

CLAIMS

1. A method for CDN inter-node encryption, comprising:

generating a symmetric key set and a key index by a key center randomly or according to a preset algorithm, and sending the symmetric key set and the key index to each Content Delivery Network (CDN) node; and

receiving, by the CDN node, the symmetric key set and the key index from the key center, determining a key from the symmetric key set according to the key index, and using the key to perform encryption and decryption processing when performing Hyper Text Transfer Protocol (HTTP) data transmission with other CDN nodes.

2. The method for CDN inter-node encryption according to claim 1, wherein:

the sending, by the key center, the symmetric key set and the key index to each CDN node comprises: sending the symmetric key set and the key index to each CDN node through an encryption channel.

3. The method for CDN inter-node encryption according to claim 1, wherein:

the using, by the CDN node, the key to perform encryption and decryption processing when performing HTTP data transmission with other CDN nodes comprises: when the CDN node is an edge node, using the key to encrypt an HTTP request, sending the encrypted HTTP request to a parent node, and using the key to decrypt an HTTP response after receiving the HTTP response from the parent node; and/or, when the CDN node is a parent node, after receiving an HTTP request from an edge node, using the key to decrypt the HTTP request, using the key to encrypt an HTTP response and sending the encrypted HTTP response to the edge node.

4. The method for CDN inter-node encryption according to claim 1, wherein:

the generating a symmetric key set and a key index by a key center, and sending the symmetric key set and the key index to each CDN node comprises:

generating, by the key center, the key index at a first time interval randomly or according to a preset algorithm, and sending the generated key index to the CDN nodes; and, generating, by the key center, the symmetric key set at a second time interval randomly or according to a preset algorithm, and sending the generated symmetric key

set to the CDN nodes, wherein the second time interval is greater than the first time interval; or

or, generating, by the key center, the key index at a first time interval randomly or according to a preset algorithm, and sending the generated key index to the CDN nodes; and, generating, by the key center, the symmetric key set at a second time interval randomly or according to a preset algorithm and generating the key index at a second time interval randomly or according to a preset algorithm, and sending the generated symmetric key set and key index to the CDN nodes, wherein the second time interval is greater than the first time interval.

5. The method for CDN inter-node encryption according to claim 1, wherein:

the key index has a range of $[x,y]$, wherein both x and y are integers and the number of symmetric keys in the symmetric key set is obtained by adding one to a difference between y and x .

6. A device for CDN inter-node encryption, applied to a key center, comprising a generation module and a sending module;

the generation module is configured to generate a symmetric key set and a key index randomly or according to a preset algorithm; and

the sending module is configured to send the key set generated by the generation module to each CDN node and further configured to send the key index generated by the generation module to each CDN node.

7. The device for CDN inter-node encryption according to claim 6, wherein:

the sending module is further configured to send the symmetric key set and the key index to each CDN node through an encryption channel.

8. The device for CDN inter-node encryption according to claim 6, wherein:

the generation module is further configured to generate the key index at a first time interval randomly or according to a preset algorithm, and generate the symmetric key set at a second time interval randomly or according to a preset algorithm, wherein the second time interval is greater than the first time interval; or, the generation module is further configured to generate the key index at a first time interval randomly or according

to a preset algorithm, and generate the symmetric key set at a second time interval randomly or according to a preset algorithm and generate the key index at a second time interval randomly or according to a preset algorithm, wherein the second time interval is greater than the first time interval.

9. A device for CDN inter-node encryption, applied to a CDN node, comprising a receiving module and an encryption processing method;

the receiving module is configured to receive a key set from a key center and further configured to receive a key index from the key center, the key set and the key index being generated randomly or according to a preset algorithm; and

the encryption processing module is configured to determine a key from the symmetric key set according to the key index and use the key to perform encryption and decryption processing when performing HTTP data transmission with other CDN nodes.

10. The device for CDN inter-node encryption according to claim 9, wherein:

the encryption processing module is further configured to: when the CDN node is an edge node, use the key to encrypt an HTTP request, send the encrypted HTTP request to a parent node, and use the key to decrypt an HTTP response after receiving the HTTP response from the parent node; and, the encryption processing module is further configured to: when the CDN node is a parent node, after receiving an HTTP request from an edge node, use the key to decrypt the HTTP request, use the key to encrypt an HTTP response and send the encrypted HTTP response to the edge node.

11. A computer-readable storage medium stored with computer programs that, when executed by a processor, implements the steps in the method according to any one of claims 1 to 5.

12. A computer apparatus, comprising a memory, a processor and computer programs that are stored on the memory and can be run on the processor, the processor implementing the steps in the method according to any one of claims 1 to 5 when executing the programs.