができる。Webクライアント１１０は、Webサイトのコンテンツを要求、処理、操作及び表示するよう構成され、論理的に相互に関連するWebページ１００のセットを含んでいる。Webクライアント１１０は、例えば、Webサーバ１３０に対して指定されたユニフォーム・リソース・ロケータ（URL）アドレスを用いてリソースを取得する要求を送信し、一般にはWebサーバ１３０のWebページ１００又はWebページの要素とともに、応答を受信する。Webクライアント１１０からの要求に応じて、Webサーバ１３０は、準備されたWebページ１００を送信したり、動的にページを生成したりすることができる。本発明のWebサーバ１３０によってクライアントに送られる各Webページには、通常のコンテンツに加えて、スクリプトが追加されている。スクリプト１４０の機能には、少なくともWebクライアント側１１０においてそのスクリプト１４０を含むWebページ１００のデータ（Webページの要素又は要素グループに関する情報、１つの要素に関する情報、特定の場合には１つの要素のコンテンツの情報を含む）を収集することが含まれる。一例では、Webページ１００の要素に関する情報には、その要素のコンテンツが含まれている。クライアント側１００におけるWebページ１００の要素及びWebページ１００の要素のコンテンツは、サーバ側１３０におけるWebページの同じバージョンの要素及びWebページ１００のこれらの要素のコンテンツとは異なる場合がある。この理由は、Webクライアント側１１０のWebページの動的リフレッシュによるものか、又は、「マン・イン・ザ・ブラウザ」攻撃によるものである。

【００４０】

１つの例示的な態様では、コントロールサーバ１５０は、スクリプトによって収集されたWebページの要素又は要素グループに関する情報を受け取る。スクリプトは、収集されたデータを「生の」状態又は変換された状態のいずれかの状態で送信することができ、送信されたデータのフォーマットは、Webサーバ１３０がWebページ１００に追加したスクリプト１４０の機能によって決定される。すなわち、実行中のスクリプトは、Webページ１００の要素に関する厳密に定義された情報を厳密に定められた形式で送信し、スクリプトの機能により決定される。あるいは、スクリプトは、Webサーバ又はコントロールサーバに、クライアント側１１０においてスクリプトの開始が成功したことについてのデータを送信し、応答として、Webページ１００のどの要素を収集する必要があるか、どのような形式によって受信者（Webサーバ１３０又はコントロールサーバ１５０に直接）に情報を送信するかについての命令を受信する。１つの例示的な態様では、スクリプト１４０は中間ノードにおいて、例えば企業のプロキシサーバによって、Webページ１００に埋め込むこともできる。

【００４１】

データ変換の主な方法には、量子化、ソート、マージ（貼り付け）、グループ化、データセットの設定、テーブルへの値の挿入、算出値、データコーディング、正規化（スケーリング）等がある。

【００４２】

一例では、データは、データ変換の結果として情報の属性を取得する。

【００４３】

スクリプトの変換方法の1つには、抽象構文木の構築、重要なオペレータのみの受信者（Webサーバ１３０又は直接コントロールサーバ１５０）への転送（送信）、スクリプト１４０の設定により又は受信者からの命令により予め決定された構成が含まれる。

【００４４】

１つの例示的な態様では、スクリプト１４０によって収集された全てのデータをコントロールサーバ１５０に転送する。コントロールサーバ１５０は、Webクライアント１１０から直接データを取得するか、又はWebサーバ１３０を介してデータを取得するよう構成される。別の形態では、コントロールサーバ１５０をWebサーバ１３０と同じネットワーク内に存在させることができる。コントロールサーバ１５０によって収集されたデータは、Webページの統計モデル２３０（又はWebページの要素の統計モデル）を構築し、異常（又は悪意のある）要素を検出するために使用される。コントロールサーバ１５０は、スクリプト１４０によって収集されたデータをN次元ベクトルに変換し、得られたベクトルをデータベース１６０に格納する。特に、コントロールサーバは、要素のコンテンツから、ハッシングアルゴリズム（CRC、MD5、MD6、SHA1、SHA2、GOST R 34.11-2012など）の１つによってハッシュを算出する。

【００４５】

コントロールサーバ１５０の解析モジュールは、受信したベクトルからクラスタ２１０を形成し、異常な要素又は要素グループを検出する。なお、異常な要素又は要素グループのコンテンツは、受信したベクトルにより表されている。この機能は、N次元空間におけるN次元ベクトルと形成されたクラスタ２１０の相互比較によって実現される。

【００４６】

例示的な一態様では、データベース１６０は、構築されたモデル及びベクトル、そして既知の悪意のある要素のハッシュを格納するよう構成される。悪意のある要素のハッシュに関する記録は、外部ソース（すなわち、すでに算出されたハッシュ）からデータベースに与えられてもよい。また、ハッシュは、異常要素のアンチウイルススキャンの結果として検出されるか、又は、Webページの悪意のある要素のコピーを格納する悪意のあるソフトウェアのリポジトリ（図示せず）から選択された、既知の悪意のある要素のコンテンツから、コントロールサーバ１５０が算出しても良い。

【００４７】

本発明で開示されたシステムは、いくつかの方法、すなわち、Webページの統計モデル２３０を構築する方法、Webページの悪意のある要素の統計モデルを構築する方法、Webページの構築されたモデル２３０を用いてWebページ１００の悪意のある要素を検出する方法、Webページの要素の構築されたモデルを用いてWebページ１００の悪意のある要素を検出する方法、ハッシュによりWebページの悪質な要素１００を検出する方法を実行することができる。これらの方法を図３に示す。

【００４８】

図３に示すように、Webページの統計モデル２３０の例示的な構築方法は、以下のように実施される。ステップ３００において、ユーザは自身のデバイスからWebサイトへアクセスする。Webクライアント１１０は、Webサーバ１３０へ送信された要求により、Webサーバ１３０からサイトのWebページを取得するよう構成される。そのプロセスの間に、Webサーバ１３０（又は中間ノード）は、Webページにスクリプト１４０を追加する。ステップ３１０において、スクリプトをWebクライアント側１１０で実行し、Webページ１００に含まれるデータを収集する。スクリプト１４０によって収集されたデータには様々な情報が含まれる。特に、スクリプト１４０は、Webページの少なくとも１つの要素（スクリプ

ト、フォームなど）のコンテンツを収集することができる。スクリプト１４０によって収集されたデータは、必要に応じて変換される。データは、スクリプト１４０自体によって、又はコントロールサーバ１５０上の処理手段によって変換される。ステップ３２０において、収集されたデータを、少なくとも１つのN次元ベクトルに変更し、ステップ３３０において保存する。ステップ３５０において、少なくとも１つのベクトルから、少なくとも１つのクラスタ２１０を形成する。ステップ３６０において、形成された少なくとも１つのクラスタ２１０に基づいて、Webページの統計モデル２３０を生成する。

【００４９】

一例では、ステップ３０１において、取得したN次元ベクトルを保存したあと、別のWebクライアント１１０にWebページ１００を取得させ、ステップ３２０において、このWebページから収集したデータに基づいてN次元ベクトルを追加的に取得させ、この後にクラスタが形成されてもよい。

【００５０】

別の例では、クラスタ２１０を作成しモデル２３０を構築した後、ステップ３０２において、別のWebクライアント１１０によりWebページ１００を取得し、このWebページからスクリプト１４０により収集されたデータに基づいてN次元ベクトルを取得し、取得したN次元ベクトルに基づいて既に作成されていたクラスタ２１０を修正（リフレッシュ）する（それらの半径、中心／重心を変更する）か、新たなクラスタ２１０を作成する。その結果、Webページの統計モデル２３０は、（クラスタ２１０の修正により）精緻化され、（新たなクラスタ２１０の作成により）補完される。スクリプト１４０が収集するデータは、それ以前のバージョン（イテレーション）において収集したデータと異なっていてもよく、例えば、Webページ１００の他の要素の情報を収集しても良い。

【００５１】

1つの例示的な態様では、ウェブページの統計モデル２３０に基づいて異常要素を検出する方法は、以下のように実施することができる。ステップ３００において、ユーザは自身のデバイスからWebサイトへアクセスする。Webクライアント１１０は、Webサーバ１３０へ送信された要求により、Webサーバ１３０からサイトのWebページを取得するよう構成される。そのプロセスの間に、Webサーバ１３０（又は中間ノード）はWebページにスクリプト１４０を追加する。ステップ３１０において、スクリプトをWebクライアント側１１０で実行し、Webページ１００に含まれるデータを収集する。スクリプト１４０によって収集されたデータには様々な情報が含まれる。特に、スクリプト１４０は、Webページの少なくとも１つの要素（スクリプト、フォームなど）のコンテンツを収集することができる。スクリプト１４０によって収集されたデータは、必要に応じて変換される。データは、スクリプト１４０自体によって、又はコントロールサーバ１５０上の処理手段によって変換される。ステップ３２０において、収集されたデータを、少なくとも１つのN次元ベクトルに変更し、ステップ３３０において保存する。

ステップ３７０において、得られたベクトルを、（例えば取得したベクトルとクラスタの中心との相互距離を測定することによって、）Webページの構築された統計モデル２３０のクラスタ及び／又は特定の統計モデルのN次元ベクトルと比較する。ステップ３８０において、比較の結果として、分析した要素を、例えば以下の場合に異常であると識別する。

（ｉ）N次元空間において、その要素のN次元ベクトルとモデルの各クラスタの中心との距離が、これらのクラスタの半径よりも大きい。

（ｉｉ）N次元空間において、その要素のN次元ベクトルとモデルの全てのクラスタの中心との間の近接度が、閾値よりも大きい。

（ｉｉｉ）N次元空間において、その要素のN次元ベクトルとモデルのクラスタのN次元ベクトルのうちクラスタ中心から最も離れたベクトルとの間の近接度が、閾値よりも大きい。

【００５２】

一例では、要素が異常であると認識されない場合、ステップ３５１において、その要素

のN次元ベクトルをWebページの統計モデル２３０に追加する。

【００５３】

別の例では、Webサーバ１３０は、Webページ１００の異常要素を検出すると、Webクライアント１１０及びユーザのデバイス１２０との接続を無効にするか、又は、その接続は維持するもののWebサーバ１３０がクライアント１１０の要求に対する応答を停止する（接続によるデータ送信を停止する）よう構成される。ステップ３９０において、データ送信が停止している間、検出されたWebページの異常要素は、悪意のある機能（危険性）の存在を理由に、コントロールサーバ１５０のアンチウィルス手段（図示せず）によりスキャンされるか、又は監視される。閾値を超える統計的有意性を有するクラスタが検出した要素の周囲に形成される場合には、検出された異常要素は安全であると識別され、接続が再確立され、セッションが継続される。

【００５４】

特定の場合、異常要素が含まれているかどうかが事前に分かっていないWebページに基づいてモデルを構築すると、競合が発生する可能性がある。すなわち、要素のN次元ベクトルがモデルのいずれのクラスタにも現れず、ジレンマ－当該ベクトルに基づいて新たなクラスタを作成するのか、又は当該ベクトルによってそのコンテンツが表されている要素を異常なものとみなすのか－が発生する。競合は、評価セクションで評価されている要素と似た（近い）要素に基づいて作成された、要素又はクラスタの統計的有意性の評価に基づいて解決される。すなわち、評価対象の要素（又は、近傍の要素、つまりN次元空間内のN次元ベクトル間の距離がある閾値よりも小さい要素）を含むWebページの数と、セクションで評価されるモデルの構築に使用されているWebページの総数との比に基づいて、セクションの範囲がページ数又はバージョン（イテレーション）数として測定される。評価セクションにおける評価されたWebページの要素の統計的有意性の値が、同一セクションにおける他の要素の統計的有意性の値（又は統計的有意性の平均値）に近い（近接性は閾値によって決定される）か、又はある閾値を超える（例えば、２０％）場合、その要素は統計的に有意であると認識され、そうでなければ（それが閾値を超えなければ）異常であると認識される。例えば、モデル構築ステップにおいて、そのベクトルが以前に形成されたクラスタ２１０のいずれにも現れないWebページ１００の要素が出現する可能性がある。したがって、例えばそのWebページ１００が含まれるセクションにおける統計的有意性を定めることによって、その要素が異常であるかどうかを決めることが必要である。一例では、所定のタイプの要素の統計的有意性の閾値は２０％である。この検証は、要素の特定の範囲において、評価された要素に近い要素が４回出現したことを示し、統計的有意性の値が閾値より小さい２％であることに対応する。したがって、評価された要素とそれに近い要素（評価された要素の周囲に形成されたクラスタ）は異常である可能性がある。特定の場合、統計的有意性の閾値を、同じ種類の要素についてのクラスタの統計的有意性の最小値と定めることもできる。例えば、モデルが２５％、３２％、４７％及び９５％の統計的有意性を有するスクリプトのクラスタを含む場合、同じ種類の要素の閾値を２５％に設定することができる。

【００５５】

1つの例示的な態様では、統計的有意性はWebページの異常要素の検出にも用いられる。例えば、これは、統計モデルが構築されていない場合、又は上述のジレンマの解決中に使用される。第１のステップにおいて、Webクライアント１１０により、Webサーバ１３０からWebページ１００を取得し、ユーザのデバイス１２０上で実行する。スクリプト１４０を含むWebページは、スクリプト１４０が実行されると、Webページ１００の少なくとも１つの要素のコンテンツに関する情報をWebクライアント側１１０で収集し、収集された情報をユーザのデバイスから送信する。前述のスクリプトは、Webクライアント１１０の助けを借りて実行され、Webページ１００の要素の少なくとも１つのコンテンツの情報をWebクライアント側１１０で収集し、収集した情報を、Webクライアント１１０がWebページ１００を受信したユーザのデバイス１２０から送信する。コントロールサーバ１５０において、デバイス１２０から受信したコンテンツに関する情報は、要素のN次元ベクトルに変

換されるよう構成されており、得られたN次元ベクトルは任意の適切な方法によってクラスタ化される。
N次元ベクトルは、Webページの各要素について、要素のグループについて、あるいは同じ種類の要素のグループについて形成される。また、異なる種類の要素がグループを形成していてもよい。クラスタ２１０が少なくとも１つのベクトルを含む場合、クラスタ２１０が形成されてから、得られたクラスタ２１０の統計的有意性を定めても良い。ここで、統計的有意性は、クラスタ２１０内のN次元ベクトルの数とWebページ１００の数との比と定めることができる。なお、ここでのWebページ１００の数は、その要素のコンテンツの情報が収集され、コントロールサーバ１５０又はWebサーバ１３０に送信されたWebページの数のことである。閾値の有意性は、上述した方法によって定められ、要素の種類、クラスタリングの方法、評価セクションの範囲等に依存させることができる。

【００５６】

１つの例示的な態様では、ユーザがWebバンキングサイト、例えばhttps://my.KaspskyBank.ru/のWebページを要求すると、要求されたWebページはスクリプト１４０を追加し、ページ１００がユーザのデバイス１２０上で実行されるWebクライアント１１０に送信される。ユーザ側のスクリプト１４０は、以下のようなWebページ上の<script>要素を収集する。

```
<script>document.documentElement.id="js";var ..../Kasperskybank/";</script>

<script src="//static.kaspersky.ru/dist/kfs/kfs.js" crossorigin="anonymous"></script>

<script src="https:// static.kaspersky.ru /ib/prod/2842c77095d860e412d7a8cf30231fd53c89fb4e/ Kasperskybank / Kasperskybank.js" crossorigin="anonymous"></script>

<script async="" src="/kfs/kfs"></script>

<script>!function(){var e=document.getElementById("before-init__noscript");e&&(e.className="ui-browser__holder-block-hide");var o=function(){try{return"withCredentials"in new XMLHttpRequest}catch(e){return!1}}();if(o){var t=function(){if(navigator.cookieEnabled)return!0;document.cookie="cookietest=1";var e=-1!=document.cookie.indexOf("cookietest=");return document.cookie="cookietest=1; expires=Thu, 01-Jan-1970 00:00:01 GMT",e}();if(t)document.body.removeChild(document.getElementById("before-init"));else{var n=document.getElementById("before-init__nocookies");n&&(n.className="ui-browser__holder-block")}}else{var r=document.getElementById("before-init__old-browser");r&&(r.className="ui-browser__holder-block")}}();</script>
```

【００５７】

src属性を持つ<script>要素の場合、スクリプト本体のロードと正規化が実行される。インラインスクリプトの場合は、正規化のみである。例えば、上述したインラインスクリプトの場合、正規化された形式は次のようになる（重要な言語構成と標準的なオブジェクト／メソッドのみを記述し、リテラルは「非人格化」している）。

```
document.documentElement.i0=v0;vari1=window.i1||{};i1.i2=v1,i1.i3=v2,i1.i4=v3,i1.i5=v4,i1.i6={i7:v5,i8:v6},i1.i9=v7;

!function(){vari0=document.getElementById(v0);i0&&(i0.i1=v1);vari2=function(){try{returnv2innewXMLHttpRequest}catch(i0){return!v3}}();if(i2){vari3=function(){if(navigator.i4)return!v4;document.cookie=v5;vari0=-v3!=document.cookie.indexOf(v6);returndocument.cookie=v7,i0}();if(i3)document.body.removeChild(document.getElementById(v8));else{vari5=document.getElementById(v9);i5&&(i5.i1=v10)}}else{vari6
```

=document.getElementById(v11);i6&&(i6.i1=v10)}}();
【００５８】
次に、スクリプト１４０は、ページ上に存在する<input>要素を収集する。
<input autocomplete="off" autocorrect="off" autocapitalize="off" class="m-login__form-field-input ng-pristine ng-invalid ng-invalid-required ng-touched" type="text" ... ng-blur="login.focus = false" placeholder="password">
【００５９】
スクリプト１４０は、<input>要素の収集されたデータを、例えば以下のように正規化しつつ変換する（属性はアルファベット順にソートされ、タグ名は切り捨てられ、属性値の空白は切り取られ、属性は「;」によりリスト化される）。
<autocapitalize=off;autocomplete=off;autocorrect=off;class=m-login__form-field-inputng-pristineng-invalidng-invalid-requiredng-touched;name=lg;ng-blur=login.focus=false;warmUp();;ng-change=input(true);ng-disabled=false;ng-keydown=login.focus=true&&$event.keyCode===13&&authUser();ng-keyup=fix(login.form.lg,$event);ng-model=login.lg;placeholder=login;spellcheck=false;style=padding:0px;;type=text;ui-focus=login.setFocus;validator=validator.lg>

<autocapitalize=off;autocomplete=off;autocorrect=off;class=m-login__form-field-inputng-pristineng-untouchedng-invalidng-invalid-required;name=pw;ng-blur=login.focus=false;ng-change=input();ng-disabled=false;ng-keydown=login.focus=true&&$event.keyCode===13&&authUser();ng-keyup=fix(login.form.pw,$event);ng-model=login.pw;placeholder=password;spellcheck=false;type=password;validator=validator.pw>
【００６０】
スクリプト１４０は、収集されたデータをコントロールサーバ１５０に送信する。コントロールサーバ１５０は、対応するモデル（全てのスクリプト要素の集合形－スクリプト型要素の統計モデル２２０）のコンテキスト内の<script>要素の収集されたデータを、以下のように処理することができる。
・各スクリプトについて、数値ベクトルを取得する（例えば、ベクトルは２次元とすることができる）。ここで、ベクトルは文字列のコード（ASCIIのような文字コードを得るため、任意の適切な符号化方法が用いられる）から計算され、収集されたデータを構成する（インラインスクリプトの場合、このデータには正規化されたスクリプトの内容が含まれ、その他の場合、データはsrc属性の内容である）。結果として得られるWebページ１００に含まれる<script>要素に対し、以下のベクトルを得ることができる。
o 16314,10816
o 2254,2598
o 16084,15036
o 356,822
o 20010,51838
・各ベクトルは、モデル２３０の２次元空間に保存され、異常がない所定の場合、全てのベクトルは以前に形成されたクラスタ内に収まる（すなわち、全てのベクトルが、以前のWebページのそれらのバージョンのスクリプト１４０から来たデータと一致する）。図４ａにおいて、ドットは分析された<script>要素を示し、着色された領域は、モデル２３０の一部として以前に作成されたモデル２２０のクラスタ２１０である。
【００６１】
コントロールサーバ１５０は、<input>要素の収集されたデータを同様の方法で処理する。結果として、視覚化は図４ｂに示す形式となる。異常要素が検出されていないため、処理は終了する。
【００６２】
１つの例では、同じページhttps://my.KasperskyBank.ru/のユーザの１人に、追加の<script>要素の形で、悪意のあるインジェクションが現れる。

<script src="https://static.kasperskyBank.ru/ib/prod/bank/malware.js" crossorigin="anonymous"></script>

【００６３】

上述した方法によって計算されたベクトルは（4560,3192）に等しく、モデルは図４Ｃ（インジェクションの内容を表した異常である現在のベクトルは、赤色にマークされる）に示される形となる。ステップ３９０において、検出された異常要素は、コントロールサーバ１５０のアンチウィルス手段によって処理され、接続自体を無効にすることができる。また同時に、モデル空間内の要素をその統計的有意性を測定するために観察しても良い。

【００６４】

本発明は、異常要素の検出だけでなく、上述のように、Webページの悪意のある要素を検出するためにも使用できる。悪意のある要素を検出するため、同一の統計的クラスタモデルを使用することができる。異常要素を検出するために使用される統計的クラスタモデルと、悪意のある要素を検出するために使用される統計的クラスタモデルとの違いには、モデルを構築するためにクラスタ化されるN次元ベクトルを作成するために用いられる情報が含まれる。異常要素を検出する場合、潜在的に危険な要素に関する情報を使用してクラスタが作成される。一方、悪意のある要素を検出する場合、既知の悪意のある要素に関する情報を使用してクラスタが作成される。つまり、統計モデルは、モデルを構築するために使用される情報のみが異なっている。したがって、図２のエンティティ２３０を、Webページの悪意のある要素の統計モデルを定めるために用いることができる。また、図１に示すシステムと同じシステムを用いてモデルを構築し、悪意のある要素を検出することができる。

【００６５】

１つの例示的な態様では、図３に示すように、Webページ２３０の悪意のある要素の統計モデルを構築する方法は、ステップ３１１において、コントロールサーバ１５０によって既知の情報を含むデータベース１６０からWebページの既知の悪意のある要素に関するデータを取得する。一態様では、これらの要素は、ステップ３９０において悪意があると認識された検出された異常要素のアンチウイルススキャンがなされた後、又は既知の悪意のあるソフトウェアのコピーを含むリポジトリを使用した後、早期に検出される。ステップ３１１においてデータベース１６０から取得されたデータは、Webページの悪意のある要素のコンテンツに関する情報（スクリプト、フォームなど）を含む。データを、必要であれば、コントロールサーバ１５０によって変換することができる。ステップ３３０において収集されたデータを、ステップ３２０において少なくとも１つのN次元ベクトルに変換することができる。ステップ３５０において、少なくとも１つのベクトルから少なくとも１つのクラスタ２１０を作成する。少なくとも１つの作成されたクラスタ２１０に基づいて、ステップ３６０において、悪意のある要素の統計モデルを構築する。

【００６６】

１つの例示的な態様では、Webページ２３０の悪意のある要素の統計モデルに基づいて悪意のある要素を検出する方法は、ステップ３００において、ユーザは自分のデバイスからWebサイトへアクセスし、Webクライアント１１０はWebサーバ１３０に要求を送信することによって、Webサーバ１３０からサイトのWebページ１００を取得する。このプロセスでは、Webサーバ１３０（又は中間ノード）によってスクリプト１４０をWebページ１００に追加してもよい。ステップ３１０において、Webクライアント側でスクリプトを実行し、Webページ１００に含まれるデータを収集する。スクリプト１４０によって収集されるデータには様々な情報が含まれる。例えば、スクリプトはWebページの少なくとも１つの要素（スクリプト、フォームなど）のコンテンツを収集することができる。スクリプト１４０によって収集されたデータは、必要に応じて変換される。データは、スクリプト１４０自体によって、又はコントロールサーバ１５０によって変換される。そして、得られたベクトルを、ステップ３７０において、Webページの悪意のある要素の構築された統計モデル２３０のクラスタ及び／又はその統計モデル２３０のN次元ベクトルと（例えば、取得したベクトルとクラスタの中心との間の相互距離を決定することによって）比較する。

ステップ３８１において、比較の結果として、分析した要素を、以下のうちの１つ又は複数を検出することで悪意のあるものであると判定する。
（１）取得したN次元ベクトルと、統計モデルの少なくとも１つのクラスタの中心との間のN次元空間における距離が、これらのクラスタの半径よりも小さい。
（２）取得したN次元ベクトルと、統計モデルの少なくとも１つのクラスタの中心との間のN次元空間における距離が、これらのクラスタの半径に等しい。
（３）取得したN次元ベクトルと、統計モデルの少なくとも１つのクラスタの中心との間の近接度が閾値未満である。
（４）取得したN次元ベクトルと、統計モデルの少なくとも１つのクラスタの中心から最も離れたN次元ベクトルの少なくとも１つとの間の近接度が閾値未満である。

【００６７】
１つの例示的な態様では、ハッシュに基づいて悪意のある要素を検出する方法は、ステップ３００において、ユーザは自分のデバイスからWebサイトへアクセスし、Webクライアント１１０がWebサーバ１３０に要求を送信することによって、Webサーバ１３０からサイトのWebページ１００を取得することを含む。ここで、Webサーバ１３０（又は中間ノード）によってスクリプト１４０をWebページ１００に追加してもよい。ステップ３１０において、Webクライアント側でスクリプトを実行し、Webページ１００に含まれるデータを収集する。スクリプト１４０によって収集されるデータには様々な情報が含まれる。例えば、スクリプトはWebページの少なくとも１つの要素（スクリプト、フォームなど）のコンテンツを収集することができる。ステップ３６１において、Webページの既知の悪意のある要素のコンテンツに関する情報から算出されたデータベース１６０から、ハッシュを取得する。ここで、情報は上述したものと同じ方法によって取得される。必要に応じて、スクリプト１４０によって収集されたデータは、スクリプト１４０自体によって、又は、コントロールサーバ１５０の処理手段によって変換される。ステップ３２１において、収集されたデータは、ハッシュの算出に用いられる。ステップ３７１において、コントロールサーバは、ステップ３１２において算出されたハッシュと、ステップ３６１においてデータベース１６０から取得したハッシュとを比較する。ステップ３８２において、コントロールサーバ１５０は、収集された情報の分析の結果として悪意のある要素を検出する。ここでは、ハッシュの比較の結果、コンテンツに関する収集された情報から算出されたハッシュとステップ３６１においてデータベース１６０から取得したハッシュが一致した場合に、要素が悪意のあるものであると判断される。

【００６８】
多くの例示的な態様において、本願に開示されている分析モジュール及び処理モジュールが実装されたWebサーバ、Webクライアント、データベース、コントロールサーバは、実際のデバイス、システム、コンポーネント、統合されたマイクロサーキットなどのハードウェアを使用して実現されるコンポーネントのグループ（特定用途向け集積回路、ASIC）、又は、フィールドプログラマブルゲートアレイ（FPGA）を含み、例えば、マイクロプロセッサシステム及びプログラム命令のセットのようなソフトウェアとハードウェアの組み合わせの形で、又はニューロシナプティックチップ上で実行することができる。システムの示された要素の機能は、ハードウェアのみによって、また、組み合わせによって実現される。後者の場合、システムの要素の機能の一部はソフトウェアによって実現され、一部はハードウェアによって実現される。特定の変形例では、要素のいくつか、又は要素の全ては、汎用コンピュータ（図５に示すものなど）のプロセッサ上に実装することができる。

【００６９】
図５は、本発明の例示的な態様に係る、Webページの異常且つ悪意のある要素を検出するためのシステム及び方法の態様の一例を示す図である。

【００７０】
図示のように、コンピュータシステム２０（パーソナルコンピュータ又はサーバ）は、ＣＰＵ２１と、システムメモリ２２と、ＣＰＵ２１に関連付けられたメモリを含む様々なシ

ステムコンポーネントを接続するシステムバス２３とを含む。当業者に理解されるように、システムバス２３は、バスメモリ又はバスメモリコントローラ、周辺バス、及び他のバスアーキテクチャと相互作用するローカルバスを含むことができる。システムメモリは、永久メモリ（ROM）２４及びランダムアクセスメモリ（RAM）を含む。基本入出力システム（BIOS）２６は、ROM２４を使用してオペレーションシステムをロードする際のように、コンピュータシステム２０の要素間で情報を転送するための基本的な手順を記憶する。

【００７１】

ハードディスク２７、磁気ディスクドライブ２８、及び光学式のドライブ３０は、ハードディスクインターフェース３２、磁気ディスクインターフェース３３、及び光学式のドライブインターフェース３４それぞれを介してシステムバス２３と接続される。ドライブ及び対応するコンピュータ情報メディアは、コンピュータ命令、データ構造体、プログラムモジュール、及びコンピュータシステム２０の他のデータを蓄積するための電源依存のモジュールである。

【００７２】

当業者であれば、コンピュータによって読み取り可能な形式でデータを記憶することのできる任意のタイプのメディア５６（ソリッド・ステート・ドライブ、フラッシュメモリカード、デジタルディスク、ランダムアクセスメモリ（RAM）等）を使用できることが理解されよう。

【００７３】

コンピュータシステム２０は、オペレーティングシステム３５が格納されるファイルシステム３６と、追加のプログラムアプリケーション３７と、他のプログラムモジュール３８と、プログラムデータ３９とを有する。コンピュータシステム２０のユーザは、キーボード４０、マウス４２、又は当業者に知られた任意の他の入力デバイス（非限定的な例：マイクロフォン、ジョイスティック、ゲームコントローラ、スキャナ等）を用いて命令を入力する。このような入力デバイスは、典型的には、シリアルポート４６を介してコンピュータシステム２０に接続され、シリアルポートはシステムバスに接続される。ただし、当業者であれば、入力デバイスを、パラレルポート、ゲームポート、又はユニバーサル・シリアル・バス（USB）など（これらに限定されない）を介して接続することができることを理解するであろう。モニタ４７又は他のタイプの表示装置もまた、ビデオアダプタ４８などのインタフェースを介してシステムバス２３に接続することができる。パーソナルコンピュータは、モニタ４７に加えて、ラウドスピーカやプリンタなどの他の周辺出力装置（図示せず）を備えていても良い。

【００７４】

コンピュータシステム２０は、１つ又は複数のリモートコンピュータ４９へのネットワーク接続により、ネットワーク環境で動作することができる。リモートコンピュータ（又は複数のコンピュータ）４９は、コンピュータシステム２０の性質を説明する上で前述した要素のほとんど又は全てを含むローカルコンピュータワークステーション又はサーバであっても良い。ルータ、ネットワークステーション、ピアデバイス、又は他のネットワークノードなどの他のデバイスも、コンピュータネットワークに存在することができるが、これに限定されるものではない。

【００７５】

ネットワーク接続は、ローカルエリアコンピュータネットワーク（LAN）５０及びワイドエリアコンピュータネットワーク（WAN）を形成することができる。このようなネットワークは、企業のコンピュータネットワーク及び社内ネットワークで使用され、一般にインターネットにアクセスする。LAN又はWANネットワーク内では、パーソナルコンピュータ２０は、ネットワークアダプタ又はネットワークインタフェース５１を介してローカルエリアネットワーク５０に接続される。ネットワークが使用される場合、コンピュータ２０は、モデム５４又はインターネットなどのワイドエリアコンピュータネットワークとの通信を可能にする当業者に周知の他のモジュールを使用することができる。モデム５４は、内部又は外部のデバイスとすることができ、シリアルポート４６によってシステムバス２

３に接続される。上記のネットワークシステムが、通信モジュールを使用して１つのコンピュータが他のコンピュータへの接続を確立するための多くの周知の方法のうちの非限定的な例であることは、当業者には理解されるだろう。

【００７６】

様々な実施形態において、ハードウェア、ソフトウェア、ファームウェア、又はこれらのあらゆる組み合わせにより、ここで説明されたシステム及び方法を実施し得る。ソフトウェアにおいて実装される場合は、方法は不揮発性コンピュータ可読メディアの１つ又は複数の命令又はコードとして保存され得る。コンピュータ可読メディアは、データストレージを含む。あくまでも例であり限定するものではないが、そのようなコンピュータ可読メディアは、RAM、ROM、EEPROM、CD-ROM、フラッシュメモリ、若しくは他のタイプの電気、磁気、光学式の記憶媒体、又はその他のメディアであってもよい。すなわち、これらによって命令又はデータ構造体という形で、要求されたプログラムコードを運ぶか又は保存することができ、汎用コンピュータのプロセッサによってアクセスすることができる。

【００７７】

様々な実施形態で、本願のシステム及びメソッドはモジュールとして実装され得る。ここで用語「モジュール」は、実世界の機器、コンポーネント、又はハードウェアを用いて実装されたコンポーネント配置であり、例えばASIC（Application Specific Integrated Circuit）、FPGA（Field-Programmable Gate Array）等の、又は例えばモジュールの機能を実行するマイクロプロセッサシステムや命令セットによる等、ハードウェアとソフトウェアの組み合わせとして実装され得る。これらは、実行中にマイクロプロセッサシステムを特定の機器に変換する。モジュールは、ハードウェア単体により促進される一定の機能とハードウェア及びソフトウェアの組み合わせによって促進される他の機能という２つの組み合わせとして実施されてもよい。モジュールの少なくとも一部又は全部は、汎用コンピュータのプロセッサにおいて実行できる（図１～４を用いて上記において詳述したもの等）。したがって、各モジュールは様々な適当な構成で実現することができて、ここに例示した特定の実装に限られるものではない。

【００７８】

なお、実施形態の通常の機能のうちの全てをここで開示しているわけではない。本発明のいずれの実施形態を開発する場合においても、開発者の具体的な目標を達成するためには実装に係る多くの決定が必要であり、これらの具体的な目標は実施形態及び開発者ごとに異なることに留意されたい。このような開発努力は、複雑で時間を要するものであるが、本発明の利益を享受し得る当業者にとっては当然のエンジニアリングであると理解されたい。

【００７９】

さらに、本明細書で使用される用語又は表現は、あくまでも説明のためであり、限定するものではない。つまり、本明細書の用語又は表現は、関連する技術分野の当業者の知識と組み合わせて、ここに示される教示及び指針に照らして当業者によって解釈されるべきであると留意されたい。明示的な記載がない限り、明細書又は特許請求の範囲内における任意の用語に対して、一般的でない又は特別な意味を持たせることは意図されていない。本明細書で開示された様々な態様は、例示のために本明細書に言及した既知のモジュールの、現在及び将来の既知の均等物を包含する。さらに、複数の態様及び用途を示し、説明してきたが、本明細書に開示された発明の概念から逸脱することなく、上述したよりも多くの改変が可能であることが、この開示の利益を有する当業者には明らかであろう。

【図１】

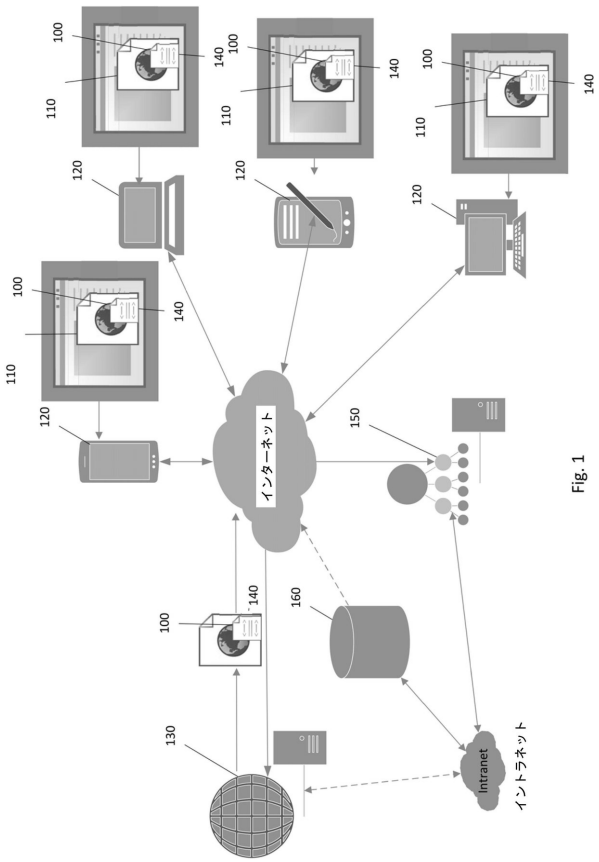

Fig. 1

【図２】

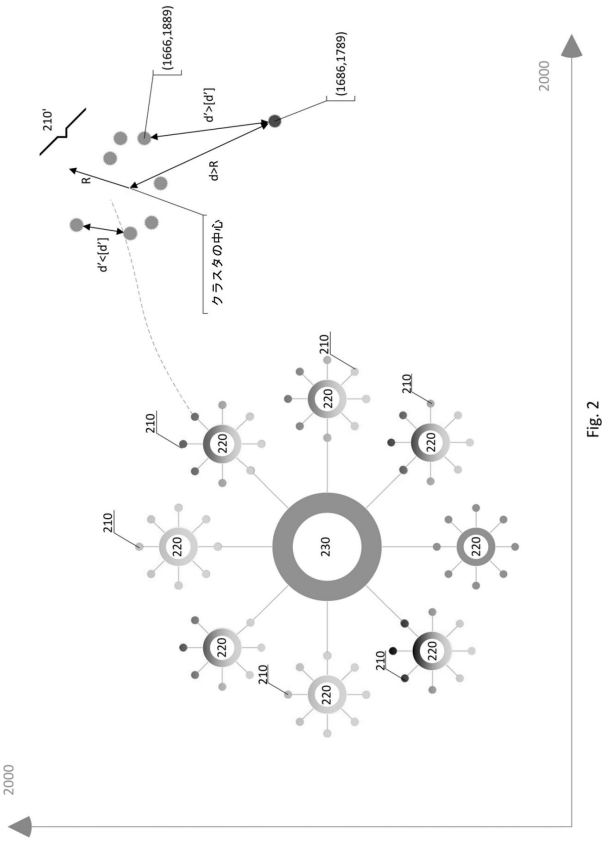

Fig. 2

【図３】

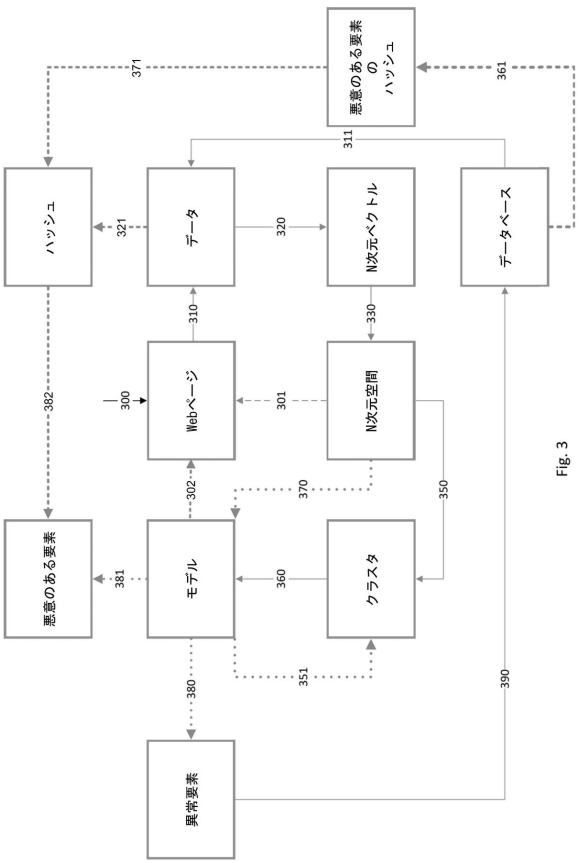

Fig. 3

【図４】

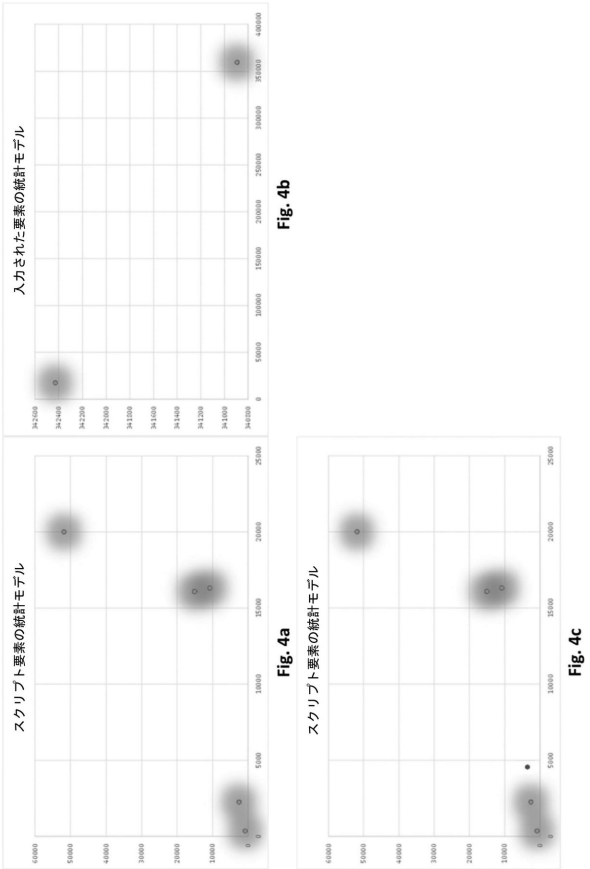

Fig. 4b

Fig. 4a

Fig. 4c

【図５】

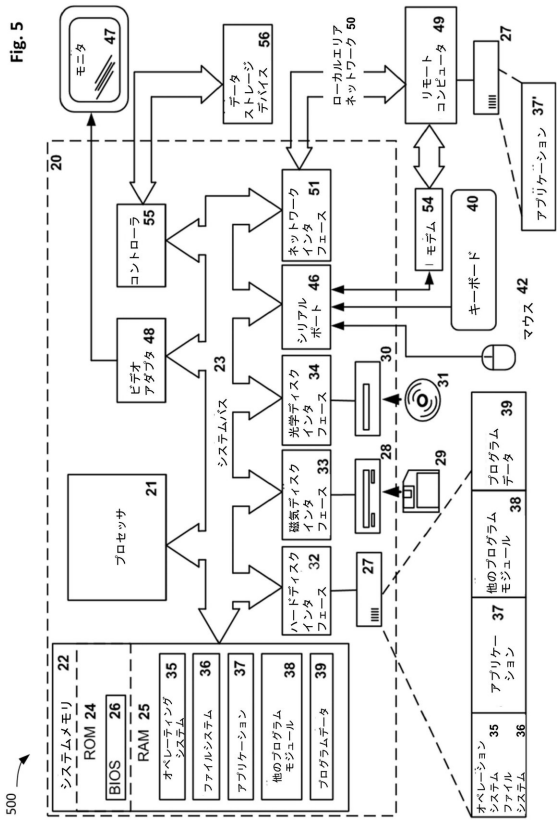
Fig. 5
500
20
システムメモリ 22
ROM 24
BIOS 26
RAM 25
オペレーティングシステム 35
ファイルシステム 36
アプリケーション 37
他のプログラムモジュール 38
プログラムデータ 39
プロセッサ 21
システムバス 23
ビデオアダプタ 48
コントローラ 55
ハードディスクインタフェース 32
磁気ディスクインタフェース 33
光学ディスクインタフェース 34
シリアルポート 46
ネットワークインタフェース 51
モニタ 47
データストレージデバイス 56
ローカルエリアネットワーク 50
リモートコンピュータ 49
モデム 54
キーボード 40
マウス 42
アプリケーション 37'
27
28
29
30
31
オペレーションシステム 35
ファイルシステム 36
アプリケーション 37
他のプログラムモジュール 38
プログラムデータ 39

フロントページの続き

(72)発明者 アントン ビー. ガルチェンコ
ロシア １２５２１２ モスクワ, レニングラドスコ ショス ３９エー／３, エーオー カスペルスキー ラボ
(72)発明者 ミハイル ブイ. ウスチノフ
ロシア １２５２１２ モスクワ, レニングラドスコ ショス ３９エー／３, エーオー カスペルスキー ラボ
(72)発明者 ヴィタリ ブイ. コンドラトフ
ロシア １２５２１２ モスクワ, レニングラドスコ ショス ３９エー／３, エーオー カスペルスキー ラボ
(72)発明者 ウラジミール エー. クスコフ
ロシア １２５２１２ モスクワ, レニングラドスコ ショス ３９エー／３, カスペルスキー ラボ エーオー

審査官 松尾 真人

(56)参考文献 米国特許第０８８２６４３９（ＵＳ，Ｂ１）
韓国公開特許第１０－２０１５－０１４４００９（ＫＲ，Ａ）
中国特許出願公開第１０２８１１２１３（ＣＮ，Ａ）
特開２０００－１３７７３２（ＪＰ，Ａ）
特表２０１６－５０８２７４（ＪＰ，Ａ）
特表２００８－５２９１０５（ＪＰ，Ａ）
特開２０１２－０８８８０３（ＪＰ，Ａ）
特開２０１０－０７９８７１（ＪＰ，Ａ）
K. Borgolte, et al., Delta: Automatic Identification of Unknown Web-based Infection Campaigns, CCS'13 Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security [online], 米国, ACM, ２０１３年１１月 ４日, P. 109-120, [平成30年8月6日検索],インターネット <URL:https://dl.acm.org/citation.cfm?id=2516725>
芝原 俊樹, リダイレクトの構造的類似性に基づく悪性Ｗｅｂページ検知手法, ＣＳＳ２０１５ コンピュータセキュリティシンポジウム２０１５ 論文集 情報処理学会シンポジウムシリーズ Ｖｏｌ．２０１５ Ｎｏ．３ ［ＣＤ－ＲＯＭ］, 日本, 一般社団法人情報処理学会, ２０１５年１０月１４日, Ｐ．４９６－５０３

(58)調査した分野(Int.Cl., ＤＢ名)
Ｇ０６Ｆ １６／００－１６／９５８
Ｇ０６Ｆ ２１／１２
Ｇ０６Ｆ ２１／５５