

PORT MULTIPLEXING METHOD AND SYSTEM COMPATIBLE WITH MULTI-PORT USE

TECHNICAL FIELD

[0001] The present invention relates to the field of communication, and in particular, relates to a port multiplexing method and system compatible with multi-port use.

BACKGROUND

[0002] A system or program communication client side uniquely identifies a server and establishes a communication channel with it through ip and port, and the process of establishing the communication channel requires different tasks or services in the same process to monitor different ports; and otherwise, abnormality of monitoring port conflict will occur in the program. The following cross-domain systems often have proxies in the middle and are systems with limited port resources:

in the system communicating via a gateway, the gateway only provides a limited number of available ports; docker machines without LAN ip between different hosts generally perform communication through the ports mapped by the hosts, and there will be multiple docker machines on the hosts at the same time, so the port use is also limited;

in existing systems or systems that use a lot of open source codes, the lack of management of port resources is very likely to cause port conflicts; and in this case, it is generally necessary to increase port management and use local proxies to forward data or use libraries with port multiplexing such as grpc to modify the codes;

however, the existing port management module has a certain cost for managing the use of multi-task ports, and due to the additional use of a data forwarding process

of the proxy process, there is a certain performance loss, which is not suitable for the scene with large communication data; using grpc and other libraries with port multiplexing to modify the codes involves a large number of modifications to the business system, which has poor compatibility with the old system.

SUMMARY

[0003] Based on the above description, an embodiment of the present application provides a port multiplexing method and system compatible with multi-port use, which solve the problems existing in the prior art.

[0004] In a first aspect, a port multiplexing method compatible with multi-port use is provided, and the method comprises the following operations:

generating a target task identification code when issuing a target task, and sending the target task identification code to a client and a server performing the target task, and performing information configuration on the client and the server;

generating by the client a virtual address and a virtual port according to the target task identification code, and establishing by the client a client mapping table with a proxy address; generating by the server a virtual address and a virtual port according to the target task identification code, and establishing by the server a server mapping table with the server address and the server port;

establishing by the client a first channel according to the client mapping table, establishing by the client a socket connection through the first channel, and sending by the client first channel information to the server; establishing by the server a second channel according to the server mapping table, and binding by the server the second channel and the socket connection into a communication channel through first channel information, wherein the communication channel is used for realizing the

communication between the client and the server.

[0005] Preferably, the number of servers performing the target task may be multiple, the operation of performing information configuration on the client and the server comprises:

distinguishing different servers that perform the target task by using proxy information and/or corresponding server information.

[0006] Preferably, the operation of establishing by the client a first channel according to the client mapping table and establishing by the client a socket connection through the first channel comprises:

establishing by the client the first channel according to the virtual address and the virtual port, wherein the virtual port is used as a first channel name, and socket connection is established according to the virtual address and the virtual port.

[0007] Preferably, the operation of establishing by the server a second channel according to the server mapping table comprises: establishing a second channel according to the virtual address and the virtual port in the server mapping table, wherein the virtual port is used as a second channel name.

[0008] Preferably, the operation of binding the second channel and the socket connection into a communication channel through first channel information comprises:

performing socket binding on the first channel and the second channel with the same name; wherein the first channel information comprises the first channel name, the first channel name is generated through the virtual port of the client; and the second channel name is generated through the virtual port of the server.

[0009] In a second aspect, a port multiplexing system compatible with multi-port use is provided, and the system comprises:

an issuing module, being configured to generate a target task identification code when issuing a target task, and send the target task identification code to a client and a server performing the target task, and perform information configuration on the client and the server;

a generating module, being configured to generate by the client a virtual address and a virtual port according to the target task identification code, and establish a client mapping table with a proxy address; the server generating a virtual address and a virtual port according to the target task identification code, and establishing a server mapping table with the server address and the server port;

a communication module, being configured to establish by the client a first channel according to the client mapping table, establish a socket connection through the first channel, and send the first channel information to the server; the server establishing a second channel according to the server mapping table, and binding the second channel and the socket connection into a communication channel through first channel information, wherein the communication channel is used for realizing the communication between the client and the server.

[0010] Preferably, the number of servers performing the target task may be multiple, and the operation of performing information configuration on the client and the server in the issuing module comprises:

distinguishing different servers that perform the target task by using proxy information and/or corresponding server information.

[0011] Preferably, the operation of establishing by the client a first channel according to the client mapping table and establishing a socket connection through the first channel in the communication module comprises:

establishing by the client the first channel according to the virtual address and the

virtual port, wherein the virtual port is used as a first channel name, and socket connection is established according to the virtual address and the virtual port.

[0012] Preferably, the operation of establishing by the server a second channel according to the server mapping table in the communication module comprises:

establishing a second channel according to the virtual address and the virtual port in the server mapping table, wherein the virtual port is used as a second channel name.

[0013] Preferably, the operation of binding the second channel and the socket connection into a communication channel through first channel information in the communication module comprises:

performing socket binding on the first channel and the second channel with the same name; wherein the first channel information comprises the first channel name, the first channel name is generated through the virtual port of the client; and the second channel name is generated through the virtual port of the server.

[0014] In the embodiment of the present application, a target task identification code is generated when issuing a target task, the target task identification code is sent to a client and a server performing the target task, and information configuration is performed on the client and the server; the client generates a virtual address and a virtual port according to the target task identification code, and establishes a client mapping table with a proxy address; the server generates a virtual address and a virtual port according to the target task identification code, and establishes a server mapping table with the server address and the server port; the client establishes a first channel according to the client mapping table, establishes a socket connection through the first channel, and sends the first channel information to the server; the server establishes a second channel according to the server mapping table, and binds the

second channel and the socket connection into a communication channel through first channel information. As can be seen from the above description, on the premise of not modifying the service code, the present application is compatible with the scenarios of large communication data and old systems at the same time by adding the concept of virtual ports, and effectively solves the application of complex systems in multiple scenarios.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] In order to explain the embodiments of the present invention or the technical solutions in the prior art more clearly, the following description will briefly introduce the drawings required in the description of the embodiments or the prior art. Obviously, the drawings in the following description are only exemplary, and other implementation drawings can be obtained by extension according to the provided drawings by those of ordinary skill in the art without any creative labor.

[0016] FIG. 1 is a flowchart diagram of a port multiplexing method compatible with multi-port use provided according to an embodiment of the present application.

[0017] FIG. 2 is a block diagram of a port multiplexing system compatible with multi-port use provided according to an embodiment of the present application.

DETAILED DESCRIPTION

[0018] In order to make the purpose, technical solutions and advantages of the present application more clear, the present application will be further explained in detail below with reference to the attached drawings and embodiments. It shall be appreciated that, the specific embodiments described here are only used to explain the present application, and are not intended to limit the present application.

[0019] In the description of the present invention, the phrase “a plurality of” means two or more unless otherwise stated. The terms such as “first” and “second” in the specification and claims of the present invention as well as the above drawings are intended to distinguish the referred objects. For the scheme with time sequence flow, such term expression need not be understood as describing a specific order or sequence; and for the scheme with device structure, such term expression does not distinguish the importance and position relationships.

[0020] In addition, the terms “including” and “having” and any variants thereof are intended to cover non-exclusive inclusion. For example, processes, methods, systems, products or equipments that comprise a series of steps or units are not necessarily limited to those steps or units that have been explicitly listed, but may also comprise other steps or units which are not explicitly listed but are inherent to these processes, methods, products or equipments, or steps or units added based on further optimization scheme of the inventive concept.

[0021] In a general LAN, the port resources are not restricted, and different algorithms and tasks are distinguished from each other by using different ports. As long as different algorithms and tasks use different ports in the business layer, the system can run normally. However, when the system runs in an environment with limited port resources such as cross-domain, the use of ports will be strictly restricted, or the ports even cannot work normally. The invention can effectively solve the above problems on the premise of not modifying the service code, and by adding the concept of virtual port and the middle mapping layer, it is compatible with the above scenarios existing in the prior art at the same time, and it can effectively solve the application of complex systems in multiple scenarios.

[0022] Specifically, please refer to FIG. 1, which shows a flowchart diagram of a

port multiplexing method compatible with multi-port use provided according to an embodiment of the present application, and the method may comprise the following steps:

S1: generating a target task identification code when issuing a target task, and sending the target task identification code to a client and a server performing the target task, and performing information configuration on the client and the server.

[0023] In a general communication mode, two parties can communicate with each other only by getting a pair of information - ip:port; while in the system with limited ports, generally an intermediate gateway imposes restriction on port resources, and thus in the embodiment of the present application, one piece of task information and three pairs of communication information are defined, wherein task_id and IP are of string type, and port is of u32 type.

[0024] First of all, the specific definitions in the embodiments of the present application are explained as follows:

1) task_id (target task identification code): a random ID generated when the system issues a task, the client and the server of the same task get the same task_id;

2) server_ip (server address) and server_port (server port): ip and port monitored at the server side, the number of the server addresses is finite and definite, so they can be statically configured;

3) proxy_ip and proxy_port: intermediate proxy addresses, such as gateway addresses or host addresses or the like;

4) virtual_ip (virtual address) and virtual_port (virtual port): virtual_ip equals to tmp_id, and virtual_port equals to a remainder after dividing the hash value of tmp_id by the maximum value of u32; tmp_id needs to be generated by information of two dimensions, the vertical dimension uses task_id to distinguish different tasks, and the

horizontal information may use proxy information or information that can distinguish different servers or the like to distinguish different target machines in the same task.

S2: generating by the client a virtual address and a virtual port according to the target task identification code, and establishing by the client a client mapping table with a proxy address; generating by the server a virtual address and a virtual port according to the target task identification code, and establishing by the server a server mapping table with the server address and the server port;

S3: establishing by the client a first channel according to the client mapping table, establishing by the client a socket connection through the first channel, and sending by the client the first channel information to the server; establishing by the server a second channel according to the server mapping table, and binding by the server the second channel and the socket connection into a communication channel through the first channel information.

[0025] The communication channel is used for realizing the communication between the client and the server. The client establishes the first channel according to the virtual address and the virtual port, wherein the virtual port is used as a first channel name, and socket connection is established according to the virtual address and the virtual port. A second channel is established according to the virtual address and the virtual port in the server mapping table, wherein the virtual port is used as a second channel name.

[0026] In the embodiment of the present application, the socket of both communication parties may be successfully established, but different services on a server use the same monitoring port and cannot be distinguished from each other, so the concepts of channel and channel_name are introduced, and a channel represents a service; for the same task, since the task_id is the same and the virtual_port is known

to each other, virtual_port is used as the channel_name.

[0027] In the embodiment of the present application, the establishment process of the communication channel is as follows:

1) the task, the task ID and the proxy_ip and proxy_port list of each proxy machine are issued, if a certain machine has no code, the corresponding position in the list is empty, and index is issued to indicate the position of the proxy address of the machine in the list;

2) the server finds out its proxy address, and finds out server_ip and server_port from the configuration according to the proxy address;

3) the server generates virtual_ip and virtual_port according to task_id, and establishes mapping with server_ip and server_port;

4) the server side keeps monitoring server_ip and server_port in the background, and waits for the client to establish a connection;

5) the client generates virtual_ip and virtual_port according to task_id, and establishes mapping with the proxy address;

the above steps make establishment of respective mapping tables of the client and the server complete, and the server side is in the connection monitoring state, and then it is started to establish the communication channel;

6) the client business layer uses virtual_ip and virtual_port to create a new channel, and uses virtual_port as the channel_name, and takes out ip:port from the respective mapping tables to establish the socket connection;

7) clientchannel sends channel_name to the server; at this time, the client-side channel has been bound with the socket, while the server-side socket is in a free state and holds the channel_name;

8) the server-side business layer uses virtual_ip and virtual_port to create a new

channel, uses virtual_port as channel_name, takes out the real monitored object from the mapping table, and registers itself on the monitored object, and the background binds the channel with the same channel_name and the free socket together;

9) after the channel is established, the client sends data to the server.

[0028] The following description presents an optional embodiment applied to the above method, specifically:

the existing three machines run an algorithm, which requires one client and two servers during the operation, the proxy addresses of the two servers are: proxy1.com:444 and proxy2.com:444, the proxy addresses at the server side are all 0.0.0.0: 8080, and the task_id of this task is task1, and roleid is used to indicate what the proxy address is when this machine acts as a server.

S1: task issuing:

1) the parameters received by the client are:

{task_id: task1, roleid:0, iplist: [null, proxy1.com:444,proxy2.com:444]},

because client roleid is 0 and there is no proxy address, bit 0 of iplist is set to null;

2) the parameters received by the server1 are:

{task_id: task1, roleid:1, iplist: [null, proxy1.com:444,proxy2.com:444]};

3) the parameters received by the server2 are:

{task_id: task1, roleid:2, iplist: [null, proxy1.com:444,proxy2.com:444]};

S2: generating virtual information and establishing a mapping table;

1) the client needs to establish a mapping to the server1 and the server2 by using either task_id+roleId or task_id+proxy addresses, here task_id+roleId is used, and for the server with roleId of 1, virtual_ip and virtual_port are generated; because virtual_port is the result of virtual_ip hash, virtual_port1 is used instead of virtual information in the mapping table for convenience of exhibition herein.

2) the server1 needs to establish the mapping of the monitoring address, get proxy1.com:444 through roleid, get the monitoring address 0.0.0.0:8080 by looking up configuration through proxy1.com:444, and generate the virtual information virtual_port1 as described above;

3) the server2 similarly sets the monitoring address 0.0.0.0:8080, and generates the virtual information virtual_port2 as described above;

the client mapping table is as follows:

Virtual addresses	Real addresses
virtual_port1	proxy1.com:444
virtual_port2	proxy2.com:444

the server1 mapping table is as follows:

Virtual addresses	Real addresses
virtual_port1	0.0.0.0:8080

the server2 mapping table is as follows:

Virtual addresses	Real addresses
virtual_port2	0.0.0.0:8080

4) the 8080 mapping relationship from proxy1.com:444 to server1 is set in advance in gateway and other equipments;

S3: establishing a communication channel:

1) the client actively establishes a connection, such as the server1 establishes a connection: a new channel is created, and the channel_name is set to be virtual_port1, the proxy address proxy1.com:444 is obtained through virtual_port1, and socket connection is established; then channel_name is sent;

2) the server1 creates a new channel, sets the channel_name to be virtual_port1, and gets the object with the monitoring address of 8080 from the mapping table through virtual_port1, and gets the socket with the same channel_name in the free state according to the channel_name, and thus the client successfully establishes connection with the related channel of the server1, and data can be received and transmitted normally.

As can be known from the above description, the present application uses multi-dimensional information to generate virtual information such as virtual_ip and virtual_port; generates virtual_port by the two-dimensional information of task_id and role id as mentioned above; and establishes the mapping relationship of m:n between virtual information and real information, and uses the virtual information as channel_name to establish a double-ended communication channel; m:n relationship is established between virtual_port and proxy_port as mentioned above, in special circumstances where there is only one gateway proxy, a many-to-one mapping relationship is established, which effectively utilizes system resources.

[0029] As shown in FIG. 2, an embodiment of the present application further provides a port multiplexing system 200 compatible with multi-port use. The system 200 comprises:

an issuing module 201, being configured to generate a target task identification code when issuing a target task, and send the target task identification code to a client and a server performing the target task, and perform information configuration on the client and the server;

a generating module 202, being configured to generate by the client a virtual address and a virtual port according to the target task identification code, and establish a client mapping table with a proxy address; the server generating a virtual address

and a virtual port according to the target task identification code, and establishing a server mapping table with the server address and the server port;

a communication module 203, being configured to establish by the client a first channel according to the client mapping table, establish a socket connection through the first channel, and send the first channel information to the server; the server establishing a second channel according to the server mapping table, and binding the second channel and the socket connection into a communication channel through first channel information, wherein the communication channel is used for realizing the communication between the client and the server.

[0030] In one optional embodiment of the present application, the number of servers performing the target task may be multiple, and the operation of performing information configuration on the client and the server in the issuing module comprises:

distinguishing different servers that perform the target task by using proxy information and/or corresponding server information.

[0031] In one optional embodiment of the present application, the operation of establishing by the client a first channel according to the client mapping table and establishing a socket connection through the first channel in the communication module 203 comprises: establishing by the client the first channel according to the virtual address and the virtual port, wherein the virtual port is used as a first channel name, and socket connection is established according to the virtual address and the virtual port.

[0032] In one optional embodiment of the present application, the operation of establishing by the server a second channel according to the server mapping table in the communication module 203 comprises: establishing a second channel according to

the virtual address and the virtual port in the server mapping table, wherein the virtual port is used as a second channel name.

[0033] In one optional embodiment of the present application, the operation of binding the second channel and the socket connection into a communication channel through first channel information in the communication module 203 comprises: performing socket binding on the first channel and the second channel with the same name; wherein the first channel information comprises the first channel name, the first channel name is generated through the virtual port of the client; and the second channel name is generated through the virtual port of the server.

[0034] The port multiplexing system compatible with multi-port use provided in the embodiment of the present application is used to realize the port multiplexing method compatible with multi-port use described above. Please refer to the definition of the port multiplexing method compatible with multi-port use described above for the specific definition of the port multiplexing system compatible with multi-port use, and this will not be further described herein. Each part of the port multiplexing system compatible with multi-port use described above may be realized in whole or in part by software, hardware and combination thereof. The above modules may be embedded in or independent of the processor in the equipment in the form of hardware, or they may be stored in the memory in the equipment in the form of software so as to be called by the processor to execute the operations corresponding to the modules described above.

[0035] The technical features of the above-mentioned embodiments may be arbitrarily combined, and for briefness of the description, not all possible combinations of the technical features in the above-mentioned embodiments are described; however, the combinations of these technical features shall all be

considered as within the scope recorded in this specification as long as there is no contradiction in the combinations.

[0036] The above-mentioned embodiments only represent several implementations of the present application, and they are described more specifically and in detail, but these embodiments should not thus be understood as limiting the scope of the present application for patent. It shall be noted that, several modifications and improvements can be made by those of ordinary skill in the art without departing from the concept of the present application, and all of these modifications and improvements are within the scope claimed in the present application. Therefore, the scope claimed in the present application for the patent shall be governed by the appended claims.

CLAIMS

1. A port multiplexing method compatible with multi-port use, being characterized in that, the method comprising:

generating a target task identification code when issuing a target task, and sending the target task identification code to a client and a server performing the target task, and performing information configuration on the client and the server to define one piece of task information and three pairs of communication information;

generating by the client a virtual address and a virtual port according to the target task identification code, and establishing by the client a client mapping table with a proxy address; generating by the server a virtual address and a virtual port according to the target task identification code, and establishing by the server a server mapping table with the server address and the server port;

establishing by the client a first channel according to the client mapping table, establishing by the client a socket connection through the first channel, and sending by the client the first channel information to the server; establishing by the server a second channel according to the server mapping table, and binding by the server the second channel and the socket connection into a communication channel through first channel information, wherein the communication channel is used for realizing the communication between the client and the server;

wherein the task information is a random ID generated when a task is issued, the client and the server of the same task get the same ID;

wherein the communication information comprises:

server address and server port;

proxy address and proxy port;

virtual address and virtual port; wherein the virtual address equals to tmp_id, and

the virtual port equals to a remainder after dividing the hash value of tmp_id by the maximum value of u32; the tmp_id is generated by information of vertical dimension and horizontal dimension; the vertical dimension uses the task information to distinguish different tasks, and the horizontal dimension uses proxy information and the virtual port is used as channel name;

wherein the operation of binding the second channel and the socket connection into a communication channel through first channel information comprises:

performing socket binding on the first channel and the second channel with the same name; wherein the first channel information comprises the first channel name, the first channel name is generated through the virtual port of the client; and the second channel name is generated through the virtual port of the server.

2. The method according to Claim 1, the number of servers performing the target task may be multiple, being characterized in that, the operation of performing information configuration on the client and the server comprises:

distinguishing different servers that perform the target task by using proxy information and/or corresponding server information.

3. The method according to Claim 1, being characterized in that, the operation of establishing by the client a first channel according to the client mapping table and establishing by the client a socket connection through the first channel comprises:

establishing by the client the first channel according to the virtual address and the virtual port, wherein the virtual port is used as a first channel name, and the socket connection is established according to the virtual address and the virtual port.

4. The method according to Claim 1, being characterized in that, the operation of establishing by the server a second channel according to the server mapping table comprises:

establishing a second channel according to the virtual address and the virtual port in the server mapping table,

wherein the virtual port is used as a second channel name.

5. A port multiplexing system compatible with multi-port use, being characterized in that, the system comprising:

an issuing module, being configured to generate a target task identification code when issuing a target task, and send the target task identification code to a client and a server performing the target task, and perform information configuration on the client and the server to define one piece of task information and three pairs of communication information;

a generating module, being configured to generate by the client a virtual address and a virtual port according to the target task identification code, and establish a client mapping table with a proxy address; the server generating a virtual address and a virtual port according to the target task identification code, and establishing a server mapping table with the server address and the server port;

a communication module, being configured to establish by the client a first channel according to the client mapping table, establish a socket connection through the first channel, and send first channel information to the server; the server establishing a second channel according to the server mapping table, and binding the second channel and the socket connection into a communication channel through the first channel information, wherein the communication channel is used for realizing the communication between the client and the server;

wherein the task information is a random ID generated when the system issues a task, the client and the server of the same task get the same ID;

wherein the communication information comprises:

server address and server port;

proxy address and proxy port;

virtual address and virtual port; wherein the virtual address equals to tmp_id, and the virtual port equals to a remainder after dividing the hash value of tmp_id by the maximum value of u32; the tmp_id is generated by information of vertical dimension and horizontal dimension; the vertical dimension uses the task information to distinguish different tasks, and the horizontal dimension uses proxy information and the virtual port is used as channel name;

wherein the operation of binding the second channel and the socket connection into a communication channel through first channel information comprises:

performing socket binding on the first channel and the second channel with the same name; wherein the first channel information comprises the first channel name, the first channel name is generated through the virtual port of the client; and the second channel name is generated through the virtual port of the server.

6. The system according to Claim 5, the number of servers performing the target task may be multiple, being characterized in that, the operation of performing information configuration on the client and the server in the issuing module comprises:

distinguishing different servers that perform the target task by using proxy information and/or corresponding server information.

7. The system according to Claim 5, being characterized in that, the operation of establishing by the client a first channel according to the client mapping table and establishing a socket connection through the first channel in the communication module comprises:

establishing by the client the first channel according to the virtual address and the virtual port, wherein the virtual port is used as a first channel name, and socket

connection is established according to the virtual address and the virtual port.

8. The system according to Claim 5, being characterized in that, the operation of establishing by the server a second channel according to the server mapping table in the communication module comprises:

establishing a second channel according to the virtual address and the virtual port in the server mapping table, wherein the virtual port is used as a second channel name.