

(51) International Patent Classification:
G06F 11/30 (2006.01)

(21) International Application Number:
PCT/CN2019/084693

(22) International Filing Date:
26 April 2019 (26.04.2019)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **NOKIA TECHNOLOGIES OY** [FI/FI];
Karaportti 3, 02610 Espoo, (FI).

(71) Applicant (for LC only): **NOKIA TECHNOLOGIES
(BEIJING) CO., LTD.** [CN/CN]; Unit 18, Room 1001,
Level 10, East Tower, World Financial Center, No.1,
East Third Ring Middle Road, Chaoyang District, Beijing
100020 (CN).

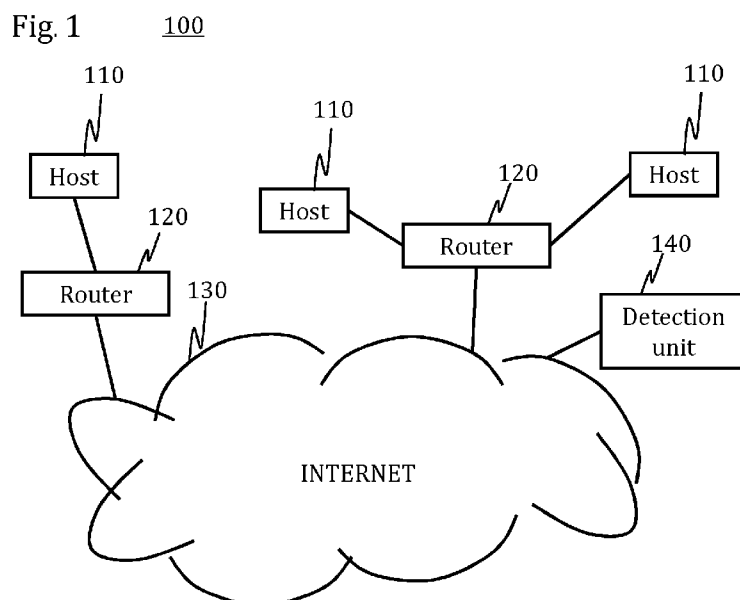
(72) Inventors: **JING, Xuyang**; Xidian University Building 69,
No. 403, No.2 South Taibai Road, Xi'an, Shaanxi 710071
(CN). **YAN, Zheng**; Xidian University Building 69, No.
403, No.2 South Taibai Road, Xi'an, Shaanxi 710071 (CN).

(74) Agent: **KING & WOOD MALLESONS**; 20th Floor,
East Tower, World Financial Centre, No.1 Dongsanhuan
Zhonglu, Chaoyang District, Beijing 100020 (CN).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ,
CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,
HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP,
KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME,
MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ,
OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA,
SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ,
UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ,
TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK,
EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,
MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: HOST CARDINALITY MEASURING FOR NETWORK ANOMALY DETECTION



(57) Abstract: Host cardinality is measured in a process including: applying a multi-dimensional reversible data structure to detect abnormal hosts in a network. The host cardinality is divided into destination cardinality, destination port cardinality and source cardinality. Corresponding apparatus and computer program and memory storing the computer program are also disclosed.

TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

HOST CARDINALITY MEASURING FOR NETWORK ANOMALY DETECTION

TECHNICAL FIELD

5 [0001] Various example embodiments relate to host cardinality measuring for network anomaly detection.

BACKGROUND

 [0002] This section illustrates useful background information without admission of any technique described herein representative of the state of the art.

10 [0003] Accurately and timely network traffic measuring assists network security. Such a measuring may be crucial for anomaly detection and defense, attack attribution, and network information forensics, among others. Among these applications, monitoring host cardinality is one useful method in network traffic measurement. The cardinality of a host represents the number of distinct peers it
15 communicates with. In this document, Destination Cardinality (DC) refers to a number of distinct destination IP addresses that a host is connected to, Destination Port Cardinality (DPC) refers to a number of distinct destination ports that a host is connected to, and Source Cardinality (SC) refers to a number of distinct source addresses that a host is connected with. DC and DPC can be used to identify
20 attacking hosts while SC can be used to detect a victim in attack detection and defense.

 [0004] A host with high DC or DPC, known as super source/spreader/point or high SC known as a super destination, is a kind of host which relates to many network anomalies. For example, during host scanning such
25 as horizontal scan and vertical scan in worm propagation, a compromised host may try to connect to as many as possible different destination IP addresses or ports in a short time. In spam distributing, a malicious host sends same content to many distinct destination IP addresses. Both of these hosts are super points. Distributed Denial of Service (DDoS) flooding attacks are launched by employed
30 massive hosts that have installed malware programs to flood a victim for the purpose of occupying its resources. Currently, in order to avoid being detected, attackers often use a large number of source IP addresses each to send few packets

to the victim. In this case, the destination IP address with high SC is a candidate for a victim. Thus, identifying the host with high DC/DPC/SC may help detecting and defending some typical network attacks.

[0005] A naive and accurate method for measuring host cardinality would be to maintain a table to track all distinct hosts with which a host communicates. However, this method would cause excessive memory consumption and content access due to large amount of networking traffic data, especially in high-speed and/or large networks.

10 SUMMARY

[0006] Various aspects of examples of the invention are set out in the claims.

[0007] According to a first example aspect of the present invention, there is provided a method of measuring host cardinality, comprising:

15 [0008] applying a multi-dimensional reversible data structure to detect abnormal hosts in a network; and

[0009] wherein the host cardinality is divided into destination cardinality, destination port cardinality and source cardinality.

20 [0010] Various abnormal network events of hosts may be detected from high cardinality. The abnormal network events may include any one or more of:

[0011] conducting worm propagation, conducting spam distribution and being exposed to a distributed denial of service flooding attack.

25 [0012] The multi-dimensional reversible data structure may be a probabilistic data structure configured for counting distinct communications among hosts. The probabilistic data structure may use three dimensions to separately record traffic information about a source IP address, a destination IP address and a destination port.

30 [0013] The method may comprise performing an update operation. In the update operation, when a new packet arrives, the probabilistic data structure may be updated by setting a bit of a corresponding buckets where the packet locates. This operation may be realized by using data-oriented hash functions. The data-oriented hash functions may be based on Chinese remainder theory. The updating

may be performed in a packet data network to monitor network traffic at real-time.

[0014] The method may further comprise a point estimation operation. In the point estimation operation, for a given source or destination IP address, respective source or destination cardinality may be queried using a probabilistic counting with a given level of accuracy. This operation may allow locating abnormal hosts with high cardinality.

[0015] The method may further comprise a combination operation. In the combination operation, a plurality of the multi-dimensional reversible data structures with same parameters may be combined into a single one using bitwise-OR operator. This operation may be used to support a distributed collection of network traffic and subsequent centralized analysis.

[0016] The method may further comprise reversible calculation operation. The reversible calculation operation may be performed on detecting abnormal hosts with high cardinality. The reversible calculation operation may attempt reversibly reconstructing original IP addresses. This may avoid completing an exhaustive exploration of all possible addresses.

[0017] The method may comprise network anomaly detection. The network anomaly detection may comprise any one or more of the following:

[0018] 1) performing the update operation to record traffic information about source IP addresses, destination IP addresses and ports;

[0019] 2) performing an adaptive high cardinality host identification algorithm based on the traffic data recorded in the multi-dimensional reversible data structure;

[0020] 3) Reversibly calculating with the multi-dimensional reversible data structures to reconstruct original IP addresses of abnormal hosts that hold high cardinality;

[0021] 4) performing anomaly attribution to classify anomaly types for selecting one or more suitable anomaly defense or mitigation mechanisms.

[0022] The multi-dimensional reversible data structure may be a SuperSketch.

[0023] The network security measurement method may be used to measure an aggregate security level of a network.

[0024] According to a second example aspect of the present invention, there is provided an apparatus comprising a memory and a processor that are configured to cause the apparatus to perform the method of the first example aspect.

5 [0025] According to a third example aspect of the present invention, there is provided a computer program comprising computer executable program code configured to execute any method of the first example aspect.

[0026] The computer program may be stored in a computer readable memory medium.

10 [0027] Any foregoing memory medium may comprise a digital data storage such as a data disc or diskette, optical storage, magnetic storage, holographic storage, opto-magnetic storage, phase-change memory, resistive random access memory, magnetic random access memory, solid-electrolyte memory, ferroelectric random access memory, organic memory or polymer
15 memory. The memory medium may be formed into a device without other substantial functions than storing memory or it may be formed as part of a device with other functions, including but not limited to a memory of a computer, a chip set, and a sub assembly of an electronic device.

[0028] Different non-binding example aspects and embodiments of the
20 present invention have been illustrated in the foregoing. The embodiments in the foregoing are used merely to explain selected aspects or steps that may be utilized in implementations of the present invention. Some embodiments may be presented only with reference to certain example aspects of the invention. It should be appreciated that corresponding embodiments may apply to other
25 example aspects as well.

BRIEF DESCRIPTION OF THE DRAWINGS

[0029] For a more complete understanding of example embodiments of the present invention, reference is now made to the following descriptions taken in
30 connection with the accompanying drawings in which:

[0030] Fig. 1 shows an architectural drawing of a system of an example embodiment;

[0031] Fig. 2 shows a flow chart of a process of an example embodiment;

[0032] Fig. 3 shows a block diagram of an apparatus of an example embodiment;

[0033] Fig. 4 presents a diagram of a Sketch;

5 [0034] Fig. 5 presents the structure of SuperSketch;

[0035] Fig. 6 presents update operation of SuperSketch;

[0036] Fig. 7 presents point estimation process for destination cardinality and destination port cardinality; and

[0037] Fig. 8 presents point estimation process for Source Cardinality.

10

DETAILED DESCRIPTION OF THE DRAWINGS

[0038] An example embodiment of the present invention and its potential advantages are understood by referring to Figs. 1 through 8 of the drawings. In this document, like reference signs denote like parts or steps.

15 [0039] Various example embodiments are next described.

[0040] Fig. 1 shows an architectural drawing of a system 100 of an example embodiment. The system 100 comprises a plurality of hosts 110, each of which may be either or both sending or receiving hosts. Fig. 1 further shows a plurality of routers 120, the Internet 130 to which the routers 120 connect the hosts 110, and a detection unit 140. The detection unit 140 is drawn as a single box illustrating a functionality, although in some embodiments the detection unit 140 is distributed to two or more separate units or implemented using virtual or cloud computing.

[0041] In an example embodiment, the detection unit 140 is implemented in a firewall or a router. In an example embodiment, the detection unit 140 is implemented in a serving gateway (SGW). In an example embodiment, the detection unit 140 is implemented in an evolved packet data gateway (ePDG). In an example embodiment, the detection unit 140 is implemented in a packet data network gateway (PGW) of an LTE core network. In an example embodiment, the detection unit 140 is implemented in an access and mobility function (AMF). In an example embodiment, the detection unit 140 is implemented in a non-3GPP interworking function (N3IWF) of a 5G core network.

[0042] Fig. 2 shows a flow chart of a process of an example embodiment illustrating a method of measuring host cardinality, comprising:

[0043] 200. applying a multi-dimensional reversible data structure to detect abnormal hosts in a network; wherein the host cardinality is divided into
5 destination cardinality, destination port cardinality and source cardinality.

[0044] 205. detecting various abnormal network events of hosts from abnormal cardinality.

[0045] 210. performing an update operation in which, when a new packet arrives, the probabilistic data structure is updated by setting a bit of a
10 corresponding buckets where the packet locates, using data-oriented hash functions that are based on Chinese remainder theory.

[0046] 215. Performing a point estimation operation for a given source or destination IP address, wherein respective source or destination cardinality is queried using a probabilistic counting with a given level of accuracy.

[0047] 220. performing a combination operation, wherein a plurality of
15 the multi-dimensional reversible data structures with same parameters are combined into a single one using bitwise-OR operator, e.g., to support a distributed collection of network traffic and subsequent centralized analysis.

[0048] 225. performing reversible calculation operation on detecting
20 abnormal hosts with high cardinality, which reversible calculation operation attempts reversibly reconstructing original IP addresses.

[0049] 230. performing network anomaly detection comprising any one or more of the following: 1) performing the update operation to record traffic information about source IP addresses, destination IP addresses and ports; 2)
25 performing an adaptive high cardinality host identification algorithm based on the traffic data recorded in the multi-dimensional reversible data structure; 3) Reversibly calculating with the multi-dimensional reversible data structures to reconstruct original IP addresses of abnormal hosts that hold high cardinality; 4) performing anomaly attribution to classify anomaly types for selecting one or
30 more suitable anomaly defense or mitigation mechanisms.

[0050] 235. using a SuperSketch as the multi-dimensional reversible data structure.

[0051] 240. Measuring an aggregate security level of a network.

[0052] Fig. 3 shows a block diagram of an apparatus 300 of an example embodiment, such as the detection unit 140. The apparatus 300 comprises an input/output function, a processing function 320, an optional user interface 330, a memory 340 and computer program code 350 stored in the memory 340. The processor 320 is configured to control the operation of the apparatus 300 based on the computer program code 350 when executing the program code 350.

[0053] The input/output function 310 310 comprises, for example, a local area network (LAN) port such as a base 100/base 1000 Ethernet port; a data bus; a wireless local area network (WLAN) unit; cellular data communication unit; or satellite data communication unit. The processing function 320 comprises, for example, any one or more of: a master control unit (MCU); a microprocessor; a digital signal processor (DSP); an application specific integrated circuit (ASIC); a field programmable gate array; a microcontroller; a portion of one or more virtualized or cloud computing processing cores.

[0054] An example embodiment is based data summarization technique called Sketch. The Sketch is a probabilistic data summarization technique composed of H hash tables of size M , as shown in Fig. 4. Incoming network traffic is modeled as pairwise items encompassing a *key* and an associated *value*, where the key can be one or more fields in packet headers, and the value can be any accumulative feature, such as the size of a packet, the number of packets, etc. Each row i is associated with an independent hash function h_i that maps an incoming key into a hashing space of $\{0, 1, 2, \dots, M-1\}$. The data whose keys are hashed to the same values will be put in the same bucket and the values will be added up. Usually, H hash functions in a sketch are chosen from the family of k -universal hash functions

$$h(x) \equiv \sum_{i=0}^{k-1} (a_i x^i + b_i) \bmod r \bmod M \quad (1)$$

defined in Equation (1),

where r is an arbitrary prime (e.g., Mersenne prime numbers are chosen for fast implementation), $a_i (\neq 0)$ and b_i are randomly selected from the set of $\{0, 1, \dots, r-1\}$, M is the width of sketch. Using k -universal hash functions, the probability that two keys are aggregated in the same bucket over H hash tables is $(1/M)^{kH}$.

[0055] A SuperSketch is described as $\mathbf{SS} = (SS_1, SS_2, \dots, SS_N)$, as shown in Fig. 5. Each SS_i is a bit cube (three-dimensional bit arrays) with the size of $p_i \times p_i \times u_i$, where $1 \leq i \leq N$, p_i and u_i are integers. Multiple bit cubes (or, more typically rectangular cuboids) with different sizes used here aim to accurately collect as much traffic information as possible and reversibly recover abnormal keys. The height, length and width of SS_i are represented as L_1 , L_2 and L_3 , respectively. In SS_i , $SS_i[j][\cdot][\cdot]$ represents j -th row in L_1 , $SS_i[\cdot][k][\cdot]$ represents k -th column in L_2 , $SS_i[\cdot][\cdot][l]$ denotes l -th column in L_3 and $SS_i[j][k][l]$ denotes bit value recorded in (j, k, l) bucket, where $i \in \{1, 2, \dots, N\}$, $j \in \{0, 1, 2, \dots, p_i - 1\}$, $k \in \{0, 1, 2, \dots, p_i - 1\}$, and $l \in \{0, 1, 2, \dots, u_i - 1\}$.

[0056] With *SuperSketch*, we select the *key* as Source IP address in L_1 , Destination IP address in L_2 , and Destination port in L_3 . The space of source IP addresses is the same as the space of destination IP addresses. Therefore, L_1 and L_2 of SS_i are associated with the same hash functions, $h_i(x) \equiv (ax + b) \bmod r \bmod p_i$, where $i \in \{1, 2, \dots, N\}$, r is an arbitrary prime number, $a (\neq 0)$ and b are randomly selected from the set of $(0, 1, \dots, r - 1)$, $p_1, p_2, p_3, \dots, p_N$ are selected as pair-wise coprime numbers around an integer P to guarantee the probability of hash collision in a certain range and they satisfy $p_1 > p_2 > p_3 \dots > p_N$. In an embodiment, descending order of $p_1, p_2, p_3, \dots, p_N$ is used to improve the accuracy of recording and reversely recovering because the larger the p_1 is, the less probability of two different keys hashed into the same bucket. L_3 of SS_i is associated with the following hash functions, $h'_i(x) \equiv (a'x + b') \bmod u_i$, where u_i is a prime number near an integer U . That the prime number is near an integer is in an example embodiment a difference not greater than 1, 5, 10 or 20 %, or not greater than 1, 5, 10, 20, 50, or 100.

[0057] In an example embodiment, $h_i(x)$ of L_1 and L_2 and $h'_i(x)$ of L_3 are data-oriented hash functions configured to obtain the location index using randomly hashing. In an example embodiment, we set $a = a' = 1$, $b = b' = 0$ and r is a large arbitrary prime number but less than the greatest one of $p_1, p_2, p_3, \dots, p_N$.

[0058] In an example embodiment, hash functions associated with L_1 , L_2 and L_3 are:

$$\begin{cases} L_1 \text{ and } L_2: h_i(x) \equiv x \bmod p_i, i \in \{1, 2, \dots, N\} \\ L_3: h'_i(x) \equiv x \bmod u_i, i \in \{1, 2, \dots, N\} \end{cases} \quad (2)$$

[0059] In an example embodiment, following four basic operations are applied in SuperSketch:

[0060] (1) Update Operation

[0061] Update operation of *SuperSketch* is used to collect traffic information. Initially, all bits in *SuperSketch* are set to zero. When a new packet (S_{IP} , D_{IP} , D_P) arrives, each $SS_i (1 \leq i \leq N)$ is updated by setting the bit to the corresponding bucket where the packet locates. The update formula is described as follows:

$$SS_i[h_i(S_{IP})][h_i(D_{IP})][h'_i(D_P)] = 1, 1 \leq i \leq N \quad (3)$$

[0062] The update process of *SuperSketch* is described in Algorithm 1 and shown in Fig. 6 (the marked bucket is updated). Notably, the packets come from a specified source IP address are hashed into the same row in L_1 in $SS_i (1 \leq i \leq N)$ but may be hashed into different buckets due to the different destination IP address or port number.

[0063] The aforementioned algorithm 1 for SuperSketch update is implemented in an example embodiment as follows:

1. **Initialize** $SS = (SS_1, SS_2, \dots, SS_N)$:
2. $SS_i[j][k][l] = 0, i \in \{1, 2, \dots, N\}, j \text{ and } k \in \{0, 1, 2, \dots, p_i - 1\}, l \in \{0, 1, 2, \dots, u_i - 1\}$
3. **Update:**
4. New arriving packet (S_{IP} , D_{IP} , D_P)
5. **for** $i = 1$ to N **do**
6. $height = h_i(S_{IP})$
7. $length = h_i(D_{IP})$
8. $width = h'_i(D_P)$
9. $SS_i[height][length][width] = 1$
10. **end for**

[0064] *Update time:* for each SS_i , it needs to take four times of calculation to update a bucket (from line 6 to 9 in Algorithm 1). The total times of calculation for SS is thus only $4 \times N$.

[0065] (2) Point Estimation Operation

[0066] In an example embodiment, point estimation operation of *SuperSketch* is used to return an approximate estimation of host cardinality for an

IP addresses. Given a specified source IP address e , $e \in \mathbf{E}$ and a destination IP address q , $q \in \mathbf{Q}$, where $\mathbf{E}$ and $\mathbf{Q}$ denote the set of all available IP addresses in the network.

[0067] There are two point estimation operations for e , namely (e) and (e) . Both of them are the estimation results over $\mathbf{SS} = (SS_1, SS_2, \dots, SS_N)$.

[0068] We take the example of the point estimation process of $DC_i(e)$ and $DPC_i(e)$ in SS_i ($1 \leq i \leq N$), as shown in Fig. 7. The estimation process is described in Algorithm 2. First, we calculate $h_i(e)$ to locate the row on L_1 where the packets come from e are hashed into. After that, a two dimensional bit array that records host cardinality information is obtained (L_3 is a row, L_2 is a column). For estimating $DC_i(e)$, as shown in Step 3 (a), a bitwise-OR operation is conducted on the two dimensional bit array from the first row to the last row. At last, we obtain a bit array of size p_i used to estimate $DC_i(e)$. We use v to denote the number of zero bit buckets in estimation bit array. The estimation bit array can be viewed as a bitmap. A famous probabilistic counting algorithm operated on bitmap can be applied to calculate $DC_i(e)$:

$$DC_i(e) = -p_i \ln\left(\frac{v}{p_i}\right) \quad (4)$$

[0069] $DC(e)$ is the mean value of set $\{DC_1(e), DC_2(e), \dots, DC_N(e)\}$:

$$DC(e) = \frac{\sum_{i=1}^N DC_i(e)}{N} \quad (5)$$

[0070] Estimation bit array (L_3 is a row, L_2 is a column) of size u_i of $DPC_i(e)$ is obtained by taking the bitwise-OR operation from the first column to the last column, as shown Step 3 (b) in Fig. 7. Based on the same method, $DPC(e)$ is estimated as follows:

$$DPC(e) = \frac{\sum_{i=1}^N -u_i \ln\left(\frac{v}{u_i}\right)}{N} \quad (6)$$

[0071] There is only one point estimation operation for q , namely (q) . Fig. 8 shows the estimation process of $SC_i(q)$. The process is similar to the estimation process of $DC_i(e)$, which we omit here.

[0072] Algorithm 2: Point Estimation Process of *SuperSketch*

1. Input:

2. *SuperSketch* that records traffic information $\mathbb{S} = (\mathbb{S}_1, \mathbb{S}_2, \dots, \mathbb{S}_N)$

3. A source IP address $\mathbb{S}$

4. **Output:** ($\mathbb{S}$)

5. **Begin:**

6. **Point estimation of ($\mathbb{S}$):**

7. **for** $\mathbb{S} = 1$ to $\mathbb{S}$ **do**

8. calculate $h_i(\mathbb{S})$ to locate a row on L_1

9. conduct slice to get a two dimensional bit array (L_3 is a row, L_2 is a column)

10. conduct a bitwise OR operation from a first row to a last row

11. calculate a number of zero bit buckets $\mathbb{S}$

12. $\mathbb{S}_i(\mathbb{S}) = -\mathbb{S}_i \mathbb{S}(\mathbb{S}/\mathbb{S}_i)$

13. **end for**

14. $DC(e) = \sum_{i=1}^N DC_i(e)/N$

15. **End**

[0073] (3) Combination Operation

[0074] Multiple *SuperSketchs*, namely $\mathbf{SS}^1 = (\mathbb{S}_1^1, SS_2^1, \dots, SS_N^1)$, $\mathbf{SS}^2 = (\mathbb{S}_1^2, SS_2^2, \dots, SS_N^2)$, ... $\mathbf{SS}^T = (\mathbb{S}_1^T, SS_2^T, \dots, SS_N^T)$ with the same parameters can be combined into a single one $\mathbf{SS} = (SS_1, SS_2, \dots, SS_N)$ through conducting a combination operation. The combination process is described as follows:

$$SS_i[j][k][l] = SS_i^1[j][k][l] \oplus SS_i^2[j][k][l] \oplus \dots \oplus SS_i^T[j][k][l] \quad (7)$$

where $SS_i^t[j][k][l]$ ($1 \leq t \leq T$) denotes the bit value of the corresponding bucket in $\mathbf{SS}^t$, $1 \leq i \leq N$, $j \in \{0, 1, 2, \dots, p_j - 1\}$, $k \in \{0, 1, 2, \dots, p_k - 1\}$, $l \in \{0, 1, 2, \dots, s_l - 1\}$, and $\oplus$ is a bitwise-OR operator.

[0075] The combination operation of *SuperSketch* allows us to conduct a distributed collection of network traffic using *SuperSketch* and then carry out a centralized analysis. The combined *SuperSketch* has global network traffic information as if all the network traffic passes through the same router.

[0076] (4) Reversible Calculation Operation

[0077] The reversible calculation operation aims to reversely recovering a set of keys that exhibit abnormal. Taking an example of reversibly calculating S_{IP} ,

we first discuss the simplest scenario that assumes one S_{IP} exhibits abnormal in L_1 in $SS_i (1 \leq i \leq N)$. There are only N marked abnormal rows over SS , denoted as $c_1, c_2, \dots, c_N$. We get the following simultaneous congruence:

$$h_1(S_{IP}) \equiv c_1 \bmod p_1; h_2(S_{IP}) \equiv c_2 \bmod p_2; \dots; h_N(S_{IP}) \equiv c_N \bmod p_N \quad (8)$$

5 **[0078]** Based on Chinese Remainder Theory, S_{IP} can be determined with the following expression:

$$S_{IP} \equiv \sum_{i=1}^N Q_i Q'_i c_i \bmod p \quad (9)$$

where $p = p_1 p_2 \dots p_N$, $Q_i = p/p_i$, and $Q_i Q'_i \equiv 1 \bmod p_i, i = (1, 2, \dots, N)$.

[0079] The solution of formula (9) is unique in the case of p is greater than or equal to the size of E . However, under the general situation, assume w ($w > 1$) source IP addresses exhibit abnormal. It means that there are w marked abnormal rows in L_1 of SS_i . If we use similar recover principle used in the simplest scenario, we will get w^N different possible combinations of abnormal rows selected from $SS_i (1 \leq i \leq N)$. This not only leads to heavy computational burden of reversibly calculating but also increases false positive rate.

[0080] In order to solve this problem, we employ the notion of modular coefficient notation in number theory to record the remainders for further row number combination. For an S_{IP} , its modular coefficient notation is $(h_1(S_{IP}), h_2(S_{IP}), \dots, h_N(S_{IP}))$, $i = 1, 2, \dots, N$. That is to say, modular coefficient notation records the row number in L_i that S_{IP} locates in $S_i (1 \leq i \leq N)$. We append a *flag* to the row $SS_i[h_i(S_{IP})][\cdot][\cdot]$ to record the modular information (row number) of S_{i+1} obtained from modular coefficient notation, namely $flag = h_{i+1}(S_{IP}), i = 1, 2, \dots, N - 1$. Each row in L_i of S_i is represented as $SS_i[h_i(S_{IP})][\cdot][\cdot](flag) (1 \leq i \leq N - 1)$.

[0081] Therefore, when there are many marked abnormal rows in $SS_i (1 \leq i \leq N)$, we match the modular information (row number) recorded by the *flag* from SS_i to SS_N with precalculated modular coefficient notation of the keys that have been hashed into the *SuperSketch*. After obtaining matched combinations of abnormal rows, we can accurately recover the abnormal keys using CRT based on the matched modular coefficient notation. The calculation process is same as the

calculation process of the simplest scenario. The rationality of this design is that if S_{IP} is malicious, it will exhibit abnormal in $SS_1, SS_2, \dots, SS_N$.

[0082] Network Traffic Measurement Based on SuperSketch

[0083] High Cardinality Host Identification and Recovery

5 **[0084]** For detecting super sources (or super destinations), we need to first locate super rows in L_1 (or super columns in L_2) in $SS_1, SS_2, \dots, SS_N$ which caused by highly possible super sources (or super destinations). We take the example of identifying super source. For identifying super destination, the process is similar, which we omit here. In $SS_i (1 \leq i \leq N)$, for any row $SS_i[j][\cdot][\cdot]$ ($0 \leq j \leq p_i - 1$, a source IP address are hashed into), its DC and DPC are calculated by the point estimation process and following functions:

$$DC_i(SS_i[j][\cdot][\cdot]) = -p_i \ln\left(\frac{v_{DC}}{p_i}\right) \quad (10)$$

$$DPC_i(SS_i[j][\cdot][\cdot]) = -u_i \ln\left(\frac{v_{DPC}}{u_i}\right) \quad (11)$$

where v_{DC} and v_{DPC} denotes the number of zero-bit buckets in estimation bit array (defined in subsection 5.2). We define the total number of destination cardinality as $F1_i$ and the total number of destination port cardinality as $F2_i$ in L_1 of SS_i , which are computed as follows:

$$F1_i = \sum_{j=1}^{p_i} DC_i(SS_i[j][\cdot][\cdot]) \quad (12)$$

$$F2_i = \sum_{j=1}^{u_i} DPC_i(SS_i[j][\cdot][\cdot]) \quad (13)$$

[0085] $SS_i[j][\cdot][\cdot]$ is considered as a super row when it satisfies at least one of the following conditions:

$$\begin{cases} DC_i(SS_i[j][\cdot][\cdot]) \geq \varepsilon F1_i \\ DPC_i(SS_i[j][\cdot][\cdot]) \geq \varepsilon F2_i \end{cases} \quad (14)$$

where $0 < \varepsilon < 1$ is a tunable percentage threshold.

[0086] After super rows are detected in each $SS_i (1 \leq i \leq N)$, we get N lists that stored abnormal row number. Thus, original addresses of malicious host are obtained after using reversible calculation operation of SuperSketch.

25 **[0087] Anomaly Attribution**

[0088] After detecting anomaly with high cardinality (super sources or super destinations), we need to attribute anomaly types to select proper defense mechanism, for example, determining the host whether is suffering from DDoS flooding attacks or conducting worm scanning (horizontal scan and vertical scan).

5 Descriptions of these anomalies are introduced in the following Table I.

Table I. Anomaly Descriptions

Anomaly types	Description
DDoS flooding attacks	Many S_{IP} s send attack packets to a D_{IP} on a specific port
Horizontal scan	A S_{IP} sends scan packets to many D_{IP} s on a specific port
Vertical scan	A S_{IP} sends scan packets to a D_{IP} on many ports

[0089] For a given super source se ($se \in E$), the attribution rules are described as follows:

10 **[0090]** If $(se)/(se) > 10$, then se conducts horizontal scan. The reason is that under the horizontal scan, the se contacts with as many as distinct destination IP addresses on a specific port as possible, which leads to high (se) but low (se) .

[0091] If $(se)/(se) > 10$, then se conducts vertical scan. The reason is that under the vertical scan, the se contacts with various ports of a target host, which
15 leads to low (se) but high (se) .

[0092] For a given super destination, it is being attacked by DDoS flooding attacks with high probability due to the basic characteristic of DDoS flooding attacks that overwhelm a victim using numerous infected hosts.

[0093] In summary, we describe the process of using *SuperSketch* to
20 measure host cardinality for network anomaly detection in Algorithm 3.

[0094] Algorithm 3: Network Anomaly Detection with *SuperSketch*

1. Use update operation of *SuperSketch* ($\mathbb{S} = (\mathbb{S}_1, \mathbb{S}_2, \dots, \mathbb{S}_N)$) to record traffic information.
- 25 2. For each $\mathbb{S}_i (1 \leq i \leq N)$ in *SuperSketch*, locate abnormal rows (columns) in $L_1 (L_2)$ using high cardinality identification method.
3. Call reversible calculation operation (9) of *SuperSketch* to compute abnormal source addresses (or destination addresses).
4. For each abnormal source address (destination address), conduct anomaly

attribution to identify anomaly types.

5. Take defense mechanisms according to anomaly types.
6. Return to step 1

[0095] Network Security Measurement

5 **[0096]** For measuring the whole security level of a network, we need to calculate the proportion of the abnormal source addresses (destination addresses) in total source addresses (destination addresses) over a period of time. The larger the proportion is, the worse security status of a network. We take the example of calculating the proportion of the abnormal source addresses in total source
 10 addresses, denoted as pro_s . The calculation process of the proportion of the abnormal destination addresses in total destination addresses (denoted as pro_d) is the same as the calculation process of pro_s , which we omit here.

[0097] High pro_s means that there is a large number of malicious traffic entered the network. We define the number of abnormal source addresses is
 15 Num_{as} and the number of distinct source addresses is Num_s in a time interval. Num_{as} can be obtained in high cardinality host identification and recovery step. Initially, all flags in $SS_1, SS_2, \dots, SS_{N-1}$ are set as Null. When a packet with new source address arrives, the flag of the row where this packet locates in L_1 will be set to an integer. Therefore, Num_s can be approximatively estimated by employing the
 20 similar idea of probabilistic counting algorithm. The algorithm of calculating pro_s is described as below:

[0098] Algorithm 4: Calculation process of the proportion of abnormal addresses in total addresses

- 25 **1.** Input:
 - 2.** *SuperSketch* that records traffic information $\mathbf{SS} = (SS_1, SS_2, \dots, SS_N)$
 - 3.** Num_{as} obtained in high cardinality host identification and recovery step
- 4. Output:** pro_s
- 5.** Begin:
- 30 **6. for** $i = 1$ to $N - 1$ **do**
 - 7.** count the number of flags in L_1 that are Null, denoted as v_{null}
 - 8.** the number of distinct source addresses estimated by SS_i is

$$9. \quad Num\_s_i = -p_i \ln \left(\frac{V_{Null}}{p_i} \right)$$

10. end for

$$10. \quad Num\_s = \frac{\sum_{i=1}^{N-1} Num\_s_i}{N-1}$$

$$11. \quad pro_s = Num\_as / Num\_s$$

5 12. End

[0099] After obtaining pro_s and pro_d , we get the following abnormal degree of source addresses (denoted as $degree_s$) and abnormal degree of destination addresses (denoted as $degree_d$):

$$degree_s = \begin{cases} \text{High, if } pro_s \geq \delta_1 \\ \text{Medium, if } \delta_2 \leq pro_s < \delta_1 \\ \text{Low, if } 0 \leq pro_s < \delta_2 \end{cases}$$

$$degree_d = \begin{cases} \text{High, if } pro_d \geq \varphi_1 \\ \text{Medium, if } \varphi_2 \leq pro_d < \varphi_1 \\ \text{Low, if } 0 \leq pro_d < \varphi_2 \end{cases}$$

10 [0100] where $\varnothing_1$, $\varnothing_2$, $\varnothing_1$ and $\varnothing_2$ are preset measurement thresholds. These thresholds and corresponding judgement criteria are set per implementation such that desired level of abnormal event judgement is attained.

[0101] Therefore, the whole security level of the network can be determined based on the following judgements:

15 Table II. Judgements for network security measurement

$degree_s$	$degree_d$		
	High	Medium	Low
High	Very serious	Serious	Serious
Medium	Serious	Low serious	Low serious
Low	Serious	Low serious	Moderate

[0102] A network administrator can select a suitable defense mechanisms based on the result of the measurement method. For a large scale heterogeneous network, its security level can be measured by aggregating the security levels of its sub-networks as calculated based on the above method.

[0103] Without in any way limiting the scope, interpretation, or application of the claims appearing below, a technical effect of one or more of the

example embodiments disclosed herein is ability to perform real-time and distributed network traffic collection also in a high-speed network environment to collect network traffic at real-time thanks the use of data-oriented hash functions.

5 **[0104]** Another technical effect of one or more of the example embodiments disclosed herein is that the efficient combination operation of enables distributed network traffic collection. Distributed network traffic collection covers more traffic information, and outperforms single point measurement methods in measuring host cardinality.

10 **[0105]** Another technical effect of one or more of the example embodiments disclosed herein is enabling finding abnormal host addresses and further enabling network administrator to take proper mitigation actions.

15 **[0106]** Another technical effect of one or more of the example embodiments disclosed herein is enabling identifying of super sources and super destinations simultaneously instead of building various sketches with different keys. Another technical effect of one or more of the example embodiments disclosed herein is enabling detection of high-rate and low-rate DDoS flooding attacks.

20 **[0107]** Another technical effect of one or more of the example embodiments disclosed herein is versatility as three dimensions can be used to separately record traffic information about source IP address, destination IP address and destination port. It is further possible to apply three point estimation operations on recorded traffic, namely Destination Cardinality, Destination Port Cardinality and Source Cardinality.

25 **[0108]** Another technical effect of one or more of the example embodiments disclosed herein is that the sketch data structure can be extended to conduct more network functions. First, it not only supports to collect packet-level (except the content in packet payload) data and flow-level traffic data but also has the ability of collecting connection-level traffic data. Second, it can involve more traffic features to detect other network anomalies that exhibit obvious abnormalities at network/transport layer.

30

[0109] Another technical effect of one or more of the example embodiments disclosed herein is providing a general data structure that supports

detecting attackers that need to conduct a lot of communications with others. The key can be selected as any source identifier in L_1 , any destination identifier in L_2 and any other identifiers in L_3 . Thus, it can be deployed in many network environments to collect data and then perform anomaly detection, such as using SuperSketch in the Internet to constrain worm propagation and bot communications, in Long Term Evolution (LTE) network to eliminate spam distribution, in Wireless Sensor Network (WSN) to prevent fake information spread. For a heterogeneous network, the network administrator can fuse the measurement results obtained from each sub-network to measure the whole security level of the heterogeneous network.

[0110] Embodiments of the present invention may be implemented in software, hardware, application logic or a combination of software, hardware and application logic. In an example embodiment, the application logic, software or an instruction set is maintained on any one of various conventional computer-readable media. In the context of this document, a "computer-readable medium" may be any non-transitory media or means that can contain, store, communicate, propagate or transport the instructions for use by or in connection with an instruction execution system, apparatus, or device, such as a computer, with one example of a computer described and depicted in Fig. 3. A computer-readable medium may comprise a computer-readable storage medium that may be any media or means that can contain or store the instructions for use by or in connection with an instruction execution system, apparatus, or device, such as a computer.

[0111] If desired, the different functions discussed herein may be performed in a different order and/or concurrently with each other. Furthermore, if desired, one or more of the before-described functions may be optional or may be combined.

[0112] Although various aspects of the invention are set out in the independent claims, other aspects of the invention comprise other combinations of features from the described embodiments and/or the dependent claims with the features of the independent claims, and not solely the combinations explicitly set out in the claims.

[0113] It is also noted herein that while the foregoing describes example embodiments of the invention, these descriptions should not be viewed in a limiting sense. Rather, there are several variations and modifications which may be made without departing from the scope of the present invention as defined in the
5 appended claims.

WHAT IS CLAIMED IS:

1. A method of measuring host cardinality, comprising:
applying a multi-dimensional reversible data structure to detect abnormal
5 hosts in a network;
wherein the host cardinality is divided into destination cardinality,
destination port cardinality and source cardinality.
2. The method of claim 1, wherein the multi-dimensional reversible data
10 structure is a probabilistic data structure configured for counting distinct
communications among hosts.
3. The method of claim 1 or 2, wherein the probabilistic data structure uses
three dimensions to separately record traffic information about a source IP
15 address, a destination IP address and a destination port.
4. The method of any one of preceding claims, wherein various abnormal
network events of hosts are detected from abnormal cardinality.
- 20 5. The method of claim 4, wherein the abnormal network events include any
one or more of:
conducting worm propagation;
conducting spam distribution;
being exposed to a distributed denial of service flooding attack.
25
6. The method of any one of preceding claims, wherein the multi-
dimensional reversible data structure is a probabilistic data structure configured
for counting distinct communications among hosts.
- 30 7. The method of any one of preceding claims, wherein the probabilistic data
structure uses three dimensions to separately record traffic information about a
source IP address; a destination IP address; and a destination port.

8. The method of any one of preceding claims, wherein the method further comprises performing an update operation.

5 9. The method of claim 8, wherein the update operation, when a new packet arrives, the probabilistic data structure is updated by setting a bit of a corresponding buckets where the packet locates.

10 10. The method of claim 9, wherein the probabilistic data structure is updated by setting a bit of a corresponding buckets where the packet locates by using data-oriented hash functions.

11. The method of claim 10, wherein the data-oriented hash functions are based on Chinese remainder theory.

15 12. The method of any one of claims 8 to 11, wherein the updating is performed in a packet data network to monitor network traffic at real-time.

20 13. The method of any one of preceding claims, wherein the method further comprises a point estimation operation.

14. The method of claim 13, wherein in the point estimation operation, for a given source or destination IP address, respective source or destination cardinality is queried using a probabilistic counting with a given level of accuracy.

25 15. The method of any one of preceding claims, wherein the method further comprises a combination operation.

30 16. The method of claim 15, wherein in the combination operation, a plurality of the multi-dimensional reversible data structures with same parameters are combined into a single one using bitwise-OR operator.

17. The method of any one of preceding claims, wherein the method further comprises a reversible calculation operation.

18. The method of claim 17, wherein the reversible calculation operation is performed for detecting abnormal hosts with abnormal cardinality.

19. The method of claim 17 or 18, wherein the reversible calculation
5 operation attempts reversibly reconstructing original IP addresses.

20. The method of any one of preceding claims, wherein the method further comprises a network anomaly detection.

10 21. The method of claim 20, wherein the network anomaly detection comprises any one or more of the following:

performing the update operation to record traffic information about source IP addresses, destination IP addresses and ports;

performing an adaptive high cardinality host identification algorithm
15 based on the traffic data recorded in the multi-dimensional reversible data structure;

reversibly calculating with the multi-dimensional reversible data structures to reconstruct original IP addresses of abnormal hosts that hold high cardinality;

20 performing anomaly attribution to classify anomaly types for selecting one or more suitable anomaly defense or mitigation mechanisms.

22. The method of any one of preceding claims, wherein the multi-dimensional reversible data structure is a SuperSketch.

25

23. The method of any one of preceding claims, further comprising using the network security measurement method to measure an aggregated security level of a network.

30 24. An apparatus comprising a memory and a processor that are configured to cause the apparatus to perform the method of any one of preceding claims.

25. A computer program comprising computer executable program code

configured to execute any method of any one of claims 1 to 23.

26. A computer readable memory medium comprising the computer program of claim 25.

Fig. 1

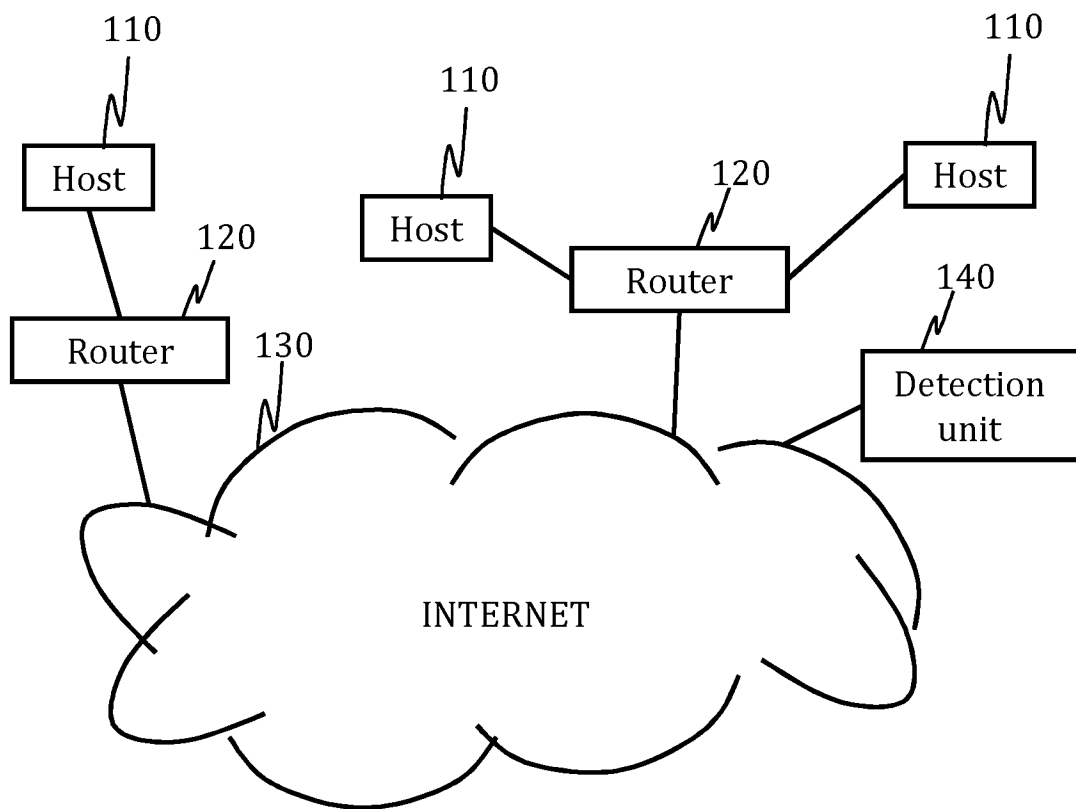


Fig. 3

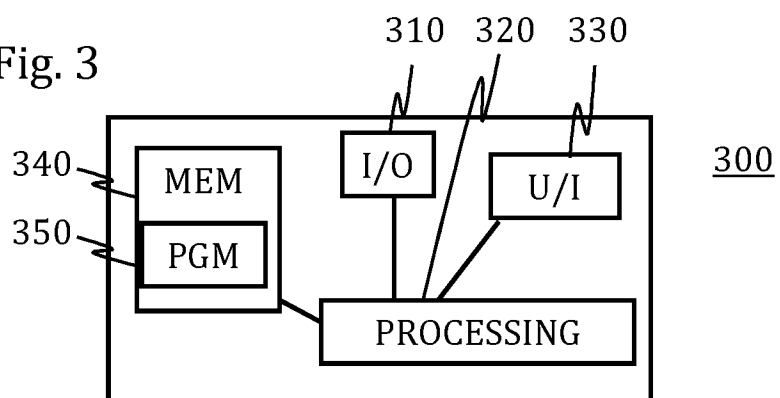
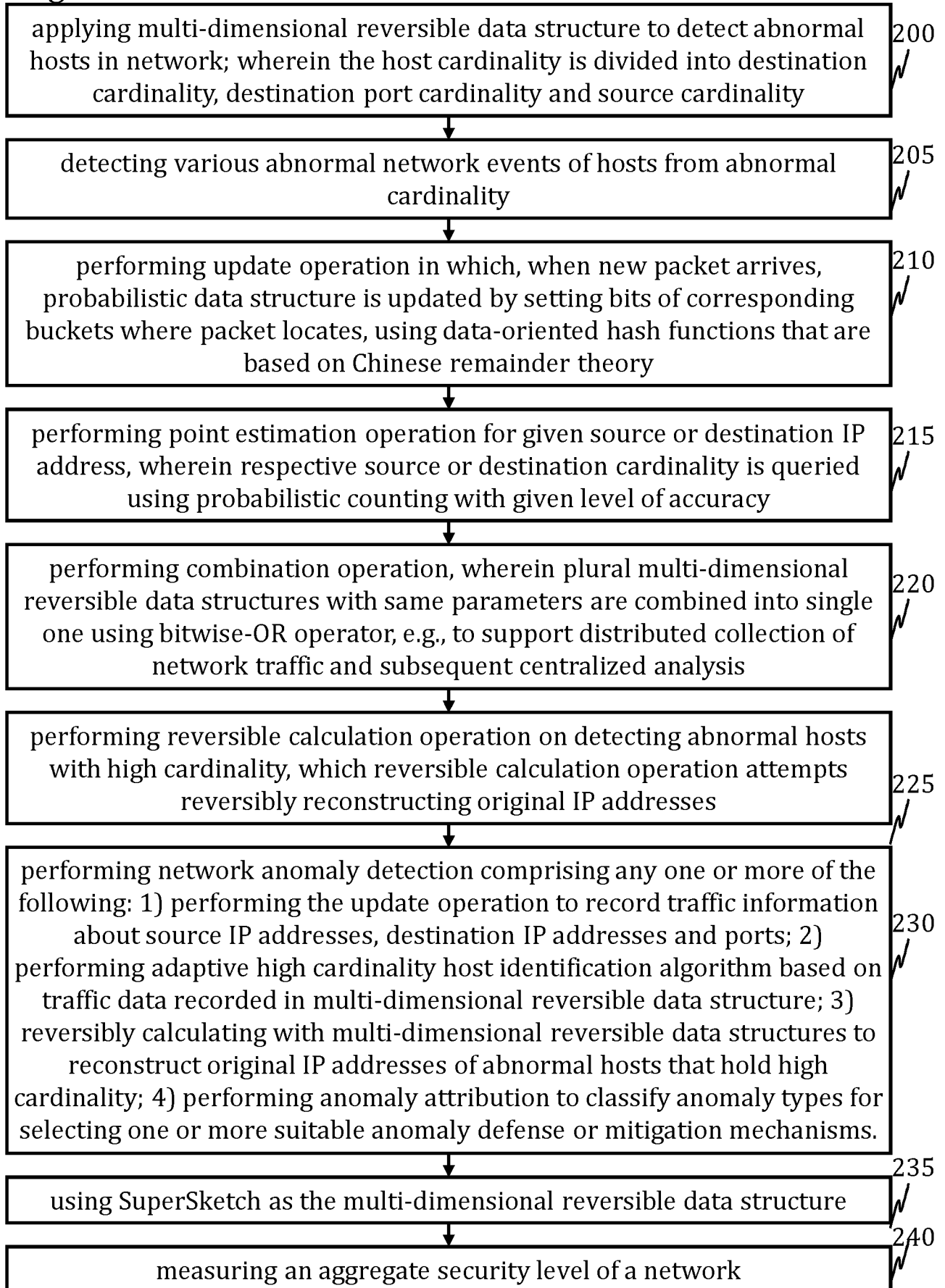


Fig. 2



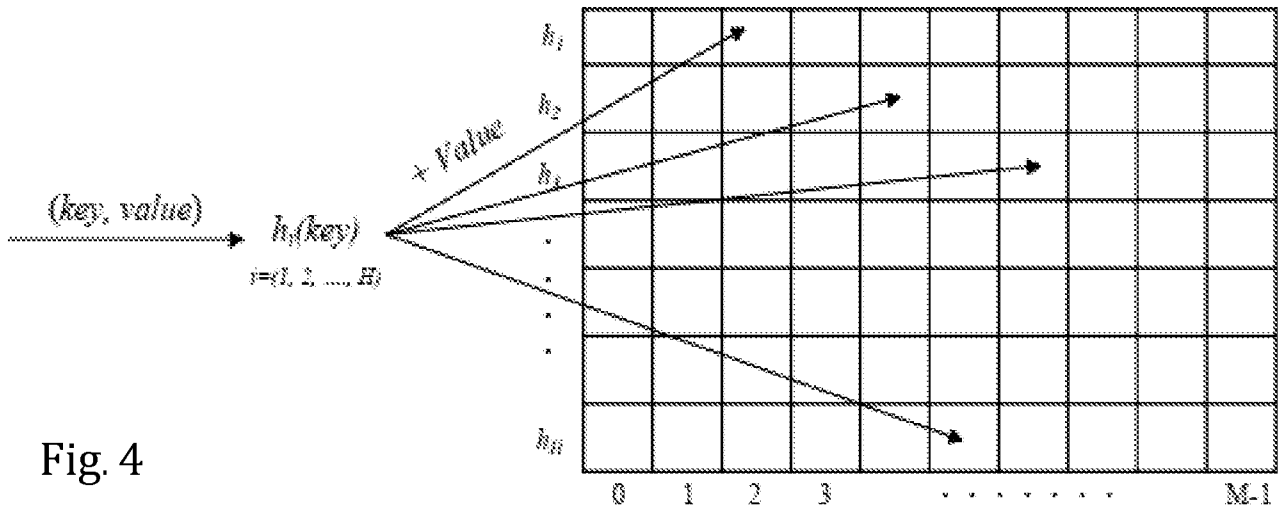


Fig. 4

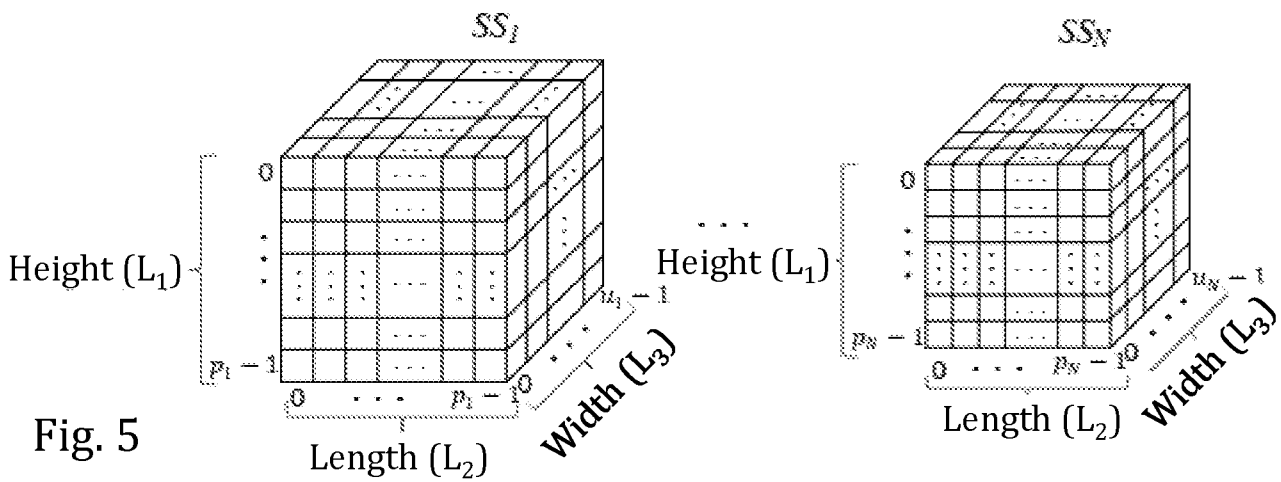


Fig. 5

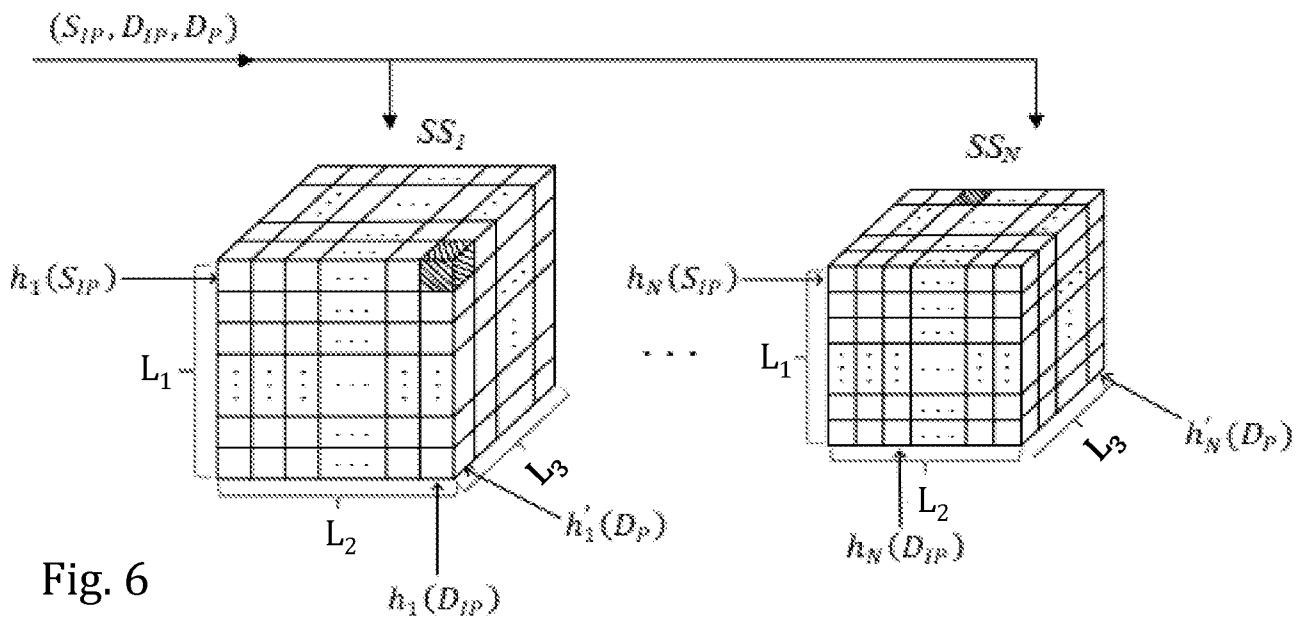


Fig. 6

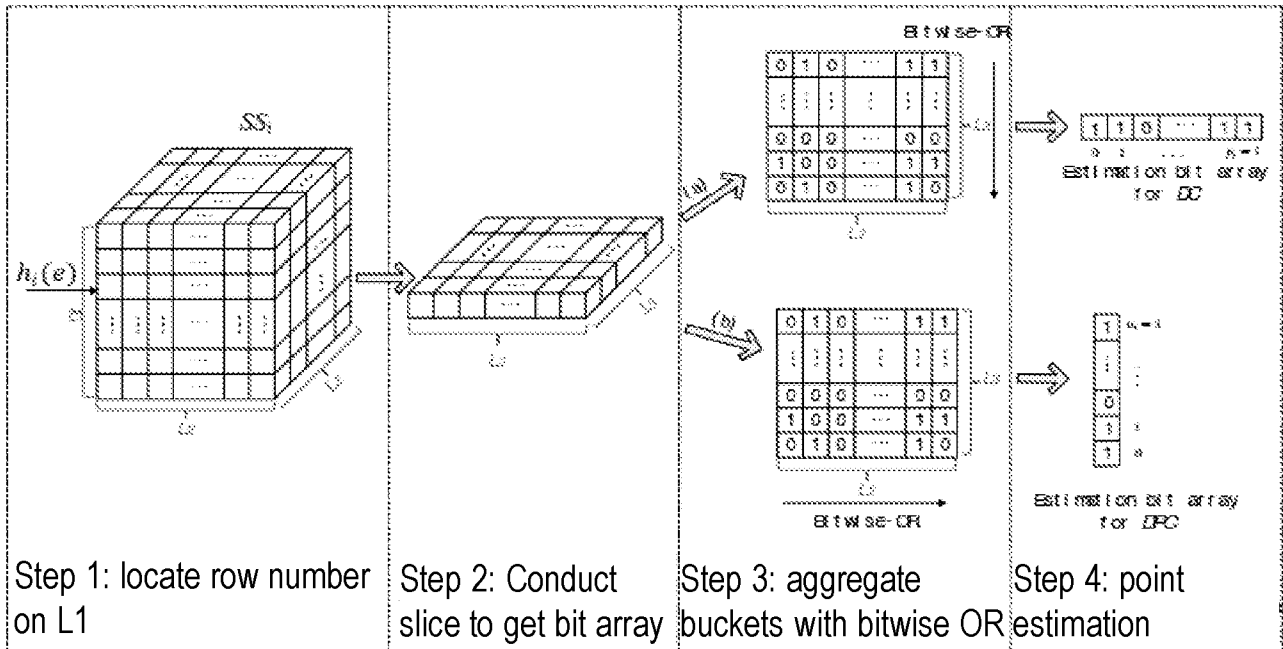


Fig. 7

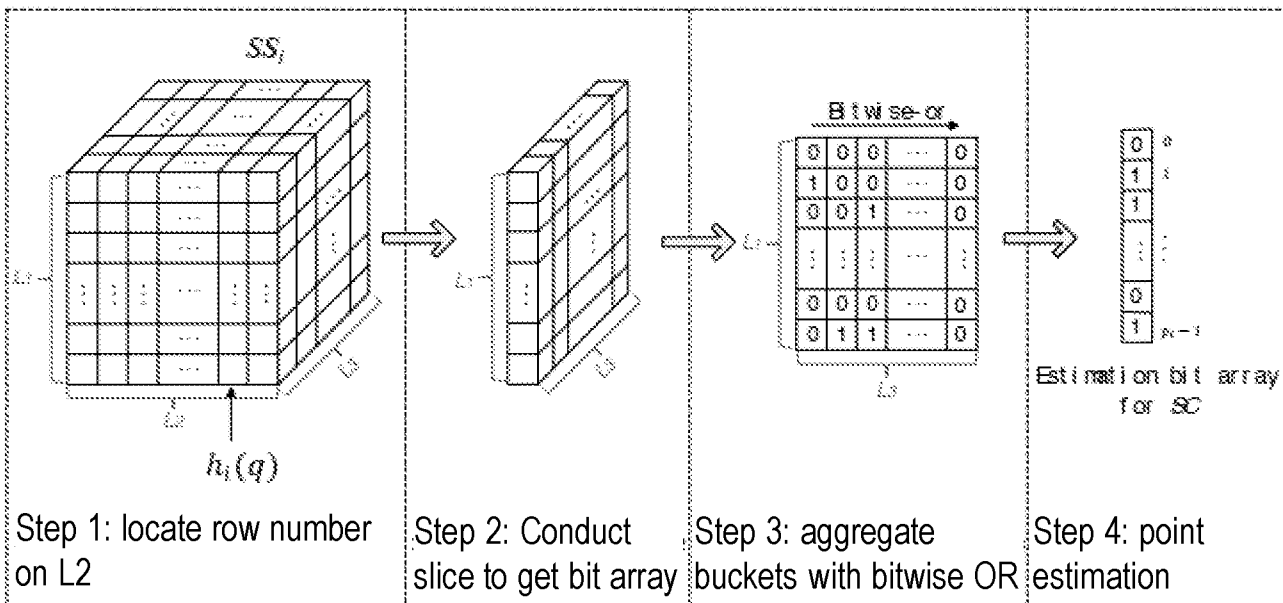


Fig. 8

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2019/084693

A. CLASSIFICATION OF SUBJECT MATTER

G06F 11/30(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F; H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS,CNXTX,CNKI,VEN,3GPP,IEEE:host, cardinality, reversible, abnormal+, destination, source, port

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Yang Liu et al. "Identifying High-Cardinality Hosts from Network-Wide Traffic Measurements" <i>IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING</i> , Vol. 13, No. 5, 16 April 2015 (2015-04-16), sections 2 and 4, abstract	1-26
A	US 8406132 B2 (CAO JINet al.) 26 March 2013 (2013-03-26) the whole document	1-26
A	US 2017250953 A1 (MICROSOFT TECHNOLOGY LICENSING LLC) 31 August 2017 (2017-08-31) the whole document	1-26
A	KR 20180065678 A (LG DISPLAY CO LTD) 18 June 2018 (2018-06-18) the whole document	1-26

☐ Further documents are listed in the continuation of Box C.☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

02 January 2020

Date of mailing of the international search report

16 January 2020

Name and mailing address of the ISA/CN

National Intellectual Property Administration, PRC
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088
China

Authorized officer

LI,Fan

Facsimile No. (86-10)62019451

Telephone No. 86-(010)-62089572

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2019/084693

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	8406132	B2	26 March 2013	US	2009296594	A1	03 December 2009
US	2017250953	A1	31 August 2017	CN	108701187	A	23 October 2018
				WO	2017146961	A1	31 August 2017
				EP	3420487	A1	02 January 2019
KR	20180065678	A	18 June 2018	None			