I687828

(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 I687828 B**

(45)公告日： 中華民國 109 (2020) 年 03 月 11 日

(21)申請案號：108119096 (22)申請日： 中華民國 108 (2019) 年 05 月 31 日

(51)Int. Cl.： *G06F21/14 (2013.01)* *G06F21/55 (2013.01)*
H04L12/801 (2013.01) *H04L29/06 (2006.01)*

(30)優先權：2019/05/02 美國 16/402,004

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

(72)發明人：林盈達 LIN, YING-DAR (TW) ；裴 進軍 TIEN, QUAN-BUI (VN) ；賴裕昆 LAI, YU-KUEN (TW) ；賴源正 LAI, YUAN-CHENG (TW)

(74)代理人：楊長峯

(56)參考文獻：

CN 1164065C CN 104391788A
CN 107749810A CN 107948009A

審查人員：潘世光

申請專利範圍項數：9 項 圖式數：8 共 29 頁

(54)名稱

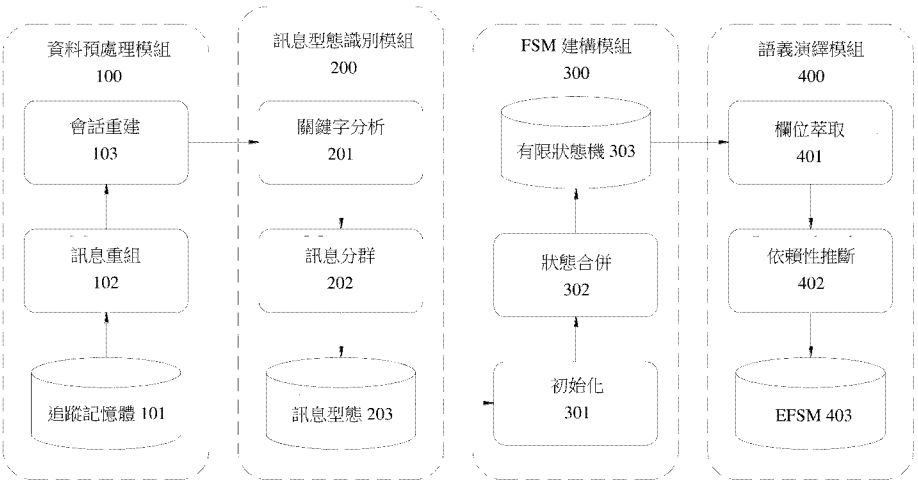
藉由從封包追蹤到擴展有限狀態機的逆向工程的自動協議測試方法

(57)摘要

本發明揭露了一種藉由從封包追蹤到擴展有限狀態機的逆向工程的自動協議測試方法。該方法包括以下步驟：解析複數個封包以萃取複數個會話；進行關鍵字分析和分群演算法以獲得協議訊息；初始化協議訊息並合併等效狀態以獲得有限狀態機；萃取協議訊息的欄位和值以獲得複數個子資料集，並在有限狀態機上添加資料保護和記憶體集以獲得擴展有限狀態機。

An automatic protocol test method by reverse engineering from packet traces to extended finite state machine is disclosed. The method includes following steps: parsing the plurality of packets to extract a plurality of sessions; conducting a keyword analysis and a clustering algorithm to obtain protocol messages; initializing the protocol messages and merging equivalent states to obtain a finite state machine; extracting fields and values of the protocol messages to obtain a plurality of sub-datasets and adding a data guard and set of memories on the finite state machine to obtain the extended finite state machine.

指定代表圖：



第 1 圖

符號簡單說明：

100:資料預處理模組

101:追蹤記憶體

102:訊息重組

103:會話重建

200:訊息型態識別模組

201:關鍵字分析

202:訊息分群

203:訊息型態

300:FSM 建構模組

301:初始化

302:狀態合併

303:有限狀態機

400:語義演繹模組

401:欄位萃取

402:依賴性推斷

403:EFSM

專利案號：108119096

申請日：

IPC 分類：

公告本

I687828

【發明摘要】

【中文發明名稱】藉由從封包追蹤到擴展有限狀態機的逆向工程的自動協議測試方法

【英文發明名稱】 AUTOMATIC PROTOCOL TEST METHOD BY REVERSE ENGINEERING FROM PACKET TRACES TO EXTENDED FINITE STATE MACHINE

【中文】

本發明揭露了一種藉由從封包追蹤到擴展有限狀態機的逆向工程的自動協議測試方法。該方法包括以下步驟：解析複數個封包以萃取複數個會話；進行關鍵字分析和分群演算法以獲得協議訊息；初始化協議訊息並合併等效狀態以獲得有限狀態機；萃取協議訊息的欄位和值以獲得複數個子資料集，並在有限狀態機上添加資料保護和記憶體集以獲得擴展有限狀態機。

【英文】

An automatic protocol test method by reverse engineering from packet traces to extended finite state machine is disclosed. The method includes following steps: parsing the plurality of packets to extract a plurality of sessions; conducting a keyword analysis and a clustering algorithm to obtain protocol messages; initializing the protocol messages and merging equivalent states to obtain a finite state machine; extracting fields and values of the protocol messages to obtain a plurality of sub-datasets and adding a data guard and set of memories on the finite state machine to obtain the extended finite state machine.

【指定代表圖】第1圖

【代表圖之符號簡單說明】

100：資料預處理模組

101：追蹤記憶體

102：訊息重組

103：會話重建

200：訊息型態識別模組

201：關鍵字分析

202：訊息分群

203：訊息型態

300：FSM建構模組

301：初始化

302：狀態合併

303：有限狀態機

400：語義演繹模組

401：欄位萃取

402：依賴性推斷

403：EFSM

【特徵化學式】無

【發明說明書】

【中文發明名稱】藉由從封包追蹤到擴展有限狀態機的逆向工程的自動協議測試方法

【英文發明名稱】 AUTOMATIC PROTOCOL TEST METHOD BY REVERSE ENGINEERING FROM PACKET TRACES TO EXTENDED FINITE STATE MACHINE

【技術領域】

【0001】 本揭露一般涉及自動協議測試方法，尤其涉及藉由從封包追蹤到擴展有限狀態機（EFSM）的逆向工程來獲得協議規範的自動協議測試方法。

【先前技術】

【0002】 協議規範有助於網路檢測系統，並且對於協議模糊器和測試案例產生工具的開發是必不可少的。入侵檢測系統通常依賴於解析器來基於協議規範執行深度封包檢查。在測試領域，智能模糊器需要具備在正確狀態下產生正確訊息的知識。漏洞發現工具還利用協議行為模型來產生非法和意外模式。然而，獲得協議規範是一項繁瑣且耗時的任務。即使對於開放協議，也需要花費時間來分析和翻譯打開的文檔以制定行為模型。因此，近日提出了一種基於規範推斷的自動協議逆向工程。

【0003】 通訊協議逆向工程的方法可以基於輸入分為兩個方向：執行追蹤（應用程式推斷）和網路追蹤（網路推斷）。由於原始程式碼不可用，很難執行應用程式推斷。大多數情況下，只有網路流量可用於分析網路伺服器提供商

方面的惡意軟體和殭屍網路。網路推斷方法只能依靠協議追蹤來主動重建行為模型並推斷出具有顯著精確結果的協議訊息格式。然而，傳統的逆向工程僅關注控制流量程而不考慮資料流量。

【0004】已知的通訊協議逆向工程技術具有局限性和缺點。因此，本發明人藉由從封包追蹤到擴展有限狀態機的逆向工程提供自動協議測試方法，以解決這些缺點並提高工業實用性。

【發明內容】

【0005】鑑於上述技術問題，本發明的一個目的是提供一種自動協議測試方法，藉由從封包追蹤到擴展有限狀態機的逆向工程，能夠同時考慮協議訊息的控制層和資料層，從而為安全性和測試應用提供更強大以及更合適的模型。

【0006】根據本揭露的一個目的，提供了一種藉由從封包追蹤到擴展有限狀態機（EFSM）的逆向工程的自動協議測試方法。此自動協議測試方法包括以下步驟：依序輸入包含複數個資料封包的流量追蹤；解析複數個封包以萃取複數個會話並重構複數個會話以獲得協議訊息；對協議訊息進行關鍵字分析和分群演算法以識別多種訊息型態；初始化協議訊息以形成初始會話序列並合併等效狀態以獲得具有一組狀態、一組轉換、一組輸入和一組輸出的有限狀態機（FSM）；萃取協議訊息的欄位和值以獲得複數個子資料集並在欄位上添加約束以產生資料保護，在有限狀態機上推斷資料保護和記憶體集以獲得擴展有限狀態機。

【0007】較佳地，可以根據源地址、源埠、目的地址、目的埠和傳輸層協議的型態來解析複數個封包。

【0008】較佳地，關鍵字分析可以包括先驗(Apriori)關鍵字分析以找到高頻和緊密序列串。

【0009】較佳地，分群演算法可以包括基於距離的K-means分群演算法。

【0010】較佳地，初始會話序列可以由前綴樹接受器以不同的路徑排列。

【0011】較佳地，等效狀態可以由相同的k-tail轉換合併。

【0012】較佳地，可以藉由Needleman和Wunch序列比對來處理協議訊息以萃取欄位。

【0013】較佳地，協議訊息中的欄位的值可以由Daikon演算法保持以推斷約束。

【0014】較佳地，對欄位的約束可以包括檢查欄位的有效值、訊息內依賴性和訊息間依賴性。

【0015】如前所述，根據本揭露藉由從封包追蹤到擴展有限狀態機的逆向工程的自動協議測試方法可以具有如下的一個或複數個優點：

【0016】1、自動協議測試方法能夠藉由利用有效的逆向工程技術推斷網路協議的行為模型，並且僅需要網路追蹤來重建忠實模型，這樣該方法可以廣泛應用於不同的網路協議的認定。

【0017】2、自動協議測試方法可以涵蓋協議訊息的控制流量和資料流量兩者，以提高協議模糊器或測試案例產生工具的品質，從而為安全性和測試應用提供更強大以及更合適的模型。

【0018】3、自動協議測試方法可以考慮訊息間依賴性和訊息內依賴性，從而可以減少用於保存訊息資料的記憶體大小以提供有效的操作。

【圖式簡單說明】

【0019】第1圖係繪示根據本揭露的自動協議測試方法的處理模型的示意圖。

【0020】第2圖是根據本揭露的自動協議測試方法的流程圖。

【0021】第3圖是根據本揭露的FSM建構模型的流程圖。

【0022】第4圖係繪示根據本揭露的從會話序列獲得的初始FSM的示意圖。

【0023】第5圖是根據本揭露的合併演算法的流程圖。

【0024】第6圖是根據本揭露的語義推斷模組的流程圖。

【0025】第7圖係繪示根據本發明的Needleman和Wunsch序列比對演算法的示意圖。

【0026】第8圖係繪示根據本揭露的Daikon演算法的示意圖。

【實施方式】

【0027】為了便於瞭解技術特徵、本揭露的內容和優點以及可以執行的有效性，具體實施方式下面將參考圖式藉由實施例詳細說明本發明。另一方面，這裡使用的圖僅僅是為了示意和輔助說明書，但是，在實施本揭露之後，不必是真實比例和精確配置。因此，不應該根據圖式的比例和配置來解釋而將本揭露的範圍限制在實際實施中。

【0028】根據本發明的實施例，可以使用各種型態的操作系統、計算平台、電腦程式和/或通用機器來執行這裡描述的組件、處理步驟和/或資料結構。此外，本領域具有通常知識者將認知到，在不脫離本文揭露的發明構思的範圍

和精神的情況下，也可以使用通用性較差的裝置，例如固線式裝置(hardwired devices)、現場可程式化邏輯閘陣列（FPGAs）、專用積體電路（ASICs）等。在包括一系列處理步驟的方法由電腦或機器執行並且那些處理步驟可以儲存為機器可讀的一系列指令的情況下，它們可以儲存在例如電腦記憶體裝置（例如，ROM（唯讀記憶體）、PROM（可編程唯讀記憶體）、EEPROM（電子抹除式可複寫唯讀記憶體）、快閃記憶體、跳轉驅動器等）、磁儲存媒體（例如，磁帶、磁盤驅動器等）、光學儲存媒體（例如，CD-ROM、DVD-ROM、紙卡和紙帶等）和其他已知型態的程式記憶體等有形媒體上。

【0029】 第1圖係繪示根據本揭露的自動協議測試方法的處理模型的示意圖。如圖所示，處理模型包括資料預處理模組100、訊息型態識別模組200、FSM建構模組300和語義演繹模組400。上述模組可以編程並保存在電腦的記憶體裝置中或雲端伺服器中。當執行自動協議測試方法時，處理器可以執行命令並訪問上述模型以處理本方法。

【0030】 首先，資料預處理模組100收集流量追蹤並保存在追蹤記憶體101中。特定協議的流量追蹤必須藉由訊息重組102和包括萃取和清理的會話重建103的步驟進行預處理。訊息型態識別模組200使用關鍵字分析201來萃取協議關鍵字，並使用訊息分群202藉由基於距離的演算法將訊息分群成組。每個組被視為不同的訊息型態203。結果進一步饋送到FSM建構模組300，以藉由使用初始化301和狀態合併302演算法來推斷有限狀態機（FSM）303。在推斷出正確的FSM 303之後，語義演繹模組400提供欄位萃取401以萃取包含觀察到的訊息中的訊息欄位的值的子資料集。藉由依賴性推斷402執行進一步分析以搜索訊息中的欄位的相關性並推導以在每個轉換上形成資料保護。每個轉換的資料保護和記憶體

集(set of memories)在FSM 303中合併以獲得擴展有限狀態機（EFSM）403。每個模型的詳細描述將由以下實施例呈現。

【0031】 第2圖是根據本揭露的自動協議測試方法的流程圖。如圖所示，該方法包括以下步驟（S1-S5）：

【0032】 步驟S1：依序輸入包含複數個資料封包的流量追蹤。請參考第1圖中提到的過程模型，資料預處理模組100可以處理各種形式的流量追蹤，例如TCP轉儲文件或pcap、pcapng。在本實施例中，模組從TCP轉儲文件中獲取流量追蹤作為輸入。可以收集流量追蹤並將其保存在電腦的記憶體中以進行以下分析。

【0033】 步驟S2：解析複數個封包以萃取複數個會話並重建複數個會話以獲得協議訊息。解析訊息以基於5元組萃取會話：源地址、源埠、目的地地址、目的地埠和傳輸層協議的型態。然後，如果需要則組裝分段訊息，也移除重複和重傳。可以在缺少碎片訊息的情況下使用時間間隔啟發式。繼續解析剩餘的訊息以忽略不相關的訊息，僅保留包含目標應用協議的訊息的有效負載以進一步分析。

【0034】 步驟S3：對協議訊息進行關鍵字分析和分群演算法，以識別多種訊息型態。為了提高訊息型態識別的品質，訊息型態識別模組200基於混合關鍵字分析的方法和基於距離的方法，其目的是利用優勢來克服彼此的限制。門檻值和關鍵字誤解的問題藉由基於距離度量的疊代K-means分群來解決。藉由萃取的關鍵字系列嘗試在K-means之前具有分群的限制。

【0035】 首先執行關鍵字分析201以發現頻繁出現的字符串的協議的關鍵字。在本實施例中，該組件可以採用Apriori演算法。然而，基於關鍵字的方法

不限於這種演算法。Kolmogorov-Smirnov檢驗、統計t檢驗和方差分佈可以包括在本方法中。Apriori關鍵字分析基於AutoReEngine的修改以識別關鍵字。基本上，該算法藉由Apriori方法疊代地找到具有穩定位置方差的高頻和緊密字節（或字符串）序列。在每次疊代中，僅保留頻率高於預定門檻值的閉合序列並將其視為關鍵字。

【0036】 在關鍵字萃取過程之後，在資料集中觀察到的關鍵字序列可以用作訊息型態的區分格式。每個關鍵字系列都是一個組。並且還在執行K-means演算法之前確定簇的數量。距離度量基於定義為$1-\frac{|a\cap b|}{|a\cup b|}$的Jaccard索引，其中a、b是訊息的字符數組。

【0037】 K-means分群演算法有助於在第一步驟中校準關鍵字萃取。例如，由於Linux文件系統的操作，字符串“CWD /”被檢測為關鍵字。（大多數命令的形式為“CWD / pub”、“CWD / root”、“CWD / conference”.....）。由於“CWD /”是關鍵字，因此訊息“CWD lib”和“CWD acld.tar.gz”h被分群為不正確的簇，並且應該藉由K-means分群演算法進行校準。為了克服缺少關鍵字的問題，該步驟保留未確定的訊息集並重複此過程。由於減小了資料集的大小，因此可以揭示原始資料集中具有低頻率的關鍵字。

【0038】 步驟S4：初始化協議訊息以形成初始會話序列並合併等效狀態以獲得具有一組狀態、一組轉換、一組輸入和一組輸出的有限狀態機。該模組的目標是在識別具有轉換以完成EFSM的資料保護之前推斷具有4元組的傳統FSM模型：(S, I, O, σ)。提議的方法論由兩部分組成：藉由GK-TAIL合併機制從追蹤和狀態合併FSM初始化。該步驟可以構造兩個FSM，一個用於客戶端，一個用於伺服器。在本節中，提供了伺服器端和客戶端的FSM模型推斷過程。

【0039】第3圖是根據本揭露的FSM建構模型的流程圖。如圖所示，FSM建構模型包括以下步驟（S401-S404）：

【0040】步驟S401：標記步驟。為了獲得更好的FSM結構，在處理之前進行資料集標記和清理。由於訊息被分組到隔離的集群中，標記組件命名其集群並將會話表示為標籤序列。如果客戶端請求訊息(i_k)並且響應訊息(o_k)來自伺服器（$i_k, o_k \in T$，其中T是訊息型態集），則會話可以表示為以下序列：$Ses = \{(i_1, o_1), (i_2, o_2) \dots (i_m, o_m)\}$。

【0041】步驟S402：調整步驟。由於追蹤是從現實世界的流量中採用的，因此某些會話不完整（例如遺失），並且由於資料封包交換和網路延遲，某些訊息會出現排序問題。因此，清理資料技術，如重新排序，缺失值推斷用於修復遺失的訊息。以不完整的FTP會話$\{(USER, 331), (PASS,\ x), \dots (QUIT, 500)\}$為例，我們可以很容易地基於資料集中的高頻對$(PASS, 230)$推斷“230”訊息的缺失值。

【0042】步驟S403：PTA初始化步驟。在此處理步驟中，資料集被饋送到初始化模組以建構接受所有會話序列的前綴樹接受器（Prefix Tree Acceptor，PTA）。對於每個會話序列，該過程從根開始，然後僅沿著樹向下移動。如果沒有現有路徑，則會創建新路徑。PTA節點集可以是初始狀態集，訊息的標籤是輸入和輸出的集合。初始轉換集是PTA轉換的集合。請參考第4圖，其係繪示根據本揭露的從會話序列獲得的初始FSM的示意圖。會話(Ses_1-Ses_u)由PTA方法按不同路徑排列。

【0043】步驟S404：合併步驟。藉由基於k-tail機制合併等效狀態來疊代地改進初始FSM。直覺是協議狀態機總是在相同的狀態下暴露相同的行為。換句

話說，如果在某個狀態下提交相同的輸入，機器將產生相同的輸出。由於協議狀態機的確定性特徵，可以合併兩個等效狀態的k-tail中的下一個狀態。請參考第5圖，其是根據本發明的合併演算法的流程圖。程式定義為以下步驟（S411-S415）：

【0044】步驟S411：計算k-tails作為下一轉換的集合。對於每個狀態s∈S，我們將k-tails定義為下一個轉換的集合。

【0045】步驟S412：初始化{Ei}由等效狀態組成的列表集。

【0046】步驟S413：比較狀態對的k-tail以初始化由等效狀態組成的列表集。

【0047】步驟S414：合併$E_i \cup E_j$。以上面的列表作為輸入，如果兩個集合共享至少一個元素直到沒有兩個集合可以合併，則疊代地合併兩個集合。在此步驟結束時，我們獲得由等效狀態組成的集合列表，並且沒有兩個集合由相同的狀態組成。

【0048】步驟S415：最終狀態集。對於每個集合，創建新狀態作為集合中所有狀態的代表。然後，對於每個原始轉換，在兩個新狀態之間添加新轉換，其相應地表示初始FSM的起始狀態和結束狀態。

【0049】步驟S5：萃取協議訊息的欄位和值以獲到複數個子資料集，並在欄位上添加約束以產生資料保護，在有限狀態機上推斷資料保護和記憶體集以獲得擴展有限狀態機。由於定義了具有4元組(S, I, O, σ)的FSM，我們開始在每次轉換時產生資料保護和記憶體(δ, M)，以完成具有6個元組$(S, I, O, \sigma, \delta, M)$協議的FSM。為此，我們研究協議執行如何處理特定狀態下即將到來的訊息以產生響應。首先，協議執行識別訊息的型態並將訊息分解為欄位（或資料）。然後，

它檢查每個欄位的值的驗證，以確定它是否執行進一步的操作。例如，訊息“*埠65,240,180,205,56,56*”，FTP伺服器識別請求命令是埠(PORT)然後解析以獲取資料（參數）(*“65,240,180,205,56,56”*)，包括前四個數字是IP地址的編碼並且剩下的就是埠號的編碼。FTP伺服器檢查保持這些數字的約束是否為（0, 255）保持這些數字的範圍。在EFSM使用Daikon推斷約束保持某些欄位的所有值之前，採用欄位萃取然後建構包含所有欄位值的子資料集。

【0050】 請參考第5圖，其是根據本發明的語義推導模組的流程圖。 該過程包括以下步驟（S501-S505）：

【0051】 步驟S501：格式萃取。格式萃取步驟是從訊息簇中推斷出協議的訊息格式。推斷每種訊息型態的格式以獲得欄位並準備由每個欄位的有效值組成的子資料集。訊息格式的自動逆向工程的示例性執行是多個Needleman和Wunsch（N＆W）序列對齊演算法。由於原始多重N＆W的高計算複雜性，使用基於預先建構的引導樹來決定順序的漸進對齊來用於對準過程。結果由共識字符串組成，顯示動態欄位和靜態欄位。靜態欄位大多是我們之前萃取的協議關鍵字系列。

【0052】 步驟S502：子資料集建構器。子資料集建構器步驟是藉由從原始軌跡中萃取來為每個包含有效值的欄位建構子資料集。

【0053】 藉由獲得追蹤中的所有觀察值並藉由響應訊息型態進行分組來建構每個動態欄位的子資料集。第7圖中給出了埠訊息對齊的圖示示例，其係繪示了根據本揭露的Needleman和Wunsch序列對齊演算法的示意圖。如圖所示，左側是追蹤中的訊息和N＆W對齊進度，右側是根據追蹤對應響應代碼“200”的每個欄位的子資料集。此後，挖掘欄位的子資料集以推導協議語義。

【0054】 步驟S503：資料保護推斷。資料保護推斷步驟可以將子資料集作為輸入來推導出對每個欄位的約束。這些約束被視為EFSM模型轉換的部分資料保護。該組件的示例性演算法是Daikon。Daikon可以將子資料集中的大量值推導成更簡單的不變量，可以用作資料保護。

【0055】 由於訊息被剖析，我們假設訊息i由欄位f組成，$i = [f_1, f_2, \ldots f_t]$，如前所述，資料保護充當輸入訊息的謂詞。然後我們可以大致定義資料保護功能如下：

$$\delta(i,m) = \delta(f_1, f_2, \ldots f_k, m) \approx \bigwedge_1^t Pr_u(f_u) \& \bigwedge_1^t Itr_{uv}(f_u, f_v) \& \bigwedge_1^t Ite_u(f_u, m)$$

【0056】 參數列出如下：

【0057】 $Pr_u(f_u) \rightarrow \{0,1\}$是資料驗證的欄位$f_u$的資料謂詞。此函數檢查欄位的值是否有效。它部分代表了該領域的語法。

【0058】 $Itr_{uv}(f_u, f_v) \rightarrow \{0,1\}$是兩個欄位$f_u, f_v$或訊息內依賴性的值之間的約束。

【0059】 $Ite_u(f_u, m) \rightarrow \{0,1\}$是欄位$f_u$的值與儲存在先前訊息中的記憶體m之間的約束。它是訊息間的依賴關係。

【0060】 為了更好地瞭解該過程，我們來看一下每類約束的一些例子。FTP埠命令引數(argument)中的IP地址和埠號格式是第一種型態的正則表達式。校驗和方向欄位（例如長度和偏移）是訊息內依賴性，而cookie（在HTTP協議中）是訊息間依賴性。由於近似地重新形成資料保護，所提出的方法通常產生欄位的資料謂詞並基於追蹤搜索訊息間/訊息內依賴性。

【0061】 EFSM依賴於Daikon演算法來產生每個欄位的資料保護。Daikon將子資料集中的大量值推導為更簡單的不變量，其EFSM可用作資料保護。因此，我們對欄位的資料謂詞定義如下：

$$Pr_u(v) = \begin{cases} 1 \ if \ v \in Daikon(f_u) \\ 0 \end{cases}$$

【0062】 請參考第8圖，其係繪示根據本揭露的Daikon演算法的示意圖。如圖所示，Daikon處理了埠命令資料的f_1字節的子資料集，發現f_1的值必須在（0, 255）的範圍內。類似地，TYPE命令的參數在可枚舉集的{“A”, “I”}中。如果未驗證此約束，則機器回復代碼為500而不是代碼200。

【0063】 原則上，Daikon將原始執行軌跡或變量值作為輸入，並找到所有觀察到的變量值的最佳匹配屬性（規則）。

【0064】 步驟S504：欄位依賴性檢測器。欄位依賴性檢測器步驟搜索相同訊息或交叉訊息中的欄位之間的依賴關係。表示欄位關係的訊息間依賴性的術語規定了不同訊息中的另一個欄位的屬性，例如cookie（HTTP）和序列號（TCP）。訊息內依賴性是一個訊息內的欄位之間的相關性，例如一致性欄位作為校驗和或方向欄位作為長度和偏移。

【0065】 在一個執行中，Pearson係數可以用於測量欄位的所有屬性對的依賴性的強度，然後將Daikon應用於潛在候選者以推斷關係。

【0066】 為了執行資料保護功能，該步驟繼續識別一個或兩個訊息中的欄位之間的關係。基本思想是我們利用Pearson係數來衡量所有欄位屬性對的依賴關係的強度，然後將Daikon應用於潛在候選者以推斷關係。

【0067】該過程由欄位的值、其長度以及在協議訊息中的偏移量所組成，目的是擷取不同型態的關係。此外，還考慮了IP地址、埠號等其他欄位。在追蹤中的會話上疊代地計算所有觀察到的欄位屬性對。

【0068】然後，基於每個觀察到的欄位屬性對(X,Y)計算Pearson乘積矩相關係數$\rho(X,Y)$。$\rho(X,Y)$的絕對值表示線性關係的強度。如果$|\rho(X,Y)|$接近1，然後X和Y之間通常存在線性關係。如果該值接近0，則它們大多是獨立的。例如，在HTTP協議和IP地址中資料的Content-Length和length屬性的$|\rho|$值為1。這是因為它們是線性相關的。最後，該步驟藉由應用Daikon演算法簡化了這些依賴性，從而可以推導出$(Y = aX + b)$的線性關係。該步驟將與Itr函數相同的訊息中的欄位之間的依賴關係分類。對於兩個訊息中的欄位之間的依賴關係，將其歸類為Ite函數。

【0069】步驟S505：記憶體推斷。記憶體推斷步驟是定義EFSM模型的記憶體集。記憶體可以被定義為保持在狀態中以供將來交互的欄位的值。欄位的值，例如Ite函數代表訊息間依賴性。或者，可以將先前訊息的每個欄位的值指定為狀態的記憶體。

【0070】如前所述，我們需要在定義的FSM的每次轉換時推斷出2元組(δ, M)資料保護和記憶體。現在藉由合併Pr、Itr和Ite函數很容易推斷資料保護功能δ。根據上述步驟，我們只需要更新狀態中的記憶體。記憶體可以被定義為保持在狀態中以供將來交互的欄位的值。欄位的值，例如Ite函數代表訊息間依賴性。或者，可以將先前訊息的每個欄位的值指定為狀態的記憶體。但是，在訊息序列之後記憶體集的大小將是巨大的。為了減小大小，我們建立了一對訊息間依賴關係。最初，包括上述環境欄位在內的每個後續訊息的欄位值都保存

在記憶體中。然後，如果在前面的訊息中沒有使用值，則消除那些值。一旦將來出現訊息間依賴關係對中的雙欄位之一，則保留相應的記憶體。剩下的記憶體是狀態的記憶體。然後獲得EFSM的6元組。

【0071】 另外，可以將測試序列產生和測試資料產生添加到自動協議測試方法。為了產生一致性測試案例，開發了一個簡單的測試案例產生原型，它將推斷的EFSM模型作為輸入。在一個實施例中，唯一輸入/輸出（UIO）可用於產生一組測試序列，其保證每個狀態至少被檢查一次。由於內部狀態可觀察性是主要問題，因此唯一輸入/輸出（UIO）序列被廣泛用於測試領域。該技術產生輸入序列和相應的輸出，其將指定狀態與剩餘狀態區分開。UIO方法用於為EFSM的每個狀態產生測試序列，這些狀態能夠在以後轉換為測試步驟（或測試場景）。

【0072】 為了測試資料流量，每個狀態的測試套件由UIO序列的每個輸入訊息上的資料值變異產生。（輸入訊息的）每個欄位的值被設計為遭受資料保護的可能情況（允許或不允許轉換）。可以將測試資料分配給測試序列中的狀態或轉換以完成測試案例。

【0073】 為了評估當前的擴展有限狀態機，選擇了FTP的四種協議、SMTP和Bittorent。網路追蹤的資料集是從公共可用和自擷取源收集的。網路追蹤由超過210,000條1,800個會話的訊息組成。但是，這些追蹤仍包含格式錯誤的訊息。需要額外的步驟來消除非法訊息。為了測量提議的EFSM的有效性，制定了若干指標。成對的Precision和Recall值都用於評估分群演算法對訊息型態識別的有效性。對於推斷的EFSM的品質，該過程使用正確性和覆蓋範圍分數，並提出一個新的方法來擷取EFSM與傳統FSM相比的強大功能。

【0074】在FTP協議的關鍵字分析中，所提出的EFSM檢測到伺服器端的所有26個命令。其餘命令（根據RFC 959中的命令總共33個命令）由於缺少追蹤而無法萃取。SMTP的關鍵字分析結果導致理想的精確分數。然而，訊息回收的分數為97%，因為有兩個孤立的EHLO群集和HELO訊息從同一群集中錯過分類。相反地，Bittorrent協議中BIT FIELD和REQUEST的訊息類似，我們的方法無法找到關鍵字來區分並將它們合併為一個集群。因此，精確度僅為94%，而訊息回收幾乎是完美的。

【0075】為了與現有作品進行比較，已經選擇了AutoReEngine。AutoReEngine在我們的資料集上應用了不同的門檻值，並保持最佳結果進行比較。結果表明，提高了本方法的準確性。對於所有協議，EFSM的成對精度和訊息回收值均高於AutoReEngine。疊代關鍵字分析有助於找到FTP的缺失關鍵字。K-means分群演算法有助於校準過度特定的萃取關鍵字問題。因此，FTP的精度和訊息回收值都得到了顯著改善。對於SMTP和Bittorent，EFSM仍然與AutoReEngine保持相同的準確性。這是因為他們都分享了關鍵字分析的主要思想，而K-means在這些情況下並沒有多大幫助。即便如此，對於訊息型態識別中的大多數協議，本方法優於AutoReEngine。

【0076】為了評估推斷的EFSM的品質，考察了k-tails參數對最終EFSM簡潔性的影響，以討論正確性、覆蓋範圍和行為準確率。具有k-tails參數的每個協議資料集的EFSM狀態數顯示，具有k-tails =1的我們的EFSM的EFSM模型在所有協議中都很簡潔。FTP、SMTP、Bittorrent理想模型中的狀態數為5、8、5和9。k-tail合併機制顯著減少了狀態數。k-tail參數僅影響簡潔性，而不影響推斷模型的品質。

【0077】 為了判斷覆蓋範圍和正確性，應用k-folds =5的參數。這意味著20%的會話都有待測試的痕跡。接受和拒絕的比例是4：1。結果，確認了我們的EFSM的FSM部分品質。SMTP的簡單性和資料集的豐富性是我們的推斷模型可以覆蓋近100%規範的原因，並且推斷模型接受所有有效會話。同樣，Bittorrent的覆蓋範圍也很高。具有大量命令集的FTP的合理複雜性導致最低覆蓋範圍為91%。在訓練集中隨機忽略測試集中的一些命令（轉換），以致EFSM無法學習。正確性從不低於90%，這保證了學習模型接近真實模型。

【0078】 以相同的方式，k-folds=5用於計算行為準確度。本質上，EFSM依賴於Daikon來產生資料保護，包括資料的有效語法和依賴性。在FTP的許多資料約束中，對埠命令參數的約束很重要。冒號字符吐出的前四個數字應映射到伺服器的IP地址以及埠號應大於255。檢測到Bittorrent的資料保護之一是hash_id的前20個字節在HANDSHAKE訊息兩者中應該是一致的。

【0079】 平均而言，用於FTP協議的推斷的EFSM的所產生輸出的89%與預期輸出匹配。與此同時，沒有資料保護和記憶體的FSM只能正確產生78%的輸出。這意味著推斷模型的行為幾乎與每個狀態的真實模型相當。其餘錯誤與追蹤中不存在的訊息有關。

【0080】 本揭露採用EFSM，靜態分析方法以網路追蹤的形式推斷出EFSM形式的協議的行為模型。推斷模型已經配備了資料值約束（稱為資料保護）。這些約束由Daikon從相關分析技術推導出的樣本和記憶體中推導出來。此外，EFSM利用K-tail機制來提高傳統FSM（控制流量）操作的準確性。

【0081】 儘管已經藉由參考圖式描述了本揭露中的特定實施例的手段，在不脫離申請專利範圍中闡述的本揭露的範圍和精神的情況下，本領域具有通常

知識者可以對其做出許多修改和變化。修改和變化應該在由本揭露的說明書限制的範圍內。

【符號說明】

100：資料預處理模組

101：追蹤記憶體

102：訊息重組

103：會話重建

200：訊息型態識別模組

201：關鍵字分析

202：訊息分群

203：訊息型態

300：FSM建構模組

301：初始化

302：狀態合併

303：有限狀態機

400：語義演繹模組

401：欄位萃取

402：依賴性推斷

403：EFSM

S1、S2、S3、S4、S5、S401、S402、S403、S404、S411、S412、S413、S414、S415、S501、S502、S503、S504、S505：步驟

【發明申請專利範圍】

【第1項】一種藉由從封包追蹤到擴展有限狀態機的逆向工程的自動協議測試方法，該自動協議測試方法包括以下步驟：

依序輸入包含複數個封包的流量追蹤；

解析該複數個封包以萃取複數個會話並重構該複數個會話以獲得一協議訊息；

對該協議訊息進行關鍵字分析和分群演算法以識別複數個訊息型態；

初始化該協議訊息以形成初始會話序列並合併等效狀態以獲得具有一組狀態、一組轉換、一組輸入和一組輸出的一有限狀態機；

萃取該協議訊息的欄位和值以獲得複數個子資料集並在該欄位上添加約束以產生資料保護，在該有限狀態機上推斷該資料保護和記憶體集以獲得一擴展有限狀態機。

【第2項】如申請專利範圍第 1 項所述的自動協議測試方法，其中根據源地址、源埠、目的地址、目的埠和傳輸層協議的型態來解析該複數個封包。

【第3項】如申請專利範圍第 1 項所述的自動協議測試方法，其中該關鍵字分析包括先驗(Apriori)關鍵字分析以找到高頻和緊密序列串。

【第4項】如申請專利範圍第 1 項所述的自動協議測試方法，其中該分群演算法包括基於距離的 K-means 分群演算法。

【第5項】如申請專利範圍第 1 項所述的自動協議測試方法，其中該初始

會話序列由一前綴樹接受器以不同的路徑排列。

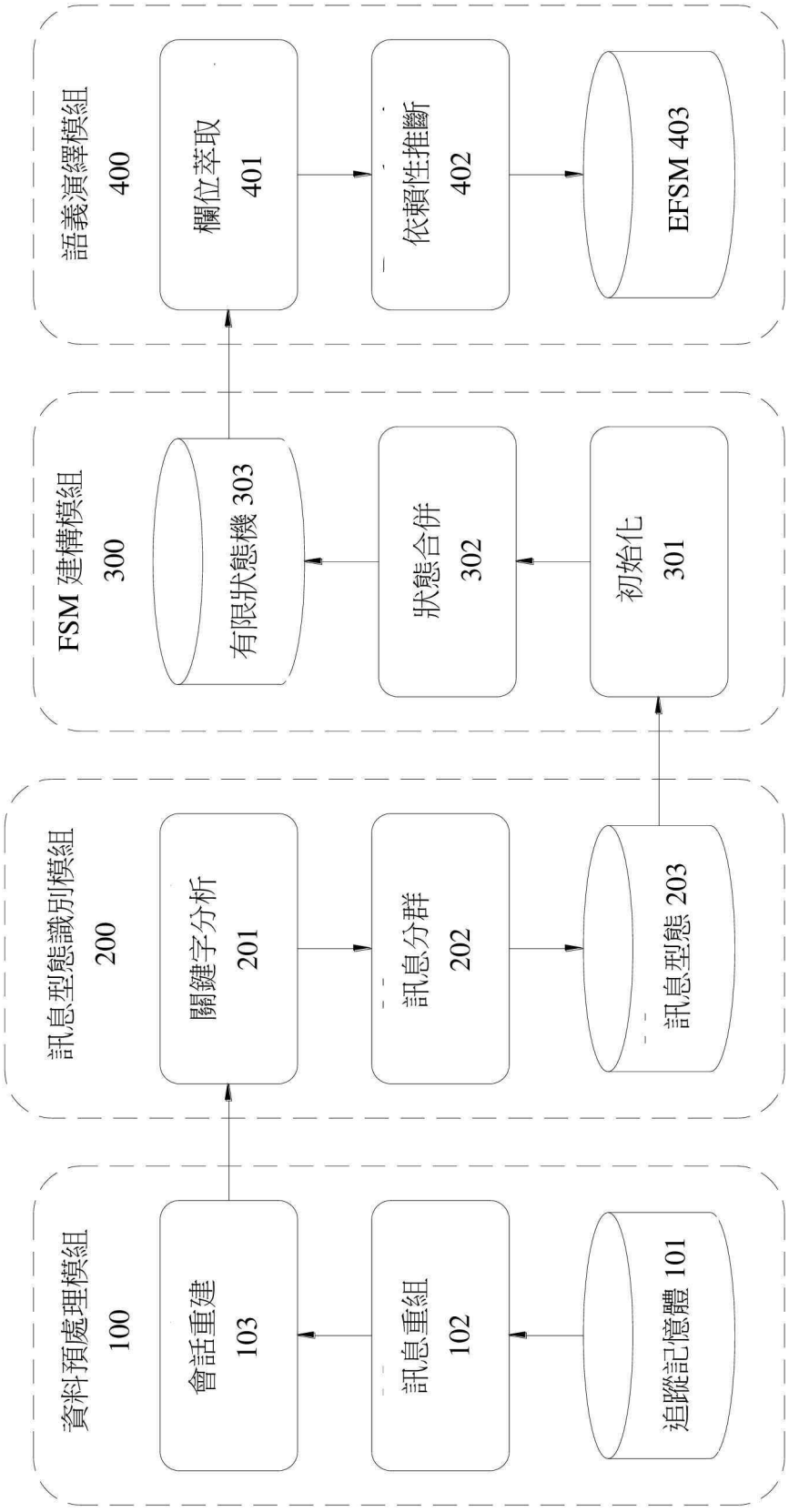
【第6項】如申請專利範圍第 1 項所述的自動協議測試方法，其中該等效狀態由相同的 k-tail 轉換合併。

【第7項】如申請專利範圍第 1 項所述的自動協議測試方法，其中該協議訊息由 Needleman 和 Wunch 序列比對處理以萃取該欄位。

【第8項】如申請專利範圍第 1 項所述的自動協議測試方法，其中在該協議訊息中的該欄位的該值由 Daikon 演算法保持以推斷該約束。

【第9項】如申請專利範圍第 1 項所述的自動協議測試方法，其中在該欄位的該約束包括檢查該欄位的有效值、訊息內依賴性和訊息間依賴性。

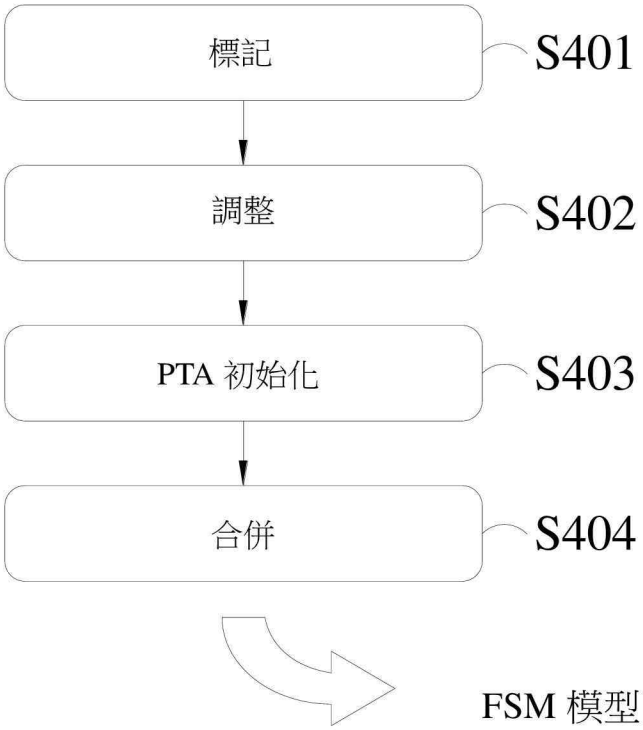
【發明圖式】


資料預處理模組
100
會話重建
103
訊息重組
102
追蹤記憶體 101
訊息型態識別模組
200
關鍵字分析
201
訊息分群
202
訊息型態 203
FSM 建構模組
300
有限狀態機 303
狀態合併
302
初始化
301
語義演繹模組
400
欄位萃取
401
依賴性推斷
402
EFSM 403

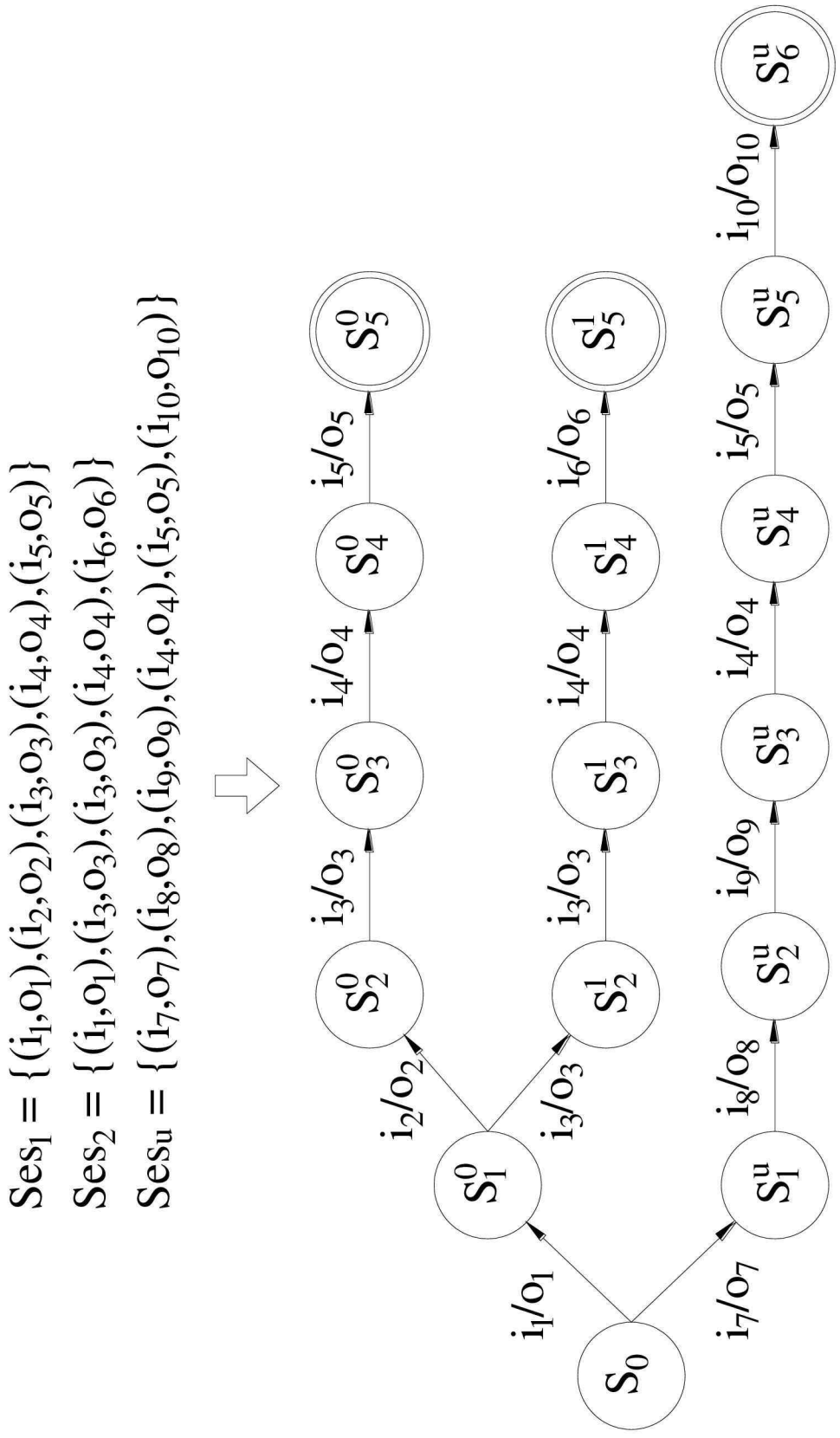
第 1 圖


依列輸入包含複數個資料封包的流量追蹤
S1
解析複數個封包以萃取複數個會話並重構複數個會話以獲得協議訊息
S2
對協議訊息進行關鍵字分析和分群演算法以識別複數種訊息類型
S3
初始化協議訊息以形成初始會話序列並合併等效狀態以獲得具有一組狀態、一組轉換、一組輸入和一組輸出的有限狀態機
S4
萃取協議訊息的欄位和值以獲得複數個子資料集，並在欄位上添加約束以產生資料保護，在有限狀態機上推斷資料保護和記憶體集以獲得擴展有限狀態機
S5

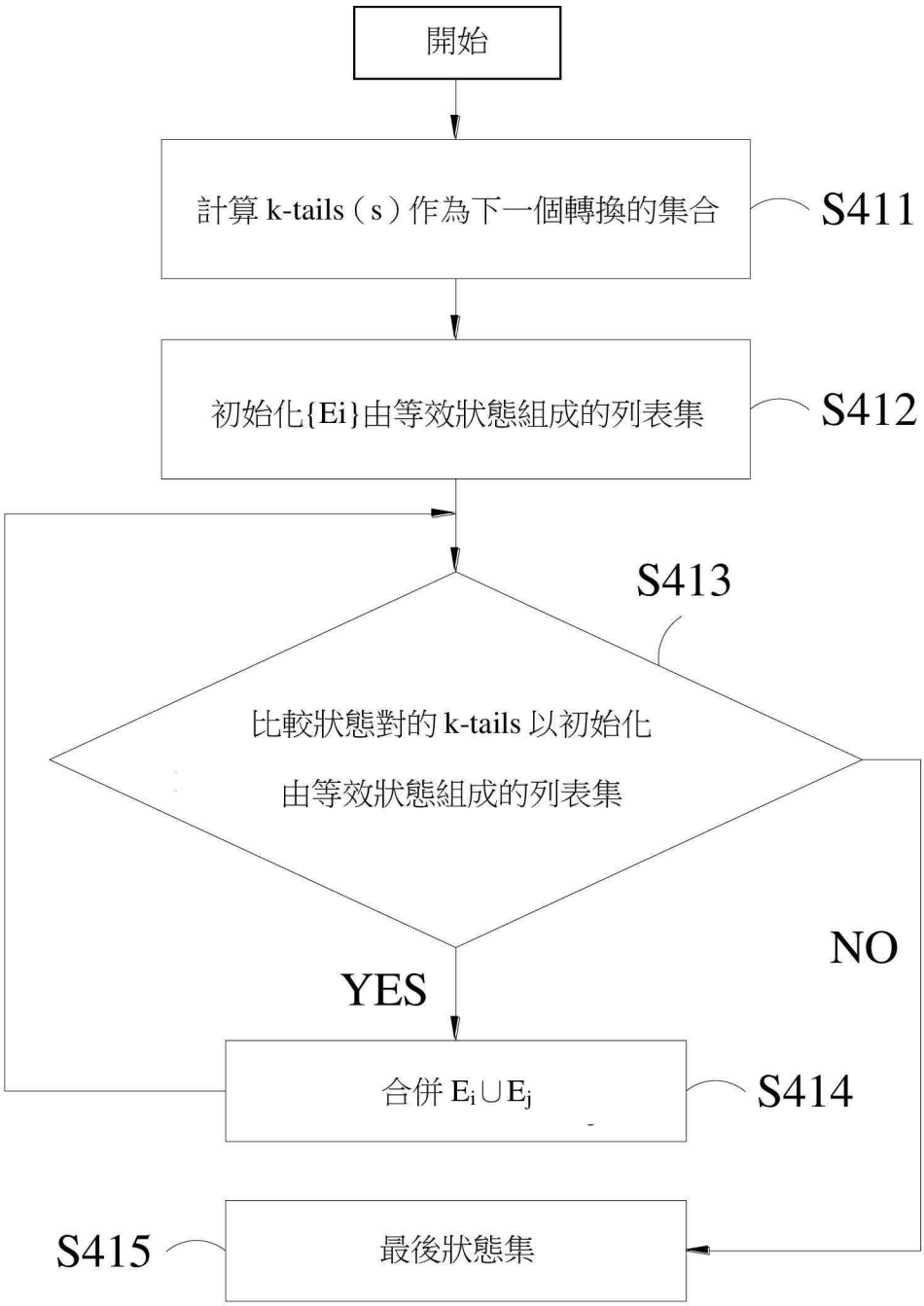
第 2 圖



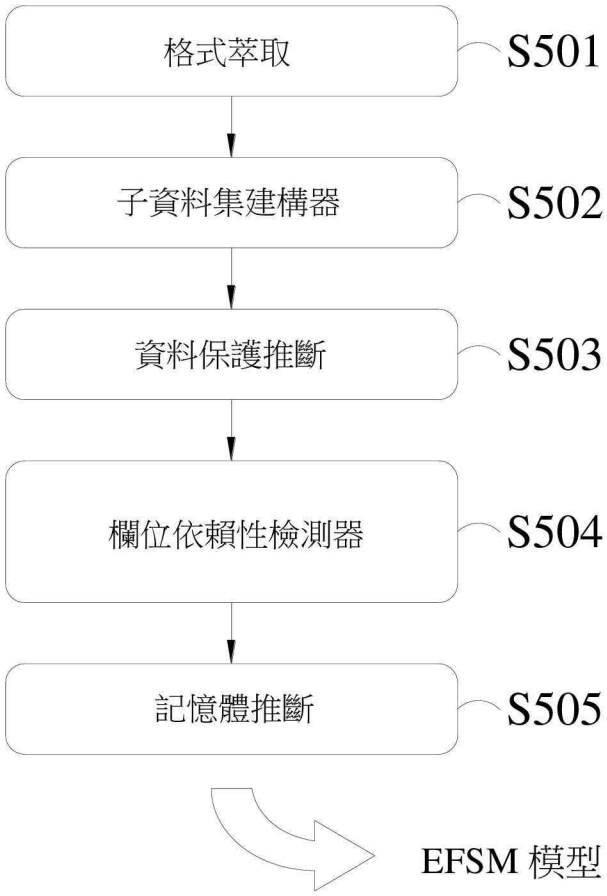
第 3 圖


Ses1 = {(i1,o1),(i2,o2),(i3,o3),(i4,o4),(i5,o5)}
Ses2 = {(i1,o1),(i3,o3),(i3,o3),(i4,o4),(i6,o6)}
Sesu = {(i7,o7),(i8,o8),(i9,o9),(i4,o4),(i5,o5),(i10,o10)}
S0
i1/o1
i7/o7
S1 0
S1 u
i2/o2
i3/o3
i8/o8
i3/o3
i9/o9
S2 0
S2 1
S2 u
i3/o3
i4/o4
i4/o4
i4/o4
S3 0
S3 1
S3 u
i5/o5
i6/o6
i5/o5
S4 0
S4 1
S4 u
i10/o10
S5 0
S5 1
S5 u
S6 u

第 4 圖


開始
計算 k-tails（s）作為下一個轉換的集合
S411
初始化{Ei}由等效狀態組成的列表集
S412
S413
比較狀態對的 k-tails 以初始化
由等效狀態組成的列表集
NO
YES
合併 $E_i \cup E_j$
S414
S415
最後狀態集

第 5 圖


格式萃取
S501
子資料集建構器
S502
資料保護推斷
S503
欄位依賴性檢測器
S504
記憶體推斷
S505
EFSM 模型

第 6 圖

序列
埠 152,79,91,8,243,24
埠 130,70,195,93,13,168
埠 99,152,5,130,130,143
N&W
比對序列
埠 152,79_,91_,89_,243,24_
埠 130,70_,195,93_,13_,168
埠 99_,152,5__,130,130,143
共識:
埠 ___,___,___,___,___
埠
埠 f1 , f2 , f3 , f4 , f5 , f6
動態欄位
靜態欄位
子-資料集 (f1) = {152,130,99 ...}
子-資料集 (f2) = {79,70,152 ...}
子-資料集 (f3) = {91,195,5 ...}
子-資料集 (f4) = {8,93,130 ...}
子-資料集 (f5) = {243,13,130 ...}
子-資料集 (f6) = {24,168,143 ...}

第 7 圖

埠 '/200
$(f_1=152, f_2=79, f_3=91, f_4=8, f_5=243, f6=24),$
$(f_1=130, f_2=70, f_3=195, f_4=93, f_5=13, f6=168),$
…
…
$(f_1=99, f_2=152, f_3=5, f_4=130, f_5=130, f6=143),$
S_0
S_1
Daikon
埠 '/200
$0 \leq f_1 < 256$
$0 \leq f_2 < 256$
$0 \leq f_3 < 256$
$0 \leq f_4 < 256$
$0 < f_5 < 256$
$0 < f_6 < 256$
S_0
S_1
型態 '/200
arg="T",arg="T",arg
="A",....arg="T"
S_0
S_1
Daikon
型態 '/200
arg∈{"A","T"}
S_0
S_1

第 8 圖