

IDENTIFIER ADDRESSING METHOD FOR HIERARCHICAL STRUCTURE

Related Application

This application claims priority to Chinese Patent Application No. 2020113263621, filed on November 24, 2020, and Chinese Patent Application No. 2021105002713, filed on May 8, 2021, entitled "IDENTIFIER ADDRESSING METHOD FOR HIERARCHICAL STRUCTURE", which are incorporated herein by reference in their entirety.

Technical Field

The present invention relates to the field of network technologies and communication technologies, and in particular, to an identifier addressing method for a hierarchical structure.

Background

With the development of new technologies and applications such as internet of things, mobile access, distributed information processing, and massive streaming media, the performance of current computer networks based on a TCP/IP architecture gradually approaches the limit, and the contradiction between the massive demand for information content services and network bandwidth bottleneck keeps increasing. A network narrow waist problem that occurs due to the structure of IP networks has become an important factor hindering the development of application layers in current networks. An existing internet TCP/IP architecture has many problems in terms of scalability, mobility, multi-homing, and traffic engineering. An IP address now used in the internet has a dual meaning of location information and identity information, which confuses the functional boundary between a locator and an identifier and has the problem of IP address semantic overload. This naming mechanism is also the crux that restricts the development of network technology.

In response to these problems, academia and standards organizations have strengthened their research on new future internet architectures. At the beginning of this century, guided by the main needs of the internet, the information-centric networking (ICN) technology was proposed to build a network architecture with information/content as the center and decouple the relationship between information and location, to increase the information storage capacity of a network and enhance content acquisition, mobility support, and content-oriented security mechanisms from the network level. Typical projects in the ICN series include NDN, DONA,

NetInf, PURSUIT, and the like. The International Internet Engineering Task Force (IETF) started the Locator/ID Separation Protocol (LISP) working group in 2009, proposing the idea of identifier and address separation to solve the problem of routing systems. The European Telecommunications Standards Institute (ETSI) published "NGP 004: Evolved Architecture for mobility using Identity Oriented Networks" in 2018, which supports native mobility based on the idea of name and address separation. A new network architecture based on the idea of identifier and address separation has become a new idea to solve the problems of conventional TCP/IP networks. After location and identifier separation is implemented, how to use the characteristics of uniqueness and persistence of identifiers to design a routing mechanism to meet the needs of future internet development has become a current research hotspot.

Identifier-based routing methods may be categorized into an identifier-based direct routing method and an address-based routing method after identifier-based resolution and addressing. In the identifier-based direct routing method, an identifier of an object is directly mapped into a route, that is, a location and a resolution path of the identifier are fused with a retrieval path of routing information. Typical projects include CCN/NDN. In the address-based routing method after identifier-based resolution and addressing, an identifier is first mapped into an address through a name resolution system, and then an address-based routing method is used. Typical projects include DONA and MoblityFirst.

The first method does not need to rely on resolution services and network addresses, and is based entirely on identifiers for routing and forwarding. However, the method is not more compatible with existing network devices and network architectures, and there is a large number of identifier routing tables, which causes scalability and route convergence efficiency problems. The second method can be more compatible with existing network devices and network architectures, and the underlying mechanism is still based on address routing. However, the method needs to rely on an external resolution and mapping system, and the security and efficiency of the system affects a routing process.

Summary

In view of existing problems of an existing identifier routing algorithm, the present invention provides an identifier addressing method for a hierarchical structure. Network domain

layering is performed based on a management domain and a control domain. A controller stores a mapping between an identifier and an address of an entity in the control domain. A border router of the management domain stores a matching relationship between an identifier of an entity in the management domain and the control domain and an association relationship between an identifier and an associated border router, to implement hierarchical storage. In addition, dynamic resolution and addressing of an identifier in a message transmission process are implemented in a manner of intra-domain and inter-domain hierarchical resolution. This effectively resolves scalability and search efficiency problems caused by a large number of identifier-address mapping tables.

In view of the foregoing problems, the present invention provides an identifier addressing method for a hierarchical structure, wherein an architecture in which an identifier and an address are separate, each network entity in a network is uniquely identified by using an identifier unrelated to a network address; the entire network is formed by a plurality of management domains, each management domain may be divided into a plurality of control domains, and one controller is provided in each control domain; network nodes participating in identifier resolution and routing comprise, but are not limited to: an edge router of a control domain and a border router of a management domain; the controller stores a mapping between an identifier and an address of an entity in a control domain, and the border router of the management domain stores a matching relationship between an identifier and a control domain of an entity and an association relationship between an identifier and an associated border router in the management domain; and the method comprises, but is not limited to:

step (1).receiving a request message carrying an identifier of a requesting network entity;

step (2).performing identifier resolution on an edge router of a control domain to obtain an address, and if the address obtained by resolution is a final destination address, ending the process; otherwise, it is the address of a certain border router in a management domain where the network entity is located, and performing step (3);

step (3). on the border router in the management domain, performing matching between the identifier of an entity in the present management domain and a controller storing the identifier related mapping relationship, and if a matched controller is found, performing

identifier resolution and addressing in the matched controller, and returning to step (2); otherwise, performing step (4); and

step (4). on the border router of the management domain, obtaining the intermediary border router which storing the mapping relationship between the identifier and management domain; and on the intermediary border router, obtaining, by means of an index, a border router in the management domain associated with the identifier, and performing step (3).

As an improvement to the foregoing method, wherein the network entity comprises, but is not limited to: a host, data, and a service; and a method for generating the identifier comprises, but is not limited to: a flat naming method, a hierarchical naming method, and a multi-semantic naming method.

As an improvement to the foregoing method, the controller is configured to implement the following functions:

(1). the controller stores and maintains the mapping between an identifier and an address of an entity in a control domain;

(2). the controller provides a registration and deregistration function of the mapping between an identifier and an address of an entity in the control domain;

(3). the controller performs information exchange with one or multiple border routers of the management domain that the controller belongs to, to determine a matching relationship between an identifier and a controller;

(4). when a message sent by a router is received, the controller examines whether a destination address in the message is a valid address;

(5). for a message with a destination address being an invalid address, the controller can perform identifier resolution;

(6). for a message with a destination address being a valid address, the controller can re-resolve an identifier based on a specific trigger mechanism;

(7). the controller resolves all identifiers, wherein if the controller stores the mapping related to an identifier, the address obtained by resolution is a final destination address; or if the controller does not store the mapping related to the identifier, the address obtained by resolution is a preset target address, comprising, but not limited to: an address of a border router and a

default forwarding address; and

(8). the controller makes a delivered flow table that carries the address obtained by resolution, wherein a router may fill or update a destination address field in a message based on the delivered flow table.

As an improvement to the foregoing method, the entity in the control domain is a network entity with a unique identifier, and comprises, but is not limited to: a network device, a service, and data; if the entity is a network device, the network device is first connected to a network by the control domain; and if the entity is data, or a service, a network device in which the entity is located is first connected to a network by the control domain, that is, a first control domain through which a message sent by the device in which the entity is located passes after the message enters the network is referred to as a control domain in which the entity is located.

As an improvement to the foregoing method, when a destination address field in a message is empty or a filled address is a predetermined address, the destination address in the message is an invalid address; otherwise, the destination address in the message is a valid address, wherein the predetermined address is an address selected from reserved addresses of an address space, is a non-routing address, and comprises a reserved address in an IP network.

As an improvement to the foregoing method, the specific trigger mechanism can trigger re-resolving and addressing of an identifier, and comprises, but is not limited to: a message carries a trigger flag bit.

As an improvement to the foregoing method, the edge router of the control domain refers to the first router through which a message passes after the message is forwarded into a range of the control domain, and is located at a periphery of the control domain, and the edge router of the control domain is configured to implement the following functions:

(1). after receiving a message with a destination address field being an invalid address, the edge router of the control domain can report and forward the message to a controller to which the edge router belongs, and request the controller to deliver and forward a flow table;

(2). for a message with a destination address field being an invalid address, the edge router of the control domain can fill or update the destination address field in the message based on a flow table delivered by a controller, wherein the updated destination address may comprise, but

is not limited to: a final destination address, and an address of a border router of a management domain to which the controller belongs;

(3). for a message with a destination address field being a valid address, the edge router of the control domain can report and forward, based on a specific trigger mechanism, the message to a controller to which the edge router belongs, and request the controller to deliver and forward a flow table;

(4). for a message with a destination address field being a valid address, the edge router of the control domain can update the destination address field in the message based on a flow table delivered by a controller; and

(5). the edge router of the control domain can perform routing and forwarding based on a flow table delivered by a controller.

As an improvement to the foregoing method, the border router of the management domain refers to a device configured to forward messages between management domains, and is a type of border device, and the border router of the management domain is configured to implement the following functions:

(1). the border router of the management domain has one index, uniquely representing the border router;

(2). the border router of the management domain stores and maintains a mapping between an index and an address of all border routers;

(3). the border router of the management domain stores and maintains the matching relationship between an identifier and a control domain of an entity in the management domain;

(4). the border router of the management domain maps an identifier of a network entity to an index of one or more border routers by using a predetermined method, wherein these mapped border routers are referred to as intermediary border routers identified by the entity, and the predetermined method comprises, but is not limited to: table lookup and function mapping;

(5). the border router of the management domain initiates registration and deregistration requests of the association relationship between an identifier and a border router to an intermediary border router;

(6). the border router of the management domain stores and maintains the association

relationship between an identifier and an associated border router;

(7). the border router of the management domain can add or delete the matching relationship between an identifier and a control domain based on a registration or deregistration request of an identifier;

(8). the border router of the management domain adds or deletes the association relationship between an identifier and a border router based on a registration or deregistration request of an identifier;

(9). the border router of the management domain resolves an identifier, and updates the destination address field in a message based on a resolution result, wherein the address obtained by resolution comprises, but is not limited to: an address of a controller, an address of an edge router, and an address of a border router; and

(10). if a received message is from a network device in a management domain in which the border router of the management domain is located, the border router of the management domain first performs matching between an identifier and a control domain in the management domain, and if matching does not succeed, performs intermediary border router mapping.

As an improvement to the foregoing method, in the matching relationship between an identifier and a control domain of an entity in the management domain, the mapping between an identifier and an address of a network entity corresponding to the identifier is stored and maintained on a controller of the control domain, and the matching relationship between an identifier and a control domain is stored and maintained on a border router of a management domain to which the network entity belongs.

As an improvement to the foregoing method, the association relationship between an identifier and an associated border router refers to a mapping between an identifier of an entity and an address of a border router that registers the relationship.

As an improvement to the foregoing method, the method further includes: a registration step of an identifier, specifically includes:

registering, by an entity, the mapping between an identifier and an address with a controller of a control domain in which the entity is located;

storing, by the controller, the mapping between an identifier and an address of the entity;

spreading a registration request to a border router of a management domain to which the entity belongs;

storing, by the border router, a matching relationship between the mapping and a control domain;

mapping, by the border router, an identifier of the network entity to an index of an intermediary border router;

spreading the registration request to the intermediary border router to which the identifier is mapped; and

storing, by the intermediary border router, the association relationship between an identifier and an associated border router.

As an improvement to the foregoing method, step (2) specifically includes the following steps:

step (201). when a destination address field in a message received by an edge router is an invalid address or an edge router is subject to a specific trigger mechanism, reporting and forwarding the message to a controller to which the edge router belongs to request to deliver and forward a flow table;

step (202). after the controller receives the message, performing identifier resolution;

step (203). delivering, by the controller, a flow table, wherein the flow table carries address information obtained by resolution; and

step (204). filling or updating, by the edge router, the destination address field in the message based on the delivered flow table.

As an improvement to the foregoing method, step (3) specifically includes the following steps:

step (301). performing, by the border router, control domain matching in the management domain based on a received message; and

step (302). if a matched control domain exists, performing identifier resolution and addressing in the matched controller, and performing step 201); and if matching does not succeed, performing intermediary border router mapping and addressing in the management domain, and performing step (4).

As an improvement to the foregoing method, step (4) specifically includes the following steps:

step (401). mapping an identifier to an index of an intermediary border router on the border router of the management domain;

step (402). forwarding a message to an intermediary border router;

step (403). obtaining an associated border router of an identifier on the intermediary border router; and

step (404). forwarding the message to the associated border router, performing control domain matching and addressing in the management domain, and performing step (301).

As an improvement to the foregoing method, the method further includes: a deregistration step of an identifier, specifically including:

deregistering, by an entity, the mapping between an identifier and an address from a controller of a control domain in which the entity is located;

deleting, by the controller, the mapping between an identifier and an address of the entity;

spreading a deregistration request to a border router of a management domain to which the entity belongs;

deleting, by the border router, the matching relationship between the mapping and a control domain;

mapping, by the border router, the identifier of the network entity to an index of an intermediary border router;

spreading the deregistration request, to the intermediary border router to which the identifier is mapped; and

deleting, by the intermediary border router, the association relationship between an identifier and an associated border router

The advantages of the present invention are as follows.

In the method in the present invention, network domain layering is performed based on a management domain and a control domain. A controller stores a mapping between an identifier and an address of an entity in the control domain. A border router of the management domain stores a matching relationship between an identifier of an entity in the management domain and

the control domain and an association relationship between an identifier and an associated border router, to implement hierarchical storage. In addition, dynamic resolution and addressing of an identifier in a message transmission process are implemented in a manner of intra-domain and inter-domain hierarchical resolution. This effectively resolves scalability and search efficiency problems caused by a large number of identifier-address mapping tables.

Brief Description of Drawings

FIG. 1 is a schematic diagram of a network structure;

FIG. 2 is a flowchart of identifier registration according to the present invention;

FIG. 3 is a specific flowchart of identifier registration according to the present invention;

FIG. 4 is a schematic diagram of network processing of registering a mapping between an EID1 and an IP1 at a service end according to the present invention;

FIG. 5 is a flowchart of identifier deregistration according to the present invention;

FIG. 6 is a specific flowchart of identifier deregistration according to the present invention;

FIG. 7 is a schematic diagram of network processing of deregistering an EID1-IP1 mapping at a service end according to the present invention;

FIG. 8 is a flowchart of identifier resolution and routing according to the present invention;

FIG. 9 is a schematic diagram of a network resolution and routing process of requesting an identifier EID1 by user 1;

FIG. 10 is a schematic diagram of a network resolution and routing process of requesting an identifier EID1 by user 2;

FIG. 11 is a schematic diagram of a network resolution and routing process of requesting an identifier EID1 by user 3; and

FIG. 12 is a flowchart of identifier registration according to the present invention.

DESCRIPTION OF EMBODIMENTS

To make the objectives, technical solutions, and advantages of the present invention clearer, the technical solutions of the present invention will be clearly and completely described below in conjunction with the accompanying drawings of the present invention.

An identifier is a globally unique identifier that is unrelated to a network address that can represent a network entity, and includes, but is not limited to, a data identifier. The network

address is a network-routable address information of a location of a network entity, and includes, but is not limited to, information such as an IP address. In the described embodiments, a content identifier is used as an example and is represented by an entity ID (EID), and the network address is represented by an IP address. Video content Movie is deployed at a service end and has an identifier of EID1. A network address of the service end is IP1. The service end publishes the video content by registering an EID1-IP1 mapping with a network. A user obtains the video content by using the identifier EID1. When the service end does not provide video content, the service end needs to deregister the EID1-IP1 mapping from the network.

In the described embodiments, all controllers in the network need to maintain one EID-IP mapping table. A BGP router separately maintains one bloom filter on all controllers in an autonomous system to which the BGP router belongs. The bloom filter is used for quick matching between an identifier and a control domain. To support a deletion function, counting bloom filtering is used in this embodiment. Based on a hash function and the nearest distance principle, an identifier may be mapped to an index of a border router. FIG. 1 is a schematic diagram of a network structure according to the present invention.

FIG. 2 and FIG. 3 are flowcharts of identifier registration. FIG. 4 is a schematic diagram of network processing of registering an EID1-IP1 mapping at a service end.

A process of identifier registration includes the following steps:

S1101: The service end sends a registration request of an EID1-IP1 mapping to a controller C1.

S1102: The controller C1 inserts the mapping in a local mapping table.

S1103: The controller C1 spreads registration information to border routers BGP1 and BGP2.

S1104: The border routers BGP1 and BGP2 insert the EID1-IP1 mapping in storage spaces of respectively maintained bloom filters corresponding to the controller C1.

S1105: Map EID1 to an index of an intermediary border router by using a hash function.

S1106: Spread the registration information to an intermediary border router BGP4.

S1107: Insert association relationships between EID1 and BGP1 and BGP2 into the intermediary border router BGP4, where a registration process ends.

FIG. 5 and FIG. 6 are flowcharts of identifier deregistration. FIG. 7 is a schematic diagram of network processing of deregistering the EID1-IP1 mapping at a service end.

A process of identifier deregistration includes the following steps:

S1201: The service end sends a deregistration request of the EID1-IP1 mapping to the controller C1.

S1202: The controller C1 deletes the mapping in the local mapping table.

S1203: The controller C1 spreads deregistration information to the border routers BGP1 and BGP2.

S1204: The border routers BGP1 and BGP2 delete the EID1-IP1 mapping in the storage spaces of the respectively maintained bloom filters corresponding to the controller C1.

S1205: Map EID1 to an index of an intermediary border router by using a hash function.

S1206: Spread the deregistration information to the intermediary border router BGP4.

S1207: Delete the association relationships between EID1 and BGP1 and BGP2 in the intermediary border router BGP4, where a deregistration process ends.

FIG. 8 is a flowchart of identifier-based addressing. FIG. 9 is a schematic diagram of an addressing process of requesting a network entity corresponding to an identifier EID1 by a user 1.

The addressing process of requesting the network entity corresponding to the identifier EID1 by the user 1 includes the following steps.

S1301: The user 1 makes a request, where a message carries the identifier EID1 of the request.

S1401: After receiving the message, an edge router ER4 examines whether a destination address field is an invalid address; if the destination address field is an invalid address, reports and forwards the message to a controller C1, and performs step S1402; and if the destination address field is a valid address, directly forwards the message based on a flow table.

S1402: After receiving the message, the controller C1 performs identity resolution to obtain an address IP1.

S1403: The controller C1 adds, to a delivered flow table, the address IP1 obtained by resolution.

S1404: The edge router ER4 updates the destination address field in the message to IP1, where the addressing process ends.

FIG. 10 is a schematic diagram of an addressing process of requesting a network entity corresponding to an identifier EID1 by a user 2.

The addressing process of requesting the network entity corresponding to the identifier EID1 by the user 2 includes the following steps.

S1301: The user 2 makes a request, where a message carries the identifier EID1 of the request.

S1401: After receiving the message, an edge router ER4 examines whether a destination address field is an invalid address; if the destination address field is an invalid address, reports and forwards the message to a controller C2, and performs step S1402; and if the destination address field is a valid address, directly forwards the message based on a flow table.

S1402: After receiving the message, the controller C2 performs identity resolution, where an entry related to EID1 is not found in the controller C2, and therefore an address of a border router BGP1 is used as a resolution address.

S1403: The controller C2 adds, to a delivered flow table, the address of the border router BGP1.

S1404: An edge router ER2 updates the destination address field in the message to the address of BGP1, and forwards the message to the border router BGP1.

S1501: Perform quick search and matching of EID1 respectively in bloom filters corresponding to the controllers C1, C2, and C3 on the border router BGP1, find that C1 includes the identifier related to EID1, and choose one edge router ER5 in a control domain in which C1 is located.

S1502: Update the destination address field in the message to an address of ER5, and forward the message to ER5.

In the control domain in which the controller C1 is located, steps S1401 to S1404 are performed again as follows:

S1401: After receiving the message, an edge router ER5 examines whether a destination address field is an invalid address; if the destination address field is an invalid address, reports

and forwards the message to a controller C1, and performs step S1402; and if the destination address field is a valid address, directly forwards the message based on a flow table.

S1402: After receiving the message, the controller C1 performs identity resolution to obtain an address IP1.

S1403: The controller C1 adds, to a delivered flow table, the address IP1 obtained by resolution.

S1404: The edge router ER5 updates the destination address field in the message to IP1, where the addressing process ends.

FIG. 11 is a schematic diagram of an addressing process of requesting a network entity corresponding to an identifier EID1 by a user 3.

The addressing process of requesting the network entity corresponding to the identifier EID1 by the user 3 includes the following steps.

S1301: The user 3 makes a request, where a message carries the identifier EID1 of the request.

S1401: After receiving the message, an edge router ER3 examines whether a destination address field is an invalid address; if the destination address field is an invalid address, reports and forwards the message to a controller C4, and performs step S1402; and if the destination address field is a valid address, directly forwards the message based on a flow table.

S1402: After receiving the message, the controller C4 performs identity resolution, where an entry related to EID1 is not found in the controller C4, and therefore an address of a border router BGP3 is used as a resolution address.

S1403: The controller C4 adds, to a delivered flow table, the address of the border router BGP3.

S1404: The edge router ER3 updates the destination address field in the message to the address of BGP3, and forwards the message to the border router BGP3.

S1501: Perform quick search and matching of EID1 on a bloom filter corresponding to the controller C4 on the border router BGP3, finds that C4 does not include the identifier related to EID1, and perform step S1601.

S1601: Map EID1 to an index of an intermediary border router by using a hash function

on the border router BGP3.

S1602: Update the destination address field in the message to an address of an intermediary border router BGP4 corresponding to the index, and forward the message to the border router BGP4.

S1603: Perform quick search of EID1 in bloom filters corresponding to the border routers in the intermediary border router BGP4, and determine that an associated border router is BGP1.

S1604: Update the destination address field in the message to an address of the associated border router being BGP1, and forward the message to BGP1, and perform step (1501).

In an autonomous system AS1, steps S1501 and S1502 are performed again as follows:

S1501: Perform quick search and matching of EID1 respectively in the bloom filters corresponding to the controllers C1, C2, and C3 on the border router BGP1, find that C1 includes the identifier related to EID1, and choose one edge router ER5 in a control domain in which C1 is located.

S1502: Update the destination address field in the message to an address of ER5, and forward the message to ER5.

In the control domain in which the controller C1 is located, steps S1401 to S1404 are performed again as follows:

S1401: After receiving the message, an edge router ER5 examines whether a destination address field is an invalid address; if the destination address field is an invalid address, reports and forwards the message to a controller C1, and performs step S1402; and if the destination address field is a valid address, directly forwards the message based on a flow table.

S1402: After receiving the message, the controller C1 performs identity resolution to obtain an address IP1.

S1403: The controller C1 adds, to a delivered flow table, the address IP1 obtained by resolution.

S1404: The edge router ER5 updates the destination address field in the message to IP1, where the addressing process ends.

A flowchart of identifier-based addressing of the present invention is shown in FIG. 12. Finally, it should be noted that the foregoing embodiments are merely intended for describing

the technical solutions of the present utility model rather than limiting the present utility model. Although the present utility model is described in detail with reference to the utility model, persons of ordinary skill in the art should understand that they may still make modifications or equivalent replacements to the technical features of the present utility model without departing from the spirit and scope of the technical solutions of the technical solutions of the present utility model. These modifications or equivalent replacements shall all fall within the scope utility model the claims the present utility model.

Finally, it should be noted that the foregoing embodiments are merely intended for describing the technical solutions of the present invention rather than limiting the present invention. Although the present invention is described in detail with reference to the embodiments, persons of ordinary skill in the art should understand that they may still make modifications or equivalent replacements to the technical features of the present invention without departing from the spirit and scope of the technical solutions of the technical solutions of the present invention. These modifications or equivalent replacements shall all fall within the scope of the claims the present invention.

CLAIMS

1. An identifier addressing method for a hierarchical structure, wherein in an architecture in which an identifier and an address are separate, each network entity in a network is uniquely identified by using an identifier unrelated to a network address; the entire network is formed by a plurality of management domains, each management domain may be divided into a plurality of control domains, and one controller is provided in each control domain; network nodes participating in identifier resolution and routing comprise, but are not limited to: an edge router of a control domain and a border router of a management domain; the controller stores a mapping between an identifier and an address of an entity in a control domain, and the border router of the management domain stores a matching relationship between an identifier and a control domain of an entity and an association relationship between an identifier and an associated border router in the management domain; and the method comprises, but is not limited to:

step (1).receiving a request message carrying an identifier of a requesting network entity;

step (2).performing identifier resolution on an edge router of a control domain to obtain an address, and if the address obtained by resolution is a final destination address, ending the process; otherwise, it is the address of a certain border router in a management domain where the network entity is located, and performing step (3);

step (3). on the border router in the management domain, performing matching between the identifier of an entity in the present management domain and a controller storing the identifier related mapping relationship, and if a matched controller is found, performing identifier resolution and addressing in the matched controller, and returning to step (2); otherwise, performing step (4); and

step (4). on the border router of the management domain, obtaining the intermediary border router which storing the mapping relationship between the identifier and management domain; and on the intermediary border router, obtaining, by means of an index, a border router in the management domain associated with the identifier, and performing step (3).

2. The identifier addressing method for a hierarchical structure according to claim 1, wherein the network entity comprises, but is not limited to: a host, data, and a service; and a

method for generating the identifier comprises, but is not limited to: a flat naming method, a hierarchical naming method, and a multi-semantic naming method.

3. The identifier addressing method for a hierarchical structure according to claim 1, wherein the controller is configured to implement the following functions:

(1). the controller stores and maintains the mapping between an identifier and an address of an entity in a control domain;

(2). the controller provides a registration and deregistration function of the mapping between an identifier and an address of an entity in the control domain;

(3). the controller performs information exchange with one or multiple border routers of the management domain that the controller belongs to, to determine a matching relationship between an identifier and a controller;

(4). when a message sent by a router is received, the controller examines whether a destination address in the message is a valid address;

(5). for a message with a destination address being an invalid address, the controller can perform identifier resolution;

(6). for a message with a destination address being a valid address, the controller can re-resolve an identifier based on a specific trigger mechanism;

(7). the controller resolves all identifiers, wherein if the controller stores the mapping related to an identifier, the address obtained by resolution is a final destination address; or if the controller does not store the mapping related to the identifier, the address obtained by resolution is a preset target address, comprising, but not limited to: an address of a border router and a default forwarding address; and

(8). the controller makes a delivered flow table that carries the address obtained by resolution, wherein a router may fill or update a destination address field in a message based on the delivered flow table.

4. The identifier addressing method for a hierarchical structure according to claim 3, wherein the entity in the control domain is a network entity with a unique identifier, and comprises, but is not limited to: a network device, a service, and data; if the entity is a network device, the network device is first connected to a network by the control domain; and if the

entity is data, or a service, a network device in which the entity is located is first connected to a network by the control domain, that is, a first control domain through which a message sent by the device in which the entity is located passes after the message enters the network is referred to as a control domain in which the entity is located.

5. The identifier addressing method for a hierarchical structure according to claim 3, wherein when a destination address field in a message is empty or a filled address is a predetermined address, the destination address in the message is an invalid address; otherwise, the destination address in the message is a valid address, wherein the predetermined address is an address selected from reserved addresses of an address space, is a non-routing address, and comprises a reserved address in an IP network.

6. The identifier addressing method for a hierarchical structure according to claim 3, wherein the specific trigger mechanism can trigger re-resolving and addressing of an identifier, and comprises, but is not limited to: a message carries a trigger flag bit.

7. The identifier addressing method for a hierarchical structure according to claim 5, wherein the edge router of the control domain refers to the first router through which a message passes after the message is forwarded into a range of the control domain, and is located at a periphery of the control domain, and the edge router of the control domain is configured to implement the following functions:

(1). after receiving a message with a destination address field being an invalid address, the edge router of the control domain can report and forward the message to a controller to which the edge router belongs, and request the controller to deliver and forward a flow table;

(2). for a message with a destination address field being an invalid address, the edge router of the control domain can fill or update the destination address field in the message based on a flow table delivered by a controller, wherein the updated destination address may comprise, but is not limited to: a final destination address, and an address of a border router of a management domain to which the controller belongs;

(3). for a message with a destination address field being a valid address, the edge router of the control domain can report and forward, based on a specific trigger mechanism, the message to a controller to which the edge router belongs, and request the controller to deliver and forward

a flow table;

(4). for a message with a destination address field being a valid address, the edge router of the control domain can update the destination address field in the message based on a flow table delivered by a controller; and

(5). the edge router of the control domain can perform routing and forwarding based on a flow table delivered by a controller.

8. The identifier addressing method for a hierarchical structure according to claim 7, wherein the border router of the management domain refers to a device configured to forward messages between management domains, and is a type of border device, and the border router of the management domain is configured to implement the following functions:

(1). the border router of the management domain has one index, uniquely representing the border router;

(2). the border router of the management domain stores and maintains a mapping between an index and an address of all border routers;

(3). the border router of the management domain stores and maintains the matching relationship between an identifier and a control domain of an entity in the management domain;

(4). the border router of the management domain maps an identifier of a network entity to an index of one or more border routers by using a predetermined method, wherein these mapped border routers are referred to as intermediary border routers identified by the entity, and the predetermined method comprises, but is not limited to: table lookup and function mapping;

(5). the border router of the management domain initiates registration and deregistration requests of the association relationship between an identifier and a border router to an intermediary border router;

(6). the border router of the management domain stores and maintains the association relationship between an identifier and an associated border router;

(7). the border router of the management domain can add or delete the matching relationship between an identifier and a control domain based on a registration or deregistration request of an identifier;

(8). the border router of the management domain adds or deletes the association

relationship between an identifier and a border router based on a registration or deregistration request of an identifier;

(9). the border router of the management domain resolves an identifier, and updates the destination address field in a message based on a resolution result, wherein the address obtained by resolution comprises, but is not limited to: an address of a controller, an address of an edge router, and an address of a border router; and

(10). if a received message is from a network device in a management domain in which the border router of the management domain is located, the border router of the management domain first performs matching between an identifier and a control domain in the management domain, and if matching does not succeed, performs intermediary border router mapping.

9. The identifier addressing method for a hierarchical structure according to claim 5, wherein in the matching relationship between an identifier and a control domain of an entity in the management domain, the mapping between an identifier and an address of a network entity corresponding to the identifier is stored and maintained on a controller of the control domain, and the matching relationship between an identifier and a control domain is stored and maintained on a border router of a management domain to which the network entity belongs.

10. The identifier addressing method for a hierarchical structure according to claim 8, wherein the association relationship between an identifier and an associated border router refers to a mapping between an identifier of an entity and an address of a border router that registers the relationship.

11. The identifier addressing method for a hierarchical structure according to claim 1, wherein the method further comprises: a registration step of an identifier, specifically comprising:

registering, by an entity, the mapping between an identifier and an address with a controller of a control domain in which the entity is located;

storing, by the controller, the mapping between an identifier and an address of the entity;

spreading a registration request to a border router of a management domain to which the entity belongs;

storing, by the border router, a matching relationship between the mapping and a control

domain;

mapping, by the border router, an identifier of the network entity to an index of an intermediary border router;

spreading the registration request to the intermediary border router to which the identifier is mapped; and

storing, by the intermediary border router, the association relationship between an identifier and an associated border router.

12. The identifier addressing method for a hierarchical structure according to claim 11, wherein step (2) specifically comprises the following steps:

step (201). when a destination address field in a message received by an edge router is an invalid address or an edge router is subject to a specific trigger mechanism, reporting and forwarding the message to a controller to which the edge router belongs to request to deliver and forward a flow table;

step (202). after the controller receives the message, performing identifier resolution;

step (203). delivering, by the controller, a flow table, wherein the flow table carries address information obtained by resolution; and

step (204). filling or updating, by the edge router, the destination address field in the message based on the delivered flow table.

13. The identifier addressing method for a hierarchical structure according to claim 12, wherein step (3) specifically comprises the following steps:

step (301). performing, by the border router, control domain matching in the management domain based on a received message; and

step (302). if a matched control domain exists, performing identifier resolution and addressing in the matched controller, and performing step 201); and if matching does not succeed, performing intermediary border router mapping and addressing in the management domain, and performing step (4).

14. The identifier addressing method for a hierarchical structure according to claim 13, wherein step (4) specifically comprises the following steps:

step (401). mapping an identifier to an index of an intermediary border router on the border

router of the management domain;

step (402). forwarding a message to an intermediary border router;

step (403). obtaining an associated border router of an identifier on the intermediary border router; and

step (404). forwarding the message to the associated border router, performing control domain matching and addressing in the management domain, and performing step (301).

15. The identifier addressing method for a hierarchical structure according to claim 11, wherein the method further comprises: a deregistration step of an identifier, specifically comprising:

deregistering, by an entity, the mapping between an identifier and an address from a controller of a control domain in which the entity is located;

deleting, by the controller, the mapping between an identifier and an address of the entity;

spreading a deregistration request to a border router of a management domain to which the entity belongs;

deleting, by the border router, the matching relationship between the mapping and a control domain;

mapping, by the border router, the identifier of the network entity to an index of an intermediary border router;

spreading the deregistration request, to the intermediary border router to which the identifier is mapped; and

deleting, by the intermediary border router, the association relationship between an identifier and an associated border router.