I419003

(19)中華民國智慧財產局

(12)發明說明書公告本 **(11)證書號數：TW I419003 B**

(45)公告日： 中華民國 102 (2013) 年 12 月 11 日

(21)申請案號：099139009 (22)申請日： 中華民國 99 (2010) 年 11 月 12 日

(51)Int. Cl.： ***G06F21/00 (2013.01)***

(71)申請人：國立交通大學 (中華民國) NATIONAL CHIAO TUNG UNIVERSITY (TW)
新竹市大學路 1001 號

(72)發明人：江易達 CHIANG, YI TA (TW) ；林盈達 LIN, YING DAR (TW) ；吳育松 WU, YU SUNG (TW) ；賴源正 LAI, YUAN CHENG (TW)

(74)代理人：高玉駿；楊祺雄

(56)參考文獻：

TW 201035795A1 US 2008/0184369A1

US 2010/0229239A1

審查人員：林信宏

申請專利範圍項數：8 項 圖式數：4 共 0 頁

(54)名稱

自動化分析與分類惡意程式之方法及系統

A METHOD AND A SYSTEM FOR AUTOMATICALLY ANALYZING AND CLASSIFYING A MALICIOUS PROGRAM

(57)摘要

一種自動化分析與分類惡意程式之方法及系統，適用於自動化分析一待比對程式是否為一已知惡意程式。該方法包含：提供該已知惡意程式之系統函數編號序列；執行該待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成一待比對系統函數編號序列；運算該待比對系統函數編號序列以及該已知惡意程式之系統函數編號序列間之一相似度；及若該相似度小於等於一相似度門檻值，則判定該待比對程式與該已知惡意程式為不同程式。

A method and a system for automatically analyzing and classifying a malicious program are disclosed. The method comprises: providing the sequence of IDs of system calls of an known malicious program; executing a to-be-matched program, and simultaneously recording the sequence of IDs of system calls invoked by the to-be-matched program as a to-be-matched system calls IDs sequence; calculating the similarity between the to-be-matched system calls IDs sequence and the sequence of IDs of system calls of the known malicious program; and determining that the to-be-matched system is different from the known malicious program if the similarity is not greater than a similarity threshold.

21~29・・・步驟

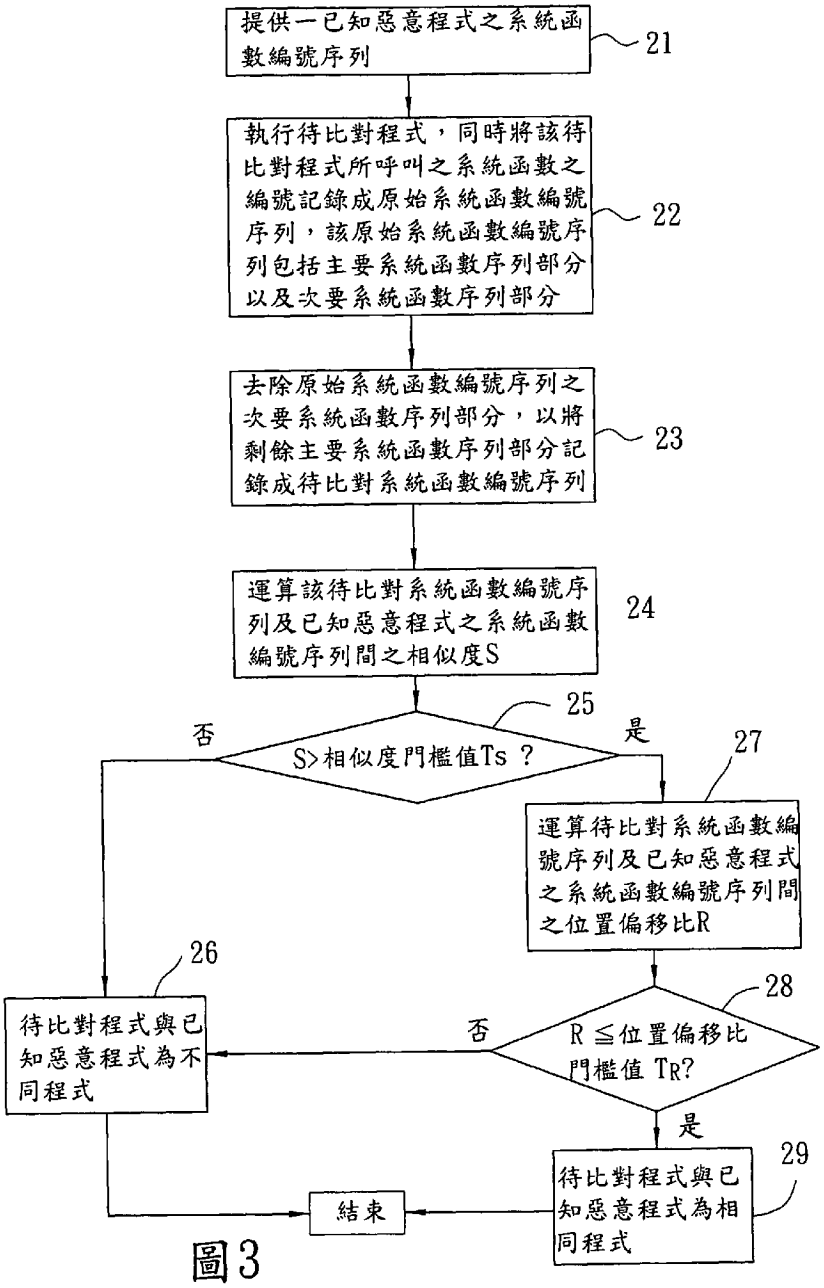
提供一已知惡意程式之系統函數編號序列
21
執行待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成原始系統函數編號序列，該原始系統函數編號序列包括主要系統函數序列部分以及次要系統函數序列部分
22
去除原始系統函數編號序列之次要系統函數序列部分，以將剩餘主要系統函數序列部分記錄成待比對系統函數編號序列
23
運算該待比對系統函數編號序列及已知惡意程式之系統函數編號序列間之相似度S
24
25
否
S>相似度門檻值Ts ？
是
27
運算待比對系統函數編號序列及已知惡意程式之系統函數編號序列間之位置偏移比R
26
28
否
R ≦位置偏移比門檻值 TR?
是
待比對程式與已知惡意程式為不同程式
29
待比對程式與已知惡意程式為相同程式
結束

圖3

公告本

發明專利說明書

（本說明書格式、順序，請勿任意更動，※記號部分請勿填寫）

※ 申請案號：99139009

※ 申請日：99.11.12　　※IPC 分類：G06F21/00 (2013.01)

一、發明名稱：(中文/英文)

自動化分析與分類惡意程式之方法及系統 / A method and a system for automatically analyzing and classifying a malicious program

二、中文發明摘要：

一種自動化分析與分類惡意程式之方法及系統，適用於自動化分析一待比對程式是否為一已知惡意程式。該方法包含：提供該已知惡意程式之系統函數編號序列；執行該待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成一待比對系統函數編號序列；運算該待比對系統函數編號序列以及該已知惡意程式之系統函數編號序列間之一相似度；及若該相似度小於等於一相似度門檻值，則判定該待比對程式與該已知惡意程式為不同程式。

三、英文發明摘要：

A method and a system for automatically analyzing and classifying a malicious program are disclosed. The method comprises: providing the sequence of IDs of system calls of an known malicious program; executing a to-be-matched program, and simultaneously recording the sequence of IDs of system calls invoked by the to-be-matched program as a to-be-matched system calls IDs sequence;

calculating the similarity between the to-be-matched system calls IDs sequence and the sequence of IDs of system calls of the known malicious program; and determining that the to-be-matched system is different from the known malicious program if the similarity is not greater than a similarity threshold.

四、指定代表圖：

(一)本案指定代表圖為：圖（ 3 ）。

(二)本代表圖之元件符號簡單說明：

21~29……步驟

五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

六、發明說明：

【發明所屬之技術領域】

本發明是有關於一種自動化分析與分類惡意程式之方法及系統，特別是指一種運用最長共同子序列(Longest Common Subsequence，LCS)演算法來自動化分析與分類惡意程式之方法及系統。

【先前技術】

在網際網路中，存在著許多由惡意程式所帶來的安全威脅。其中，殭屍網路(Botnet)中的殭屍程式(Bot)是一種很嚴重的威脅。

殭屍網路是一種自律(Autonomous)網路，其由執行攻擊者所控制之軟體代理程式之眾多受害電腦所組成。此種軟體代理程式即稱為殭屍程式。

殭屍網路通常用來進行惡毒的攻擊行為，例如垃圾電子郵件、資訊竊取等。這些攻擊行為可能導致使用網際網路時產生問題，或者導致金融損失，因此殭屍程式的偵測與移除技術便成為相關人士持續研發的方向。

傳統上分析殭屍網路之方法可分為靜態分析與動態分析兩種。靜態分析單純分析程式碼，而不考慮實際執行的情形。然而，在程式碼經過混淆變形(如加密或壓縮等)後，即無法利用靜態分析方法來進行正確分析，且此種經混淆變形之程式碼依舊可以正常執行，而對電腦產生危害。

至於，動態分析則是藉由監控系統應用程式介面(Application Program Interface，API)及其參數來進行比對工

作，然而此種動態分析作法太過粗略而可能無法比對成功，例如在比對檔名時可能因檔名會隨機改變而造成比對失敗。因此，有必要尋求解決之道。

【發明內容】

因此，本發明之目的，即在提供一種自動化分析與分類惡意程式之方法。

於是，本發明自動化分析與分類惡意程式之方法適用於自動化分析一待比對程式是否為一已知惡意程式。該方法包含下列步驟：(A)提供該已知惡意程式之系統函數編號序列；(B)執行該待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成一待比對系統函數編號序列；(C)運算該待比對系統函數編號序列以及該已知惡意程式之系統函數編號序列間之一相似度；以及(D)若該相似度小於等於一相似度門檻值，則判定該待比對程式與該已知惡意程式為不同程式。

本發明之另一目的，即在提供一種自動化分析與分類惡意程式之系統。

於是，本發明自動化分析與分類惡意程式之系統適用於自動化分析一待比對程式是否為一已知惡意程式。該系統包含一資料庫、一記錄模組以及一分析模組。該資料庫包括該已知惡意程式之系統函數編號序列。該記錄模組用以執行該待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成一待比對系統函數編號序列。該分析模組用以運算該待比對系統函數編號序列以及該已知惡意程

式之系統函數編號序列間之一相似度。若該相似度小於等於一相似度門檻值，則該分析模組判定該待比對程式與該已知惡意程式為不同程式。

本發明之功效在於，藉由三項主要技術特徵來達到高辨識率，即藉由依序進行之系統函數編號序列之片段辨識、最長共同子序列相似度分析，以及位置偏移分析等三階段，來比對待比對程式與已知惡意程式是否為相同程式，。

【實施方式】

有關本發明之前述及其他技術內容、特點與功效，在以下配合參考圖式之一個較佳實施例的詳細說明中，將可清楚的呈現。

在本發明被詳細描述之前，要注意的是，在以下的說明內容中，類似的元件是以相同的編號來表示。

參閱圖 1，本發明自動化分析與分類惡意程式之系統 1 之較佳實施例適用於自動化分析一待比對程式 9 是否為一已知惡意程式(如殭屍程式等)，其中該待比對程式 9 原本儲存於一儲存空間(圖未示)中。在本較佳實施例中，本發明軟體系統 1 係藉由三項主要技術特徵來達到高辨識率，即藉由依序進行之系統函數(System Call)編號(ID)序列之片段辨識 (Segment Identification)、最長共同子序列 (Longest Common Subsequence，LCS)相似度分析，以及位置偏移(Shift)分析等三階段，來比對待比對程式 9 與已知惡意程式是否為相同程式。此外，即使該待比對程式 9 利用混淆工

具(或稱封裝工具(Packer))偽裝，本發明技術亦能成功辨識出該經偽裝的待比對程式 9 與已知惡意程式是否為相同程式。

該系統 1 包含一資料庫 13、一記錄模組 11 以及一分析模組 12。

該資料庫 13 包括預先建置完成之已知惡意程式之系統函數編號序列。

該記錄模組 11 用以執行該待比對程式 9，同時將該待比對程式所呼叫之系統函數之編號記錄成一包括一主要系統函數序列部分及一次要系統函數序列部分之原始系統函數編號序列，且繼而去除該原始系統函數編號序列之次要系統函數序列部分，以將剩餘的主要系統函數序列部分記錄成該待比對系統函數編號序列。

該分析模組 12 用以藉由運算該待比對系統函數編號序列以及該已知惡意程式之系統函數編號序列間之一相似度 S 以及一位置偏移比 R，來分析該待比對程式 9 與該已知惡意程式為相同或不同程式。

參閱圖 1~4，以下將以本發明自動化分析與分類惡意程式之方法之較佳實施例，來更詳細說明本發明技術之細節。

如圖 2 所示，一般來說，殭屍程式在執行時所呼叫的系統函數可分為一主要系統函數部分(如圖 2 中的經封裝之程式部分 93)以及一次要系統函數部分，其中該次要系統函數部分包括程式載入器部分 91、拆封載入器部分 92 以及程

式退出處置器部分 94。在程式載入器部分 91 中，作業系統(如 Windows OS)載入器載入必要的動態連結庫(DLL)，並配置記憶體空間等。在拆封載入器部分 92 中，拆封載入器準備一用以執行原始程式之環境，例如將被壓縮的二進位程式碼(Binary)拆封成文字片段。在經封裝之程式部分 93 中，殭屍程式之主程式呼叫主要系統函數。在程式退出處置器部分 94 中，會有一些系統函數被用來釋放所配置的資源，且退出程式。此外，不同的混淆工具(即封裝工具)會引入不同的系統函數，然而為了要維持原始程式的功能，故在經封裝之程式部分 93 中總是會包括由原始程式所呼叫的系統函數。因而本發明即利用此特性，使得殭屍程式即使經混淆工具偽裝，卻仍然可以被本發明之技術辨識出。

如圖 3 步驟 21 所示，本發明自動化分析與分類惡意程式之方法之較佳實施例一開始需先在資料庫 13 中建置許多已知惡意程式之對應系統函數編號序列 131。

接著，如步驟 22 所示，該記錄模組 11 執行該待比對程式 9，同時將該待比對程式 9 所呼叫之系統函數之編號記錄成一原始系統函數編號序列，其中該原始系統函數編號序列包括一主要系統函數序列部分 111 以及次要系統函數序列部分 112，且該主要系統函數序列部分 111 係與圖 2 中的經封裝之程式部分 93(即主要系統函數部分)相對應，而該次要系統函數序列部分 112 係與圖 2 中的程式載入器部分 91、拆封載入器部分 92 及程式退出處置器部分 94(合稱為次要系統函數部分)相對應。

然後，如步驟 23 所示，該記錄模組 11 去除該原始系統函數編號序列之次要系統函數序列部分 112，以將剩餘的主要系統函數序列部分 111 記錄成該待比對系統函數編號序列 19。在本較佳實施例中，由於殭屍程式(即該待比對程式 9)在執行時所呼叫的系統函數中只有經封裝之程式部分 93 與本發明方法中後續的最長共同子序列相似度分析及位置偏移分析有關，且在程式載入器部分 91、拆封載入器部分 92 及程式退出處置器部分 94 中所呼叫的系統函數幾乎對於所有執行檔來說都是相同的，因此該記錄模組 11 可根據資料庫 13 中已知惡意程式之系統函數編號序列 131，將程式載入器部分 91、拆封載入器部分 92 及程式退出處置器部分 94 等三個無關的片段去除，且只使用經封裝之程式部分 93 來進行最長共同子序列相似度分析及位置偏移分析，藉以提高本發明技術對該待比對程式 9 的辨識率。

如上所述，當本發明方法完成去除步驟 23 時，記錄模組 11 便會輸出待比對系統函數編號序列 19 至分析模組 12。如圖 2、4 所示，該待比對系統函數編號序列 19 係對應至待比對程式 9 之經封裝之程式部分 93，且記錄模組 11 係從該經封裝之程式部分 93 之開始到結束，依呼叫時間先後順序，排序經封裝之程式部分 93 所呼叫的一連串系統函數 10(如圖 4 中的 NtClose、NtCreateFile、NtDeleteFile 及 NtLoadKey 等)，因而產生該待比對系統函數編號序列 19，例如在圖 4 中，此編號序列 19 為(1, 10, 11, 12, ...)。

在本發明較佳實施例中，該記錄模組 11 係利用 Pin 工

具從執行中的待比對程式 9 中擷取出待比對程式 9 所呼叫的系統函數及其編號(ID)。請參閱 http://www.pintool.org，Pin 可在執行檔執行時動態加入程式碼，使得將 Pin 附加到一正執行中的程序是可能的。

接著，如圖 3 步驟 24 所示，該分析模組 12 運算該待比對系統函數編號序列 19 以及該已知惡意程式之系統函數編號序列 131 間之一相似度 S(X, Y)，其中：

相似度 $S(X,Y) = L/\min(|X|, |Y|)$，

其中 X 為該待比對系統函數編號序列 19，Y 為該已知惡意程式之系統函數編號序列 131，L 為 X 及 Y 之最長共同子序列(Longest Common Subsequence，LCS)之長度，$\min(|X|, |Y|)$為 X 及 Y 中較短序列之長度。由於 $L \leqq \min(|X|, |Y|)$，故相似度 S(X, Y)的值介於 0 與 1 間，其中 S(X, Y)=1 表示 X 為 Y 的變體(Variant)，或者 Y 為 X 的變體。此外，該分析模組 12 係利用一相似度門檻值 T_S 來判定待比對系統函數編號序列 19 以及已知惡意程式之系統函數編號序列 131 是相同或不相同。

例如，假設已知惡意程式之系統函數編號序列 Y 為(1, 2, 3, 4, 5)，且待比對系統函數編號序列 X 為(1, 10, 11, 12, 2, 3, 18, 4, 20, 21, 5)，則序列 X 與 Y 之 LCS 為(1, 2, 3, 4, 5)，L 等於 5，且 $\min(|X|, |Y|)$也等於 5，因此 S(X, Y)=1，且 X 為 Y 的變體。

接著，如步驟 25 所示，該分析模組 12 判定其所運算出的相似度 S(X, Y)是否大於一相似度門檻值 T_S，其中在本

發明之較佳實施例中，該相似度門檻值 T_S 例如可為 60%。

若該步驟 25 之判定結果為否，亦即相似度 S(X, Y)小於等於相似度門檻值 T_S，則接著如步驟 26 所示，該分析模組 12 判定該待比對程式與已知惡意程式為不同程式。

反之，若該步驟 25 之判定結果為是，亦即相似度 S(X, Y)大於相似度門檻值 T_S，則接著如步驟 27，該分析模組 12 進一步運算 LCS 之各元素在該待比對系統函數編號序列 19 中之位置序列$(a_1,a_2,a_3,...,a_L)$，且運算 LCS 之各元素在已知惡意程式之系統函數編號序列 131 中之位置序列$(b_1,b_2,b_3,...,b_L)$。然後，該分析模組 12 運算位置序列$(a_1,a_2,a_3,...,a_L)$以及$(b_1,b_2,b_3,...,b_L)$間之 L 個位置差$(a_1-b_1,a_2-b_2,a_3-b_3,...,a_L-b_L)$中為相異數字之位置差之個數 N，其中該 N 為正整數。然後，該分析模組 12 再利用如下公式算出位置偏移比 R：

位置偏移比 R=N/L。

例如，同樣以上述已知惡意程式之系統函數編號序列 Y 為(1, 2, 3, 4, 5)，待比對系統函數編號序列 X 為(1, 10, 11, 12, 2, 3, 18, 4, 20, 21, 5)，且 LCS 為(1, 2, 3, 4, 5)之情況為例，LCS 之各元素在該已知惡意程式之系統函數編號序列 131 中之位置序列為(1, 2, 3, 4, 5)，且 LCS 之各元素在待比對系統函數編號序列 19 中之位置序列為(1, 5, 6, 8, 11)，因此兩個位置序列間的位置差序列為(0, 3, 3, 4, 6)，且 N=4，位置偏移比 R=4/5=80%。

接著，如步驟 28 所示，該分析模組 12 判定所運算出

的位置偏移比 R 是否小於等於一位置偏移比門檻值 T_R，其中在本發明之較佳實施例中，該位置偏移比門檻值 T_R 例如可為 6%。

若步驟 28 之判定結果為是，亦即位置偏移比 R 小於等於該位置偏移比門檻值 T_R，則接著如步驟 29 所示，該分析模組 12 判定待比對程式 9 與該已知惡意程式為相同程式。

反之，若步驟 28 之判定結果為否，亦即位置偏移比 R 大於該位置偏移比門檻值 T_R，則接著如步驟 26 所示，該分析模組 12 判定待比對程式 9 與該已知惡意程式為不同程式。同樣以前述待比對程式 9 與該已知惡意程式間的系統函數編號序列之相似度 S(X, Y)=1 且位置偏移比 R=80%之範例來做說明，即便待比對程式 9 與該已知惡意程式間具有 100%的系統函數編號序列相似度 S，然而由於兩者間的位置偏移比 R 為 80%，其遠大於位置偏移比門檻值 T_R，故分析模組 12 仍會將待比對程式 9 與已知惡意程式判定為不同程式。此表示雖然已知惡意程式之經封裝之程式部分 93(圖 2)所呼叫的系統函數全部都出現在待比對程式 9 之經封裝之程式部分 93 所呼叫的系統函數中，但由於該待比對程式 9 額外呼叫了其他系統函數，使得待比對程式 9 之待比對系統函數編號序列 19 中較該已知惡意程式額外插入了 10, 11, 12, 18, 20, 21 等 6 個序列元素，造成同樣出現在待比對系統函數編號序列 19 及已知惡意程式之系統函數編號序列 131 中的序列元素 2, 3, 4, 5 之位置偏移，導致位置偏移比 R 遠大於位置偏移比門檻值 T_R，因而即使系統函數編號序列

之相似度 S(X, Y)=100%，該分析模組 12 仍會將待比對程式 9 與該已知惡意程式二者間判定為不同程式。

綜上所述，本發明自動化分析與分類惡意程式之方法及系統係藉由三項主要技術特徵來達到高辨識率，即藉由依序進行之系統函數編號序列之片段辨識、最長共同子序列相似度分析，以及位置偏移分析等三階段，來比對待比對程式與已知惡意程式是否為相同程式，故確實能達成本發明之目的。

惟以上所述者，僅為本發明之較佳實施例而已，當不能以此限定本發明實施之範圍，即大凡依本發明申請專利範圍及發明說明內容所作之簡單的等效變化與修飾，皆仍屬本發明專利涵蓋之範圍內。

【圖式簡單說明】

圖 1 是一系統方塊圖，說明本發明自動化分析與分類惡意程式之系統之較佳實施例；

圖 2 是一示意圖，說明殭屍程式在執行時所呼叫的系統函數可分為程式載入器部分、拆封載入器部分、經封裝之程式部分以及程式退出處置器部分等四部分；

圖 3 是一流程圖，說明本發明自動化分析與分類惡意程式之方法之較佳實施例；以及

圖 4 是一示意圖，說明本發明中的記錄模組從圖 3 中的經封裝之程式部分之開始到結束，依呼叫時間先後順序，排序經封裝之程式部分所呼叫的一連串系統函數，因而產生待比對系統函數編號序列。

【主要元件符號說明】

1……自動化分析與分類惡意程式之系統

10……一連串系統函數

11……記錄模組

12……分析模組

13……資料庫

131……已知惡意程式之系統函數編號序列

19……待比對系統函數編號序列

21~29……步驟

9……待比對程式

91……程式載入器部分

92……拆封載入器部分

93……經封裝之程式部分

94……程式退出處置器部分

七、申請專利範圍：

1. 一種自動化分析與分類惡意程式之方法，適用於自動化分析一待比對程式是否為一已知惡意程式，該方法包含下列步驟：

(A)提供該已知惡意程式之系統函數編號序列；

(B)執行該待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成一待比對系統函數編號序列；

(C)運算該待比對系統函數編號序列以及該已知惡意程式之系統函數編號序列間之一相似度，其中該相似度 $S(X,Y) = L/\min(|X|,|Y|)$，X 為該待比對系統函數編號序列，Y 為該已知惡意程式之系統函數編號序列，L 為 X 及 Y 之最長共同子序列之長度，$\min(|X|,|Y|)$ 為 X 及 Y 中較短序列之長度；以及

(D)若該相似度小於等於一相似度門檻值，則判定該待比對程式與該已知惡意程式為不同程式。

2. 根據申請專利範圍第 1 項所述之自動化分析與分類惡意程式之方法，其中若該相似度 S(X,Y)大於該相似度門檻值，則該方法還包含下列步驟：

(E)運算該最長共同子序列之各元素在該待比對系統函數編號序列中之位置序列$(a_1,a_2,a_3,\ldots,a_L)$，且運算該最長共同子序列之各元素在該已知惡意程式之系統函數編號序列中之位置序列$(b_1,b_2,b_3,\ldots,b_L)$；

(F)運算該位置序列$(a_1,a_2,a_3,\ldots,a_L)$以及該位置序列$(b_1,b_2,b_3,\ldots,b_L)$間之 L 個位置差$(a_1-b_1,a_2-b_2,a_3-b_3,\ldots,a_L-b_L)$

中為相異數字之位置差之個數 N；

(G)運算該待比對系統函數編號序列以及該已知惡意程式之系統函數編號序列間之一位置偏移比 R=N/L；

(H)若該位置偏移比 R 大於一位置偏移比門檻值，則判定該待比對程式與該已知惡意程式為不同程式；及

(I)若該位置偏移比 R 小於等於該位置偏移比門檻值，則判定該待比對程式與該已知惡意程式為相同程式。

3. 根據申請專利範圍第 2 項所述之自動化分析與分類惡意程式之方法，其中該相似度門檻值為 60%，且該位置偏移比門檻值為 6%。

4. 根據申請專利範圍第 1 項所述之自動化分析與分類惡意程式之方法，其中該待比對程式所呼叫的系統函數包括一主要系統函數部分以及一次要系統函數部分，且該(B)步驟包括下列子步驟：

(B-1)在執行該待比對程式時，同時將該待比對程式所呼叫之系統函數之編號記錄成一原始系統函數編號序列，其中該原始系統函數編號序列包括一主要系統函數序列部分以及一次要系統函數序列部分；以及

(B-2)去除該原始系統函數編號序列之次要系統函數序列部分，以將剩餘的主要系統函數序列部分記錄成該待比對系統函數編號序列。

5. 一種自動化分析與分類惡意程式之系統，適用於自動化分析一待比對程式是否為一已知惡意程式，該系統包含：

一資料庫，包括該已知惡意程式之系統函數編號序列；

一記錄模組，用以執行該待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成一待比對系統函數編號序列；以及

一分析模組，用以運算該待比對系統函數編號序列以及該已知惡意程式之系統函數編號序列間之一相似度，其中該相似度 $S(X,Y) = L/\min(|X|,|Y|)$，X 為該待比對系統函數編號序列，Y 為該已知惡意程式之系統函數編號序列，L 為 X 及 Y 之最長共同子序列之長度，$\min(|X|,|Y|)$為 X 及 Y 中較短序列之長度，若該相似度小於等於一相似度門檻值，則該分析模組判定該待比對程式與該已知惡意程式為不同程式。

6. 根據申請專利範圍第 5 項所述之自動化分析與分類惡意程式之系統，其中若該相似度 $S(X,Y)$大於該相似度門檻值，則該分析模組還用以運算該最長共同子序列之各元素在該待比對系統函數編號序列中之位置序列$(a_1,a_2,a_3,\ldots,a_L)$，運算該最長共同子序列之各元素在該已知惡意程式之系統函數編號序列中之位置序列$(b_1,b_2,b_3,\ldots,b_L)$，運算該位置序列$(a_1,a_2,a_3,\ldots,a_L)$及該位置序列$(b_1,b_2,b_3,\ldots,b_L)$間之 L 個位置差$(a_1-b_1,a_2-b_2,a_3-b_3,\ldots,a_L-b_L)$中為相異數字之位置差之個數 N，以及運算該待比對系統函數編號序列及該已知惡意程式之系統函數編號序列間之一位置偏移比 R=N/L，其中若該位置偏移

比 R 大於一位置偏移比門檻值，則該分析模組判定該待比對程式與該已知惡意程式為不同程式，且若該位置偏移比 R 小於等於該位置偏移比門檻值，則該分析模組判定該待比對程式與該已知惡意程式為相同程式。

7. 根據申請專利範圍第 6 項所述之自動化分析與分類惡意程式之系統，其中該相似度門檻值為 60%，且該位置偏移比門檻值為 6%。

8. 根據申請專利範圍第 5 項所述之自動化分析與分類惡意程式之系統，其中該待比對程式所呼叫的系統函數包括一主要系統函數部分以及一次要系統函數部分，該記錄模組在執行該待比對程式時，同時將該待比對程式所呼叫之系統函數之編號記錄成一包括一主要系統函數序列部分及一次要系統函數序列部分之原始系統函數編號序列，且繼而去除該原始系統函數編號序列之次要系統函數序列部分，以將剩餘的主要系統函數序列部分記錄成該待比對系統函數編號序列。

六、圖式：

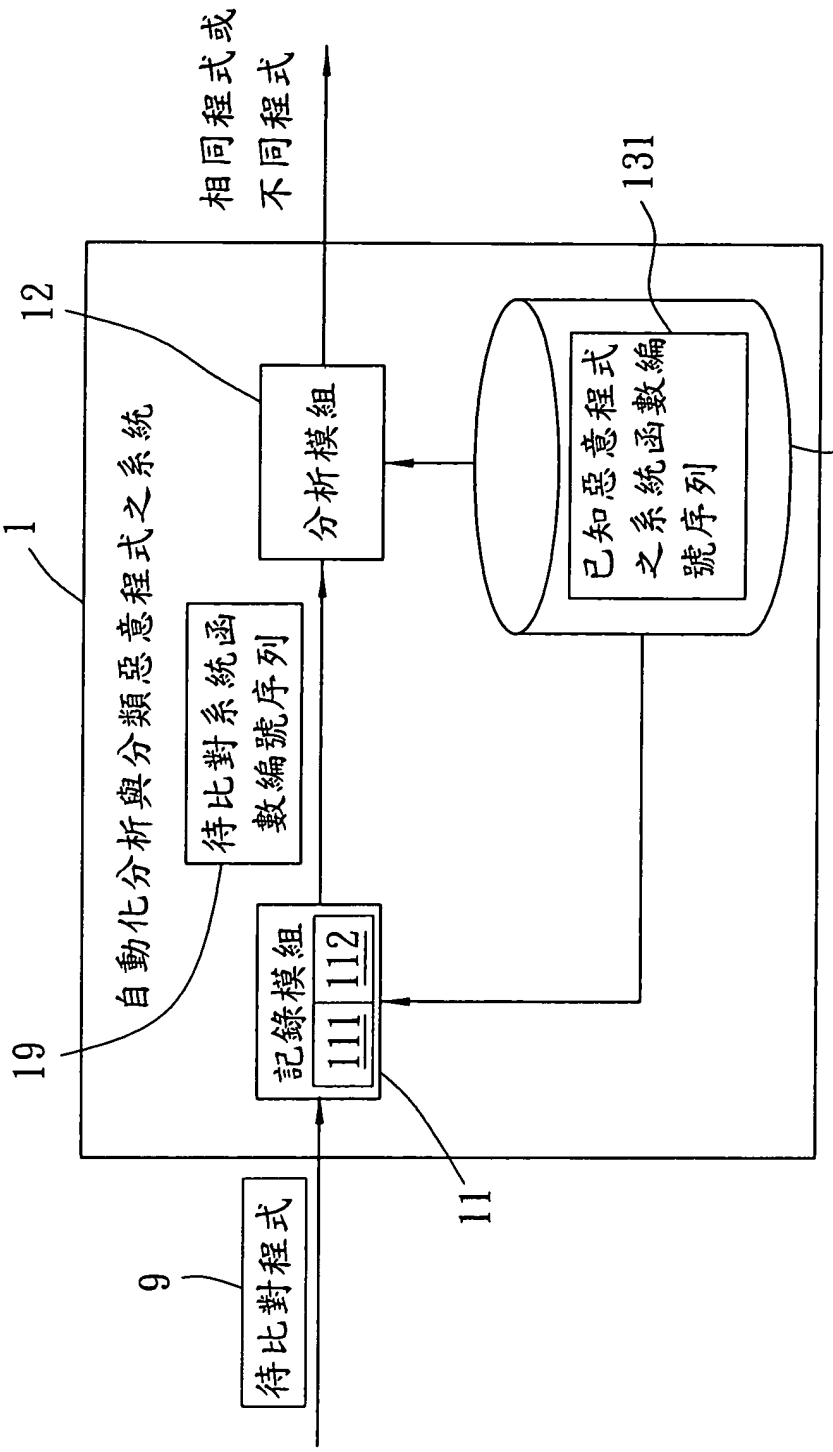
相同程式或
不同程式
12
131
1
分析模組
已知惡意程式
之系統函數編
號序列
13
自動化分析與分類惡意程式之系統
待比對系統函
數編號序列
19
記錄模組
111
112
11
9
待比對程式

圖1

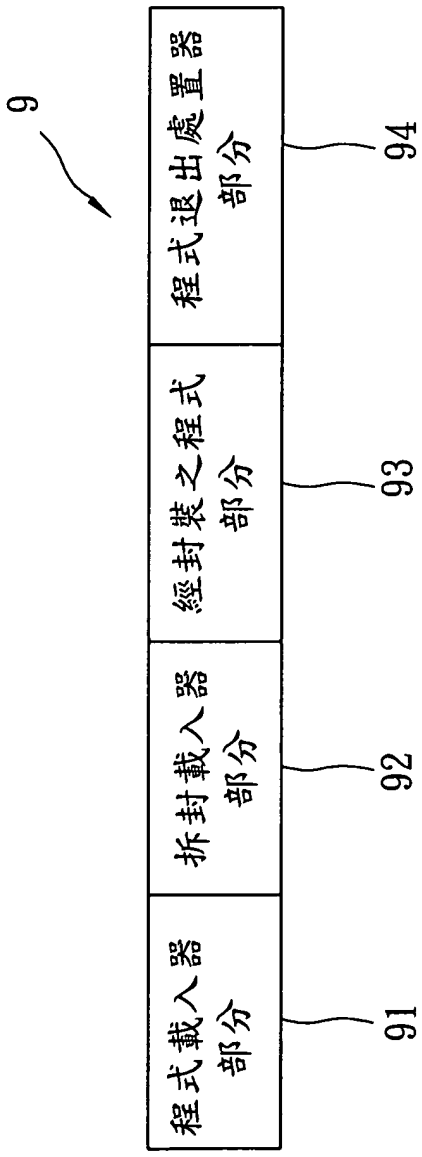

圖2

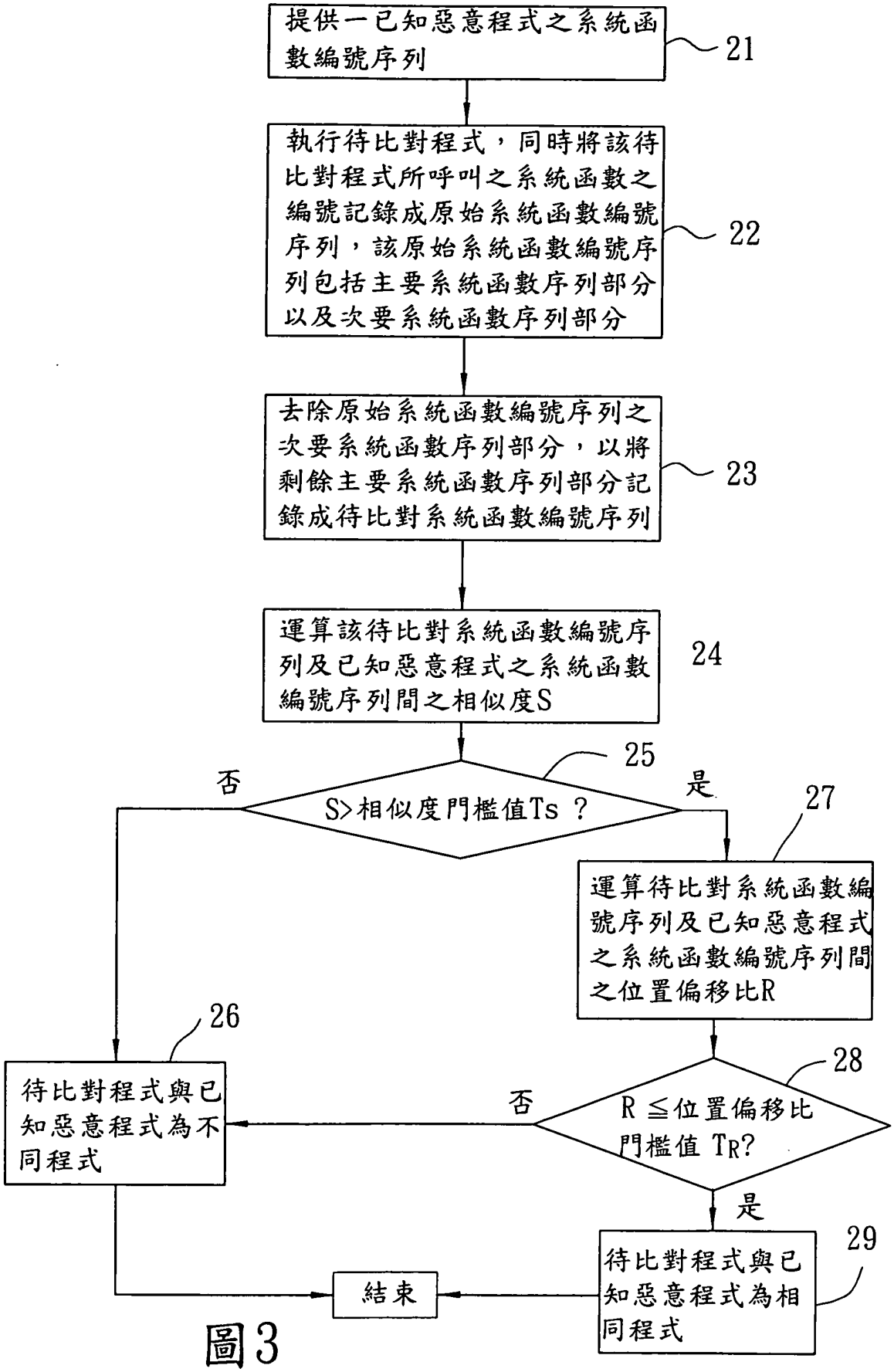
提供一已知惡意程式之系統函數編號序列
21
執行待比對程式，同時將該待比對程式所呼叫之系統函數之編號記錄成原始系統函數編號序列，該原始系統函數編號序列包括主要系統函數序列部分以及次要系統函數序列部分
22
去除原始系統函數編號序列之次要系統函數序列部分，以將剩餘主要系統函數序列部分記錄成待比對系統函數編號序列
23
運算該待比對系統函數編號序列及已知惡意程式之系統函數編號序列間之相似度S
24
25
否
S>相似度門檻值Ts ？
是
27
運算待比對系統函數編號序列及已知惡意程式之系統函數編號序列間之位置偏移比R
26
待比對程式與已知惡意程式為不同程式
28
否
R ≦位置偏移比門檻值 TR?
是
29
待比對程式與已知惡意程式為相同程式
結束

圖3

10

系統函數	NtClose	NtCreateFile	NtDeleteFile	NtLoadKey	....
順序	1	2	3	4	....
19 編號	1	10	11	12	....

圖4