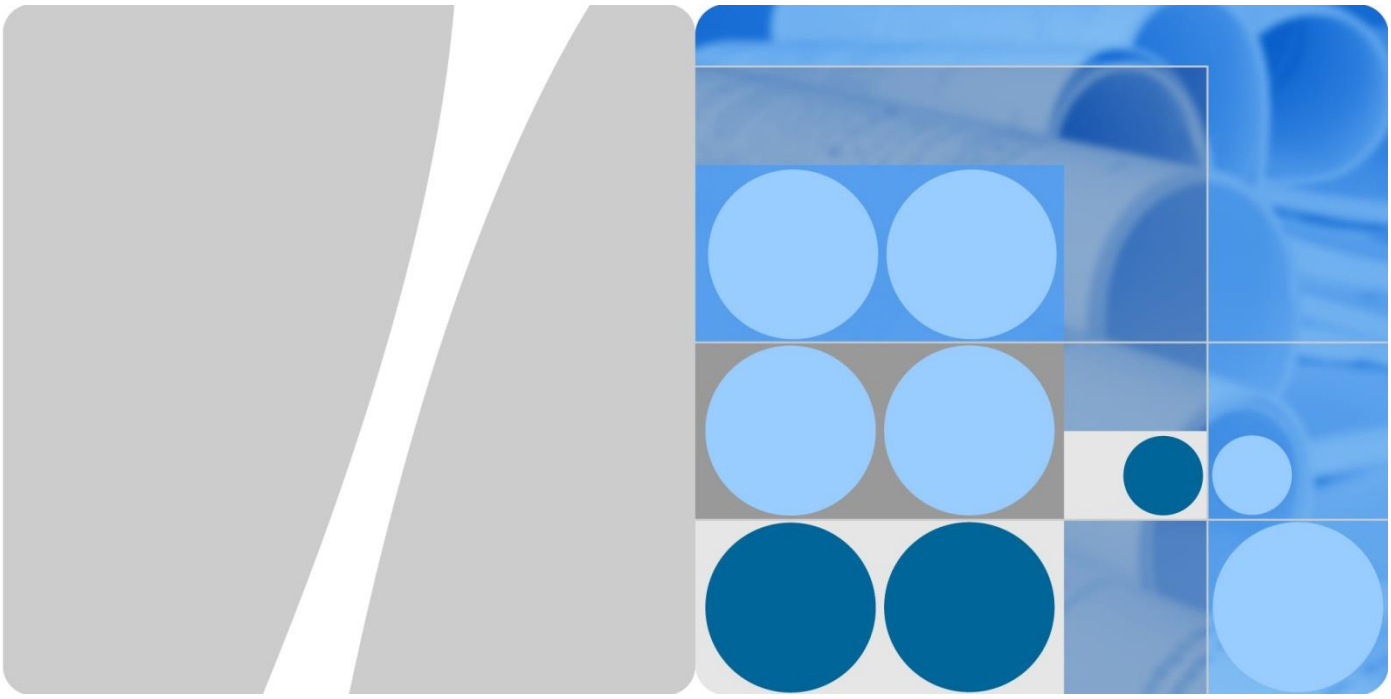


Huawei USG6000 NGFW

SLB Technical White Paper

Issue V1.0

Date 2017-2-28

Huawei Technologies Co., Ltd.

Trademarks and Permissions


and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: http://www.huawei.com

Email: support@huawei.com

Contents

Huawei USG6000 NGFW
SLB Technical White Paper

Keyword:

NGFW, SLB

Abstract:

This document describes the principle and solution of the SLB technology for Huawei NGFW.

Abbreviation	Full Spelling
NGFW	Next-Generation Firewall
SLB	Server Load Balancing

1 Background

With the rapid development of Internet, network data access traffic also rapidly increases. For example, the access volumes of data centers, large-scale portal websites, and enterprise application services far exceed the capabilities of existing servers. As such, services or applications of end users fail to get quick responses, causing business losses to service providers. So how to rapidly deploy services for high availability and rapid response?

Scalability: In response to the preceding issues, the administrator needs to expand server capabilities in a timely manner to compensate for capability insufficiencies. The administrator can also purchase servers with higher performance to replace existing servers. This solution, however, increases the costs and invalidates existing servers.

Periodicity or temporality: External application services feature periodical changes. For example, the 12306 website (Chinese official website for booking train tickets) may bear an extremely large volume of traffic before and after the Spring Festival period. After this period, however, the server usage significantly decreases. Xiaomi's case is another example. Every time Xiaomi rolls out a new mobile phone product available for panic buying, its server bears a large volume of traffic. After that, the traffic significantly decreases. Purchasing a new server and using it only in traffic peak hours is a resource waste.

High availability: During external service and application provision, the experience of end customers cannot be compromised after a single server becomes down or overloaded. The administrator shall sense the potential issue of the server and properly distribute new traffic.

The server load balancing (SLB) mechanism properly addresses the preceding issue. In this mechanism, multiple servers are deployed, and traffic is properly distributed among them so that the usage of the server performance can be maximized and that the application system stability and availability can be guaranteed.

2 Concepts and Principles

2.1 SLB Concepts and Principles

Using the SLB technology, software or network devices can be configured before a group of servers that implement the same or similar functions to properly distribute traffic for the servers, or to switch the service requests that have been distributed to a faulty server to other functional servers. The following part describes several basic concepts of SLB.

- **Virtual server**

  Virtual servers are virtualized servers that can be accessed by external clients in the SLB mechanism. A virtual server consists of the IP address and port number. User access traffic is passed on to the virtual server through a public or private network. Then the virtual server forwards the traffic to a background physical server based on the policy.

- **Physical server and physical server group**

  The physical server is the device that actually provides the service and consists of the actual IP address and port number. The physical server responds to and processes user requests and returns the results to the customer.

  A physical server group is a cluster of physical servers.

- **LB algorithm**

  The LB device distributes service traffic to different physical servers based on a certain algorithm. For example, the SLB module provides round robin (RR), weighted round robin (WRR), least connection (LC), and weighted least connection (WLC), source IP-based hash, and weighted source IP-based hash. Users can select an LB algorithm based on the specific usage scenario.

- **Sticky session algorithm**

  Sticky session is a mechanism on the LB module that can identify the correlation of interactions between the client and server and ensure that a series of correlated access request are always allocated to the same server during LB. Sticky session algorithms currently available are source IP address-based sticky session, HTTP cookie-based sticky session, and session ID-based sticky session.

- **Health surveillance and limit on the maximum number of connections**

  To guarantee the availability and reliability of the physical server, a server health status surveillance mechanism is in need. For example, as for the most common web application servers, HTTP can be used to configure the specific surveillance URL and interval for health status surveillance over the physical server. If the health status of a server is abnormal, subsequent new sessions are no longer sent to this server. After the

health status of this server becomes normal again, subsequent new sessions are re-allocated to this server.

The limit on the maximum number of connections supported by a physical server can be configured to prevent the server from becoming down or failing to respond after the number of connections exceeds the performance.

The limit on the maximum number of connections supported by a virtual server can be configured to guarantee the number of service connections that can be provided externally.

- **SSL uninstallation and LB after uninstallation**

  For the sake of security, an increasing number of web servers are using HTTPS for transmission. Compared with common HTTP traffic processing, the processing of encrypted HTTPS traffic consumes more server resources and has higher requirements on the web server performance. In addition, the FW cannot extract the URL or HOST field from encrypted HTTPS traffic for refined traffic scheduling. As a result, the value of the LB function cannot be maximized.

  In SSL uninstallation, the FW serves as the SSL proxy server and takes over SSL data encryption and decryption. The intranet server can directly read the restored HTTP traffic, without having to install a specific drive program. This helps implement smooth transfer and greatly reduces the load on the intranet server.

- **LB and security policy**

  While implementing traffic LB based on a matched specific LB policy, the FW also provides the corresponding security policy. After load-balanced traffic matches a security policy, security scanning and detection, such as antivirus, IPS, URL filtering, DLP, and AAPT, can be implemented.

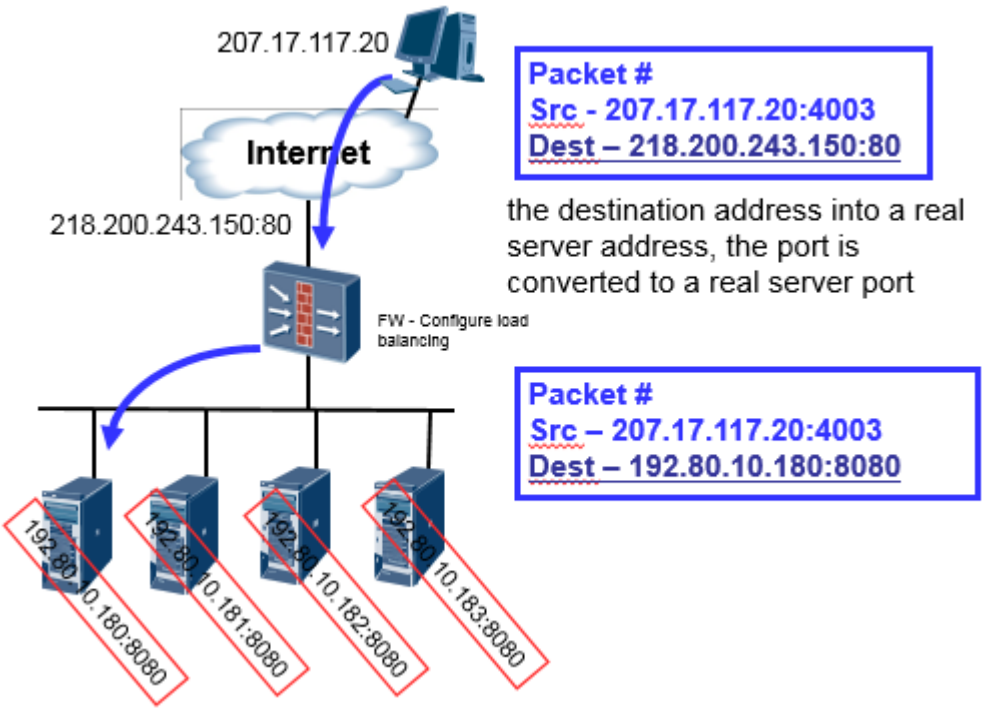
3 Typical Applications

3.1 Layer 4-based SLB

Layer 4-based SLB is an LB mode based on destination NAT. After receiving request packets sent to the virtual server, the load balancer selects a specific physical server based on the configured LB algorithm.

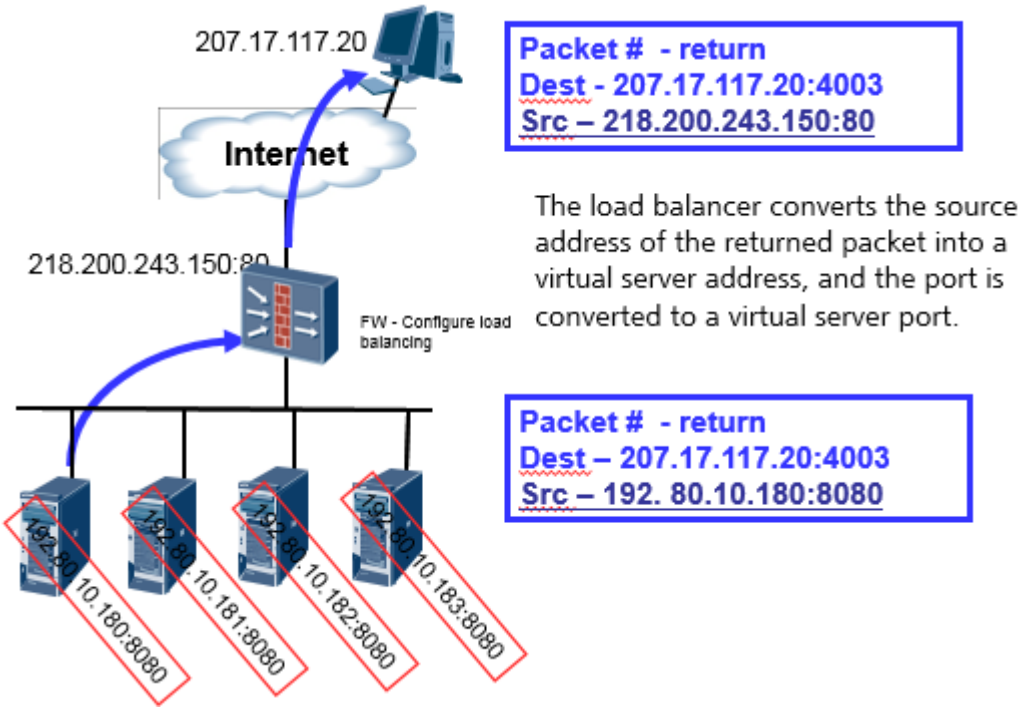
As shown in the following figure, the VIP of the virtual server is 218.200.243.150, the VPORT is 80, and a request packet from the client at 207.17.117.20 is received. The LB module selects a background physical server, such as the one at 192.80.10.180, based on the configured LB algorithm. The FW replaces the destination address of the received packet with the address of the physical server and replaces the port of the received packet with the port of the physical server.

Figure 3-1 Layer 4-based SLB (inbound)



After the FW receives the response packet from the physical server, the LB module replaces the source address of the data packet with the address of the virtual server and replaces the port of the data packet with the port of the virtual server.

Figure 3-2 Layer 4-based SLB (outbound)



The Layer 4-based LB scenario is simple in deployment and has basically the same forwarding performance as the FW.

If the client must interact with the server for multiple times to complete a specific operation or the server stores relevant user service information that may be used the next time, using this server to process all requests of the user is desired. In this scenario, a sticky session mode shall be provided. The source IP address-based mode can be configured to ensure that all packets from 207.17.117.20 are sent to the server at 192.80.10.180.

3.2 Layer 7-based SLB

Layer 4-based SLB cannot obtain application layer packets and therefore cannot implement application layer-based LB. Naturally, it cannot implement URL- and Host-based upper layer LB.

Layer 7-based SLB requires proxy at both the client and server ends.

The virtual server receives the client request and establishes a corresponding TCP connection with the client. Then the virtual server parses the corresponding application layer packet, locates the matched LB policy, and selects a specific background server based on the LB policy. Then the FW establishes a TCP connection with this background server. Layer 7-based SLB enriches user usage scenarios but also proposes higher requirements on the performance.

3.3 HTTP Content-based LB

The physical server group is matched based on the content in the HTTP packet header. Available matching policies are as follows:

- URL: The physical server group is matched based on the resource obtaining path. This policy applies to scenarios where servers provide different resource obtaining paths.
- HOST: The physical server group is matched based on the host name carried in the packet header. This mode applies to scenarios where servers provide different request host names.
- Cookie: The physical server group is matched based on information about a certain field carried in the cookie.
- Referer: The physical server group is matched based on Referer information.

3.4 Sticky Session

- Source IP address-based sticky session

  The source IP address-based sticky session function is usually used in Layer 4 server LB applications to ensure that service packets with the same source IP address can be allocated to the same server.

  After the SLB service of the FW receives the first request of a service initiated by a client, it employs the LB algorithm to select a server to process this request, establish the sticky session entry, and records the information about the server allocated to the client. Within the lifecycle of the entry, subsequent service packets from the same source IP address are always sent to this server for processing.

- HTTP cookie-based sticky session

  The HTTP cookie-based sticky session function is usually used in Layer 7 server LB applications and ensures that packets of the same session are allocated to the same server based on the cookie field information carried in the HTTP request packet from the client. It falls into three modes of cookie insertion, cookie rewriting, and cookie passive.

  Cookie insertion: The response packet of the server does not carry the **Set-Cookie** field that contains the server information. The SLB module adds the **Set-Cookie** field that contains the server information to the packet. In this way, the next request packet of the client carries the cookie field that contains the server information. The SLB module parses the server information in the cookie field and sends the request packet to the corresponding server accordingly.

  Cookie rewriting: The response packet of the server carries an empty **Set-Cookie** field. The SLB module rewrites the value of this cookie field with one that contains the server information. In this way, the next request packet of the client carries the cookie field that contains the server information. The SLB module parses the server information in the cookie field and sends the request packet to the corresponding server accordingly.

  Cookie passive: The response packet of the server carries the **Set-Cookie** field that contains the server information. As for subsequent request packets from the client, the SLB module parses the server information in the cookie field and sends the request packet to the corresponding server accordingly.

  The preceding HTTP cookie, cookie rewriting, and cookie passive modes require settings on the server end so that the **Set-Cookie** field with the corresponding feature can be returned.

- SSL session ID-based sticky session

The SSL session ID-based sticky session function is usually used in Layer 7 server LB applications to ensure that service packets with the same SSL session ID can be allocated to the same server in the SSL access system environment.

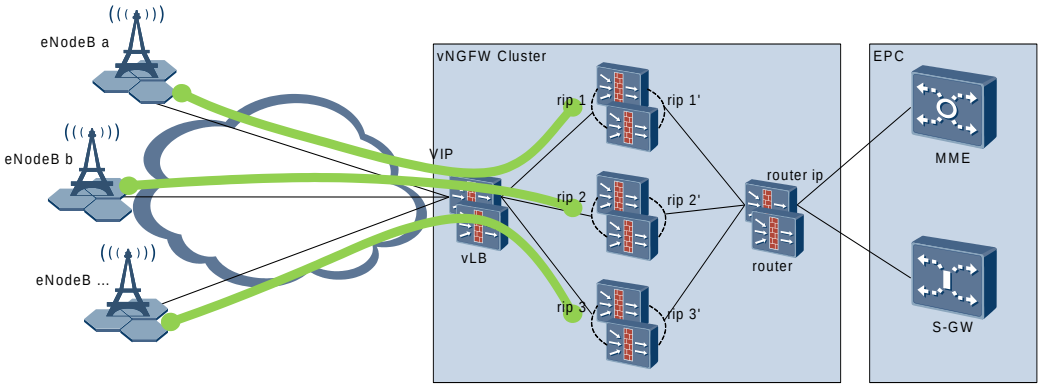
When the client initiates the first SSL access request, the SLB employs the LB algorithm to select a server to process the request, establishes a sticky session entry, and records the information about the server allocated to the client. Within the lifecycle of the entry, subsequent service packets with the same source SSL session ID are always sent to this server for processing.

3.5 IPsec LB

Two vLB devices (LB function of the vNGFW) are deployed to implement vNGFW LB. The eNodeB initiates a tunnel connection towards the VIP of the vLB. The vLB implements NAT between the eNodeB and vSEG (RIP1 to RIP3 in the figure). Actually, the vSEG (RIP1 to RIP3 in the figure) processes IKE negotiation and IPsec encryption/decryption and encapsulation.

The IP address observed by the eNodeB is the VIP on which vLB is not implemented. The management platform can also dynamically perform RIP capacity expansion or deletion by detecting the device operating status under the current RIP.

Figure 3-3 IPsec LB

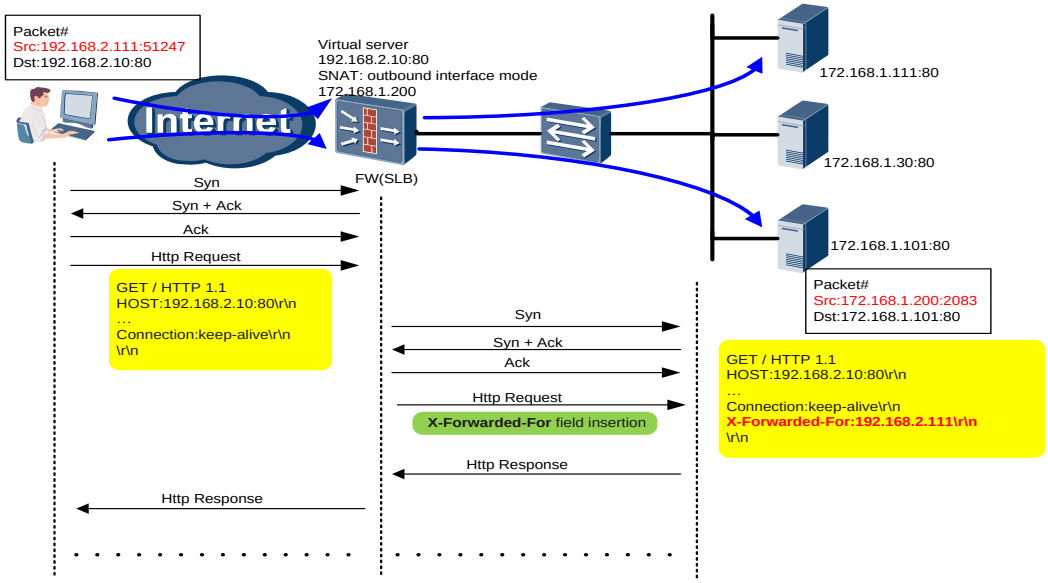


3.6 Source Address-based Sticky Session

As for the Layer 7 SLB module, the provided sticky client IP address refers to the real IP address of the client transferred by inserting the **X-Forwarded-For** field into the header of the HTTP request packet that arrives at the physical server. **X-Forwarded-For: x.x.x.x** is such an example.

The Layer 7 SLB function provides the sticky client IP address. In this case, the SLB serves as a proxy that establishes a TCP connection with both the client and server. After obtaining the HTTP request packet sent from the client, the SLB inserts the **X-Forwarded-For** header information and sends the real client IP address to the server.

Figure 3-4 Source address-based sticky session

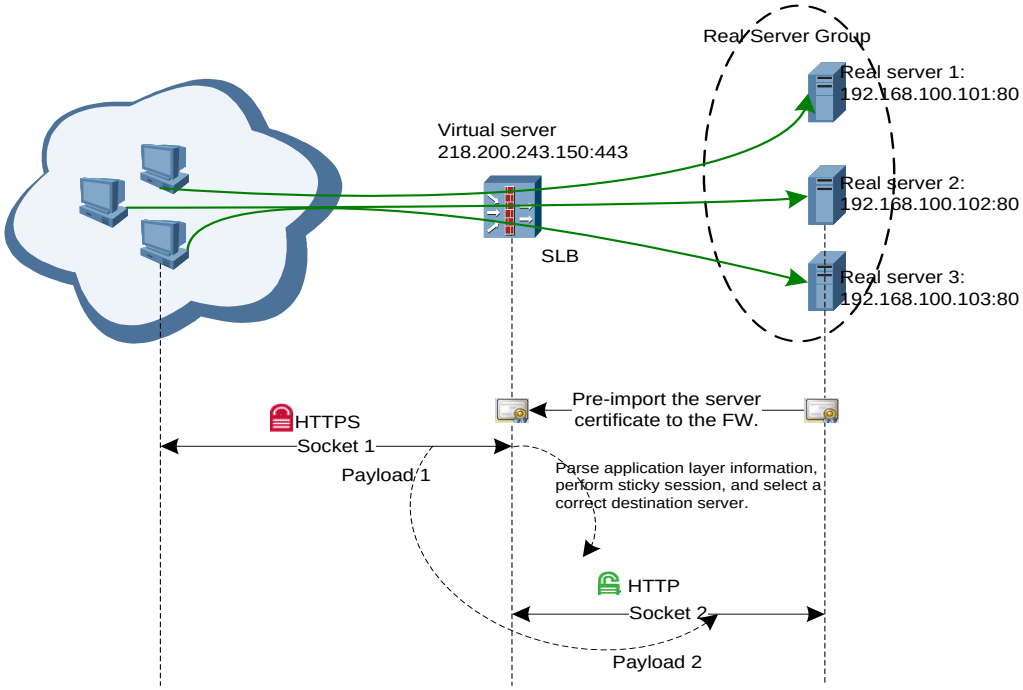


3.7 SSL Uninstallation

In SSL uninstallation, the FW serves as the SSL proxy server and takes over SSL data encryption and decryption. The intranet server can directly read the restored HTTP traffic, without having to install a specific drive program. This helps implement smooth transfer and greatly reduces the load on the intranet server. In addition, for encrypted HTTPS traffic, the FW cannot extract the URL, HOST, or cookie field for more refined traffic scheduling. After the SSL uninstallation function is enabled, HTTPS traffic can be restored to HTTP traffic. The FW can have such LB functions as HTTP scheduling policy and sticky session enabled for more refined traffic scheduling.

As shown in the following figure, SSL uninstallation requires the pre-import of the HTTPS server certificate to the FW. The client sends the HTTPS packet to the FW. The FW uses the pre-imported certificate for SSL handshake authentication with the client, restores HTTPS traffic from the client to HTTP traffic, and sends the restored traffic to the HTTP server in the interval physical server group. After receiving the response packet from the HTTP server, the FW encrypts HTTP traffic to HTTPS traffic and forwards the encrypted traffic to the client to complete the entire interaction process.

Figure 3-5 SSL uninstallation



4 Application Implementation

XX University uses Founder software for its educational management system. This management software is deployed on 20 servers virtualized through VMWare. The address segment of the physical server group is 192.168.0.37 to 192.168.0.56, and the corresponding port is 3906. Considering the performance of the blade server, the upper limit of the processing performance for each server is 1000. Once this upper limit is exceeded, the entire education management system may become down. The LB algorithm employs the LC mode. When a server is overloaded, another server that is currently optimal is selected based on the LB algorithm.

The upper limit on the number of virtual server connections is set to 20 x 1000. After this upper limit is exceeded, no corresponding connections are established.

Considering that students may access the educational management system through the extranet, the access source address may be SNATed. In this case, sticky session in cookie insert mode is selected, and the aging time is set to 300 seconds.