



1.7 Large Load VPN by Clustering

SoftEther VPN Server supports the clustering function, which enables multiple VPN Servers to be administered as a single VPN Server and realize load balancing and fault tolerance among the various VPN Servers.

For a more detailed description of the clustering functions, see [3.9 Clustering](#).

1.7.1 Necessity of Clustering

SoftEther VPN Server is VPN server software that equips with superior performance and functions. Although the power a single computer can exhibit by hardware resource is always limited, the throughput and number of simultaneous connections that can be supported by a single VPN Server that differs according to the hardware performance of the computer running VPN Server. No matter how much the server hardwares are optimized and speeded up, hardware performance limitations ultimately exist, and more processing cannot be executed on a single computer.

Using the clustering function of SoftEther VPN Server enables you to consolidate multiple VPN Servers as a single cluster. VPN connection source computers that attempt to connect to the cluster (usually connection from VPN Client, but there may also be cascade connection from VPN Server / Bridge, etc., in some cases) are automatically connected to one of the VPN Servers in the cluster by cluster controller. At this point, the cluster controller decides the load balancing algorithm by operation mode of connection destination of Virtual Hub.

[1-7-1.png](#)

Processing large amounts of VPN connections by clustering.

If one of the computer operates in the cluster experiences trouble, such as fault and stops running, it will directly connect to other cluster computers by participating in the cluster

and VPN communication processing continues. At this point, it appears that the VPN communication from VPN connection source has stopped instantaneously, but it will be restored right away. Because of the processing the Virtual Hub that had conducted by VPN Server up to that point is executed, communication will continue automatically by avoiding the trouble without the VPN Server administrator or VPN users performing any special processing at all.

In the case where processing cannot be carried out with a single SoftEther VPN Server using these features, large amount of simultaneous connections that decreases throughput dramatically. And it can be processed in parallel by properly balancing the load or in the case a server in the cluster stops, also processing can be taken over by another server, so it can be used effectively in large scale environments or environments demanding high reliability.

[1-7-2.png](#)

Load balancing.

Two types of computers that participate in SoftEther VPN Server clusters: cluster controllers and cluster member servers.

Cluster Controller

A cluster controller is a special computer. Each cluster of server requires only one cluster controller. The cluster controller manages all other computers that participate in the cluster (cluster member servers) and conducts important processing to maintain compatibility among the various servers.

If you are about to construct a cluster of VPN Servers using SoftEther VPN Server, at first, set as the cluster controller one of the server computer and other server computers will be connected to the cluster controller.

Cluster Member Server

All computers which participating in the cluster other than the cluster controller are cluster member servers. Cluster member servers cannot operate on their own, but by executing the cluster control connection to the cluster controller to the SoftEther VPN Server cluster will begin to operate as a single unit based on the connection destination cluster controller.

[1-7-3.png](#)

Cluster controller and cluster member servers.

1.7.2 Applications of Clustering

The cluster function of SoftEther VPN Server operates optimally with primarily two applications: large scale remote access VPN Server and large scale Virtual Hub hosting VPN Server. It will also function correctly, if it is necessary to use the two applications combined.

1.7.3 Large Scale Remote Access VPN Server

The SoftEther VPN Server clustering function can be used when construct a remote access VPN server by using SoftEther VPN Server to connect computers at a remote location to company LAN. Also it is expected to be an extremely large number of

connections or while high reliability is required and you want to shorten stop time as much as possible for remote access VPN server hardware fault, etc.

A cluster of VPN Servers is constructed, a static Virtual Hub is established within for instance of static Virtual Hub generated by VPN server, load balancing is automatically carried out for large quantities of users that attempt remote access to the network and are connected to the proper VPN Server computer in the cluster by bridge connection of network such as physical company LAN. At this time the user does not have to be aware that whom connects to the cluster and no special operation will be required. Also, as a result of load balancing, the same communication can be carried out as when connected to any VPN Server computer. In case hardware fault occurs for the connection destination VPN Server computer or if the server needs to be temporarily shut off or restarted for adding hardware or updating the operating system, that computer is already connected and when reconnected the VPN session is automatically switched to another VPN Server so communication can continue.

This secures scalability and fault tolerance for remote access VPN Server.

Also, if there are multiple physical LANs to be remote accessed, you can create multiple static Virtual Hubs and can connect each respective Virtual Hub to the physical LAN by local bridge connection.

For application examples of the clustering function of large scale remote access VPN servers, see [10.8 Build a Large Scale Remote Access VPN Service](#).

1-7-4.png

Large scale remote access VPN Server.

1.7.4 Large Scale Virtual Hub Hosting VPN Server

You can effectively use the clustering function when using a large amount of Virtual Hubs for hosting with SoftEther VPN Server. The clustering function of SoftEther VPN Server is used if Internet Service Providers or the IT department of large corporations offer Virtual Hub function for customers or users, if there are many Virtual Hubs, or if there are many VPN sessions to be simultaneously connected.

A cluster can be of multiple VPN Servers can be constructed and the exact amount of dynamic Virtual Hubs can be created within it. In the case of such a configuration, if VPN Client or VPN Bridge in a remote location connects to VPN Server by VPN connection or cascade connection, the connection destination Virtual Hub creates an instance for one of the VPN Servers operating in the cluster and communication within that Virtual Hub is possible. Load is automatically balanced for Virtual Hub or VPN connection session for the Virtual Hub. At this time the user does not have to be aware that he is connected to the cluster and no special operation is required. In case hardware fault occurs for the connection destination VPN Server computer or if the server needs to be temporarily shut off or restarted for adding hardware or updating the operating system, that computer is already connected and when reconnected the VPN session is automatically switched to another VPN Server so communication can continue (at this time, Virtual Hub instance is also automatically switched to another server). Just as with a conventional Virtual Hub, because no communication at all is carried out among Virtual Hubs individually, independence of Virtual Hubs is maintained. Also, administrator authority for each Virtual Hub can be transferred to the customer or user.

For application examples of the clustering function of large scale Virtual Hub hosting VPN servers, see [10.9 Build a Large Scale Virtual Hub Hosting Service](#).

1-7-5.png

Large scale Virtual Hub hosting VPN Server.

See Also

- [3.9 Clustering](#)
- [10.8 Build a Large Scale Remote Access VPN Service](#)
- [10.9 Build a Large Scale Virtual Hub Hosting Service](#)