ROMÂNIA

(19) OFICIUL DE STAT
PENTRU INVENŢII ŞI MĂRCI
Bucureşti

(11) **RO 132010 A2**

(51) Int.Cl.
G06F 11/30 (2006.01);
H04L 12/26 (2006.01)

(12) # CERERE DE BREVET DE INVENŢIE

(21) Nr. cerere: **a 2015 01028**

(22) Data de depozit: **22/12/2015**

(41) Data publicării cererii:
30/06/2017 BOPI nr. **6/2017**

(71) Solicitant:
• **IXIA**, *A CALIFORNIA CORPORATION, 26601 WEST AGOURA ROAD, CALABASAS, CALIFORNIA, US*

(72) Inventatori:
• **NISTUR MARIUS PAVEL**, *STR.MITROPOLITUL VARLAAM NR.88, AP.4, SECTOR 1, BUCUREŞTI, B, RO;*
• **NICULESCU SILVIU IONUŢ**, *ALEEA DIHAM NR.1, BL.B16, SC.1, ET.7, AP.47, BUCUREŞTI, B, RO;*
• **ŞTEFAN ALEXANDRU-BOGDAN**, *STR.GRIGORE IONESCU NR.83, BL.44, AP.5, SECTOR 2, BUCUREŞTI, B, RO*

(74) Mandatar:
RATZA ŞI RATZA SRL, *B-DUL A.I. CUZA, NR. 52-54, SECTOR 1, BUCUREŞTI*

(54) # METODE, SISTEME ŞI SUPORT CITIBIL DE CALCULATOR PENTRU DIAGNOSTICAREA REŢELEI

(57) **Rezumat:**

Invenţia se referă la o metodă, un sistem şi un suport citibil de către calculator, destinate diagnosticării unei reţele de calculatoare. Metoda de diagnosticare a reţelei, conform invenţiei, constă în configurarea unei multitudini de noduri de diagnosticare (**108**, **110**, **112**) pentru a observa comportamentul traficului de date asociat cu un sistem de testat (**SUT**), observarea comportamentului traficului de date asociat cu sistemul de testat (**SUT**) şi detectarea problemei acestuia, identificarea, folosind informaţii despre topologia sistemului de testat (**SUT**), a unui nod de reţea asociat cu problema sistemului de testat, declanşarea unuia dintre nodurile de diagnosticare (**108**, **110**, **112**) pentru a obţine informaţii referitoare la nod, de la nodul de reţea, în care nodul de diagnosticare foloseşte cel puţin un protocol de comunicaţie pentru a interoga nodul de reţea cu privire la informaţiile legate de acesta, şi diagnosticarea problemei sistemului de testat (**SUT**) folosind datele referitoare la nod. Sistemul de diagnosticare a reţelei, conform invenţiei, cuprinde cel puţin un procesor şi un controler de diagnosticare (**102**) implementat utilizând cel puţin un procesor, în care controlerul de diagnosticare (**102**) este configurat să execute metoda conform invenţiei. Suportul citibil de către calculator, conform invenţiei, are stocate instrucţiuni executabile de calculator care, atunci când sunt executate de un procesor, comandă calculatorul să efectueze etapele metodei conform invenţiei.

Revendicări: 20
Figuri: 5

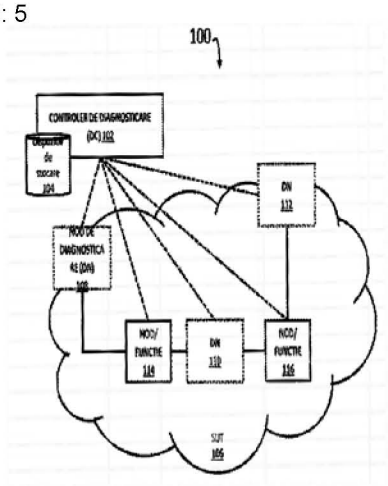

Fig. 1

RO 132010 A2

METODE, SISTEME ȘI SUPORT CITIBIL DE CALCULATOR PENTRU DIAGNOSTICAREA REȚELEI

Domeniul tehnic de aplicare

Prezenta invenție se referă la testarea rețelelor de calculatoare. Mai precis, prezenta invenție se referă la metode, sisteme și suport citibil de calculator pentru diagnosticarea rețelei.

Stadiul anterior al tehnicii

În general, operatorii de rețea testează nodurile de rețea înainte de implementarea acestora în rețelele reale. De exemplu, un operator de rețea mobilă poate testa o nouă generație de echipamente de comunicații folosind o rețea de testare și/sau mai multe platforme de testare înainte de implementarea echipamentului de comunicații într-o rețea reală (cum ar fi, non-testată). Cu toate acestea, chiar și cu testare prealabilă, uneori rețelele reale pot prezenta diverse aspecte care trebuie să fie depanate sau diagnosticate și rezolvate rapid. Pentru a diagnostica și rezolva astfel de probleme poate fi costisitor, se consumă timp și resurse. Mai mult decât atât, aceste probleme pot afecta eficiența utilizatorului final și/sau pot cauza întreruperi de rețea.

În consecință, există o nevoie de metode, sisteme și suport citibil de calculator îmbunătățite pentru diagnosticarea rețelei.

Expunerea pe scurt a invenţiei

Sunt dezvăluite metode, sisteme și suport citibil de calculator pentru diagnosticarea rețelei. Conform unei metode, aceasta este realizată la un controler de diagnosticare implementat, folosind cel puțin un procesor. Metoda include configurarea unei multitudini de noduri de diagnosticare pentru a observa comportamentul traficului de date asociat cu un sistem de testat (SUT). Metoda include, de asemenea, observarea, folosind nodurile de diagnosticare, a comportamentul traficului de date asociat cu SUT. Metoda include suplimentar detectarea, folosind comportamentul traficului de date, a problemei SUT. Metoda include, de asemenea identificarea, folosind informații despre topologia SUT, a unui nod de rețea în SUT asociat cu problema SUT. Metoda include suplimentar declanșarea unuia dintre nodurile de diagnosticare astfel încât să se

obţină informaţii referitoare la nod de la nodul de reţea, în care nodul de diagnosticare foloseşte cel puţin un protocol de comunicaţie pentru a interoga nodul de reţea despre informaţiile legate de acesta. Metoda include, de asemenea, diagnosticarea, folosind datele referitoare la nod, problemei SUT.

Conform unui sistem, acesta conţine un controler de diagnosticare implementat care foloseşte cel puţin un procesor. Controlerul de diagnosticare este configurat pentru a configura o multitudine de noduri de diagnosticare pentru a observa comportamentul traficului de date asociate cu un SUT, pentru a observa, folosind nodurile de diagnosticare, comportamentul traficului asociat cu SUT, pentru a detecta, folosind comportamentul de trafic, o problemă SUT, pentru a identifica, folosind informaţii despre topologia SUT, un nod de reţea în SUT asociat cu problema SUT, pentru a declanşa unul dintre nodurile de diagnosticare ca să se obţină informaţiile referitoare nod de la nodul de reţea, în care nodul de diagnosticare foloseşte cel puţin un protocol de comunicaţii pentru a interoga nodul reţelei despre informaţiile referitoare la nod şi pentru a diagnostica, folosind informaţiile referitoare nod, problema SUT.

Obiectele prezentei invenţii pot fi puse în aplicare de software-ul în combinaţie cu hardware şi/sau firmware. De exemplu, obiectele descrise aici pot fi implementate de software-ul executat de un procesor. Conform unui exemplu de realizare, obiectele prezentei invenţii pot fi implementate folosind un suport non tranzitoriu citibil de calculator care are stocate instrucţiuni executabile de calculator, astfel că atunci când sunt executate de procesor, comandă calculatorul să efectueze etapele. În mod exemplar, suportul citibil de calculator adecvat pentru punerea în aplicare a obiectelor prezentei invenţii, include dispozitive non-tranzitorii, cum ar fi dispozitive cu memorie disc, dispozitive cu memorie cip, dispozitive logice programabile şi circuite integrate specifice. În plus, un suport care poate fi citit de calculator care implementează prezenta invenţie poate fi dispus pe un singur dispozitiv sau platformă informatică sau poate fi distribuit pe mai multe dispozitive sau platforme de calcul.

Aşa cum este utilizat aici, termenul "nod" se referă la o platformă informatică fizică care include cel puţin un procesor, o interfaţă de reţea şi memorie.

Aşa cum sunt utilizaţi aici, fiecare din termenii "funcţie", "motor" şi "modul" se referă la hardware care de asemenea poate include software şi/sau firmware pentru implementarea problemei(lor) descrise aici.

Prezentarea pe scurt a desenelor explicative

Se dă în continuare mai multe exemple de realizare a invenției, în legătură cu Figurile 1 – 5 care reprezintă:

- Figura 1 este o diagramă care prezintă un mediu pentru diagnosticare de rețea, în conformitate cu un exemplu de realizare a prezentei invenții;
- Figurile 2A-2B sunt diagrame care prezintă un algoritm de diagnosticare, în conformitate cu un exemplu de realizare a prezentei invenții;
- Figura 3 este o diagramă care prezintă comunicații asociate cu diagnosticarea de rețea, în conformitate cu un exemplu de realizare a prezentei invenții;
- Figura 4 este o diagramă care prezintă comunicații asociate cu diagnosticarea de rețea, conform unui alt exemplu de realizare a prezentei invenții; și
- Figura 5 este o diagramă care prezintă un metodă de diagnosticare de rețea, în conformitate cu un exemplu de realizare a prezentei invenții.

Descrierea detaliată

Sunt dezvăluite metode, sisteme și suport citibil de calculator pentru diagnosticare rețelei. Diagnosticarea rețelei necesită de obicei un operator de rețea care să execute manual numeroase teste și încercări pentru a diagnostica caracteristicile sau problemele pe baza rezultatelor testelor și informațiilor de configurare cunoscute. Cu toate acestea, deoarece astfel de diagnosticări sunt realizate manual și pentru că unele probleme nu sunt evidente din rezultatele testelor specifice, de obicei, aceste diagnosticări de rețea necesită un volum semnificativ de lucru și pot fi mari consumatoare de timp, ineficiente și predispuse la erori.

În conformitate cu unele aspecte ale prezentei invenții, sunt descrise tehnici, metode sau mecanisme pentru automatizarea diagnosticării problemelor de rețea și pentru a preveni astfel de probleme prin simularea de trafic real și prin utilizarea rezultatelor pentru depanarea unui sistem de testat (SUT) (de exemplu, unul sau mai multe dispozitive de rețea). De exemplu, un instrument de diagnosticare sau un algoritm corespunzător poate identifica cu precizie o cauza inițială a uneia sau mai multor probleme, printr-o abordare non-intruzivă în monitorizarea unei rețele reale, cum ar fi aceea care utilizează script-uri (scenarii) pentru a obține informații de stare și/sau alte

informații referitoare la dispozitiv din dispozitivele de rețea care utilizează unul sau mai multe protocoale de comunicații suportate de dispozitivele de rețea.

În mod avantajos, în conformitate cu unele aspecte ale prezentei invenții, depanarea și/sau rezolvarea problemelor rețelelor pot fi îmbunătățite prin utilizarea de noduri de diagnosticare care execută unul sau mai multe script-uri (scenarii de rețea) pentru obținerea fermă și în mod automat a informațiilor de stare și/sau a altor informații legate de dispozitiv de la unul mai multe dispozitive (de exemplu, noduri de și/sau funcții de rețea). Prin utilizarea script-urilor și/sau a unui sau mai mulți algoritmi de diagnosticare, o rețea mare care conține un număr semnificativ de dispozitive poate fi diagnosticată rapid cu puțină sau deloc intervenție umană, reducând astfel costurile și timpii morți în rețea.

Figura 1 este o diagramă care prezintă un mediu **100** informatic pentru diagnosticare de rețea, în conformitate cu un exemplu de realizare a obiectului prezentei invenții. Referitor la Figura 1, mediul **100** informatic poate include un controler **102** de diagnosticare, un dispozitiv **104** de stocare a datelor, nodurile **108-112** de diagnosticare (DN-uri) și un sistem **106** de testat (SUT) care include nodurile și/sau funcțiile de rețea (noduri/funcții) **114 -116**.

Controlerul **102** de diagnosticare poate reprezenta orice entitate adecvată (de exemplu, una sau mai multe platforme informatice sau un instrument de testare implementate folosind cel puțin un procesor) pentru efectuarea diverselor aspecte legate de diagnosticarea sau rezolvarea problemelor asociate cu SUT **106**. Conform unor exemple de realizare concrete, controlerul **102** de diagnosticare poate, de asemenea, efectua testarea SUT **106** sau poate să fie implicat în testarea SUT **106**.

În unele exemple de realizare, controlerul **102** diagnosticare poate configura unul sau mai multe noduri DN **108-112** pentru a genera și transmite trafic de testare la SUT **106** și/sau pentru a observa traficul în rețea, pentru a genera metrici de legate de trafic și pentru a efectua diagnosticarea pe una sau mai multe porțiuni ale SUT **106**. De exemplu, controlerul **102** de diagnosticare și/sau DN-urile **108-112** pot include funcționalități pentru utilizarea script-urilor de testare pentru configurarea uneia sau mai multor testări ale SUT **106** și pot utiliza, de asemenea, script-uri de diagnosticare (de exemplu, într-o interfață de linie de comandă (CLI) legată de limbajul de criptare) ca să se obțină de date de la nodurile sau funcțiile de rețea în SUT **106**. În acest

exemplu, fiecare script de diagnosticare poate utiliza diferite sintaxe, comenzi și/sau limbaj de programare, pe baza unui tip de nod care va fi diagnosticat și/sau pe baza protocolului sau interfeței de comunicații utilizate.

Dispozitivul **104** de stocare a datelor poate reprezenta orice entitate adaptată (de exemplu, un suport non-tranzitoriu care poate fi citit de calculator, memoria încorporată sau un dispozitiv de memorie) pentru stocarea datelor asociate cu testarea și/sau diagnosticarea. De exemplu, dispozitivul **104** de stocare a datelor poate stoca diverse script-uri de diagnosticare, script-uri de testare, caracteristici de funcționare SUT (de exemplu, informații de configurare, informații despre capacitatea de performanță, informații de resurse interne etc.), informații de conectivitate SUT (de exemplu, informații despre starea legăturii de comunicație, informații despre port, etc), informații de interfață de comunicație legate de SUT (de exemplu, informații sub protocolul de interfață de comunicații referitor la managementul rețelei SUT, informații sub protocolul de management rețea simplă (SNMP), informații sub protocolul de rețea criptografic (SSH), informații sub protocolul de rețea care se folosește în Internet precum și în rețele de calculatoare tip LAN la comunicația textuală, bidirecțională și interactivă, Telnet, informații sub protocolul de transfer hypertext (HTTP), informații sub protocolul de arhitectură software a rețelei (REST) informații despre interfața pentru programarea de aplicații (API), informații despre proprietatea interfeței pentru programarea de aplicații API, etc.). În unele exemple de realizare, dispozitivul **104** de stocare a datelor poate fi accesat de către controlerul de diagnosticare și/sau de alte entități. În unele exemple, dispozitivul **104** de stocare a datelor poate fi amplasat pe o singură platformă sau dispozitiv informatic sau poate fi distribuit pe mai multe dispozitive sau platforme informatice.

În unele exemple de realizare, controlerul **102** de diagnosticare poate configura un SUT **106** și poate să valideze apoi SUT **106** sau o porțiune a acestuia, cum ar fi elementele individuale de rețea sau de noduri/funcții. În astfel de realizări, această configurație poate, opțional, să fie realizată utilizând un controler SDN sau un controler de proprietate care are una sau mai multe interfețe de programare de aplicații (API) expuse și care este susținută de controlerul **102** de diagnosticare.

În unele exemple de realizare de realizare, controlerul **102** de diagnosticare poate primi informații despre o configurare preexistentă a SUT (de exemplu, de la un

controler de rețea sau de la o bază de date aferentă) și poate configura DN-rile **108-112** pentru validarea configurației SUT. În unele exemple de realizare, controlerul **102** de diagnosticare poate modifica anumite aspecte ale unei configurații pentru diverse scopuri de testare și măsurare. În astfel de exemple de realizare, controlerul **102** de diagnosticare poate schimba configurația modificată înapoi la configurația sa inițială.

SUT **106** poate include nodurile/funcțiile **114-116** și/sau alte entități pentru realizarea uneia sau mai multor funcțiuni sau servicii. De exemplu, SUT **106** poate include nodurile/funcțiile **114-116** pentru conversia pachetelor de date de tip VoIP de la un format la alt format, în cazul în care fiecare nod/funcție în SUT **106** îndeplinește una sau mai multe funcții asociate cu transformarea pachetelor de tip VoIP. Într-un alt exemplu, SUT **106** poate include nodurile/funcțiile **114-116** pentru furnizarea de comunicații media între două rețele. În acest exemplu, nodul/funcția **116** poate îndeplini o funcție firewall și nodul/funcția **114** poate îndeplini o funcție de server media. Fiecare dintre nodurile/funcțiile **114-116** poate reprezenta orice entitate adecvată (de exemplu, software-ul stocat într-o memorie și/sau care este executat folosind cel puțin un procesor) pentru efectuarea uneia sau mai multor funcții de rețea. Fiecare dintre nodurile/funcțiile **114-116** poate fi o construcție logică implementată folosind hardware sau resursele fizice de la una sau mai multe locații, dispozitive și/sau platforme. De exemplu, folosind cel puțin un procesor de pe o primă platformă informatică sau de pe un computer dedicat să fie utilizat ca și server și desemnat să fie instalat într-un cadru de lucru (server rack) și de pe memoria de la o a doua platformă informatică sau server rack, nodul/funcția **116** poate îndeplini funcții de server web, de exemplu, care recepționează cererile prin protocolul de transfer hypertext (HTTP) și care furnizează răspunsuri HTTP. Într-un alt exemplu, folosind procesoarele și memoria de la o aceeași platformă informatică sau server rack, nodul/funcția **114** și nodul/funcția **116** pot îndeplini funcții de criptare și respectiv, de decriptare.

Fiecare dintre DN **108-112** poate reprezenta orice entitate adecvată (de exemplu, software-ul stocat într-o memorie și/sau executat folosind cel puțin un procesor) pentru efectuarea diverselor aspecte legate de testarea SUT **106** și/sau de diagnosticare ale diferitelor comunicații sau rețele potențiale, asociate cu SUT **106**. De exemplu, DN **108** poate acționa ca un punct final (de exemplu, o aplicaţie pentru calculator care imită funcționarea unei unităţi optice CD/DVD (daemon) sau un serviciu care se execută de

cel puțin un procesor) care primește de la controlerul de diagnosticare informații de configurare testare sau un controler de testare și generează trafic în funcție de informațiile de configurare testare recepționate.

În unele exemple de realizare, fiecare dintre DN **108-112** poate fi implementat folosind hardware sau resurse fizice de la una sau mai multe locații, dispozitive, și/sau platforme. De exemplu, DN **110** poate fi implementat pe o platformă informatică aceeași ca și DN **112** și nodul/funcția **116**. Într-un alt exemplu, DN **110** poate fi implementat pe platforme informatice diferite de DN **112** și de nodul/funcția **116**.

În unele exemple de realizare, DN-urile **108-112** pot testa diverse noduri/funcții, de exemplu, un nod/funcție de echilibrarea încărcării, un nod/funcție de detectare a intruziunilor, un nod/funcție de protecție la intruziune, un nod /funcție antivirus, un nod/funcție firewall, un nod/funcție antispam, un nod de comutație sau un nod/funcție de rutare. În unele exemple de realizare, DNS **108-112** poate acționa sau apărea altor entități, cum ar fi DN-urile, nodurile, funcțiile, rețele sau calculatoarele principale. În unele exemple de realizare, testarea DN **108-112** poate fi transparentă sau necunoscută de alte entități (de exemplu, nodurile/funcțiile **114-116**), în SUT **106**.

În unele exemple de realizare, DN-urile **108-112** pot fi configurate pentru a genera și/sau transmite trafic de testare pentru testarea SUT **106** sau a unui sau mai multor dintre nodurile/funcțiile **114-116**. În unele exemple de realizare, informațiile de generare trafic pot fi determinate pe baza cerințelor de testare, a funcționalității unui nod/funcție și/sau a altor factori. De exemplu, DN **108** poate fi configurat să acționeze ca un generator de trafic pentru testarea nodului/funcției **114** care îndeplinește o funcție firewall. În acest exemplu, controlerul **102** de diagnosticare poate configura DN **108** pentru a genera trafic care pare să provină de la diferite rețele, inclusiv rețelele locale și de la distanță. Continuând cu acest exemplu, DN **108** poate transmite traficul generat de nodul/funcția **114**, caz în care un DN **110** ulterior poate analiza ieșirea de la nodul/funcția **114**.

În unele exemple de realizare, DN-urile **108-112** pot fi configurate pentru validarea de la un capăt la altul a funcționalității SUT **106**. De exemplu, DN **108** poate fi situat la un început de SUT **106** și DN **112** poate fi situat la un capăt final al SUT **106**. În acest exemplu, DN **108** poate transmite trafic de testare printr-un port de ieșire la nodul/funcția **114** și apoi, după prelucrare, nodul/funcția **114** poate transmite trafic la

nodul/funcția **116** prin DN **110** înainte de a fi trimis de la nodul/funcția **116** la DN **112**. Continuând cu acest exemplu, DN **112** poate analiza ieșirea pentru erori sau alte probleme potențiale, pe baza rezultatelor de ieșire așteptate pentru traficul de testare. În unele exemple de realizare, DN-urile **108-112** pot fi configurate să acționeze un dispozitiv hardware care oferă o modalitate de a accesa datele care trec printr-o rețea de calculatoare pentru monitorizarea traficului și/sau observarea comportamentului rețelei. De exemplu, controlerul **102** de diagnosticare poate configura DN **110** pentru a acționa ca un dispozitiv hardware care oferă o modalitate de a accesa datele care trec printr-o rețea în SUT **106,** pentru analiza de trafic, copierea traficului, generarea metricilor legate de trafic și/sau altor funcții legate de dispozitivul hardware de rețea.

În unele exemple de realizare, DN **108-112** pot fi configurate pentru a efectua funcții de diagnosticare. De exemplu, fiecare dintre DN **108-112** poate fi configurat cu un motor sau un algoritm de diagnosticare capabile de diagnosticarea problemelor asociate cu SUT **106** care comunică cu unul sau mai multe noduri/funcții **114-116** folosind unul sau mai multe protocoale de management sau de comunicații, programe sau interfețe de comunicații. În acest exemplu, algoritmul de diagnosticare poate încerca să comunice utilizând o multitudine de protocoale de comunicații (de exemplu, un protocol de management SNMP sau HTTP) sau mecanisme și poate învăța sau se poate adapta singur în funcție de setările preconfigurate, datele istorice și/sau preferințele utilizatorilor.

În unele exemple de realizare, un motor de diagnosticare sau un algoritm corespunzător poate accesa și interpreta scripturi de diagnosticare capabile să comunice și/sau să facă diagnosticarea problemelor asociate cu diferite tipuri de noduri/funcții. De exemplu, un algoritm de diagnosticare care se execută la DN **108** poate accesa și utiliza diverse scripturi de diagnosticare pentru a comunica (sau pentru a încerca să comunice) cu diferite noduri folosind diferite protocoale de comunicație. În acest exemplu, un prim script de diagnosticare poate fi declanșat de către algoritmul de diagnosticare atunci când se încearcă să se comunice cu ajutorul unui protocol SNMP, un al doilea script de diagnosticare poate fi declanșat de către algoritmul de diagnosticare atunci când se încearcă să se comunice cu ajutorul unui protocol SSH, și tot un al doilea script de diagnosticare poate fi declanșat de către algoritmul de diagnosticare atunci când se încearcă să se comunice cu ajutorul unui protocol HTTP,

un protocol Cmd (protocolul de comenzi IP folosind linia de comandă) sau un protocol criptografic de rețea Bash.

În unele exemple de realizare, un script de diagnosticare poate fi dispozitivul specific. De exemplu, un prim script de diagnosticare ar putea fi pentru comunicație și pentru diagnosticarea problemelor asociate cu nodul/funcția **114** (de exemplu, un nod/funcție firewall sau un dispozitiv de comutație) și un al doilea script de diagnosticare poate fi pentru comunicație și pentru diagnosticarea problemelor asociate cu nodul/funcția **116** (de exemplu, un server web). În acest exemplu, fiecare script de diagnosticare poate solicita diferite tipuri de informații, în funcție de funcționalitatea sau capacitățile nodului/funcției ce va fii diagnosticat.

În unele exemple de realizare, DN-urile **108-112** pot fi configurate pentru conectivitatea dinamică (de exemplu, inserarea, activarea sau dezactivarea). De exemplu, DN **108** poate reprezenta o imagine virtuală (de exemplu, un client virtual), care poate fi introdusă în mod dinamic în jurul sau în SUT **106** pe baza diverșilor factori, cum ar fi cerințele de testare, problemele detectate de trafic, condițiile de rețea sau momentul zilei.

În unele exemple de realizare, după ce DN **108-112** sunt instalate în mediul **100** informatic, fiecare dintre DN **108-112** se poate înregistra cu controlerul **102** de diagnosticare sau cu un sistem corespunzător. În unele exemple de realizare, fie în mod automat, fie pe baza datelor introduse de utilizator, controlerul **102** de diagnosticare poate crea o topologie a SUT **106** și poate iniția testarea lui folosind diferite scripturi de trafic la un anumit interval de timp. De exemplu, fiecare script de trafic poate implica diferite tipuri de performanță a traficului tranzitat, de exemplu, care include trafic de voce, video și amestecuri de aplicații.

În unele exemple de realizare, după o fază de simulare a traficului, controlerul **102** de diagnosticare poate fi capabil să determine cu precizie care segment de rețea are probleme de rețea. De exemplu, DC poate primi metrici referitoare la trafic și/sau alte informații de la DN-urile **108-112** și poate analiza aceste informații pentru a determina dacă anumite rute, noduri, funcții și/sau tipuri de trafic nu au acționat cum era de așteptat sau anticipat, de exemplu, pe baza topologiei cunoscute, a informațiilor istorice, a cerințelor sau pragurilor de testare, etc.

În unele exemple de realizare, după detectarea unei sau mai multor probleme, informațiile relevante (de exemplu, informațiile referitoare la testare) pot fi trimise la DN-uri (de exemplu, DN **108-112**) care sunt la margini sau într-un anumit traseu sau segment de rețea problematice. Utilizând informațiile recepționate de la controlerul **102** de diagnosticare și/sau alte informații relevante, fiecare dintre DN-uri poate utiliza un motor sau un algoritm de diagnosticare pentru efectuarea de diagnosticări în încercarea de a identifica cauza problemelor detectate și de a oferi soluții potențiale. Detalii suplimentare cu privire la un exemplu de algoritm de diagnosticare este discutat mai jos în legătură cu Figura 2.

Se va aprecia că Figura 1 este pentru scopuri ilustrative și că diferite entități descrise, locațiile și/sau funcțiile lor descrise mai sus în legătură cu Figura 1 pot fi modificate, alterate, adăugate sau eliminate. De exemplu, un dispozitiv (de exemplu, un calculator care include cel puțin un procesor cuplat la o memorie) poate include funcționalitatea nodului/funcției **114** și DN **108**.

Figurile 2A-2B reprezintă o diagramă care prezintă un algoritm de diagnosticare în conformitate cu un exemplu de realizare a obiectului descris aici. În unele exemple de realizare, un algoritm de diagnosticare, menționat ca și un algoritm de depanare, poate include logica de program pentru depanarea problemelor asociate cu SUT **106** sau cu nodurile/funcțiile acestuia. De exemplu, un algoritm de diagnosticare poate utiliza unul sau mai multe protocoale de comunicații și/sau script-uri automate pentru comunicația cu unul sau mai multe noduri/funcții (de exemplu, nodurile/funcțiile **114-116**).

Conform Figurii 2A, în etapa **201**, datele de intrare de la testare, de exemplu, metricile legate de trafic, informațiile de stare și/sau alte informații asociate cu SUT **106**, pot fi utilizate pentru a deduce, deriva sau determina diverse caracteristici de intrare pentru efectuarea de diagnosticări referitoare la SUT. De exemplu, intrările se pot baza pe rezultatele testelor și/sau pe alte informații și pot include informații prin Internet Protocol (IP) sau prin protocoalele de trafic utilizate (de exemplu, protocolul de control al transmisiei (TCP) și/sau prin protocolul datagramelor de utilizator (UDP)), precum și informații despre problemele potențiale (probleme de conectivitate, probleme de randament, de întârziere de rețea sau de depășire a unui prag acceptabil de bruiaj (jitter) din exterior, întreruperi de conexiuni, semnale proaste Wi-Fi, etc).

În etapa **202**, se poate determina dacă a avut loc o problemă de conectivitate, probabil în timpul testării. De exemplu, atunci când un transfer de date măsurat între punctele finale este mai mic decât o valoare prezisă, pot fi efectuate etape suplimentare pentru a verifica SUT sau o conexiune a unui dispozitiv corespunzător.

În unele exemple de realizare, dacă o problemă de conectivitate a avut loc probabil în timpul testării, se poate efectua etapa **203**. În unele exemple de realizare, dacă o problemă de conectivitate a nu a avut loc în timpul testării, se poate realiza etapa **227**.

În etapa **227**, poate fi analizată și/sau verificată disponibilitatea traficului. De exemplu, algoritmul **200** de diagnosticare care se execută la DN **108** poate iniția o comandă "traceroute" (urmărire traseu) de la punctul final local și poate analiza răspunsul.

În etapa **228**, poate fi detectat și raportat un dispozitiv care cauzează problema. De exemplu, algoritmul **200 de** diagnosticare care se execută la DN **108** poate utiliza un răspuns la o comandă "traceroute" pentru a detecta un dispozitiv care cauzează probleme de rețea, cum ar fi un dispozitiv ce acționează ca un obstacol de trafic.

La etapa **203**, poate fi analizată și/sau verificată conectivitatea (sau lipsa acesteia) la un dispozitiv local (de exemplu, nodul/funcția **114)**.

În etapa **204**, se poate determina dacă există o problemă de conectivitate care implică dispozitivul local. De exemplu, o comandă "ping" poate fi realizată utilizând o adresă IP asociată cu dispozitivul local. În acest exemplu, în cazul în care comanda "ping" are succes, se poate stabili că nu există probleme de conectivitate, dar în cazul în care comanda "ping" eșuează, atunci se poate stabili că există o problemă de conectivitate.

În unele exemple de realizare, dacă există o problemă de conectivitate care implică dispozitivul local, poate apărea etapa **205**. În unele exemple de realizare, dacă nu există o problemă de conectivitate care implică dispozitivul local, poate să apară etapa **206**.

În etapa **205**, poate fi raportată o cauză sau un diagnostic, de exemplu, la un utilizator, la un raport de sistem sau la o altă entitate.

În etapa **206**, poate fi analizată și/sau verificată conectivitatea la o pereche de noduri (de exemplu, controlerul **102** de diagnosticare, DN **108**, DN **110**, DN **112**, etc.).

La etapa **207**, în cazul în care conectivitatea la o pereche de noduri nu poate fi verificată, poate fi detectat și poate fi raportat un dispozitiv de blocare (de exemplu, un dispozitiv intermediar sau firewall).

În etapa **208**, se poate determina dacă dispozitivul local este accesibil folosind un protocol SNMP.

La etapa **209**, dacă dispozitivul local este accesibil folosind un protocol SNMP, poate fi stabilită o conexiune SNMP.

La etapa **210**, se pot determinat posibilele cauze ale unei sau mai multor probleme detectate folosind intrările de la testare și/sau alte informații.

Conform Figurii 2B, în etapa **211**, se poate determina dacă informațiile referitoare la dispozitive(de exemplu, nodul) obținute folosind un protocol SNMP sunt utile pentru diagnosticarea unei sau mai multor probleme detectate. De exemplu, informațiile legate de testare și problemele detectate pot fi analizate pentru a determina dacă informațiile se pot obține folosind un protocol SNMP și sunt, probabil, pentru a diagnostica o problemă detectată.

În unele exemple de realizare, dacă informațiile referitoare la dispozitiv obținute folosind un protocol SNMP sunt determinate ca fiind relevante pentru una sau mai multe probleme detectate, poate apărea etapa **212**. În unele exemple de realizare, dacă informațiile referitoare la dispozitiv obținute folosind un protocol SNMP sunt determinate ca nefiind relevante pentru una sau mai multe probleme detectate, poate să apară etapa **213**.

La etapa **212**, pot fi obținute informațiile legate de dispozitiv și poate fi raportată o cauză sau un diagnostic, de exemplu, la un utilizator, la un sistem de raportare sau la o altă entitate. De exemplu, algoritmul **200** de diagnosticare care se execută la DN **108** poate transmite un mesaj de cerere SNMP la nodul/funcția **114** pentru solicitarea de informații de configurare, de informații legate de identificatorul de obiect (OID) și/sau alte informații legate de nodul/funcția **114** și poate fi transmis de la nodul/funcția **114** până la DN **108** un mesaj de răspuns SNMP care conține informațiile solicitate.

La etapa **213**, pot fi obținute informațiile referitoare la dispozitiv. De exemplu, algoritmul **200** de diagnosticare care se execută la DN **108** poate stoca informațiile referitoare la dispozitiv obținute folosind SNMP și poate, de asemenea, încerca să recupereze informațiile suplimentare folosind interfețe de comunicații și/sau alte protocoale de comunicații.

În etapa **214**, se poate determina dacă dispozitivul local este accesibil cu ajutorul SSH.

În unele exemple de realizare, dacă informațiile referitoare la dispozitiv obținute

folosind SSH sunt determinate ca fiind relevante pentru una sau mai multe probleme detectate, poate apărea etapa **215**. În unele exemple de realizare, în cazul în care informațiile referitoare la dispozitiv obținute folosind SSH sunt determinate ca fiind nerelevante pentru una sau mai multe probleme detectate, poate apărea etapa **220**.

În etapa **215**, poate fi stabilită o sesiune SSH.

În etapa **216**, pot fi executate unul sau mai multe script-uri specifice în timpul sesiunii SSH. De exemplu, algoritmul **200** de diagnosticare care se executa la DN **108** poate executa un script care utilizează SSH sau un program relevant sau interfață pentru obținerea de informații la dispozitiv și/sau la dispozitivul bazat pe metrici.

În etapa **217**, se poate determina dacă informațiile referitoare la dispozitiv obținute folosind SSH sunt relevante pentru una sau mai multe probleme detectate. De exemplu, informațiile referitoare la dispozitiv pot indica probleme de configurare, probleme de hardware și/sau alte probleme care pot afecta capacitatea SUT-ului **106**. În unele exemple de realizare, dacă informațiile referitoare la dispozitiv obținute folosind SSH sunt determinate ca fiind relevante pentru una sau mai multe probleme detectate, poate apărea etapa **218**. În alte exemple de realizare, în cazul în care informațiile referitoare la dispozitiv obținute folosind SSH sunt determinate ca fiind nerelevante pentru una sau mai multe probleme detectate, poate apărea etapa **219**.

În etapa **218**, poate fi raportată o cauză sau un diagnostic, de exemplu, la un utilizator, la un sistem de raportare sau la o altă entitate.

În etapa **219**, o eroare sau un mesaj corespunzător indică faptul că poate fi raportat că nu este găsită o cauză care să detecteze o problemă. În unele exemple de realizare, înainte de raportarea unei erori, pot fi efectuate etape suplimentare de diagnosticare care includ obținerea de informații de la un dispozitiv local folosind interfețe de comunicație și/sau protocoale de comunicații, suplimentare.

În etapa **220**, se poate determina dacă dispozitivul local este accesibil folosind o interfață API prin HTTP. În unele exemple de realizare, dacă un dispozitiv local este accesibil folosind o interfață API prin HTTP, poate să apară etapa **221**. În unele exemple de realizare, dacă un dispozitiv local nu este accesibil folosind o interfață API prin HTTP, poate să apară etapa **226**.

În etapa **221**, poate fi stabilită o conexiune HTTP.

În etapa **222**, pot fi executate, în timpul conexiunii HTTP, unul sau mai multe script-uri specifice dispozitivului. De exemplu, algoritmul **200** de diagnosticare care se execută la DN **108** poate executa un script care utilizează o interfață API prin HTTP sau un program sau o interfață relevante pentru obținerea de informații la dispozitiv și/sau la dispozitivul bazat pe metrici.

În etapa **223**, se poate determina dacă informațiile referitoare la dispozitiv obținute folosind o interfață API prin HTTP sunt relevante pentru una sau mai multe probleme detectate. De exemplu, informațiile referitoare la dispozitiv pot indica probleme de configurare, probleme de hardware și/sau alte probleme care pot afecta capacitatea SUT-ului **106**.

În unele exemple de realizare, dacă informațiile referitoare la dispozitiv obținute folosind o interfață API prin HTTP sunt determinate ca fiind relevante pentru una sau mai multe probleme detectate, poate apărea etapa **224**. În unele exemple de realizare, în cazul în care informațiile referitoare la dispozitiv obținute folosind o interfață API prin HTTP sunt determinate ca fiind nerelevante pentru una sau mai multe probleme detectate, poate apărea etapa **225**.

În etapa **224**, poate fi raportată o cauză sau un diagnostic, de exemplu, la un utilizator, la un sistem de raportare sau la o altă entitate.

În etapa **225**, o eroare sau un mesaj corespunzător indică faptul că poate fi raportat că nu este găsită o cauză pentru a detecta o problemă. În unele exemple de realizare, înainte de raportarea unei erori, pot fi efectuate etape suplimentare de diagnosticare care includ obținerea de informații de la un dispozitiv local folosind interfețe de comunicație și/sau protocoale de comunicații, suplimentare.

Se va aprecia că Figura 2 este pentru scopuri ilustrative și că acțiuni diferite și/sau suplimentare, altele decât cele descrise în Figura 2, pot fi utilizate pentru diagnosticarea referitoare la SUT. Se va aprecia, de asemenea, că diferite acțiuni descrise aici pot să apară simultan sau într-o ordine sau secvență diferită. De exemplu, algoritmul **200** de diagnosticare poate include aspecte de învățare automată care pot evita utilizarea anumitor protocoale de comunicații, interfețe sau programe bazate pe ratele de succes istorice și/sau bazate pe rezultate care implică tipuri similare de dispozitive pentru a fi diagnosticate. Într-un alt exemplu, algoritmul **200** de

diagnosticare poate modifica o comandă prin care mecanismele de comunicație sunt utilizate pentru diferite dispozitive de diagnosticat.

Figura 3 este o diagramă care prezintă comunicațiile asociate cu diagnosticarea de rețea, în conformitate cu un exemplu de realizare a obiectului descris aici. În unele exemple de realizare, controlerul **102** de diagnosticare poate interacționa cu mediul **100** informatic pentru configurarea, previzionarea sau gestionarea DN-urilor **108-112** și/sau testării SUT **106**. De exemplu, controlerul **102** de diagnosticare poate utiliza informații de rețea în momentul transmiterii informațiilor de configurare la DN **108-112** pentru testarea SUT **106** și/sau diagnosticarea problemelor asociate cu SUT **106**.

În unele exemple de realizare, controlerul **102** de diagnosticare poate configura DN **108** și **112** pentru testarea SUT **106** într-o rețea reală. De exemplu, SUT **106** poate fi configurat pentru a procesa și/sau ruta traficul de date (de exemplu, de tip voce, video, și/sau trafic de aplicații) a unui număr de utilizatori într-o rețea reală. În acest exemplu, controlerul **102** de diagnosticare poate configura DN **108** pentru a simula trafic în rețeaua reală. Continuând cu acest exemplu, DN-urile **108** și **112** pot fi configurate pentru a observa traficul simulat și/sau traficul non-simulat și pentru a genera metrici de trafic despre traficul observat.

În unele exemple de realizare, controlerul **102** de diagnosticare poate analiza metricile referitoare la trafic și/sau alte informații de la DNS **108** și **112** pentru a determina dacă există probleme la SUT. Dacă există o problemă la SUT, controlerul **102** de diagnosticare poate oferi instrucțiuni și/sau alte informații DN-urilor **108** și **112** pentru obținerea de informații suplimentare de la elemente individuale (de exemplu, nodurile/funcțiile **114-116**) în SUT **106**. De exemplu, fiecare DN **108** și **112** poate efectua interogarea nodurilor/funcțiilor **114-116** folosind un algoritm de diagnosticare sau script-uri de diagnosticare conexe.

Conform Figurii 3, în etapa **3001**, pot fi trimise de la controlerul **102** de diagnosticare la DN **108,** informații de configurare testare (de exemplu, informații de configurare).

În etapa **3002**, informațiile de configurare testare pot fi transmise de la DN **108** la DN **112** și/sau la alte DN-uri, în mediul **100** informatic la SUT **106**.

În etapa **3003**, traficul de testare sau simulat poate fi trimis de la DN **108** la DN **112** prin SUT **106**. De exemplu, DN **108** poate acționa ca un trafic generat și poate

transmite numeroase tipuri de pachete de date care simulează una sau mai multe sesiuni de comunicație.

În etapa **3004**, după testare, rezultatele testelor pot fi trimise de la DN **108** la controlerul **102** de diagnosticare. De exemplu, DN **108** poate colecta metricile de trafic de la DN **112** și/sau poate genera rezultatele testelor pe baza acestor metrici de trafic și/sau a altor date de la diverse surse.

În etapa **3005**, controlerul **102** de diagnosticare poate analiza rezultatele testelor și/sau alte informații pentru a determina dacă există una sau mai multe probleme asociate cu SUT **106**. De exemplu, controlerul **102** de diagnosticare poate utiliza rezultatele testelor și poate consulta dispozitivul **104** de stocare a datelor despre informații de topologie de rețea relevante și/sau despre informațiile de configurare SUT.

În unele exemple de realizare de realizare, controlerul **102** de diagnosticare poate determina că există o problemă potențială asociată cu SUT **106** și poate încerca să declanșeze diagnosticarea. De exemplu, controlerul **102** de diagnosticare poate transmite un mesaj legat de diagnosticare sau informații corespunzătoare la DN **108** și/sau la DN **112** pentru obținerea informațiilor de stare și/sau a altor date de la SUT **106** sau de la nodul/funcția **114**.

În etapa **3006**, informațiile legate de diagnosticare pot fi trimise de la controlerul **102** de diagnosticare la DN **108** pentru obținerea informațiilor de stare de la nodul/funcția **114**.

În etapa **3007**, informațiile legate de diagnosticare pot fi transmise de la DN **108** la DN **112** și/sau la alt DN-uri în mediul **100** informatic sau la SUT **106**.

La etapa **3008**, DN **112** poate executa un algoritm de diagnosticare pentru obținerea de informații de stare și/sau a altor informații de la nodul/funcția **114**. În unele exemple de realizare, un algoritm de diagnosticare poate include numeroase scripturi și/sau logica de program pentru a încerca să comunice cu mai multe noduri de rețea, cu mai multe funcții de rețea sau cu entități asociate. De exemplu, un algoritm de diagnosticare poate încerca să solicite informații de stare și/sau alte date de la nodul/funcția **114** folosind o serie sau un set de protocoale de comunicații.

În etapa **3009**, poate să nu fie stabilită o legătură între DN **112** și nodul/funcția **114**. De exemplu, comunicațiile efectuate de DN **112** la nodul/funcția **114** pot fi blocate sau eliminate de către un firewall care intervine, situat între DN **112** și nodul/funcția **114**.

În etapa **3010**, DN **108** poate executa un algoritm de diagnosticare pentru obținerea de informații de stare și/sau a altor informații de la nodul/funcția **114**.

În etapa **3011**, poate fi stabilită o conexiune legată de SNMP între DN **108** și nodul/funcția **114**.

În etapa **3012**, poate fi trimis un mesaj de solicitare SNMP de la DN **108** la nodul/funcția **114**. De exemplu, un mesaj de solicitare SNMP poate cere informații de stare și/sau alte informații de la o baza de date referitoare la managementul rețelei.

În etapa **3013**, poate fi trimis un mesaj de răspuns SNMP de la nodul/funcția **114** la DN **108**. De exemplu, un mesaj de răspuns SNMP poate furniza informații de stare și/sau alte informații despre nodul/funcția **114**.

În etapa **3014**, DN **108** poate stabili dacă sunt necesare informații suplimentare de la nodul/funcția **114**. Dacă este așa, DN **108** sau un algoritmul de diagnosticare asociat poate determina să se utilizeze un alt protocol de comunicații și/sau un alt script specific dispozitivului, pentru a obține informații suplimentare.

În etapa **3015**, poate fi stabilită o conexiune legată de SSH între DN **108** și nodul/funcția **114**.

În etapa **3016**, una sau mai multe comenzi SSH pot fi trimise de la DN **108** la nodul/funcția **114** pentru obținerea de informații diverse despre nodul/funcția **114**. De exemplu, în timpul unei sesiuni SSH, o comandă SSH poate solicita diverse metrice de procesare asociate cu traficul de testare.

În etapa **3017**, informațiile pot fi trimise de la nodul/funcția **114** la DN **108** pentru a indica posibile cauze ale unei sau mai multor probleme referitoare la SUT **106**. De exemplu, ca răspuns la o comandă SSH pentru informațiile de stare, un router poate indica o problemă cauzată de faptul că un tabel de rutare este complet, fiind în imposibilitatea de a curăți intrările.

În etapa **3018**, poate fi transmis la controlerul **102** de diagnosticare, un raport de diagnosticare sau informații asociate. De exemplu, controlerul **102** de diagnosticare poate primi rapoarte multiple de diagnosticare de la unul sau mai multe DN-uri **108-112** și poate utiliza aceste rapoarte la generarea unui raport de diagnosticare SUT. În acest exemplu, raportul de diagnosticare SUT poate indica faptul că SUT **106** a prezentat o problemă de trafic cauzată de greșelile de configurare a ale portului de rețea la nodul/funcția **114** și poate indica o posibilă soluție în cazul în care un anumit

trafic de aplicație este transmis la numerele de port "3453" sau "6785" în loc de numărul de port "8080", port care primește de asemenea traficul HTTP criptat. Într-un alt exemplu, raportul de diagnosticare SUT poate indica faptul că SUT **106** a prezentat o problemă de lățime de bandă sau congestionare cauzată de eșecuri ale legăturii de comunicație într-un grup de agregare asociat cu nodul/funcția **116** și poate indica o posibilă soluție în cazul în care legăturile de comunicație sunt inspectate sau înlocuite. Se va aprecia că acțiunile și/sau comunicațiile reprezentate în Figura 3 sunt pentru scop ilustrativ și că diferite acțiuni și/sau comunicații altele decât cele descrise în Figura 3 pot fi utilizate suplimentar pentru testarea SUT **106** și/sau pentru diagnosticarea problemelor asociate cu SUT **106**. Se va aprecia de asemenea că diversele comunicații și/sau a acțiuni descrise aici pot să apară simultan sau într-o ordine sau secvență diferită. De exemplu, etapa **3008** se poate produce concomitent cu etapa **3010**.

Figura 4 este o diagramă care prezintă comunicațiile asociate cu diagnosticarea de rețea, conform unui alt exemplu de realizare a obiectului descris aici. În unele exemple de realizare, controlerul **102** de diagnosticare poate interacționa cu mediul **100** informatic pentru configurarea, previzionarea sau gestionarea DN **108-112** și/sau testarea SUT **106**. De exemplu, controlerul **102** de diagnosticare poate utiliza informațiile de rețea la transmiterea de informații de configurare la DN **108-112** pentru testarea SUT **106** și/sau pentru diagnosticarea problemelor referitoare la SUT **106**.

În unele exemple de realizare, controlerul **102** de diagnosticare poate configura DN **108-112** pentru testarea SUT **106** într-o rețea reală. De exemplu, SUT **106** poate fi configurat pentru a procesa și/sau ruta traficul de date (de exemplu, de tip voce, video, și/sau trafic de aplicații) a unui număr de utilizatori într-o rețea reală. În acest exemplu, controlerul 102 de diagnosticare poate configura DN **108** pentru a simula traficul în rețeaua reală. Continuând cu acest exemplu, DN-urile **108-112** pot fi configurate pentru a observa traficul simulat și/sau traficul non-simulat și pentru a genera metrici asociate cu traficul care afectează aproximativ traficul observat.

În unele exemple de realizare de realizare, controlerul **102** de diagnosticare poate primi metrici legate de trafic și/sau alte informații de la DN-urile **108-112** pentru a analiza și determina dacă există probleme SUT. Dacă există o problemă la SUT, controlerul **102** de diagnosticare poate oferi instrucțiuni și/sau alte informații la unul

sau mai multe DN-uri **108-112** pentru obținerea de informații suplimentare de la elemente individuale (de exemplu, noduri/funcții **114-116**) în SUT **106**. De exemplu, fiecare dintre DN-urile **108-112** poate efectua interogarea de noduri/funcții **114-116** folosind un algoritm de diagnosticare sau script-uri de diagnosticare conexe.

Conform Figurii 4, în etapa **4001**, controlerul **102** de diagnosticare poate configura DN **108** ca și un client virtual. De exemplu, un client virtual (de exemplu, software-ul rulat de un procesor) poate acționa ca și un generator de trafic pentru generarea și transmiterea traficului de testare la SUT **106**.

În etapa **4002**, controlerul **102** de diagnosticare poate configura DN **110** ca un dispozitiv hardware virtual. De exemplu, dispozitiv hardware virtual (cum ar fi software-ul rulat de un un procesor) poate acționa ca dispozitiv hardware de rețea și poate monitoriza traficul care traversează DN **110**.

În etapa **4003**, controlerul **102** de diagnosticare poate declanșa ca DN **108** să înceapă transmiterea traficului de testare. De exemplu, controlerul **102** de diagnosticare poate transmite informații de configurare care indică pachetele de testare de transmis la SUT **106** sau la porțiuni ale acestuia.

În unele exemple de realizare, după ce a primit o comandă de declanșare de la controlerul **102** de diagnosticare "începe testarea", pachetele de testare pot fi generate și transmise de la DN **108** la nodul/funcția **114** și procesate. După procesare, pachetele de testare pot fi transmise de la nodul/funcția **114** la nodul/funcția **116** prin DN **110**. Continuând cu acest exemplu, pachetele de răspuns pot fi generate și transmise de la nodul/funcția **116** la nodul/funcția **114** prin DN **110** și apoi de la nodul/funcția **114** la DN **108**.

În unele exemple de realizare, DN-urile **108-110** sau logica de program a acestora pot monitoriza datele de trafic transmise și/sau genera metrici referitoare la trafic pentru diagnosticarea potențialelor probleme asociate cu SUT **106**.

În etapa **4004**, legătura de comunicație observată și/sau metricile de trafic pot fi transmise de la DN **110** la controlerul **102** de diagnosticare. De exemplu, după ce testarea este completă sau la intervale periodice, informațiile legate de testare (de exemplu, latență, bruiaj (jitter), etc.) pot fi calculate de DN **110** și raportate la controlerul **102** de diagnosticare.

În etapa **4005**, legătura de comunicație observată și/sau metricile de trafic pot fi transmise de la DN **108** la la controlerul **102** de diagnosticare. De exemplu, după ce testarea este completă sau la intervale periodice, informațiile legate de testare (de exemplu, latență, bruiaj (jitter), etc.) pot fi calculate de DN **108** și raportate la controlerul **102** de diagnosticare.

În etapa **4006**, controlerul **102** de diagnosticare poate analiza link-ul recepționat și/sau metricile de trafic pentru a identifica comportamentul anormal (de exemplu, una sau mai multe probleme). De exemplu, controlerul **102** de diagnosticare poate determina care jitter asociat cu nodul/funcția **114** a fost semnificativ mai mare decât se aștepta sau care nod/funcție **116** nu pare să genereze pachete de răspuns adecvate.

La etapa **4007**, în cazul în care este detectat un comportament anormal, controlerul **102** de diagnosticare poate determina care nod/funcție să aleagă sau să interogheze pentru informații suplimentare. De exemplu, controlerul **102** de diagnosticare poate utiliza logica de program, informații despre SUT și/sau informații de rețea (de exemplu, informații de topologie de rețea) pentru a face propriile determinări.

În etapa **4008**, controlerul **102** de diagnosticare poate interoga (de exemplu, cerere și/sau sondaj) nodul/funcția **114** pentru informațiile de stare, metricile de trafic și/sau pentru alte informații care pot fi utile in diagnosticarea comportamentului anormal. De exemplu, controlerul **102** de diagnosticare poate utiliza un script și/sau un algoritm de diagnosticare care încearcă să obțină informații de stare de la nodul/funcția **114** folosind mai multe protocoale și/sau mecanisme (de exemplu, SNMP, o interfață REST HTTP API, comenzi SHH, etc.)

În etapa **4009**, controlerul **102** de diagnosticare poate interoga (de exemplu, cerere și/sau sondaj) nodul/funcția **116** pentru informațiile de stare, metricile de trafic și/sau pentru alte informații care pot fi utile in diagnosticarea comportamentului anormal. De exemplu, controlerul **102** de diagnosticare poate utiliza un script și/sau un algoritm de diagnosticare care încearcă să obțină informații de stare de la nodul/funcția **116** folosind mai multe protocoale și/sau mecanisme (de exemplu, SNMP, o interfață REST HTTP API, comenzi SHH, etc.)

În etapa **4010**, controlerul **102** de diagnosticare poate utiliza informațiile obținute (de exemplu, informații de stare și informații de monitorizare comportament de rețea)

pentru a genera un raport de diagnosticare SUTcare include sugerarea soluțiilor la problemele identificate.

Se va aprecia că acțiunile și/sau comunicațiile reprezentate în Figura 4 sunt pentru scop ilustrativ și că diferite alte comunicații și/sau acțiuni, altele decât cele descrise în Figura 4, pot fi utilizate suplimentar pentru testarea SUT **106** și/sau pentru diagnosticarea problemelor asociate cu SUT **106**. Se va aprecia de asemenea că diversele comunicații și/sau a acțiuni descrise aici pot să apară simultan sau într-o ordine sau secvență diferită.

Figura 5 este o diagramă care prezintă o metodă **500** pentru diagnosticarea rețelei, în conformitate cu un exemplu de realizare a obiectului descris aici. În unele exemple de realizare, metoda **500** sau porțiuni ale acesteia poate fi efectuată de către sau la controlerul **102** de diagnosticare și/sau de către sau la un alt nod sau modul (de exemplu, un controler de diagnosticare). În unele exemple de realizare, metoda **500** poate include etapele **502**, **504** și/sau **506**.

Conform metodei **500**, în etapa **502**, pot fi configurate o multitudine de noduri de diagnosticare pentru a observa comportamentul traficului asociat cu un SUT. De exemplu, controlerul **102** de diagnosticare poate transmite informații de configurare la DN **108** și DN **112** pentru testarea și diagnosticarea la SUT **106** a oricăror probleme detectate asociate SUT **106**.

În unele exemple de realizare, configurarea unei multitudini de noduri de diagnosticare poate include utilizarea informațiilor de configurare a rețelei sau a informațiilor despre topologia SUT.

În unele exemple de realizare, configurarea unei multitudini de noduri de diagnosticare include configurarea unui dispozitiv hardware de rețea pentru monitorizarea traficului de rețea în SUT.

În unele exemple de realizare, configurarea unei multitudini de noduri de diagnosticare poate include configurarea unui nod dintre nodurile de diagnosticare pentru a transmite trafic de testare și configurarea la cel puțin unul dintre nodurile de diagnosticare pentru a genera metrici de trafic asociate cu traficul de testare.

La etapa **504**, comportamentul traficului asociat cu SUT poate fi observat cu ajutorul nodurilor de diagnosticare. De exemplu, nodul **108** de diagnosticare poate observa

mesajele de răspuns de la nodul/funcția **116** la traficul inițial prin el și nodul **112** de diagnosticare poate observa mesajele de solicitare de la nodul/funcția **114**.

În etapa **506**, o problemă SUT poate fi detectată cu ajutorul comportamentului traficului. De exemplu, folosind metricile de trafic obținute de la DN **108** și DN **112**, controlerul **102** de diagnosticare poate stabili că 30% din traficul transmis de DN **108** nu răspunde.

În etapa **508**, poate fi identificat un nod de rețea în SUT asociat cu problema SUT, cu ajutorul informațiilor despre topologia SUT. De exemplu, după stabilirea că răspunsurile nu sunt recepționate pentru traficul transmis de DN **108**, controlerul **102** de diagnosticare poate determina care nod/funcție **114** și nod/funcție **116** pot fi asociate cu această problemă SUT.

În unele exemple de realizare, un nod de rețea poate include un nod de echilibrare a încărcării, un nod de detectare a intruziunilor, un nod de prevenirea intruziunilor, un nod antivirus, un nod antispam, un nod firewall, un nod de comutație sau un nod de rutare.

În etapa **510**, unul dintre nodurile de diagnosticare poate fi declanșat pentru a obține informații referitoare la nod de la nodul de rețea. Nodul de diagnosticare poate folosi cel puțin un protocol de management sau de comunicație pentru a alege nodul de rețea pentru informațiile referitoare la nod. Câteva exemple de informații legate de nod (de exemplu, informații legate de dispozitiv) pot include informațiile de configurare, informațiile de topologie, informații de securitate, protocoalele de comunicații suportate, tipuri de memorie, utilizarea memoriei, disponibilitatea de memorie, informații de producător, date de construcție, tip procesor, utilizare procesor, disponibilitate procesor, versiunea sistemului de operare, versiunea de firmware, metrici legate de nod, metrici de randament, metrici de performanță, metrici de eroare, informații de stare, statutul actual al lungimii, capacitatea de stocare, utilizarea spațiului de stocare, informații de conectivitate (de exemplu, informații de port, informații de legătură de comunicație, etc.), informații de rutare sau informații de memorie tampon.

În unele exemple de realizare, cel puțin un management sau protocol de comunicații poate include SNMP, SSH, Telnet, un protocol CLI, HTTP, și/sau REST.

În unele exemple de realizare, interogarea unui nod de rețea pentru informații asociate nodului poate include determinarea dacă un prim protocol de comunicații este utilizabil

pentru comunicația cu nodul de rețea în SUT și ca răspuns la determinarea faptului că primul protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT, solicitarea, de la nodul de rețea, a informațiilor referitoare la nodul care folosește primul protocol de comunicații.

În unele exemple de realizare, interogarea unui nod de rețea pentru informații referitoare la nod poate include, de asemenea, determinarea dacă un al doilea protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT și ca răspuns la determinarea faptului că al doilea protocolul de comunicație este utilizabil pentru comunicația cu nodul de rețea în SUT, solicitarea, de la nodul de rețea, a informațiilor referitoare la nodul care folosește al doilea protocol de comunicație.

În etapa **512**, problema SUT poate fi diagnosticată folosind informațiile referitoare la nod. De exemplu, folosind SSH sau o interfață API prin HTTP, controlerul **102** de diagnosticare poate prelua metricile de performanță asociate cu nodul/funcția **114**, de exemplu, un nod de rutare. În acest exemplu, metricile de performanță pot indica o scădere semnificativă a performanței până la de două ori în timpul testării. În acest exemplu, controlerul **102** de diagnosticare poate utiliza aceste informații împreună cu informațiile despre ce tip de trafic a fost transmis în acest timp, pentru a determina că acel nod/funcție **114** a fost configurat greșit și, ca atare, nu a reușit să ruteze în mod corespunzător traficul prin protocolul de inițiere a sesiunii (SIP).

În unele exemple de realizare, diagnosticarea unei probleme SUT poate include generarea unui raport de diagnosticare care indică problema SUT și cel puțin o potențială soluție. De exemplu, ca răspuns la determinarea că acel nod/funcție **114** a fost configurat greșit și nu a reușit să ruteze în mod corespunzător traficul prin protocolul de inițiere a sesiunii (SIP), controlerul de diagnosticare poate indica, într-un raport de diagnosticare, că o soluție potențială pentru greșelile de configurare este aceea de a adăuga intrări de rutare la nodul/funcția **114** pentru rutarea traficului SIP.

Se va aprecia că metoda **500** este pentru scopuri ilustrative și că pot fi utilizate acțiuni diferite și/sau suplimentare. Se va aprecia de asemenea că diferite acțiuni descrise aici pot să apară într-o ordine sau secvență diferită.

Trebuie remarcat faptul că menționatul controler **102** de diagnosticare și/sau funcționalitatea descrisă aici poate constitui un dispozitiv informatic cu scop special. Mai mult, controlerul **102** de diagnosticare și/sau funcționalitatea descrisă aici pot

îmbunătăți domeniul tehnologic de diagnosticare de rețea prin furnizarea de mecanisme de testare automate a SUT **106** și diagnosticarea problemelor referitoare la SUT **106**. Mai mult, controlerul **102** de diagnosticare și/sau funcționalitatea descrisă aici poate îmbunătăți domeniul tehnologic de diagnosticare de rețea prin furnizarea de mecanisme pentru obținerea informațiilor de stare și/sau a altor informații de la nodurile/funcțiile **114-116** în SUT **106**, folosind un algoritm de diagnosticare care încearcă să comunice cu ajutorul unui sau mai multor protocoale de comunicație.

Se va înțelege că diferite detalii ale obiectului prezentei invenții pot fi modificate fără a ne îndepărta de la scopul invenției. În plus, descrierea de mai sus este pentru a prezenta scopul și nu în scopul limitării, astfel că invenția revendicată este definită prin revendicările enunțate mai jos.

REVENDICĂRI

1. Metodă de diagnosticare rețea, care, la un controler de diagnosticare implementat care folosește cel puțin un procesor, constă în::
   - configurarea unei multitudini de noduri de diagnosticare pentru a observa comportamentul traficului de date asociat cu un sistem de testat (SUT);
   - observarea, folosind nodurile de diagnosticare, a comportamentul traficului de date asociat cu SUT;
   - detectarea, folosind comportamentul traficului de date, a problemei SUT;
   - identificarea, folosind informații despre topologia SUT, a unui nod de rețea în SUT asociat cu problema SUT;
   - declanșarea ca un nod dintre nodurile de diagnosticare să obțină informații referitoare la nod de la nodul de rețea, în care nodul de diagnosticare folosește cel puțin un protocol de comunicație pentru a interoga nodul de rețea despre informațiile legate de acesta;
   - diagnosticarea, folosind datele referitoare la nod, a problemei SUT.
2. Metodă, conform revendicării 1, **caracterizată prin aceea că** menționata configurare a multitudinii de noduri de diagnosticare include configurarea unui nod dintre nodurile de diagnosticare pentru a transmite trafic de testare și configurarea a cel puțin un nod dintre nodurile de diagnosticare pentru a genera metrici de trafic asociate cu traficul de testare.
3. Metodă, conform revendicării 1, **caracterizată prin aceea că** protocolul de comunicație include cel puțin un protocol de management simplu de rețea (SNMP), un protocol criptografic de rețea (SSH), un protocolul Telnet, un protocol de interfață de linie de comandă (CLI), un protocol de transfer hypertext (HTTP) sau un protocol de arhitectură software a rețelei (REST).
4. Metodă, conform revendicării 1, **caracterizată prin aceea că** interogarea nodului de rețea pentru informațiile referitoare la nod, constă în:
   - determinarea dacă un prim protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT; și
   - ca răspuns la determinarea faptului că primul protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT, solicitarea,

de la nodul de rețea, a informațiilor referitoare la nodul care folosește primul protocol de comunicații.

5. Metodă, conform revendicării 4, **caracterizată prin aceea că** mai constă :
   - determinarea dacă un al doilea protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT; și
   - ca răspuns la determinarea faptului că al doilea protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT, solicitarea, de la nodul de rețea, a informațiilor referitoare nodul care folosește al doilea protocol de comunicații.
6. Metodă, conform revendicării 1, **caracterizată prin aceea că** menționata configurare a multitudinii de noduri de diagnosticare include folosirea informațiilor de configurare a rețelei sau a informațiilor despre topologia SUT.
7. Metodă, conform revendicării 1, **caracterizată prin aceea că** menționata configurare a multitudinii de noduri de diagnosticare include configurarea unui dispozitiv hardware de rețea care permite trafic de date pentru monitorizarea traficului de rețea în SUT.
8. Metodă, conform revendicării 1, **caracterizată prin aceea că** diagnosticarea problemei SUT include generarea unui raport de diagnosticare care indică problema SUT și cel puțin o soluție posibilă.
9. Metodă, conform revendicării 1, **caracterizată prin aceea că** nodul de rețea include un nod de echilibrare a încărcării, un nod de detectare a intruziunilor, un nod de prevenirea intruziunilor, un nod antivirus, un nod antispam, un nod firewall, un nod de comutație sau un nod de rutare.
10. Sistem de diagnosticare de rețea, care, la cel puțin un procesor, conține:
    - un controler de diagnosticare implementat care folosește cel puțin un procesor, în care controlerul de diagnosticare este configurat pentru a configura o multitudine de noduri de diagnosticare pentru a observa comportamentul traficului de date asociat cu un sistem de testat (SUT), pentru a observa, folosind nodurile de diagnosticare, comportamentul traficului asociat cu SUT, pentru a detecta, folosind comportamentul traficului, o problemă SUT, pentru a identifica, folosind informații despre topologia SUT, un nod de rețea în SUT asociat cu problema SUT, pentru

a declanșa ca unul dintre nodurile de diagnosticare să obțină informații referitoare la nod de la nodul de rețea, în care nodul de diagnosticare folosește cel puțin un protocol de comunicații pentru a interoga nodul rețelei despre informațiile legate de nod și pentru a diagnostica, folosind informațiile legate de nod, problema SUT.

11. Sistem, conform revendicării 10, **caracterizat prin aceea că** menționatul controler de diagnosticare este configurat pentru a configura unul dintre nodurile de diagnosticare pentru a transmite trafic de testare și pentru a configura cel puțin unul din multitudinea de noduri de diagnosticare să genereze metrici de trafic asociate cu traficul de testare.

12. Sistem, conform revendicării 10, **caracterizat prin aceea că** protocolul de comunicație include cel puțin un protocol de management simplu de rețea (SNMP), un protocol criptografic de rețea (SSH), un protocolul Telnet, un protocol de interfață de linie de comandă (CLI), un protocol de transfer hypertext (HTTP) sau un protocol de arhitectură software a rețelei (REST).

13. Sistem, conform revendicării 1 **caracterizat prin aceea că**, menționatul controler de diagnosticare este configurat pentru a determina dacă un prim protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT și ca răspuns la determinarea faptului că primul protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT, solicitarea, de la nodul de rețea, a informațiilor referitoare la nodul care folosește primul protocol de comunicații.

14. Sistem, conform revendicării 13, **caracterizat prin aceea că** menționatul controler de diagnosticare este configurat pentru a determina dacă un al doilea protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT și ca răspuns la determinarea faptului că al doilea protocol de comunicații este utilizabil pentru comunicația cu nodul de rețea în SUT, solicitarea, de la nodul de rețea, a informațiilor referitoare la nodul care folosește al doilea protocol de comunicații.

15. Sistem, conform revendicării 10, **caracterizat prin aceea că** menționata configurare a multitudinii de noduri de diagnosticare include folosirea informațiilor de configurare a rețelei sau informațiile despre topologia SUT.

16. Sistem, conform revendicării 10, **caracterizat prin aceea că** menționatul controler de diagnosticare este configurat pentru a configura un dispozitiv hardware de rețea care permite trafic de date pentru monitorizarea traficului de rețea în SUT.
17. Sistem, conform revendicării 10, **caracterizat prin aceea că** menționatul controler de diagnosticare este configurat pentru a genera un raport de diagnosticare care indică problema SUT și cel puțin o soluție posibilă.
18. Sistem, conform revendicării 10, **caracterizat prin aceea că** nodul de rețea include un nod de echilibrare a încărcării, un nod de detectare a intruziunilor, un nod de prevenirea intruziunilor, un nod antivirus, un nod antispam, un nod firewall, un nod de comutație sau un nod de rutare.
19. Suport non tranzitoriu citibil de calculator care are stocate instrucțiuni executabile de calculator, astfel că, atunci când sunt executate de procesor, comandă calculatorul să efectueze etapele care constau în:
    - configurarea unei multitudini de noduri de diagnosticare pentru a observa comportamentul traficului de date asociat cu un sistem de testat (SUT);
    - observarea, folosind nodurile de diagnosticare, a comportamentul traficului de date asociat cu SUT;
    - detectarea, folosind comportamentul traficului de date, a problemei SUT;
    - identificarea, folosind informații despre topologia SUT, a unui nod de rețea în SUT asociat cu problema SUT;
    - declanșarea ca un nod dintre nodurile de diagnosticare să obțină informații referitoare la nod de la nodul de rețea, în care nodul de diagnosticare folosește cel puțin un protocol de comunicație pentru a interoga nodul de rețea despre informațiile legate de acesta;
    - .diagnosticarea, folosind datele referitoare la nod, a problemei SUT.
20. Suport non tranzitoriu citibil de calculator, conform revendicării 19, **caracterizat prin aceea că** menționata configurare a multitudinii de noduri de diagnosticare include configurarea unui nod dintre nodurile de diagnosticare pentru a transmite trafic de testare și configurarea a cel puțin unui nod dintre nodurile de diagnosticare pentru a genera metrici de trafic asociate cu traficul de testare.

a 2015 01028
22/12/2015

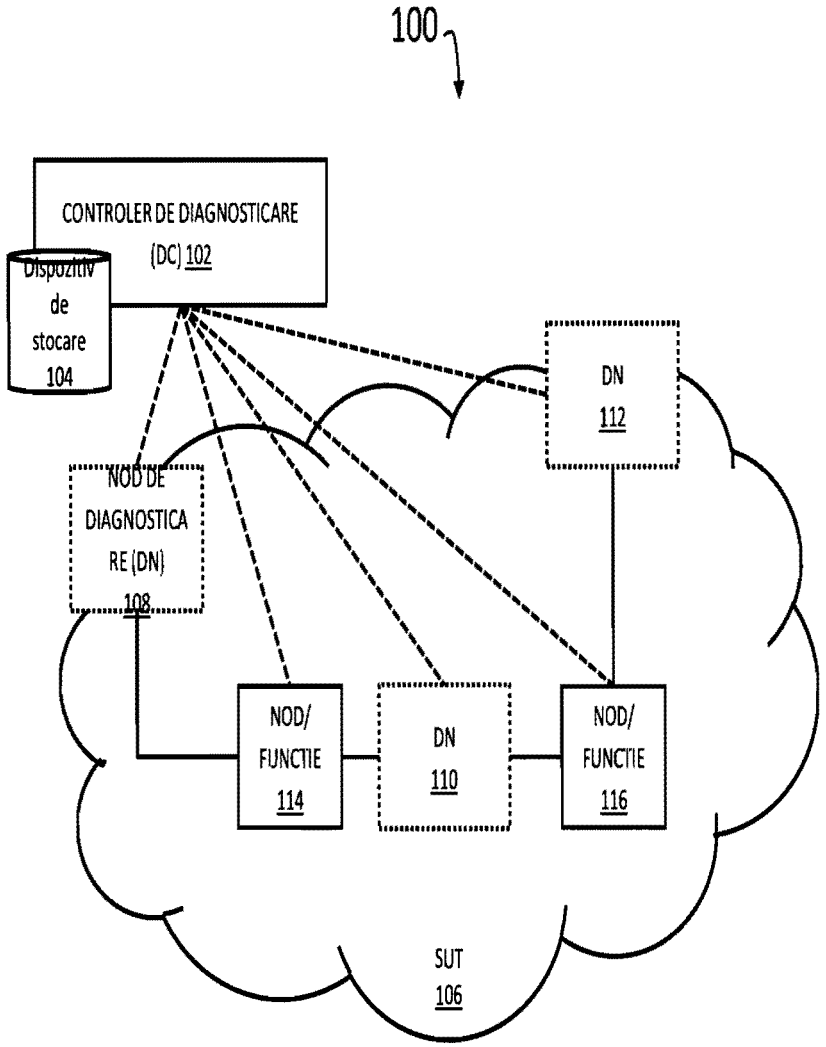

FIG. 1

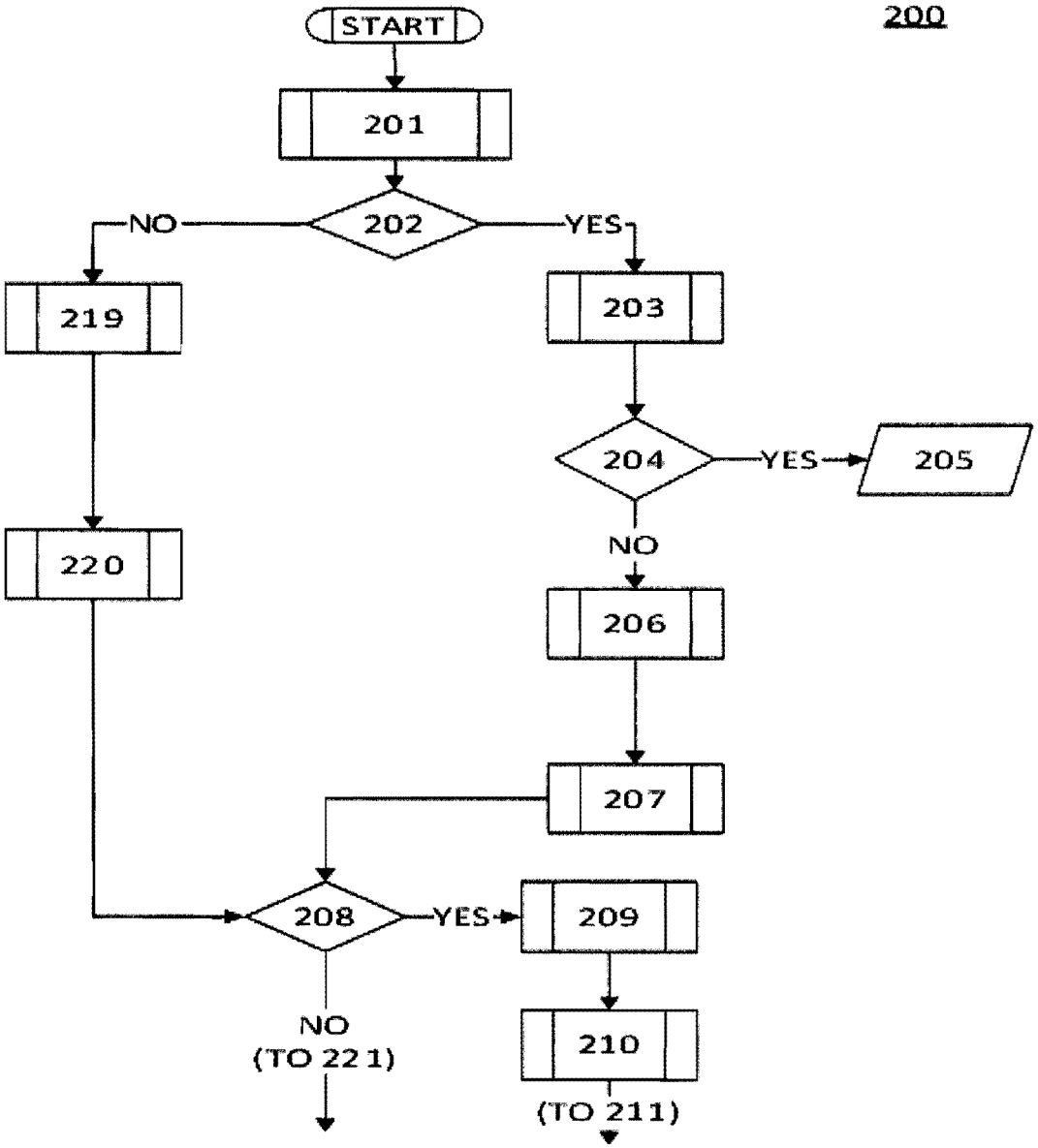
200
START
201
202
NO
YES
219
203
204
YES
205
NO
220
206
207
208
YES
209
NO
(TO 221)
210
(TO 211)

FIG. 2A

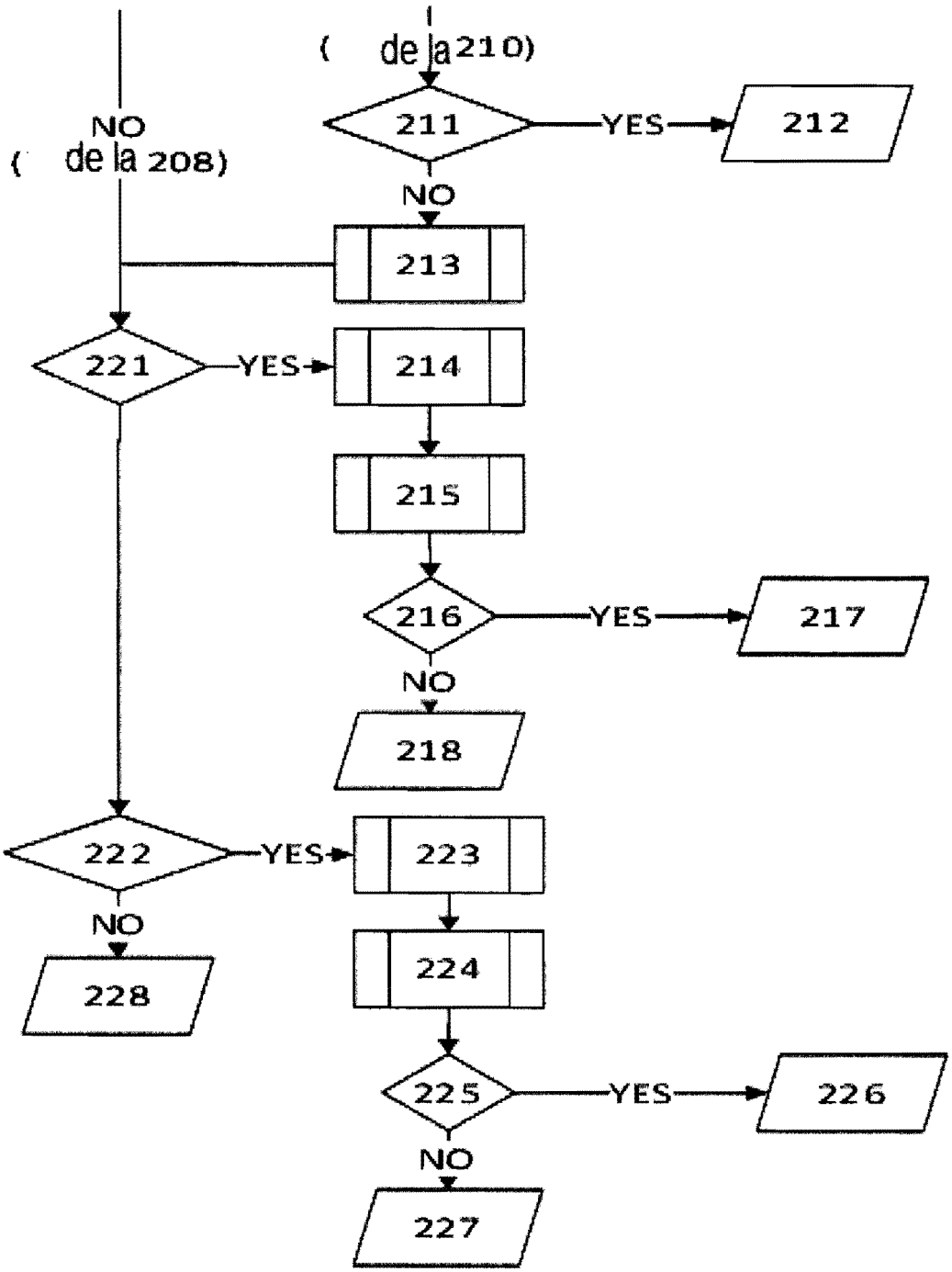
(de la 210)
211
YES
212
NO
NO
(de la 208)
213
221
YES
214
215
216
YES
217
NO
218
222
YES
223
NO
224
228
225
YES
226
NO
227

FIG. 2B

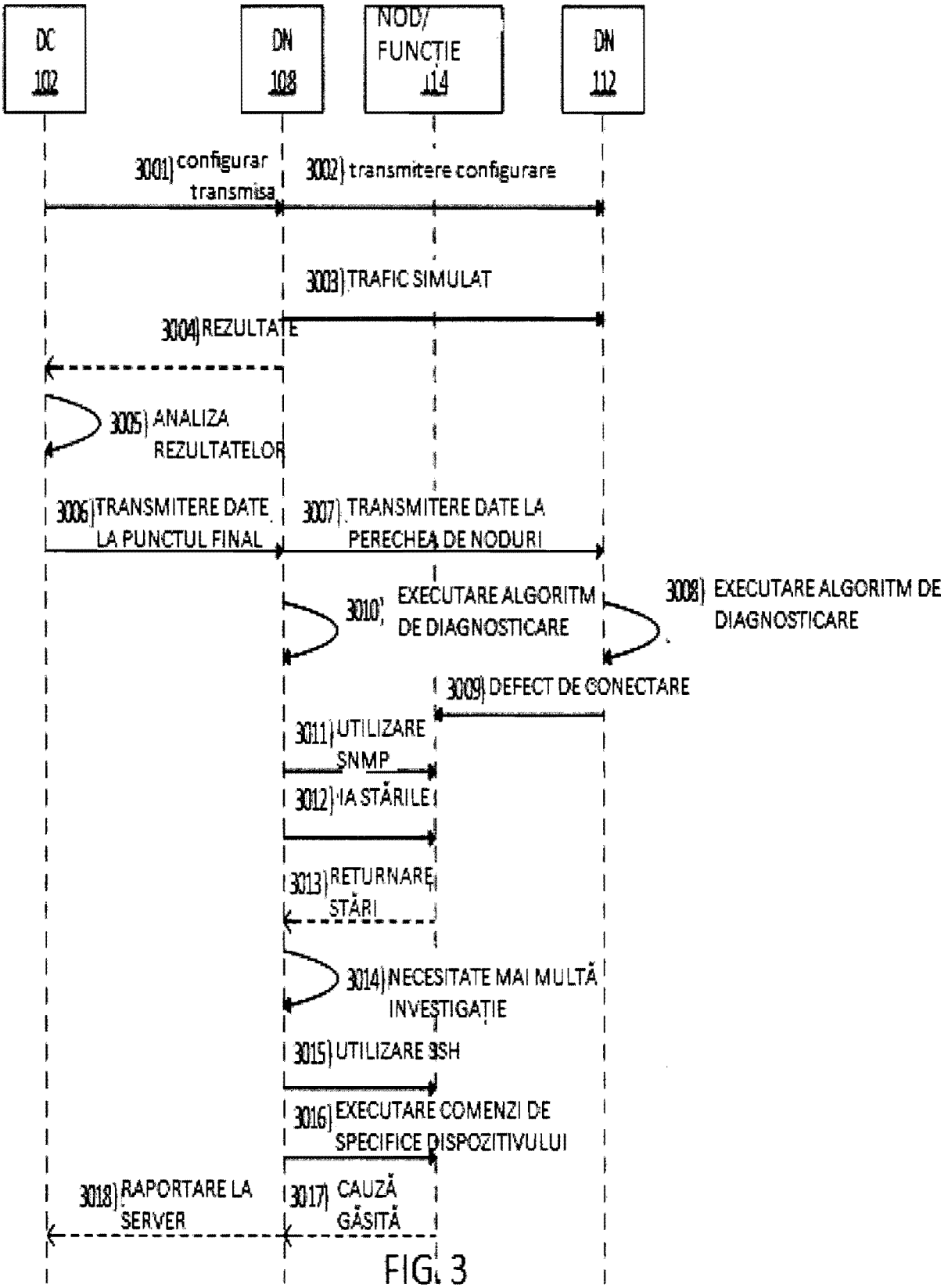
DC
102
DN
108
NOD/
FUNCTIE
114
DN
112
3001] configurar transmisa
3002] transmitere configurare
3003] TRAFIC SIMULAT
3004] REZULTATE
3005] ANALIZA REZULTATELOR
3006] TRANSMITERE DATE LA PUNCTUL FINAL
3007] TRANSMITERE DATE LA PERECHEA DE NODURI
3008] EXECUTARE ALGORITM DE DIAGNOSTICARE
3010] EXECUTARE ALGORITM DE DIAGNOSTICARE
3009] DEFECT DE CONECTARE
3011] UTILIZARE SNMP
3012] IA STĂRILE
3013] RETURNARE STĂRI
3014] NECESITATE MAI MULTĂ INVESTIGAȚIE
3015] UTILIZARE SSH
3016] EXECUTARE COMENZI DE SPECIFICE DISPOZITIVULUI
3017] CAUZĂ GĂSITĂ
3018] RAPORTARE LA SERVER

FIG. 3

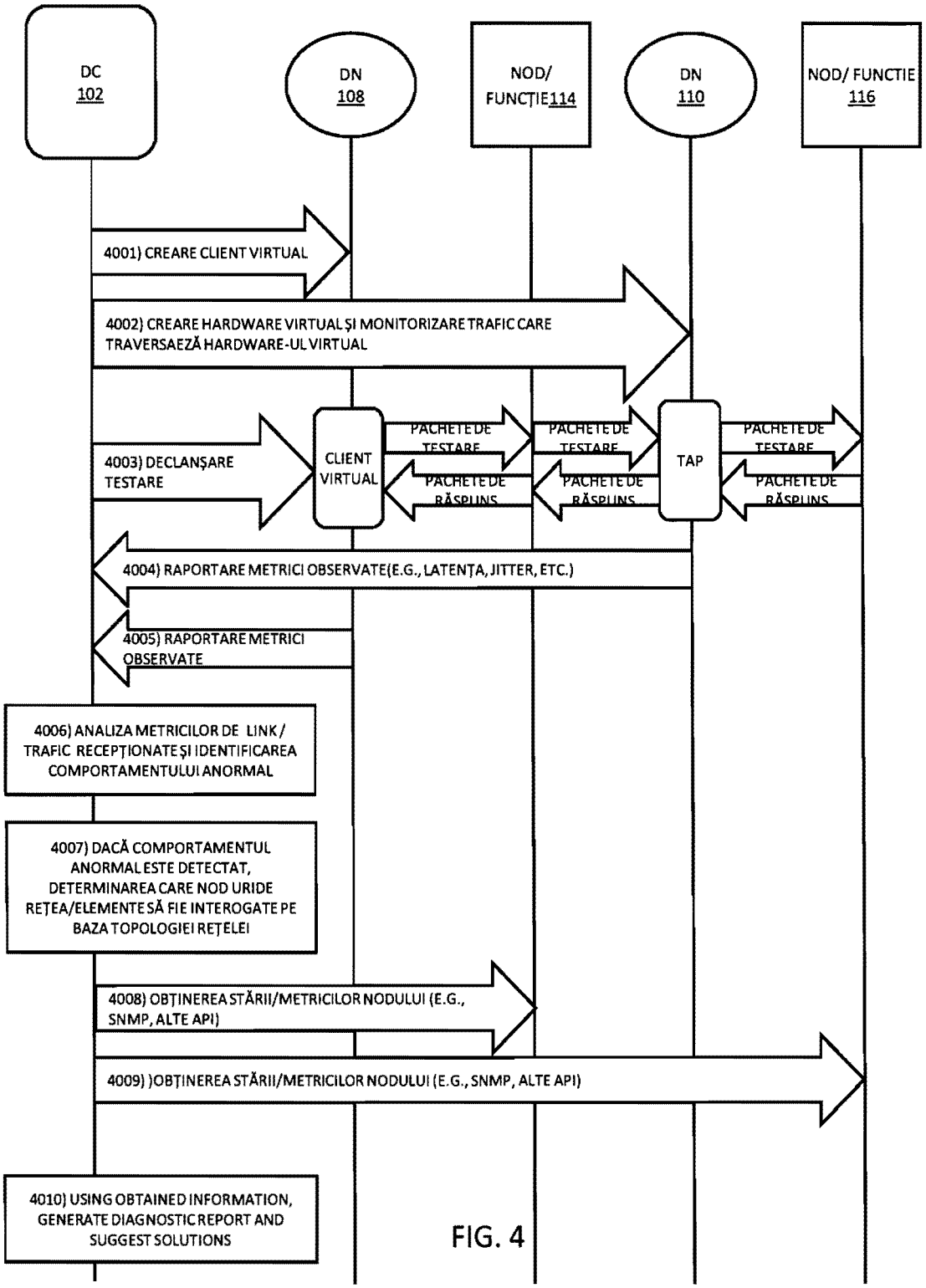
DC
102
DN
108
NOD/
FUNCȚIE114
DN
110
NOD/ FUNCTIE
116
4001) CREARE CLIENT VIRTUAL
4002) CREARE HARDWARE VIRTUAL ȘI MONITORIZARE TRAFIC CARE TRAVERSAEZĂ HARDWARE-UL VIRTUAL
4003) DECLANȘARE TESTARE
CLIENT VIRTUAL
PACHETE DE TESTARE
PACHETE DE RĂSPUNS
PACHETE DE TESTARE
PACHETE DE RĂSPUNS
TAP
PACHETE DE TESTARE
PACHETE DE RĂSPUNS
4004) RAPORTARE METRICI OBSERVATE(E.G., LATENȚA, JITTER, ETC.)
4005) RAPORTARE METRICI OBSERVATE
4006) ANALIZA METRICILOR DE LINK/ TRAFIC RECEPȚIONATE ȘI IDENTIFICAREA COMPORTAMENTULUI ANORMAL
4007) DACĂ COMPORTAMENTUL ANORMAL ESTE DETECTAT, DETERMINAREA CARE NOD URIDE REȚEA/ELEMENTE SĂ FIE INTEROGATE PE BAZA TOPOLOGIEI REȚELEI
4008) OBȚINEREA STĂRII/METRICILOR NODULUI (E.G., SNMP, ALTE API)
4009))OBȚINEREA STĂRII/METRICILOR NODULUI (E.G., SNMP, ALTE API)
4010) USING OBTAINED INFORMATION, GENERATE DIAGNOSTIC REPORT AND SUGGEST SOLUTIONS

FIG. 4

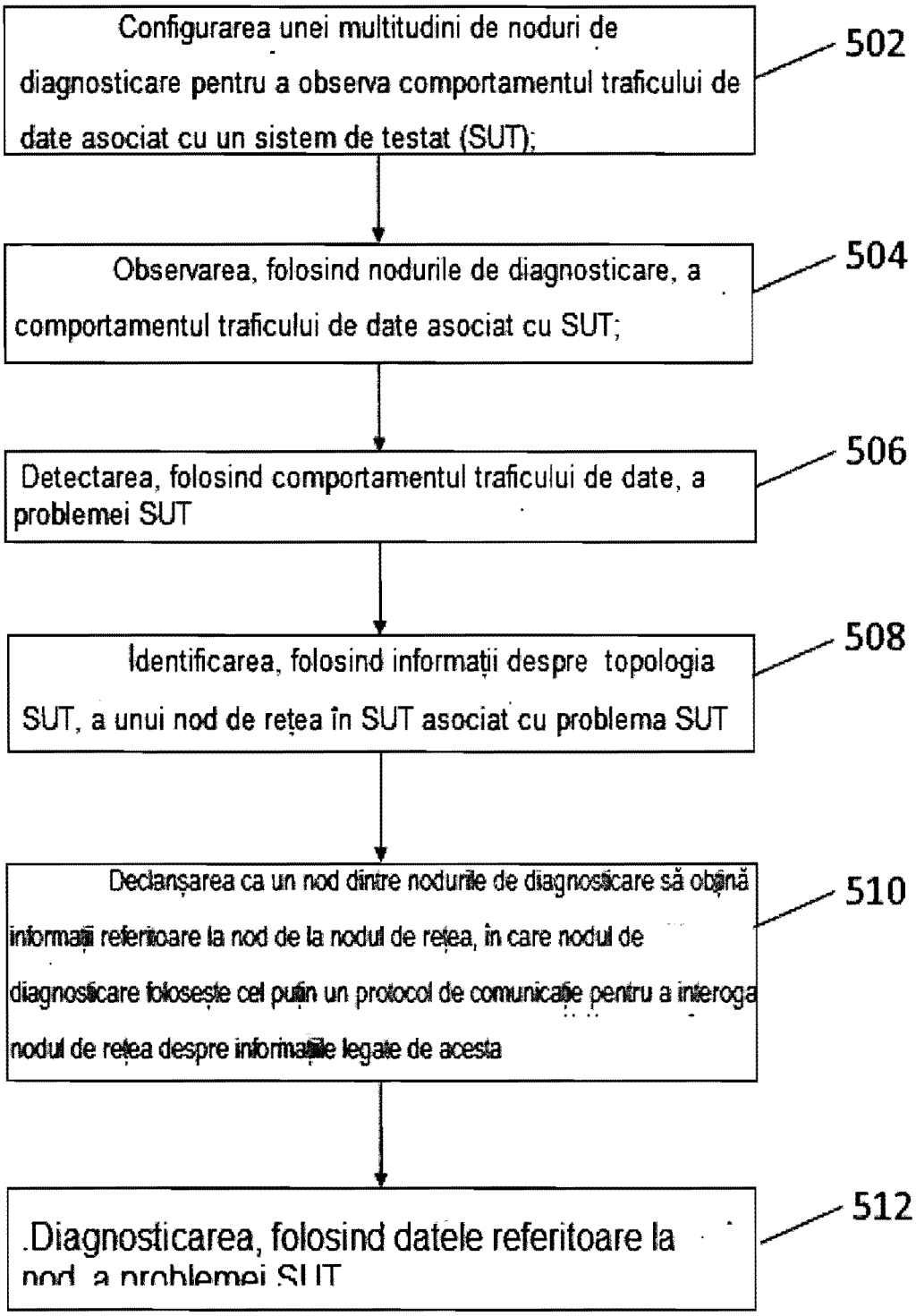
Configurarea unei multitudini de noduri de diagnosticare pentru a observa comportamentul traficului de date asociat cu un sistem de testat (SUT);
502
Observarea, folosind nodurile de diagnosticare, a comportamentul traficului de date asociat cu SUT;
504
Detectarea, folosind comportamentul traficului de date, a problemei SUT
506
Identificarea, folosind informații despre topologia SUT, a unui nod de rețea în SUT asociat cu problema SUT
508
Declanșarea ca un nod dintre nodurile de diagnosticare să obțină informații referitoare la nod de la nodul de rețea, în care nodul de diagnosticare folosește cel puțin un protocol de comunicație pentru a interoga nodul de rețea despre informațiile legate de acesta
510
.Diagnosticarea, folosind datele referitoare la
512

FIG. 5