

(51) International Patent Classification:
H04L 12/803 (2013.01)(21) International Application Number:
PCT/CN2014/095404(22) International Filing Date:
29 December 2014 (29.12.2014)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **NOKIA TECHNOLOGIES OY** [FI/FI];
Karaportti 3, FI-02610 Espoo (FI).(71) Applicant (*for LC only*): **NAVTEQ (SHANGHAI)
TRADING CO., LTD.** [CN/CN]; Room 2930, 2933 and
2942, North Tower, Kerry Center, No.1515 Nanjing Road
West, Jing'an District, Shanghai 200040 (CN).(72) Inventor: **LI, Desheng**; Room 303, Liye Building Block
E, University Science Park of Taike Park, No. 20 Qingyuan
Road, Wuxi New District, Wuxi, Jiangsu 214135 (CN).(74) Agent: **KING & WOOD MALLESONS**; 20th Floor, East
Tower, World Financial, Centre, No.1 Dongsanhuan Zhong-
glu, 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, DK, DM,
DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR,
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, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ,
GW, KM, ML, MR, NE, SN, TD, TG).**Declarations under Rule 4.17:**— *of inventorship (Rule 4.17(iv))***Published:**— *with international search report (Art. 21(3))*

(54) Title: METHOD AND APPARATUS FOR SERVER LOAD BALANCING

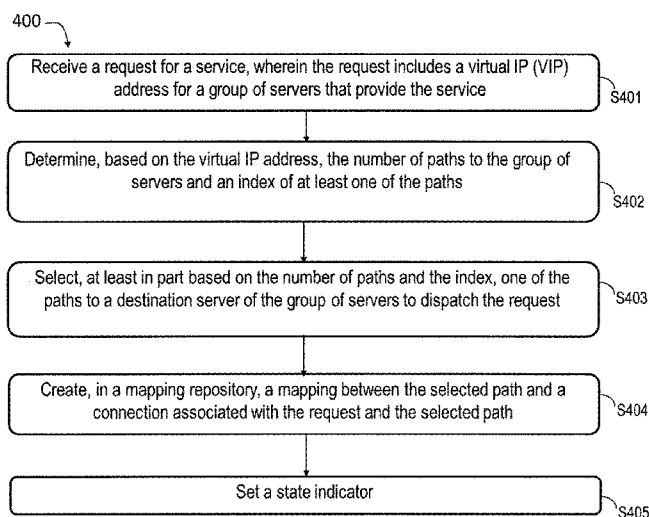


FIG. 4

(57) **Abstract:** The subject matter described herein relates to method and apparatus for server loading balancing. The method comprises receiving, at the IP gateway, a request for a service, the request including a virtual IP address of a group of servers that provides the service; determining, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths; and selecting, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers to dispatch the request. By integrating the server load balancing functionality directly and seamlessly in an IP gateway in addition to its normal functionality, requirement on extra load balancing appliance is avoided.



METHOD AND APPARATUS FOR SERVER LOAD BALANCING

TECHNICAL FIELD

5 [0001] The non-limiting and exemplary embodiments of the present disclosure generally relate to communication networks, and specifically to a method and apparatus for server load balancing in communication networks.

BACKGROUND

10 [0002] This section introduces aspects that may facilitate a better understanding of the disclosure. Accordingly, the statements of this section are to be read in this light and are not to be understood as admissions about what is in the prior art or what is not in the prior art.

15 [0003] Server load balancing (referred to as SLB) is a network technology widely used to distribute client requests for a service to back-end servers which are hosting the same service. A SLB appliance sits between the clients and the back-end servers and operates as a front-end of the service by provisioning a virtual internet protocol (VIP) address which can be addressed from the clients. The clients access the service through the VIP. The SLB appliance distributes the client requests to back-end servers according to load
20 balancing decision. Among a few of load balancing methods, there is a Layer 2 (L2) Direct Server Return (DSR) method (also called direct routing, n-path), which works as illustrated in FIG. 1.

25 [0004] As shown in FIG. 1, the SLB appliance and the servers need to be on the same L2 network segment. A VIP through which the service is accessed will be configured on both the SLB appliance and the servers. The SLB appliance is responsible for answering the Address Resolution Protocol (ARP) request for the VIP. The servers should bind the service to the VIP and not respond or advertise the ARP for the VIP. The client requests for the VIP will be directed to the SLB appliance first, and the SLB distributes the requests to the back-end servers by changing the destination Media Access Control (MAC) address

to the MAC addresses of the back-end servers accordingly. The client requests pertaining to a connection identified by the combination of a source IP (SIP) address, a destination IP (DIP) address, a protocol, a source protocol number and a destination protocol number are always delivered to the same server. The response from the back-end server will be sent to the client directly and bypass the SLB appliance completely. In addition to the VIP, each server can be configured with a real IP through which it can be addressed directly.

SUMMARY

[0005] The L2 DSR SLB method as introduced above, as well as other existing load balancing methods, has to insert an additional appliance (i.e., the SLB appliance) to existing network infrastructure to achieve the load balancing functionality. Since the client requests for the VIP have to be directed to the SLB appliance first, it increases the transmitting time and adds extra processing delay, as well as new potential points of failure. Moreover, the performance/throughput of a normal SLB appliance is usually lower comparing with a network device, such as a 10G or 40G Ethernet switch, and thus it may become the bottleneck of the system performance. On the other hand, introducing a powerful SLB appliance maybe too costive comparing to the Ethernet switches at the same scale, and thus is undesirable.

[0006] In the patent US7739398, a method of accelerating the operation of a triangulation load balancer which operates similarly as the Layer 2 DSR SLB method described above is proposed, wherein an accelerator switch is introduced to reduce the delay introduced by the SLB appliance. However, the method still requires a separate SLB appliance for load balancing purpose and the SLB is still the bottleneck before the accelerator switch learns the load balancing policy.

[0007] To diminish or eliminate at least one of the above-mentioned problems, various aspects of the present disclosure provide a method, and apparatuses for enabling server load balancing with low complexity and cost.

[0008] According to the first aspect of the present disclosure, there is provided a

method of server loading balancing on an internet protocol (IP) gateway, the method comprises receiving, at the IP gateway, a request for a service, the request including a virtual IP address of a group of servers that provide the service; determining, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths; and selecting, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers to dispatch the request.

[0009] In one embodiment, each of the paths can be defined based on a real IP address associated with each of the group of servers.

[0010] In another embodiment, the determining, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths can comprise determining the number of paths to the group of servers further based on a weight associated with each of the group of servers.

[0011] In still another embodiment, the number of paths to the group of servers and the index of at least one of the paths can be kept in an entry of a longest prefix match (LPM) table; the index of at least one of the paths points to one of multiple consecutive entries in an equal-cost multipath (ECMP) table, each of the multiple consecutive entries pointing to an entry in a Next-Hop table, and the number of the multiple consecutive entries in the ECMP table equals to the number of paths, and the selecting, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers may comprise selecting one of the paths to a destination server of the group of servers based on the a hash value in conjunction with the LPM table, the ECMP table and the Next-Hop table.

[0012] In one embodiment, the method may further comprise creating, in a mapping repository, a mapping between the selected path and a connection associated with the request and the selected path; and wherein the selecting, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of server may comprise selecting one of the paths to the destination server of the group of servers further based on the created mapping.

[0013] In another embodiment, the selecting one of the paths to a destination server of

the group of servers further based on the stored mapping may comprise selecting one of the paths to the destination server of the group of servers further based on the created mapping only when the mapping repository is indicated as available by a state indicator. In another embodiment, the selecting one of the paths to a destination server of the group of servers further based on the stored mapping may further comprise if the state indicator indicating a transition state:

[0014] - selecting one of the paths to the destination server of the group of servers without considering the created mapping, and,

[0015] - creating or updating a mapping in the mapping repository based on the selected path.

[0016] In another embodiment, the method may further comprise setting the state indicator, the setting operation comprises: setting the state indicator to a first state indicating unavailability of the mapping repository initially; reconfiguring the state indicator from the first state to the second state indicating a transition state, and starting a timer, if a server is to be added to or removed from the group of servers; reconfiguring the state indicator from the second state back to the first state if the addition or removal of the server is withdrawn before the timer expires, or if no mapping is created in the mapping repository when the timer expires; and reconfiguring the state indicator from the second state to the third state indicating available of the mapping repository, otherwise; removing a mapping from the mapping repository if the mapping is not used for a specific time period; and releasing the mapping repository and reconfiguring the state indicator from the third state to the first state, when all mappings are removed from the mapping repository.

[0017] In one embodiment, the IP gateway can be one of a router and a Layer 3 switch.

[0018] According to the second aspect of the present disclosure, there is provided an apparatus in an IP gateway for performing server loading balancing, the apparatus may comprise a receiver, configured to receive, at the IP gateway, a request for a service, the request including a virtual IP address of a group of servers that provides the service; a controller, configured to determine, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths; and a selector, configured to

select, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers to dispatch the request.

[0019] In one embodiment, each of the paths can be defined based on a real IP address associated with each of the group of servers.

5 [0020] In one embodiment, the controller can be configured to determine the number of paths to the group of servers further based on a weight associated with each of the group of servers.

10 [0021] In another embodiment, the number of paths to the group of servers and the index of at least one of the paths can be kept in an entry of a longest prefix match (LPM) table; the index of at least one of the paths points to one of multiple consecutive entries in an equal-cost multipath (ECMP) table, each of the multiple consecutive entries pointing to an entry in a Next-Hop table, and the number of the multiple consecutive entries in the ECMP table equals to the number of paths, and the selector can be configured to select one of the paths to a destination server of the group of servers based on the a hash value in
15 conjunction with the LPM table, the ECMP table and the Next-Hop table.

[0022] In one embodiment, the apparatus can further comprise a mapping repository, configured to create a mapping between the selected path and a connection associated with the request and the selected path; and wherein the selector can be configured to select one of the paths to the destination server of the group of servers further based on the created
20 mapping. In another embodiment, the selector can be configured to select one of the paths to the destination server of the group of servers further based on the created mapping only when the mapping repository is indicated as available by a state indicator. In still another embodiment, the selector can be further configured to select one of the paths to the destination server of the group of servers without considering the created mapping,
25 and, create or update a mapping between the selected path and a connection associated with the request and the selected path in the mapping repository, if the state indicator indicating a transition state.

[0023] In one embodiment, the apparatus may further comprise a state controller, configured to: set the state indicator to a first state indicating unavailability of the mapping

repository initially; reconfigure the state indicator from the first state to the second state indicating a transition state, and start a timer, if a server is to be added to or removed from the group of servers; reconfigure the state indicator from the second state back to the first state if the addition or removal of the server is withdrawn before the timer expires, or if no
5 mapping is created in the mapping repository when the timer expires; and reconfigure the state indicator from the second state to the third state indicating available of the mapping repository, otherwise; remove a created mapping from the mapping repository if the mapping is not used for a specific time period; and release the mapping repository and reconfigure the state indicator from the third state to the first state, when all mappings are
10 removed from the mapping repository.

[0024] In another embodiment, the IP gateway can be one of a router and a Layer 3 switch.

[0025] According to the third aspect of the present disclosure, there is provided an IP gateway, comprising any of the apparatus according to the second aspect of the disclosure.

15 **[0026]** According to the fourth aspect of the present disclosure, there is provided an apparatus in an IP gateway, the apparatus may comprise a processor and a memory, said memory containing instructions executable by said processor whereby said apparatus is operative to perform any method according to the first aspect of the disclosure.

20 **[0027]** According to the fifth aspect of the present disclosure, there is provided an apparatus in an IP gateway, the apparatus may comprise processing means adapted to perform the method according to the first aspect of the disclosure.

25 **[0028]** In accordance with some embodiments of the subject matter described herein, the problem can be alleviated by integrating the server load balancing functionality directly and seamlessly in an IP gateway in addition to its normal functionality. Hence requirement on extra load balancing appliance is avoided. The IP gateway can be a router or Layer 3 (L3) switch. In either case the methods can be implemented with low complexity and cost by reusing the information and mechanism currently available on the IP gateway.

[0029] By linking the load balancing functionality to existing IP routing and ECMP processing pipeline seamlessly, the proposed method reduces the implementation complexity significantly and avoids introduction of separate load balancing appliances. In addition, in some embodiments, a further enhancement with ephemeral connection mapping table and corresponding mechanism is proposed. It ensures the client requests pertaining to a connection be delivered to the same server even when new back-end servers are added to or removed from the group of servers.

[0030] This Summary is provided to introduce a selection of concepts in a simplified form. The concepts are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matters, nor is it intended to be used to limit the scope of the claimed subject matters.

BRIEF DESCRIPTION OF THE DRAWINGS

[0031] Embodiments of the subject matter described herein are illustrated by way of example and not limited in the accompanying figures in which like reference numerals indicate similar elements and in which:

[0032] FIG. 1 illustrates a schematic diagram of a server load balancing solution in prior art ;

[0033] FIG. 2 illustrates a block diagram of an environment in which embodiments of the subject matter described herein may be implemented;

[0034] FIG. 3 illustrates a schematic diagram of the relationship between a Longest Prefix Match (LPM) table, an Equal-cost multipath (ECMP) table and a Next_Hop table used for routing;

[0035] FIG. 4 illustrates a flowchart of a method for server load balancing in an IP gateway in accordance with one embodiment of the subject matter described herein;

[0036] FIG. 5 illustrates a schematic diagram of selecting a server for load balancing in

accordance with one embodiment of the disclosure;

[0037] FIG. 6 illustrates a schematic diagram of selecting a server for weighted load balancing in accordance with one embodiment of the disclosure;

[0038] FIG. 7 illustrates a schematic diagram of selecting a server in case of server removal in accordance with one embodiment of the disclosure;

[0039] FIG. 8 illustrates a schematic diagram of selecting a server based on a mapping repository in accordance with one embodiment of the disclosure;

[0040] FIG. 9 illustrates a schematic flow chart of selecting a server based on a state indicator in accordance with one embodiment of the disclosure;

[0041] FIG. 10 illustrates a schematic state transition diagram in accordance with one embodiment of the disclosure; and

[0042] FIG. 11 illustrates a block diagram of an apparatus in an IP gateway for load balancing in accordance with one embodiment of the disclosure.

DETAILED DESCRIPTION

[0043] The present disclosure will now be described more fully hereinafter with reference to the accompanying drawings, in which certain embodiments of the present disclosure are shown. This disclosure may, however, be embodied in many different forms and should not be construed as limited to the embodiments set forth herein; rather, these embodiments are provided by way of example so that this disclosure will be thorough and complete, and will fully convey the scope of the present disclosure to those skilled in the art. Like numbers refer to like elements throughout the specification.

[0044] Generally, all terms used in the claims are to be interpreted according to their ordinary meaning in the technical field, unless explicitly defined otherwise herein. All references to “a/an/the element, apparatus, component, means, or step” are to be interpreted openly as referring to at least one instance of the element, apparatus, component, means, or step unless explicitly stated otherwise. The steps of any method

disclosed herein do not have to be performed in the exact order disclosed, unless explicitly stated. The discussion above and below in respect of any of the aspects of the present disclosure is also in applicable parts relevant to any other aspect of the present disclosure.

[0045] As used herein, the term “includes” and its variants are to be read as open terms that mean “includes, but is not limited to.” The term “based on” is to be read as “based at least in part on.” The term “one embodiment” and “an embodiment” are to be read as “at least one embodiment.” The term “another embodiment” is to be read as “at least one other embodiment.” Other definitions, explicit and implicit, may be included below.

[0046] The following will discuss the details of the example embodiments of the present disclosure with reference to the accompanying drawings.

[0047] FIG. 2 shows an environment of a communication network 200 in which embodiments of the subject matter described herein may be implemented. In this example, an IP gateway 201 serves as a next hop router/gateway for groups of servers. These servers can either directly connect to the SLB GW or through a switch. Though for illustration purpose, only two servers 211 and 212 are shown, it is to be understood that there can be any suitable number of servers to be served by the IP gateway.

[0048] In case the IP gateway 201 operates as a routing device, a Layer 3 (L3) Longest Prefix Match (LPM) table can be used to determine how to route packets. A LPM searching algorithm is used to determine the longest subnet match for the given destination IP address of the packet to be routed. A matched entry indicates the next hop information the routing logic can use to construct the L2 header for the routing packet and then transmit it out from appropriate port. The Next hop information can be stored in a Next_Hop table. The entry of Next_Hop table can also be referred as next_hop. The next_hop may usually include a destination MAC address of the next hop, an egress port, etc. Usually, for each endpoint on the subnet with which the router has an interface, a next_hop can be constructed based on the information retrieved through ARP interaction.

[0049] In the IP gateway 201, a technique called Equal-cost multipath (ECMP) can also be used for routing packets along multiple paths of equal cost. If multiple equal-cost routes to the same destination (for example, the server 211) exist, ECMP can be used to

provide load balancing among the redundant paths. The LPM table used in conjunction with the ECMP table allows a packet to be forwarded along one of the equal-cost paths, based on a hashing function of a combination of a source IP address, a destination IP address, a Protocol, a Layer 4 (L4) port number in the packet header and other vendor specified factors. If a LPM lookup results in a hit, and the ECMP is indicated as enabled by for example the ECMP-FLAG field in the matched entry, the associated ECMP_PTR and ECMP_COUNT fields are used to index the ECMP table to determine which next_hop should be used. If the ECMP-FLAG bit is not set in the matched LPM entry, the L3 routing proceeds in the usual way, that is, routes without ECMP. FIG. 3 presents the relationship between the LPM, ECMP and Next_Hop tables used for routing in an example system implementation.

[0050] In a communication network as shown in FIG. 2, there can be multiple servers (for example, the servers 211 and 212) hosting same service requested by a client. To distribute client requests for the same service to the multiple servers properly, some server load balancing (SLB) method is required. Usually, this can be implemented by introducing a SLB appliance sitting between the clients and the back-end servers, and the SLB appliance can serve as the front-end of the service by provisioning a virtual IP (VIP) address of the servers through which the clients access the service. The SLB appliance distributes the client requests to the multiple servers according to certain server load balancing method. One example for such implementation has been shown in FIG. 1.

[0051] As introduced above, the SLB implementation as shown in FIG. 1 requires inserting an additional appliance (i.e., a SLB appliance) to existing network infrastructure to achieve the server load balancing functionality. Since the client requests to the VIP have to be directed to the SLB appliance first, it increases the transmitting time and adds extra processing delay, as well as new potential points of failure. Moreover, the SLB appliance may become the bottleneck of the system performance, or it may be too costly.

[0052] To diminish or eliminate at least one of the above-mentioned problems, various aspects of the present disclosure provide a method, and apparatuses for enabling server load balancing with low complexity and cost.

[0053] In accordance with an embodiment of the invention, server loading balancing functionality can be integrated directly and seamlessly into an IP gateway (for example the IP gateway 201 shown in FIG. 2) in a communication network. Hence requirement on extra server load balancing appliance is avoided. The IP gateway can be a router or Layer 3 (L3) switch. In either case the server load balancing methods can be implemented with low complexity and cost by reusing the information and mechanism currently available on the IP gateway, for example, the longest prefix match (LPM) table and the equal-cost multipath (ECMP) table currently used for routing purpose. For example, in accordance with one embodiment, the server loading balancing methods can be implemented on an Ethernet switch which uses normal commercial Ethernet ASIC. The IP gateway with server load balancing functionality integrated can be referred as a SLB GW hereafter.

[0054] Reference is now made to FIG. 4 which illustrates a flow chart of an example method 400 for server load balancing in a communication network in accordance with one embodiment of the subject matter described herein. In one embodiment, the method 400 may be implemented in an IP gateway, e.g., the IP gateway 201 as shown in FIG. 2. The IP gateway can be a router or a L3 switch, and as mentioned, after integrating the SLB functionality besides its normal functions, the IP gateway can be called a SLB GW. It is to be understood that the subject matter described herein is not limited in this regard. In alternative embodiments, the method 400 may be implemented by any other suitable entities in the communication network.

[0055] As shown, in one embodiment, the method 400 is entered at step S401, where the SLB GW receives a request for a service, wherein the request includes a virtual IP (VIP) address for a group of servers that provide the service; at step S402, the SLB GW determines, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths; and at step S403, the SLB GW selects, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers to dispatch the request received at step S401.

[0056] In one embodiment, for each group of servers, a VIP is used for clients to access

the service hosting by that group of servers. For example, the server 1 and server 2 shown in FIG. 2 can be assigned a VIP of 192.168.1.10. The VIP is included in the client request for a service. The SLB GW is responsible for answering and advertising the Address Resolution Protocol (ARP) for those VIPs.

5 [0057] In another embodiment, the SLB GW also maintains a real IP for each of the group of servers. For example, the SLB GW may create an entry in a group table for each group of servers which the SLB GW serves, wherein each entry consists of the VIP for that group and a real IP for each server of that group. However, it is to be understood that embodiments of the disclosure are not limited to such an implementation, that is, the
10 SLB can maintain the information of VIP and real IPs for the group of servers in any suitable form rather than the group table. In one embodiment, a path to each of the group of servers can be defined based on a corresponding real IP address. For example, a path to the server 1 shown in FIG. 2 can be determined based on the real IP 192.168.1.11, and another path to the server 2 shown in FIG. 2 can be determined based on the real IP
15 address 192.168.1.12. In one embodiment, the path can be determined through ARP interactions, that is, the path can be determined by deriving the Media Access Control (MAC) addresses for the groups of servers based on their real IP addresses. In one embodiment, the SLB GW also keeps the information of the path (e.g., the MAC address) in a Next-Hop table as a next-hop entry. However, it can be appreciated by those skilled
20 in the art that the path can also be defined in other way, for example, based on any suitable address existing currently or to be developed in the future through which a server can be accessed.

[0058] In one embodiment, a weight is assigned to each of the group of servers for administrative purpose, and in this embodiment, at step S402, the SLB GW can determine
25 the number of paths to the group of servers further based on the weight associated with each of the group of servers. For example, different weight can be specified for each of the servers depending on their computing capacity. In one embodiment, a server with larger computing capacity than other servers can be assigned a larger weight, and at step S402, the SLB GW can determine a larger number of paths for this server than for other
30 servers. The determined number of paths to the group of servers at step S402 can be a

sum of the weights specified for all the servers within the group.

[0059] In some embodiments, the number of paths to the group of servers and the index of at least one of the paths determined at step S402 can be kept in an entry of a longest prefix match (LPM) table, for example, they can be written into the ECMP_COUNT and ECMP_PTR fields of the LPM entry, respectively. The index of at least one of the paths (for example, the ECMP_PTR field) points to one of multiple consecutive entries in an equal-cost multipath (ECMP) table, each of the multiple consecutive entries pointing to an entry in a Next-Hop table, and the number of the multiple consecutive entries in the ECMP table equals to the number of paths (i.e., the value of the ECMP_COUNT field).

For example, assuming the server 1 with the real IP of 192.168.1.11 and the server 2 with the real IP of 192.168.1.12 shown in FIG. 2 are the group of servers under consideration, and they both are assigned a weight of "1". At step S402, the SLB can determine the number of paths to the group of servers as 2, and then two consecutive entries can be allocated in an ECMP table, with each entry directing to a next-hop entry in a Next-Hop table. Then the SLB GW can create an entry in a LPM table for the prefix 192.168.1.10/255.255.255.255 by setting the field ECMP_FLAG to "1", field ECMP_COUNT to "2" and setting the field ECMP_PTR to point to one of the two consecutive entries allocated in the ECMP table. Preferably, the ECMP_PTR can be set to point to the first entry of the two consecutive entries allocated in the ECMP table. In such an embodiment, at step S403, the SLB GW can select one of the paths to a destination server of the group of servers based on a hash value in conjunction with the LPM table, the ECMP table and the Next-Hop table.

[0060] In FIG 5, an example is shown to illustrate how the selection operation at step S403 is performed. As shown, when client request packets with the destination IP (DIP) set as the VIP 192.168.1.10 reach the SLB GW, the routing logic of the SLB GW can use the destination IP 192.168.1.10 to search the LPM table. The entry for 192.168.1.10/32 could be matched. The hash value, together with the ECMP_COUNT and the ECMP_PTR fields, determines one of the two next_hops to be used, and then the routing logic of the SLB GW can replace the destination MAC address in the request packets with the one in the selected next_hop and sends out the packets to the corresponding server.

In the example shown in FIG. 5, the hash value passes through a module logic associated with the ECMP_COUNT field, and then results in an offset value. The offset value together with the index indicated by the ECMP-PTR direct to an entry in the ECMP table, which further pointing to a next-hop entry, i.e., a MAC address of one of the group of servers. Preferably, the hash value for packets pertaining to a connection should be consistent, so the packets will be always directed to the same server. In one embodiment, the hash function may have a connection identifier as the input parameter. In one embodiment, the connection can be identified by a combination of source IP (SIP) address, destination IP (DIP) address, protocol, source protocol number and destination protocol number. In another embodiment, the hash value could be calculated based on any combination of the SIP address, the protocol type, the source protocol port number and the destination protocol number, while the DIP is not necessary to be included for the hash calculation, since it is not a differentiator for the connection, considering that all the packets matched to the LPM entry have the same destination IP. It is to be understood that, the operations shown in FIG. 5 are just for illustration, and in other embodiments, any suitable operations can be performed to select a proper server from the group of servers for server load balancing based on the number of paths and the index of one of the paths.

[0061] In above example, the SLB performance is expected to be close to Round Robin load balancing algorithm when the request packets are even from different clients. In another embodiment, a weighted-cost multipath method can be used to achieve an effect close to Weighted Round Robin load balancing algorithm. The request packets pertaining to different connections are distributed to the servers in proportion to the weight valued associated. One example is shown in FIG. 6. Assuming the computing capacity of server 2 shown in FIG. 2 is twice that of the server 1 shown in FIG. 2, and then assuming that to balance the requests (received at step S401) to the server 1 and server 2 in portion to their computing capacity, a weight of “2” is allocated to the server 2 and a weight of “1” is allocated to the server 1. In such case, at step S402, the SLB GW can determine the number of paths to be 3. In this example, the ECMP_COUNT field in the LPM entry is set to 3 accordingly. Then in the ECMP table, 3 consecutive entries are created, with two of the ECMP entries pointing to the next-hop entry associated with the

server 2, and one of the ECMP entries pointing to the next-hop entry associated with the server 1. The ECMP_PTR is set to point to the first of the 3 consecutive ECMP entries. By using similar hashing operation and module logic as shown in FIG. 5, the requests will be direct to the 3 ECMP entries with similar probability, however, since two of the ECMP
5 entries direct to the server 2, while only one ECMP entry directs to the server 1, more request will be distributed to the server 2. Thus, a weighted SLB is achieved. It can be appreciated by those skilled in the art that FIG. 6 is presented just for the purpose of illustration and it should not be considered as a limitation to the embodiments of the disclosure. In an embodiment, any suitable algorithm can be used to realize the SLB
10 based on the weight. For example, the weight value may serve as an input parameter for the hash function.

[0062] One key issue of SLB is to ensure that packets pertaining to a connection are not directed to different servers if new servers are added to the SLB group or servers are removed out of the group. In a normal SLB appliance, the persistence is achieved
15 through a session table (connection and session will be used interchangeably herein). For each newly created session, according the load balancing decision, a server is chosen to be used. And the server chosen for the session is stored in the session table. Subsequent packets pertaining to that session will use the same server stored in the session entry. The conventional SLB session table is hard to implement on normal hardware and usually
20 is implemented in software. When implemented in hardware, the capacity is limited since the session table cannot be too big.

[0063] The SLB GW described in previous embodiments only uses ECMP hash function to map a VIP to one of the servers, and thus is memory-less and session-less. Resilient hashing can ensure persistent mapping in case of server removal. One example
25 is shown in FIG. 7. As shown, when a server is removed from the group, persistent mapping for packets is achieved by replacing the next-hop entry associated with the removed server with another valid next_hop entry (i.e., an entry associated with one of the remaining servers in the group) and keeping the ECMP_COUNT in the LPM entry unchanged. Since the hash value of the packets pertaining to the connection is consistent,
30 by making the ECMP_COUNT and ECMP_PTR and the ECMP entries unchanged, the

persistence can be ensured.

[0064] However, the resilient hash as illustrated in FIG. 7 cannot ensure the persistent mapping in case a server is added to the group. Actually, even for several removals, it's not graceful, since the connection mapped to the server about to be removed will be remapped to another server before the connection closed, which will lead to the connection broken. To solve these problems, in one embodiment, a further step S404 is introduced into the method 400, where the SLB GW creates, in a mapping repository, a mapping between the selected path and a connection associated with the request and the selected path, that is, a mapping between the selected server and the connection; and in this embodiment, at step S403, the SLB GW can select one of the paths to the destination server of the group of servers further based on the created mapping. In one embodiment, the mapping repository can be a connection-next_hop mapping table, however, embodiments of the invention are not limited thereto, and in other embodiments, the mapping repository can take any suitable form, such as a database or a file. The connection-next_hop mapping table can be referred as C-N mapping table hereafter. The C-N mapping table can work together with ECMP table to achieve high performance, low memory consumption, large capacity as well as the persistence required for SLB even when a server is added to the group or removed from the group. The C-N mapping table may be used only for a transition period when the ECMP table needs to be changed.

[0065] In one embodiment, a C-N mapping table can be created for each LPM entry. It saves the mapping between a connection and the next_hop entry used to direct the packets pertaining to the connection. As introduced above, normally, a connection can be identified by a combination of a SIP address, a DIP address, protocol, source protocol number and destination protocol number. In an embodiment, the hash value of the connection identifier combination can be used to represent a bundle of connections in the C-N mapping table. In another embodiment, the hash value could be calculated based on any combination of a SIP address, the protocol type, the source protocol port number and the destination protocol number, while the DIP is not necessary to be included for the hash calculation because it is not a differentiator, considering that all the packets matched to the LPM entry have the same destination IP.

[0066] In one embodiment, at step S403, the SLB GW selects one of the paths to the destination server of the group of servers based on the created mapping in the mapping repository only when the mapping repository is indicated as available by a state indicator. For example, when the state indicator indicates unavailability of the mapping repository, the SLB may be performed, for example, just based on the LPM table, the ECMP table, the hash value and the Next-Hop table, as shown in FIGs. 5-7.

[0067] In FIG. 8, an example of selecting the server based on the mapping repository and the state indicator is illustrated. As shown, in this example, the state indicator can be maintained in a field of the LPM entry, the ECMP_ST field, for example. However, it should be noted that embodiments of the disclosure are not limited thereto, and in other embodiments, the state indicator can be stored in another table separate from the LPM table. In this example, there is another field, i.e., the CN_PTR field created in the LPM entry. It points to a C-N mapping table pertaining to the LMP entry. In the embodiment, the CN_PTR is only used when the state indicator (ECMP_ST) indicates that the C-N mapping table is available. In this example, assume that ECMP_ST can indicate one of three predefined states: ECMP, CT (Connection Tracking) and CA (Connection Assistance). The ECMP state indicates that normal SLB based on the ECMP table can be performed, that is, no mapping repository is available. The CA state indicates that SLB based on the mapping repository can be performed, that is the mapping repository is available. The CT state indicates that the mapping repository is unavailable currently, but is to be created or updated based on the server selection results. That is, the working states determine how the routing logic should handle the received request packet when the LPM entry is matched. In the example shown in FIG. 8, the ECMP_ST indicates a CT state, it can be interpreted to be a transition state, which means no mapping is available currently and normal SLB based on ECMP should be performed, however, after the server is selected, the result can be used to create or update a mapping in the mapping repository. It is to be understood that the states of ECMP, CT and CA are just presented for the purpose of illustration, and in other embodiments, different states and/or different number of states can be defined to control the utilization of the mapping repository, therefore, embodiments of the disclosure are not limited thereto.

[0068] Just for the purpose of illustration, in FIG. 9, a flow chart is shown to illustrate the operations performed at step S403 by the SLB GW depending on the state indicator (for example, the ECMP_ST in the LPM table). In this example, assume a matched LPM entry has been found for a received request at step S401. Then the ECMP_ST is checked.

5 If it indicates the ECMP state, the routing logic is exactly the same as usual. If it indicates the ECMP_CT state, the routing logic firstly uses the normal routing and ECMP process to route the request packet. Upon completion of the ECMP processing, an entry is added to the C-N mapping table, based on the connection the packet pertaining to and the next_hop entry associated with the selected server, or, if an mapping entry for the
10 connection already exists in the C-N table, the existing entry is update. Optionally, a hit bit can be set for the mapping entry. If the ECMP_ST indicates the CA state, the routing logic firstly uses the C-N mapping table to determine the next_hop. If it fails to get an entry after searching the connections in the table, the routing process will fall back to the normal ECMP process.

15 [0069] In one embodiment of the disclosure, an aging mechanism can be used to remove a mapping entry from the C-N mapping table when the entry is not used for a specific time period. In one embodiment, a hit bit can be associated with each C-N entry (i.e., each mapping). When the entry is dereferenced or updated, the hit bit is set. If the hit bit is not set for a specific time period, the entry will be removed.

20 In one embodiment, the method 400 further comprises step S405, where the state indicator is managed. That it, the SLB controls the transition between multiple working states. In one embodiment, the state indicator is managed by: setting the state indicator to a first state indicating unavailability of the mapping repository initially; reconfiguring the state indicator from the first state to the second state indicating a transition state, and
25 starting a timer, if a server is to be added to or removed from the group of servers; reconfiguring the state indicator from the second state back to the first state if the addition or removal of the server is withdrawn before the timer expires, or if no mapping is created in the mapping repository when the timer expires; and reconfiguring the state indicator from the second state to the third state indicating available of the mapping repository,
30 otherwise; removing a created mapping from the mapping repository if the mapping is not

used for a specific time period; and releasing the mapping repository and reconfiguring the state indicator from the third state to the first state, when all mappings are removed from the mapping repository.

[0070] In FIG. 10, a schematic state transition diagram is presented which illustrates an example state transition in accordance with an embodiment of the step S405 of the method 400. However, it can be understood that these states and their transitions are just shown for the purpose of illustration, and in other embodiments, any suitable states and/or transitions can be defined. As shown, initially, for example when the LPM entry for a SLB group is firstly created, the working state is set to the state of ECMP. Then when a new server is about to be added to or removed from the group, the SLB GW firstly creates an empty mapping repository, for example allocates an ephemeral mapping table with empty entry, and then sets the working state to CT and makes the CN_PTR field point to the new created mapping table. A CT timer is also started. Before the timer expires, if the addition/removal is withdrawn, then the timer is reset and the work state is set to ECMP. In the CT state, new entry might be added to the mapping table (e.g., the C-N mapping table); when the timer expires, if there is no entry in the table, the state is set to ECMP, otherwise it is set to CA. In one embodiment, the mapping in the mapping repository can be considered as aged and be removed if it is not hit for a specific time period. The CA state remains until last entry in the mapping table is aged and removed. Then the mapping table can be released and the CN_PTR is cleared and the state (e.g., ECMP_ST field) is set to ECMP.

[0071] As introduced above, in some embodiments, the mapping repository can be an ephemeral connection-next-hop mapping table. It ensures the client requests pertaining to connection to be delivered to the same server even when a server is added to or removed from the group and at the same time avoids the requirement of a large session table due to an aging mechanism.

[0072] As introduced above, in an embodiment of the disclosure, the IP gateway where the method 400 is implemented can be one of a router and a Layer 3 switch, but embodiments of the disclosure are not limited thereto. In other embodiments, the method 400 described with reference to FIGs. 4-10 can be implemented in any suitable devices.

[0073] Reference is now made to FIG. 11 which illustrates a block diagram of an apparatus 1100 in an IP GW for server load balancing in accordance with one embodiment of the subject matter described herein. The apparatus 1100 can be implemented in, e.g., the IP GW shown in FIG. 2, but embodiments of the subject matter described herein are not limited thereto. For example, in another embodiment, the apparatus can be implemented in any suitable network entity. The apparatus 1100 may perform the example methods described with reference to FIGs. 4-10, but is not limited to these methods. Then any feature presented above, e.g., the operations involved in the steps described with reference to FIG. 4 can be applied to the apparatus 1100 presented below. It is to be noted that the methods described with reference to FIG. 4 may be performed by the apparatus 1100 but is not limited to being performed by this apparatus 1100.

[0074] As shown, in one embodiment, the apparatus 1100 comprises a receiver 1101, configured to receive, at the IP gateway, a request for a service, the request including a virtual IP address of a group of servers that provide the service; a controller 1102, configured to determine, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths; and a selector 1103, configured to select, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers to dispatch the request received at step 1101.

[0075] In one embodiment, each of the paths can be defined based on a real IP address associated with each of the group of servers. For example, it can be a MAC address derived based on the real IP address through ARP interactions, as described with reference to FIG. 4 and the method 400, and thus details will not be repeated again.

[0076] In one embodiment, the controller 1102 can be configured to determine the number of paths to the group of servers further based on a weight associated with each of the group of servers. As described with reference to FIG. 4, the weight can be assigned to each of the group of servers at least based on the computing capacity, such that the selector 1103 can select a server by taking into account the server's computing capacity.

[0077] In another embodiment, the number of paths to the group of servers and the index of at least one of the paths determined by the controller 1102 can be maintained in

an entry of a longest prefix match (LPM) table; the index of at least one of the paths points to one of multiple consecutive entries in an equal-cost multipath (ECMP) table, each of the multiple consecutive entries pointing to an entry in a Next-Hop table, and the number of the multiple consecutive entries in the ECMP table equals to the number of paths, and the selector can be configured to select one of the paths to a destination server of the group of servers based on the a hash value in conjunction with the LPM table, the ECMP table and the Next-Hop table.

[0078] In one embodiment, the apparatus can further comprise a mapping repository 1104, configured to store a mapping between the selected path and a connection associated with the request and the selected path; and wherein the selector 1103 can be configured to select one of the paths to the destination server of the group of servers further based on the stored mapping. In another embodiment, the selector 1103 can be configured to select one of the paths to the destination server of the group of servers further based on the created mapping only when the mapping repository is indicated as available by a state indicator. In still another embodiment, the selector 1103 can be further configured to select one of the paths to the destination server of the group of servers without considering the created mapping, and, create or update a mapping between the selected path and a connection associated with the request and the selected path in the mapping repository, if the state indicator indicating a transition state. As described with reference to method 400 and FIGs 4-10, in one embodiment, the state indicator can indicate one of three predefined states (for example, ECMP, CT, and CA), and details will not be repeated here.

[0079] In one embodiment, the apparatus may further comprise a state controller 1105, configured to control the transition between states. For example, it may be configured to set the state indicator to a first state (for example an ECMP state) indicating unavailability of the mapping repository initially; reconfigure the state indicator from the first state to the second state (for example, CT) indicating a transition state, and start a timer, if a server is to be added to or removed from the group of servers; reconfigure the state indicator from the second state back to the first state if the addition or removal of the server is withdrawn before the timer expires, or if no mapping is created in the mapping repository when the timer expires; and reconfigure the state indicator from the second state to the third state

(for example, CA) indicating available of the mapping repository, otherwise; remove a created mapping from the mapping repository if the mapping is not used for a specific time period (for example, if the hit bit associated with the mapping is not set for a specific time period); and release the mapping repository and reconfigure the state indicator from the third state to the first state, when all mappings are removed from the mapping repository. An example for the state transition has been shown in FIG. 10.

[0080] In an embodiment, the IP gateway where the apparatus 1100 is embedded can be one of a router and a Layer 3 switch. However, embodiments of the invention are not limited thereto. The apparatus 1100 can be implemented in any suitable network entity.

[0081] As described above, the method 400 and the apparatus 1100 can be used to improve the server load balancing. It enables SLB by reusing at least some of the existing information, functions and modules already available in an existing IP gateway, and thus can achieve the SLB with low complexity and cost.

[0082] It is to be understood that, though in some embodiments of the subject matter described herein, methods and apparatus are described in the context of an IP GW, embodiments of the subject matter described herein are not limited thereto.

[0083] The modules/units included in the apparatuses 1100 may be implemented in various manners, including software, hardware, firmware, or any combination thereof. In one embodiment, one or more units may be implemented using software and/or firmware, for example, machine-executable instructions stored on the storage medium. In addition to or instead of machine-executable instructions, parts or all of the units in the apparatuses 1100 may be implemented, at least in part, by one or more hardware logic components. For example, and without limitation, illustrative types of hardware logic components that can be used include Field-programmable Gate Arrays (FPGAs), Application-specific Integrated Circuits (ASICs), Application-specific Standard Products (ASSPs), System-on-a-chip systems (SOCs), Complex Programmable Logic Devices (CPLDs), and the like.

[0084] Generally, various embodiments of the subject matter described herein may be implemented in hardware or special purpose circuits, software, logic or any combination

thereof. Some aspects may be implemented in hardware, while other aspects may be implemented in firmware or software which may be executed by a controller, microprocessor or other computing device. While various aspects of embodiments of the subject matter described herein are illustrated and described as block diagrams, flowcharts, or using some other pictorial representation, it will be appreciated that the blocks, apparatus, systems, techniques or methods described herein may be implemented in, as non-limiting examples, hardware, software, firmware, special purpose circuits or logic, general purpose hardware or controller or other computing devices, or some combination thereof.

[0085] By way of example, embodiments of the subject matter can be described in the general context of machine-executable instructions, such as those included in program modules, being executed in a device on a target real or virtual processor. Generally, program modules include routines, programs, libraries, objects, classes, components, data structures, or the like that perform particular tasks or implement particular abstract data types. The functionality of the program modules may be combined or split between program modules as desired in various embodiments. Machine-executable instructions for program modules may be executed within a local or distributed device. In a distributed device, program modules may be located in both local and remote storage media.

[0086] Program code for carrying out methods of the subject matter described herein may be written in any combination of one or more programming languages. These program codes may be provided to a processor or controller of a general purpose computer, special purpose computer, or other programmable data processing apparatus, such that the program codes, when executed by the processor or controller, cause the functions/operations specified in the flowcharts and/or block diagrams to be implemented. The program code may execute entirely on a machine, partly on the machine, as a stand-alone software package, partly on the machine and partly on a remote machine or entirely on the remote machine or server.

[0087] In the context of this disclosure, a machine readable medium may be any

tangible medium that may contain, or store a program for use by or in connection with an instruction execution system, apparatus, or device. The machine readable medium may be a machine readable signal medium or a machine readable storage medium. A machine readable medium may include but not limited to an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, or device, or any suitable combination of the foregoing. More specific examples of the machine readable storage medium would include an electrical connection having one or more wires, a portable computer diskette, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), an optical fiber, a portable compact disc read-only memory (CD-ROM), an optical storage device, a magnetic storage device, or any suitable combination of the foregoing.

[0088] Further, while operations are depicted in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Likewise, while several specific implementation details are contained in the above discussions, these should not be construed as limitations on the scope of the subject matter described herein, but rather as descriptions of features that may be specific to particular embodiments. Certain features that are described in the context of separate embodiments may also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment may also be implemented in multiple embodiments separately or in any suitable sub-combination.

[0089] Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described above. Rather, the specific features and acts described above are disclosed as example forms of implementing the claims.

WHAT IS CLAIMED IS:

1. A method of server loading balancing on an internet protocol (IP) gateway, comprising:

receiving, at the IP gateway, a request for a service, the request including a virtual IP address of a group of servers that provides the service;

determining, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths; and

selecting, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers to dispatch the request.

2. The method of claim 1, wherein each of the paths is defined based on a real IP address associated with each of the group of servers.

3. The method of claim 1, wherein the determining, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths comprising:

determining the number of paths to the group of servers further based on a weight associated with each of the group of servers.

4. The method of claim 1, wherein the number of paths to the group of servers and the index of at least one of the paths are kept in an entry of a longest prefix match (LPM) table,

the index of at least one of the paths points to one of multiple consecutive entries in an equal-cost multipath (ECMP) table, each of the multiple consecutive entries pointing to an entry in a Next-Hop table, and the number of the multiple consecutive entries in the ECMP table equals to the number of paths, and

the selecting, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers comprises selecting one of the paths to a destination server of the group of servers based on the a hash value in conjunction with the LPM table, the ECMP table and the Next-Hop table.

5. The method of claim 1, further comprising:

creating, in a mapping repository, a mapping between the selected path and a connection associated with the request and the selected path; and

5 wherein the selecting, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of server comprising:

selecting one of the paths to the destination server of the group of servers further based on the created mapping.

10 6. The method of claim 5, wherein the selecting one of the paths to a destination server of the group of servers further based on the stored mapping comprising:

selecting one of the paths to the destination server of the group of servers further based on the created mapping only when the mapping repository is indicated as available by a state indicator.

15 7. The method of claim 6, wherein the selecting one of the paths to a destination server of the group of servers further based on the stored mapping further comprising:

if the state indicator indicating a transition state

20 - selecting one of the paths to the destination server of the group of servers without considering the created mapping, and,
- creating or updating a mapping in the mapping repository based on the selected path.

8. The method of claim 5, further comprising setting the state indicator by:

25 setting the state indicator to a first state indicating unavailability of the mapping repository initially;

reconfiguring the state indicator from the first state to the second state indicating a transition state, and starting a timer, if a server is to be added to or removed from the group of servers;

30 reconfiguring the state indicator from the second state back to the first state if the addition or removal of the server is withdrawn before the timer expires, or if no mapping

is created in the mapping repository when the timer expires; and reconfiguring the state indicator from the second state to the third state indicating available of the mapping repository, otherwise;

removing a created mapping from the mapping repository if the mapping is not
5 used for a specific time period; and

releasing the mapping repository and reconfiguring the state indicator from the third state to the first state, when all mappings are removed from the mapping repository.

9. The method of any of claims 1-7, wherein the IP gateway is one of a router and
10 a Layer 3 switch.

10. An apparatus in an IP gateway for performing server loading balancing, comprising:

a receiver, configured to receive, at the IP gateway, a request for a service, the
15 request including a virtual IP address of a group of servers that provides the service;

a controller, configured to determine, based on the virtual IP address, the number of paths to the group of servers and an index of at least one of the paths; and

a selector, configured to select, at least in part based on the number of paths and the index, one of the paths to a destination server of the group of servers to dispatch the
20 request.

11. The apparatus of claim 10, wherein each of the paths is defined based on a real IP address associated with each of the group of servers.

25 12. The apparatus of claim 10, wherein the controller is configured to determine the number of paths to the group of servers further based on a weight associated with each of the group of servers.

13. The apparatus of claim 10, wherein
30 the number of paths to the group of servers and the index of at least one of the paths are kept in an entry of a longest prefix match (LPM) table,

the index of at least one of the paths points to one of multiple consecutive entries in an equal-cost multipath (ECMP) table, each of the multiple consecutive entries pointing to an entry in a Next-Hop table, and the number of the multiple consecutive entries in the ECMP table equals to the number of paths, and

5 the selector is configured to select one of the paths to a destination server of the group of servers based on the a hash value in conjunction with the LPM table, the ECMP table and the Next-Hop table.

14. The apparatus of claim 10, further comprising:

10 a mapping repository, configured to create a mapping between the selected path and a connection associated with the request and the selected path; and

wherein the selector is configured to select one of the paths to the destination server of the group of servers further based on the created mapping.

15 15. The apparatus of claim 14, wherein the selector is configured to select one of the paths to the destination server of the group of servers further based on the created mapping only when the mapping repository is indicated as available by a state indicator.

16. The apparatus of claim 14, wherein the selector is further configured to:

20 select one of the paths to the destination server of the group of servers without considering the created mapping, and, create or update a mapping in the mapping repository based on the selected path, if the state indicator indicating a transition state.

17. The apparatus of claim 14, further comprising a state controller, configured to
25 set the state indicator by:

setting the state indicator to a first state indicating unavailability of the mapping repository initially;

reconfiguring the state indicator from the first state to the second state indicating
30 a transition state, and starting a timer, if a server is to be added to or removed from the group of servers;

reconfiguring the state indicator from the second state back to the first state if the addition or removal of the server is withdrawn before the timer expires, or if no mapping is created in the mapping repository when the timer expires; and reconfiguring the state indicator from the second state to the third state indicating available of the mapping repository, otherwise;

removing a created mapping from the mapping repository if the mapping is not used for a specific time period; and

releasing the mapping repository and reconfiguring the state indicator from the third state to the first state, when all mappings are removed from the mapping repository.

18. The apparatus of any of claims 10-17, wherein the IP gateway is one of a router and a Layer 3 switch.

19. An IP gateway, comprising the apparatus according to any of claims 10-17.

20. An apparatus in an IP gateway, comprising a processor and a memory, said memory containing instructions executable by said processor whereby said apparatus is operative to perform the method according to any of Claims 1-9.

21. An apparatus in an IP gateway, comprising processing means adapted to perform the method according to any of Claims 1-9.

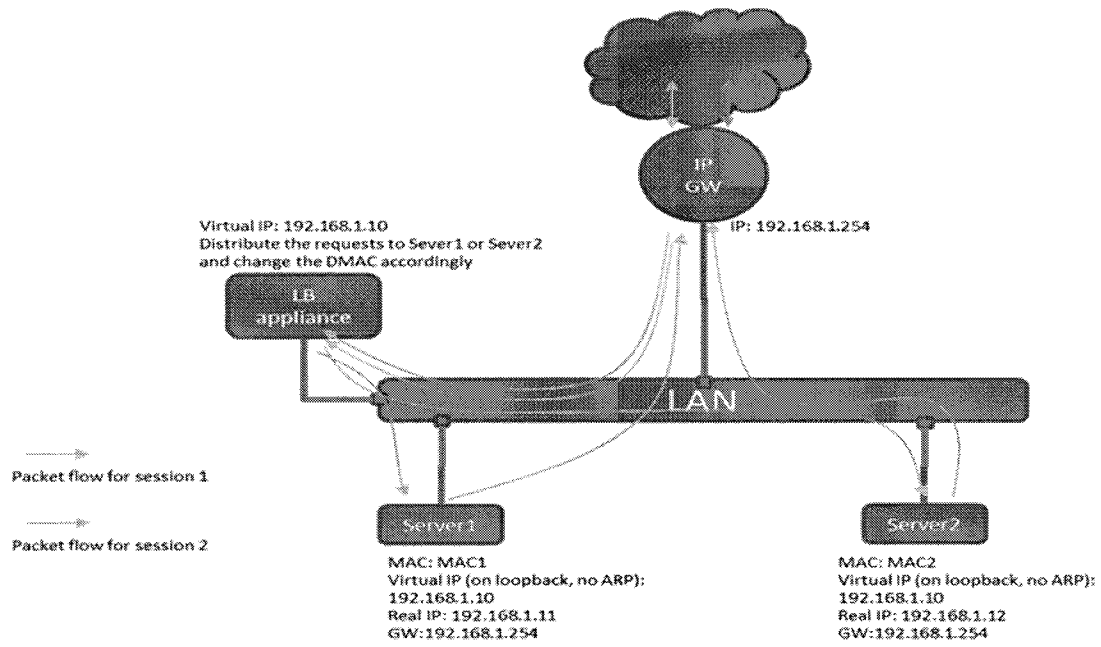


FIG. 1

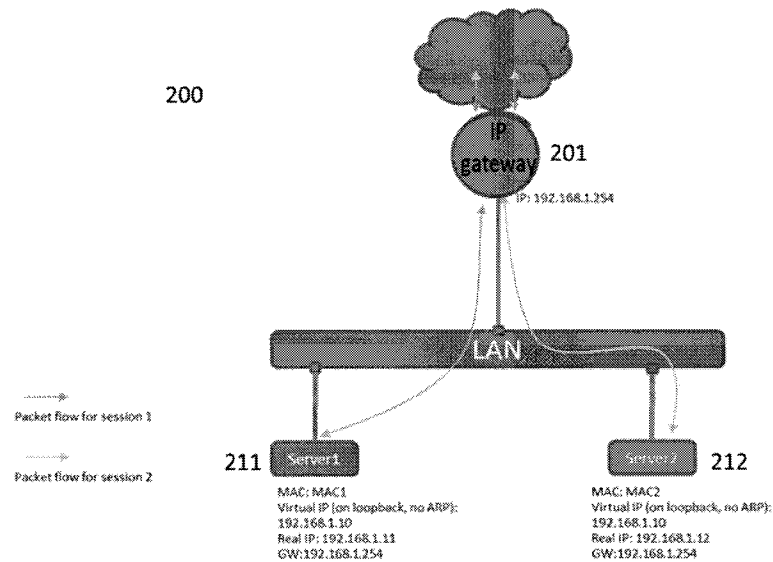


FIG. 2

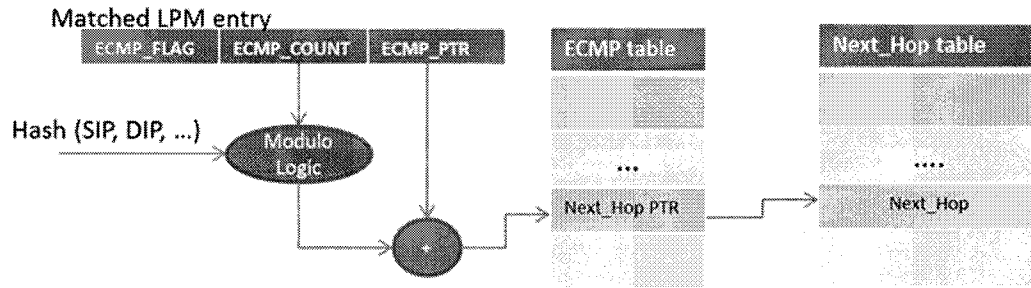


FIG. 3

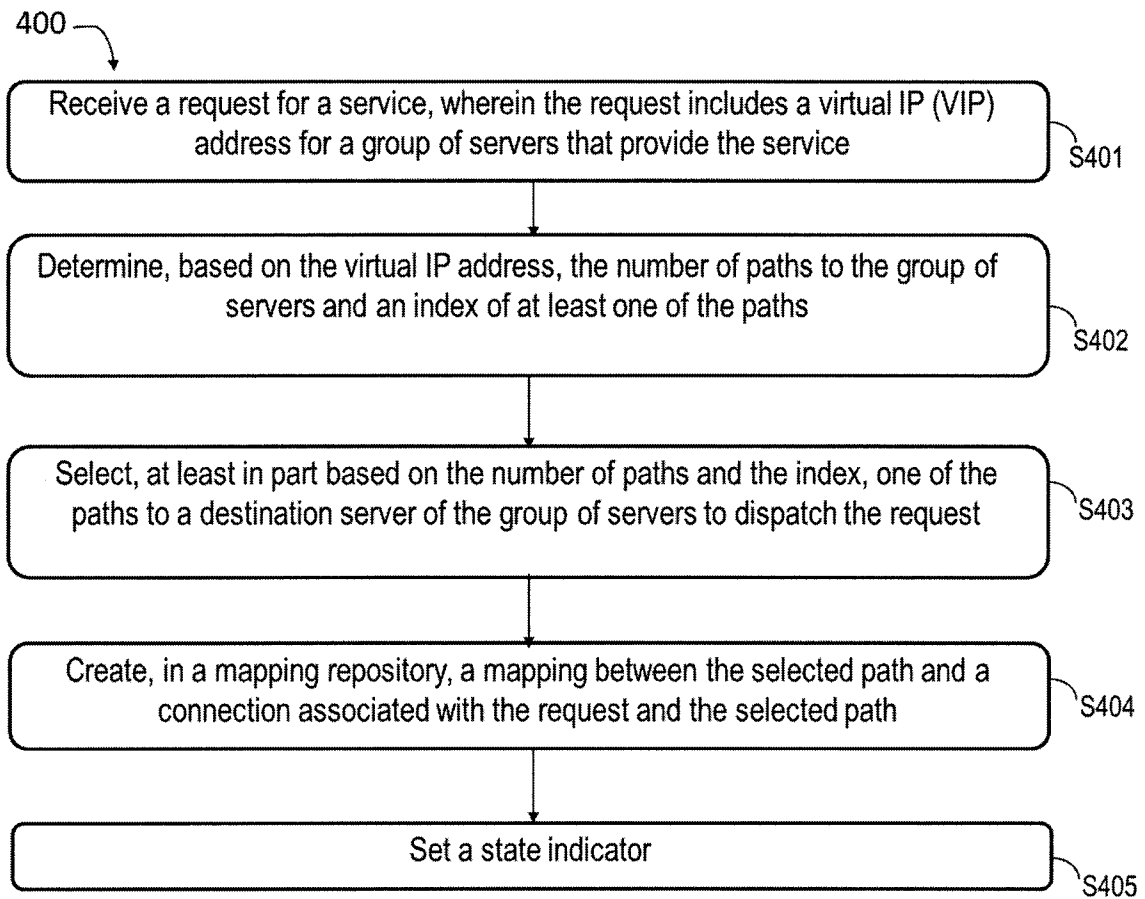


FIG. 4

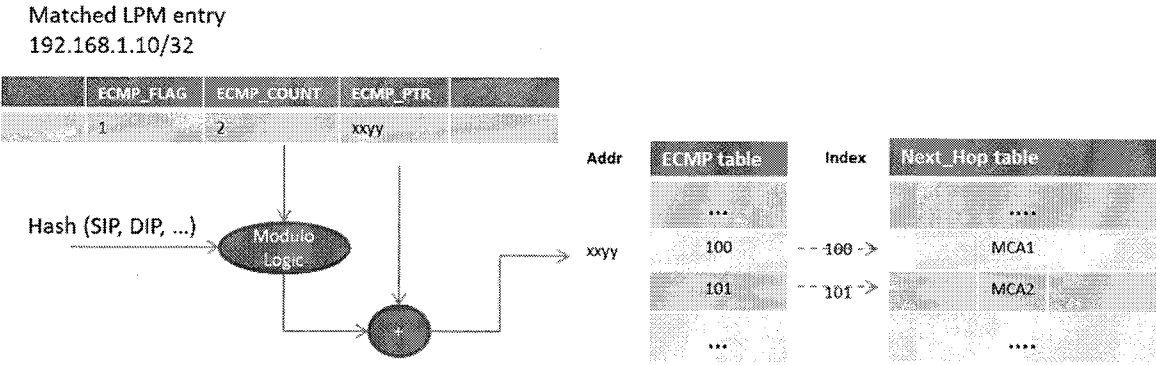


FIG. 5

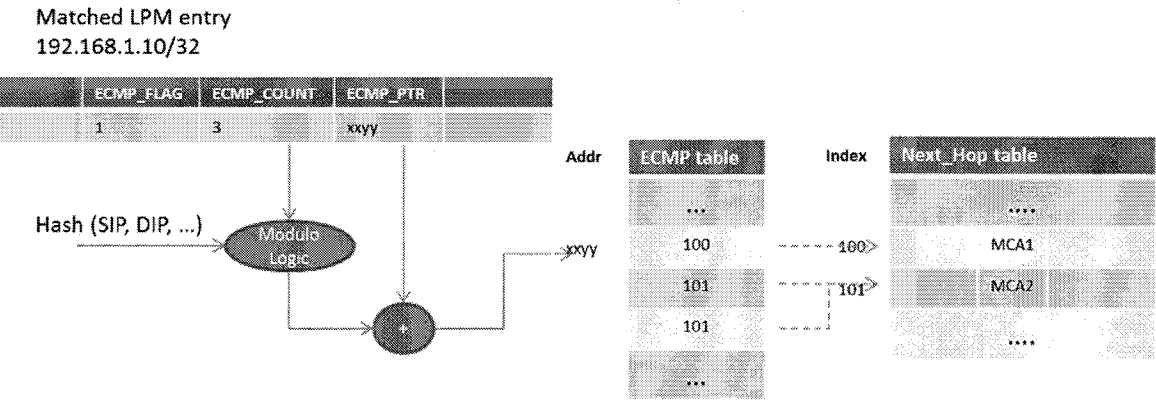


FIG. 6

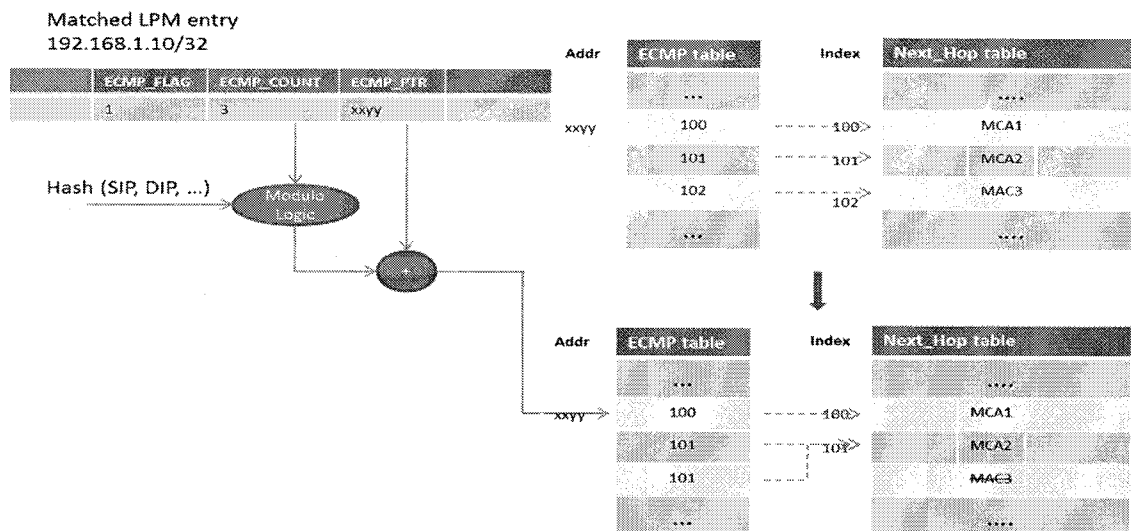


FIG. 7

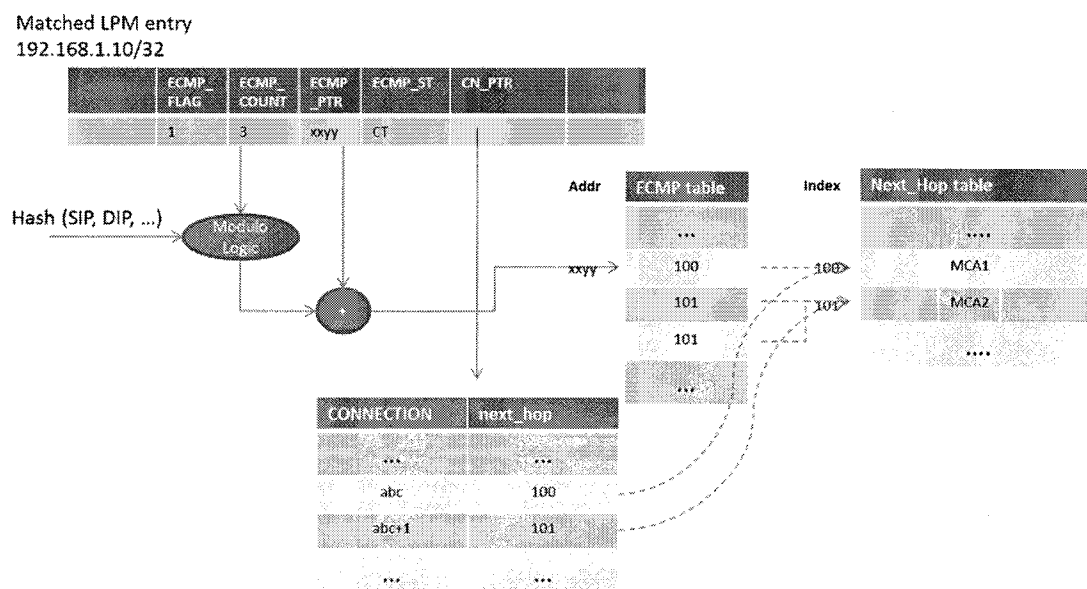


FIG. 8

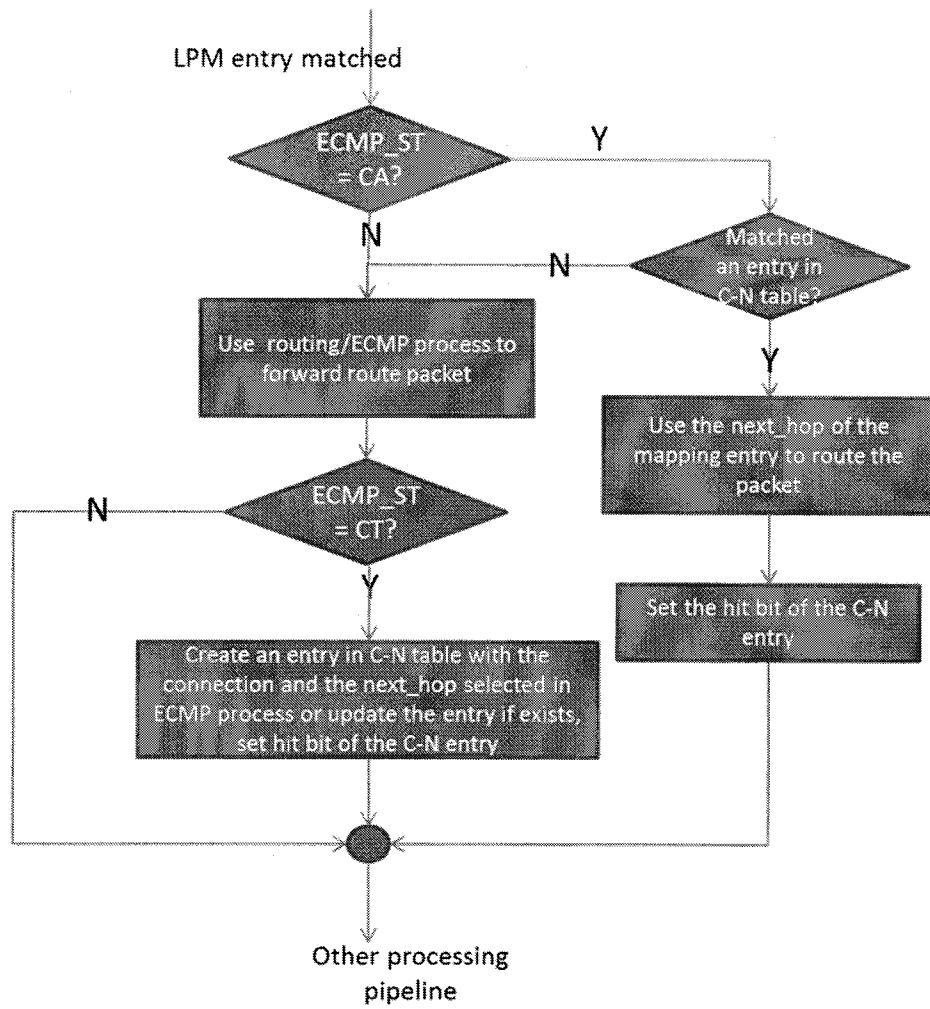


FIG. 9

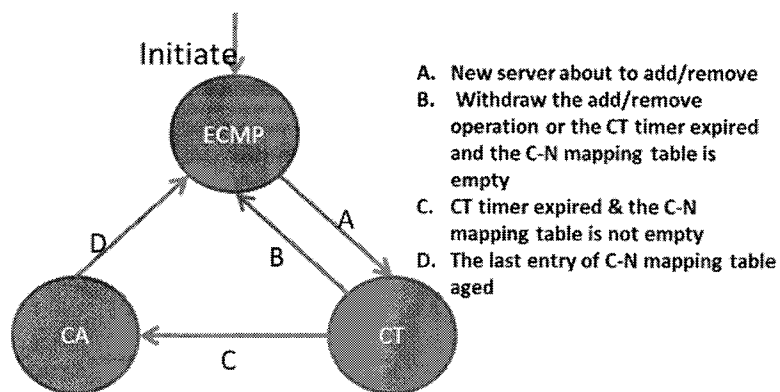


FIG. 10

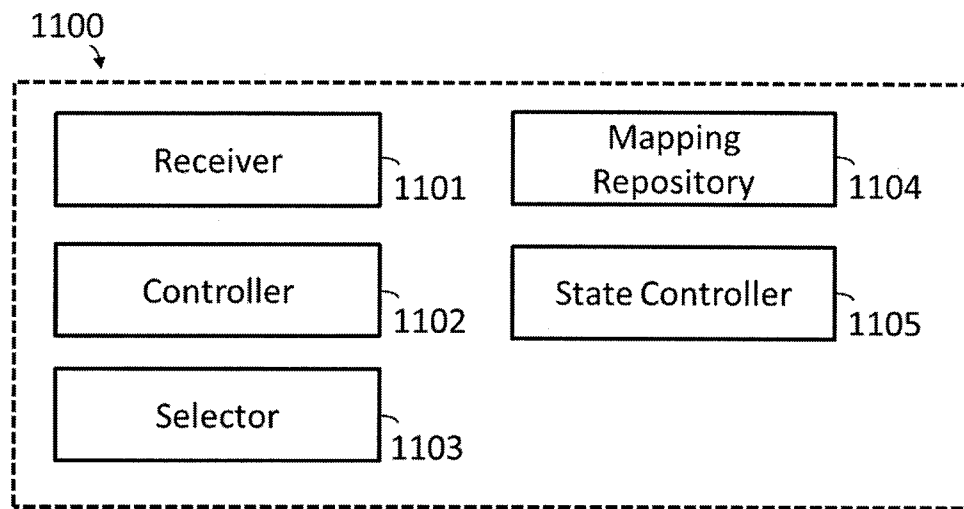


FIG. 11

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2014/095404

A. CLASSIFICATION OF SUBJECT MATTER

H04L 12/803(2013.01)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)

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)

CPRSABS,CNTXT,CNKI,VEN: server?, load w balanc+, gateway, route?, switch+, virtual w address, virtual w IP w address, path?, number

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 101404619 A (HANGZHOU H3C TECHNOLOGIES CO., LTD.) 08 April 2009 (2009-04-08) the description page 5, line 4-page 10, line 15	1-3, 9-12, 18-21
A	CN 101404619 A (HANGZHOU H3C TECHNOLOGIES CO., LTD.) 08 April 2009 (2009-04-08) the whole document	4-8, 13-17
X	CN 104079497 A (BEIJING SINOGRID SOFTWARE SYSTEMS CO., LTD.) 01 October 2014 (2014-10-01) the description paragraphs [0053]-[0102]	1-3, 9-12, 18-21
X	CN 104079497 A (BEIJING SINOGRID SOFTWARE SYSTEMS CO., LTD.) 01 October 2014 (2014-10-01) the description paragraphs [0053]-[0102]	1-3, 9-12, 18-21
X	US 2003005080 A1 (CAPITAL ONE FINANCIAL CORP. ET AL.) 02 January 2003 (2003-01-02) the description paragraphs [0015]-[0028]	1-3, 9-12, 18-21



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

201bete

Date of mailing of the international search report

15 October 2015

Name and mailing address of the ISA/CN

STATE INTELLECTUAL PROPERTY OFFICE OF THE
P.R.CHINA
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088, China

Facsimile No. (86-10)62019451

Authorized officer

HUANG,Haofu

Telephone No. (86-10)62411436

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2014/095404

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2003005080 A1 (CAPITAL ONE FINANCIAL CORP. ET AL.) 02 January 2003 (2003-01-02) the whole document	4-8, 13-17

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2014/095404

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	101404619	A	08 April 2009	CN	101404619	B	08 June 2011
CN	104079497	A	01 October 2014	None			
US	2003005080	A1	02 January 2003	WO	03003233	A1	09 January 2003
				EP	1421507	A1	26 May 2004
				AU	2002318164	A1	03 March 2003