6.3 VPN Server / VPN Bridge Management Command Reference (For Entire Server)

This section describes the commands for configuring and managing the entire VPN Server from among the commands that can be called when using vpncmd in VPN Server or VPN Bridge management mode. For information about the commands for configuring and managing a Virtual Hub selected with the **Hub** command, please refer to 6.4 VPN Server / VPN Bridge Management Command Reference (For Virtual Hub).

6.3.1 "About": Display the version information

Command Name	**About**
Purpose	Display the version information
Description	This displays the version information of this command line management utility. Included in the version information are the vpncmd version number, build number and build information.
Command-line	*About*
Arguments for "About":	
No arguments are required.	

6.3.2 "ServerInfoGet": Get server information

Command Name	**ServerInfoGet**
Purpose	Get server information
Description	This allows you to obtain the server information of the currently connected VPN Server or VPN Bridge. Included in the server information are the version number, build number and build information. You can also obtain information on the current server operation mode and the information of operating system that the server is operating on.
Command-line	*ServerInfoGet*
Arguments for "ServerInfoGet":	
No arguments are required.	

6.3.3 "ServerStatusGet": Get Current Server Status

Command Name	**ServerStatusGet**
Purpose	Get Current Server Status
Description	This allows you to obtain in real-time the current status of the currently connected VPN Server or VPN Bridge. You can get statistical information on data communication and the number of different kinds of objects that exist on the server. You can get information on how much memory is being used on the current computer by the OS.
Command-line	*ServerStatusGet*
Arguments for "ServerStatusGet":	
No arguments are required.	

6.3.4 "ListenerCreate": Create New TCP Listener

Command Name	**ListenerCreate**
Purpose	Create New TCP Listener
Description	This allows you to create a new TCP Listener on the server. By creating the TCP Listener the server starts listening for a connection from clients at the specified TCP/IP port number. A TCP Listener that has been created can be deleted by the ListenerDelete command. You can also get a list of TCP Listeners currently registered by using the ListenerList command. nTo execute this command, you must have VPN Server administrator privileges.
Command-line	*ListenerCreate [port]*
Arguments for "ListenerCreate":	
port	Using an integer, specify the newly added TCP/IP listener port number. You can also use a port number that is already being used by a different program; however the VPN Server will not be able to use it until that program ends the use of that port. Specify a port number that is within the range of 1 to 65535.

6.3.5 "ListenerDelete": Delete TCP Listener

Command Name	**ListenerDelete**
Purpose	Delete TCP Listener

Description	This allows you to delete a TCP Listener that's registered on the server. When the TCP Listener is in a state of operation, the listener will automatically be deleted when its operation stops. You can also get a list of TCP Listeners currently registered by using the ListenerList command. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ListenerDelete [port]*
Arguments for "ListenerDelete":	
port	Using an integer, specify the TCP/IP listener port number you want to delete.

6.3.6 "ListenerList": Get List of TCP Listeners

Command Name	**ListenerList**
Purpose	Get List of TCP Listeners
Description	This allows you to get a list of TCP listeners registered on the current server. You can obtain information on whether the various TCP listeners have a status of operating or error. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ListenerList*
Arguments for "ListenerList":	
No arguments are required.	

6.3.7 "ListenerEnable": Begin TCP Listener Operation

Command Name	**ListenerEnable**
Purpose	Begin TCP Listener Operation
Description	This starts the operation of stopped TCP Listeners registered on the current server. You can also get a list of TCP Listeners currently registered by using the ListenerList command. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ListenerEnable [port]*
Arguments for "ListenerEnable":	

port	Using an integer, specify the port number of the TCP/IP listener you want to start.

6.3.8 "ListenerDisable": Stop TCP Listener Operation

Command Name	**ListenerDisable**
Purpose	Stop TCP Listener Operation
Description	This stops the operation of operating TCP Listeners registered on the current server. You can also get a list of TCP Listeners currently registered by using the ListenerList command. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ListenerDisable [port]*
Arguments for "ListenerDisable":	
port	Using an integer, specify the port number of the TCP/IP listener you want to stop.

6.3.9 "ServerPasswordSet": Set VPN Server Administrator Password

Command Name	**ServerPasswordSet**
Purpose	Set VPN Server Administrator Password
Description	This sets the VPN Server administrator password. You can specify the password as a parameter. If the password is not specified, a prompt will be displayed to input the password and password confirmation. If you include the password as a parameter, this password will be displayed momentarily on the screen, which poses a risk. We recommend that whenever possible, avoid specifying this parameter and input the password using the password prompt. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ServerPasswordSet [password]*
Arguments for "ServerPasswordSet":	
password	This specifies a new password setting.

6.3.10 "ClusterSettingGet": Get Clustering Configuration of Current VPN Server

Command Name	**ClusterSettingGet**
Purpose	Get Clustering Configuration of Current VPN Server
Description	You can use this to acquire the clustering configuration of the current VPN Server. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ClusterSettingGet*
Arguments for "ClusterSettingGet":	
No arguments are required.	

6.3.11 "ClusterSettingStandalone": Set VPN Server Type as Standalone

Command Name	**ClusterSettingStandalone**
Purpose	Set VPN Server Type as Standalone
Description	Use this to set the VPN Server type as Standalone Server. Standalone server means a VPN Server that does not belong to any cluster in its current state. When VPN Server is installed, by default it will be in standalone server mode. Unless you have particular plans to configure a cluster, we recommend the VPN Server be operated in standalone mode. To execute this command, you must have VPN Server administrator privileges. Also, when this command is executed, VPN Server will automatically restart. This command cannot be run on VPN Bridge.
Command-line	*ClusterSettingStandalone*
Arguments for "ClusterSettingStandalone":	
No arguments are required.	

6.3.12 "ClusterSettingController": Set VPN Server Type as Cluster Controller

Command Name	**ClusterSettingController**
Purpose	Set VPN Server Type as Cluster Controller

Description	Use this to set the VPN Server type as Cluster Controller. A cluster controller is the central computer of all member servers of a cluster in the case where a clustering environment is made up of multiple VPN Servers. A cluster requires one computer to serve this role. The other cluster member servers that are configured in the same cluster begin operation as a cluster member by connecting to the cluster controller. To execute this command, you must have VPN Server administrator privileges. Also, when this command is executed, VPN Server will automatically restart. This command cannot be run on VPN Bridge.
Command-line	*ClusterSettingController [/WEIGHT:weight] [/ONLY:yes|no]*
Arguments for "ClusterSettingController":	
/WEIGHT	This sets a value for the performance standard ratio of this VPN Server. This is the standard value for when load balancing is performed in the cluster. Normally it is 100. For example, making only one machine 200 while the other members have a status of 100, will regulate that machine to receive twice as many connections as the other members during load balancing. Specify 1 or higher for the value. If this parameter is left unspecified, 100 will be used.
/ONLY	By specifying "yes" here, the VPN Server will operate only as a controller on the cluster and it will always distribute general VPN Client connections to members other than itself. This function is used in high-load environments. If this parameter is left unspecified, "no" will be used.

6.3.13 "ClusterSettingMember": Set VPN Server Type as Cluster Member

Command Name	**ClusterSettingMember**
Purpose	Set VPN Server Type as Cluster Member
Description	Use this to set the VPN Server type as Cluster Member Server. A cluster member server is a member computer belonging to a clustering configuration made up of multiple VPN Servers with another existing cluster controller as the center. Multiple cluster members can be added to the cluster as required. Before setting the VPN Server as a cluster member server, first ask the administrator of the cluster controller to be used for the controller's IP address and port number, the public IP address and public port number (when required) of this VPN Server and the password.

	To execute this command, you must have VPN Server administrator privileges. Also, when this command is executed, VPN Server will automatically restart. This command cannot be run on VPN Bridge.
Command-line	*ClusterSettingMember [server:port] [/IP:ip] [/PORTS:ports] [/PASSWORD:password] [/WEIGHT:weight]*
Arguments for "ClusterSettingMember":	
server:port	Specify the host name or IP address, and port number of the destination cluster controller using the parameter with the format host name:port number.
/IP	Specify the public IP address of this server. If you wish to leave public IP address unspecified, specify it like this: "/IP:none". When a public IP address is not specified, the IP address of the network interface used when connecting to the cluster controller will be automatically used.
/PORTS	Use this to specify the list of public port numbers on this server. The list must have at least one public port number set, and it is also possible to set multiple public port numbers. When specifying multiple port numbers, separate them using a comma such as "/PORTS443,992,8888".
/PASSWORD	Specify the password required to connect to the destination controller. It needs to be the same as an administrator password on the destination controller.
/WEIGHT	This sets a value for the performance standard ratio of this VPN Server. This is the standard value for when load balancing is performed in the cluster. For example, making only one machine 200 while the other members have a status of 100, will regulate that machine to receive twice as many connections as the other members. Specify 1 or higher for the value. If this parameter is left unspecified, 100 will be used.

6.3.14 "ClusterMemberList": Get List of Cluster Members

Command Name	**ClusterMemberList**
Purpose	Get List of Cluster Members
Description	Use this command when the VPN Server is operating as a cluster controller to get a list of the cluster member servers on the same cluster, including the cluster controller itself. For each member, the following information is also listed. Type, Connection Start, Host Name, Points, Number of Session, Number of

	TCP Connections, Number of Operating Virtual Hubs, Using Client Connection License and Using Bridge Connection License. This command cannot be run on VPN Bridge.
Command-line	*ClusterMemberList*
Arguments for "ClusterMemberList":	
No arguments are required.	

6.3.15 "ClusterMemberInfoGet": Get Cluster Member Information

Command Name	**ClusterMemberInfoGet**
Purpose	Get Cluster Member Information
Description	When the VPN Server is operating as a cluster controller, you can get information on cluster member servers on that cluster by specifying the IDs of the member servers. You can get the following information about the specified cluster member server: Server Type, Time Connection was Established, IP Address, Host Name, Points, Public Port List, Number of Operating Virtual Hubs, First Virtual Hub, Number of Sessions and Number of TCP Connections. This command cannot be run on VPN Bridge.
Command-line	*ClusterMemberInfoGet [id]*
Arguments for "ClusterMemberInfoGet":	
id	Specify the ID of the cluster member whose information you want to get. You can obtain the cluster member server ID by using the ClusterMemberList command.

6.3.16 "ClusterMemberCertGet": Get Cluster Member Certificate

Command Name	**ClusterMemberCertGet**
Purpose	Get Cluster Member Certificate
Description	When the VPN Server is operating as a cluster controller, you can get the public X.509 certificate of cluster member servers on that cluster by specifying the IDs of those member servers. You can save the certificate as an X.509 format file. This command cannot be run on VPN Bridge.
Command-line	*ClusterMemberCertGet [id] [/SAVECERT:cert]*
Arguments for "ClusterMemberCertGet":	

id	Specify the ID of the cluster member whose certificate you want to get. You can obtain the cluster member server ID by using the ClusterMemberList command.
/SAVECERT	Specify the file path name to save the certificate you obtained. You can save the certificate in X.509 format.

6.3.17 "ClusterConnectionStatusGet": Get Connection Status to Cluster Controller

Command Name	**ClusterConnectionStatusGet**
Purpose	Get Connection Status to Cluster Controller
Description	Use this command when the VPN Server is operating as a cluster controller to get the status of connection to the cluster controller. You can get the following information: Controller IP Address, Port Number, Connection Status, Connection Start Time, First Connection Established Time, Current Connection Established Time, Number of Connection Attempts, Number of Successful Connections, Number of Failed Connections. This command cannot be run on VPN Bridge.
Command-line	*ClusterConnectionStatusGet*
Arguments for "ClusterConnectionStatusGet":	
No arguments are required.	

6.3.18 "ServerCertGet": Get SSL Certificate of VPN Server

Command Name	**ServerCertGet**
Purpose	Get SSL Certificate of VPN Server
Description	Use this to get the SSL certificate that the VPN Server provides to the connected client. You can save the certificate as an X.509 format file.
Command-line	*ServerCertGet [cert]*
Arguments for "ServerCertGet":	
cert	Specify the file path name to save the certificate you obtained. You can save the certificate in X.509 format.

6.3.19 "ServerKeyGet": Get SSL Certificate Private Key of VPN Server

Command Name	**ServerKeyGet**
Purpose	Get SSL Certificate Private Key of VPN Server
Description	Use this to get the SSL certificate private key that the VPN Server provides to the connected client. You can save the private key as a Base 64 encoded file. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ServerKeyGet [key]*
Arguments for "ServerKeyGet":	
key	Specify the file path name to save the private key you obtained. You can save the private key in a Base 64 encoded format.

6.3.20 "ServerCertSet": Set SSL Certificate and Private Key of VPN Server

Command Name	**ServerCertSet**
Purpose	Set SSL Certificate and Private Key of VPN Server
Description	You can set the SSL certificate that the VPN Server provides to the connected client and the private key for that certificate. The certificate must be in X.509 format and the private key must be Base 64 encoded format. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ServerCertSet [/LOADCERT:cert] [/LOADKEY:key]*
Arguments for "ServerCertSet":	
/LOADCERT	Specify the X.509 format certificate file to use.
/LOADKEY	Specify the Base 64 encoded private key file for the certificate to use.

6.3.21 "ServerCipherGet": Get the Encrypted Algorithm Used for VPN Communication.

Command Name	**ServerCipherGet**
Purpose	Get the Encrypted Algorithm Used for VPN Communication.
Description	Use this to get the current setting of the algorithm used for the electronic signature and encrypted for SSL connection to be used for

	communication between the VPN Server and the connected client and the list of algorithms that can be used on the VPN Server.
Command-line	*ServerCipherGet*
Arguments for "ServerCipherGet":	
No arguments are required.	

6.3.22 "ServerCipherSet": Set the Encrypted Algorithm Used for VPN Communication.

Command Name	**ServerCipherSet**
Purpose	Set the Encrypted Algorithm Used for VPN Communication.
Description	Use this to set the algorithm used for the electronic signature and encrypted for SSL connections to be used for communication between the VPN Server and the connected client. By specifying the algorithm name, the specified algorithm will be used later between the VPN Client and VPN Bridge connected to this server and the data will be encrypted. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ServerCipherSet [name]*
Arguments for "ServerCipherSet":	
name	This specifies the encrypted and electronic signature algorithm to set. You can obtain the list of usable algorithms by using the ServerCipherGet command.

6.3.23 "Debug": Execute a Debug Command

Command Name	**Debug**
Purpose	Execute a Debug Command
Description	Runs a debug command on the running VPN Server / Bridge process. This command should be executed wjen the support staff requests to do so. Misuse of this command might cause a crash of VPN Server / Bridge running.
Command-line	*Debug [id] [/ARG:arg]*
Arguments for "Debug":	
id	Specify a debug command number.

/ARG	Specify a string to pass to the debug command. If a string contains spaces, contains the whole command by " ".

6.3.24 "Crash": Raise a error on the VPN Server / Bridge to terminate the process forcefully.

Command Name	**Crash**
Purpose	Raise a error on the VPN Server / Bridge to terminate the process forcefully.
Description	This command will raise a fatal error (memory access violation) on the VPN Server / Bridge running process in order to crash the process. As the result, VPN Server / Bridge will be terminated and resterted if it is running as a service mode. If the VPN Server is running as a user mode, the process will not automatically restarted. This command is for a situation when the VPN Server / Bridge is under a non-recoverable error or the process is in an infinite loop. This command will disconnect all VPN Sessions on the VPN Server / Bridge. All unsaved settings in the memory of VPN Server / Bridge will be lost. Before run this command, run the Flush command to try to save volatile data to the configuration file. To execute this command, you must have VPN Server / VPN Bridge administrator privileges.
Command-line	*Crash [yes]*
Arguments for "Crash":	
yes	Input "yes" for confirmation.

6.3.25 "Flush": Save All Volatile Data of VPN Server / Bridge to the Configuration File

Command Name	**Flush**
Purpose	Save All Volatile Data of VPN Server / Bridge to the Configuration File
Description	Normally, the VPN Server / VPN Bridge retains the volatile configuration data in memory. It is flushed to the disk as vpn_server.config or vpn_bridge.config periodically. The period is 300 seconds (5 minutes) by default. (The period can be altered by modifying the AutoSaveConfigSpan iten in the configuration file.) The data will be saved on the timing of shutting down normally of

	the VPN Server / Bridge. Execute the Flush command to make the VPN Server / Bridge save the settings to the file immediately. The setting data will be stored on the disk drive of the server computer. Use the Flush command in a situation that you do not have an enough time to shut down the server process normally. To execute this command, you must have VPN Server administrator privileges. To execute this command, you must have VPN Server / VPN Bridge administrator privileges.
Command-line	*Flush*
Arguments for "Flush":	
No arguments are required.	

6.3.26 "KeepEnable": Enable the Keep Alive Internet Connection Function

Command Name	**KeepEnable**
Purpose	Enable the Keep Alive Internet Connection Function
Description	This allows you to enable the Keep Alive Internet Connection Function. By using the Keep Alive Internet Connection Function for network connection environments where connections will automatically be disconnected when there are periods of no communication that are longer than a set period, it is possible to keep alive the Internet connection by sending packets to a nominated server on the Internet at set intervals. You can set a destination host name etc, by using the KeepSet command. To execute this command on a VPN Server or VPN Bridge, you must have administrator privileges.
Command-line	*KeepEnable*
Arguments for "KeepEnable":	
No arguments are required.	

6.3.27 "KeepDisable": Disable the Keep Alive Internet Connection Function

Command Name	**KeepDisable**
Purpose	Disable the Keep Alive Internet Connection Function

Description	This allows you to disable the Keep Alive Internet Connection Function. To execute this command on a VPN Server or VPN Bridge, you must have administrator privileges.
Command-line	*KeepDisable*
Arguments for "KeepDisable":	
No arguments are required.	

6.3.28 "KeepSet": Set the Keep Alive Internet Connection Function

Command Name	**KeepSet**
Purpose	Set the Keep Alive Internet Connection Function
Description	Use this to set the destination host name etc. of the Keep Alive Internet Connection Function. For network connection environments where connections will automatically be disconnected where there are periods of no communication that are longer than a set period, by using the Keep Alive Internet Connection Function, it is possible to keep alive the Internet connection by sending packets to a nominated server on the Internet at set intervals. When using this command, you can specify the following: Host Name, Port Number, Packet Send Interval, and Protocol. Packets sent to keep alive the Internet connection will have random content and personal information that could identify a computer or user is not sent. You can use the KeepEnable command or KeepDisable command to enable/disable the Keep Alive Internet Connection Function. KeepSet does not change the enabled/disabled status. To execute this command on a VPN Server or VPN Bridge, you must have administrator privileges.
Command-line	*KeepSet [/HOST:host:port] [/PROTOCOL:tcp|udp] [/INTERVAL:interval]*
Arguments for "KeepSet":	
/HOST	Specify the host name or IP address, and port number of the destination using the format "host name:port number".
/PROTOCOL	Specify either tcp or udp.
/INTERVAL	Specify, in seconds, the interval between the sending of packets.

6.3.29 "KeepGet": Get the Keep Alive Internet Connection Function

Command Name	**KeepGet**
Purpose	Get the Keep Alive Internet Connection Function
Description	Use this to get the current setting contents of the Keep Alive Internet Connection Function. In addition to the destination's Host Name, Port Number, Packet Send Interval and Protocol, you can obtain the current enabled/disabled status of the Keep Alive Internet Connection Function.
Command-line	*KeepGet*
Arguments for "KeepGet":	
No arguments are required.	

6.3.30 "SyslogEnable": Set syslog Send Function

Command Name	**SyslogEnable**
Purpose	Set syslog Send Function
Description	Use this to set the usage of syslog send function and which syslog server to use.
Command-line	*SyslogEnable [1|2|3] [/HOST:host:port]*
Arguments for "SyslogEnable":	
1|2|3	Specify, using an integer, 1, 2 or 3 for the setting to use the syslog send function. 1: Send server log by syslog. 2: Send server and Virtual Hub security logs by syslog. 3: Send server, Virtual Hub security, and packet logs by syslog.
/HOST	Specify the host name or IP address, and port number of the syslog server using the format [host name:port number]. If the port number is omitted, 514 will be used.

6.3.31 "SyslogDisable": Disable syslog Send Function

Command Name	**SyslogDisable**
Purpose	Disable syslog Send Function
Description	Use this to disable the syslog send function.
Command-line	*SyslogDisable*
Arguments for "SyslogDisable":	
No arguments are required.	

6.3.32 "SyslogGet": Get syslog Send Function

Command Name	**SyslogGet**
Purpose	Get syslog Send Function
Description	This allows you to get the current setting contents of the syslog send function. You can get the usage setting of the syslog function and the host name and port number of the syslog server to use.
Command-line	*SyslogGet*
Arguments for "SyslogGet":	
No arguments are required.	

6.3.33 "ConnectionList": Get List of TCP Connections Connecting to the VPN Server

Command Name	**ConnectionList**
Purpose	Get List of TCP Connections Connecting to the VPN Server
Description	Use this to get a list of TCP/IP connections that are currently connecting to the VPN Server. It does not display the TCP connections that have been established as VPN sessions. To get the list of TCP/IP connections that have been established as VPN sessions, you can use the SessionList command. You can get the following: Connection Name, Connection Source, Connection Start and Type. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ConnectionList*
Arguments for "ConnectionList":	
No arguments are required.	

6.3.34 "ConnectionGet": Get Information of TCP Connections Connecting to the VPN Server

Command Name	**ConnectionGet**
Purpose	Get Information of TCP Connections Connecting to the VPN Server
Description	Use this to get detailed information of a specific TCP/IP connection that is connecting to the VPN Server.

	You can get the following information: Connection Name, Connection Type, Source Hostname, Source IP Address, Source Port Number (TCP), Connection Start, Server Product Name, Server Version, Server Build Number, Client Product Name, Client Version, and Client Build Number. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ConnectionGet [name]*
Arguments for "ConnectionGet":	
name	This allows you to specify the name of the connection whose information you want to get. To get a list of connection names, you can use the ConnectionList command.

6.3.35 "ConnectionDisconnect": Disconnect TCP Connections Connecting to the VPN Server

Command Name	**ConnectionDisconnect**
Purpose	Disconnect TCP Connections Connecting to the VPN Server
Description	Use this to forcefully disconnect specific TCP/IP connections that are connecting to the VPN Server. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ConnectionDisconnect [name]*
Arguments for "ConnectionDisconnect":	
name	Specify the name of the connection to disconnect. To get a list of connection names, you can use the ConnectionList command.

6.3.36 "BridgeDeviceList": Get List of Network Adapters Usable as Local Bridge

Command Name	**BridgeDeviceList**
Purpose	Get List of Network Adapters Usable as Local Bridge
Description	Use this to get a list of Ethernet devices (network adapters) that can be used as a bridge destination device as part of a Local Bridge connection. If possible, network connection name is displayed. You can use a device displayed here by using the BridgeCreate command.

	To execute this command, you must have VPN Server administrator privileges.
Command-line	*BridgeDeviceList*
Arguments for "BridgeDeviceList":	
No arguments are required.	

6.3.37 "BridgeList": Get List of Local Bridge Connection

Command Name	**BridgeList**
Purpose	Get List of Local Bridge Connection
Description	Use this to get a list of the currently defined Local Bridge connections. You can get the Local Bridge connection Virtual Hub name and the bridge destination Ethernet device (network adapter) name or tap device name, as well as the operating status.
Command-line	*BridgeList*
Arguments for "BridgeList":	
No arguments are required.	

6.3.38 "BridgeCreate": Create Local Bridge Connection

Command Name	**BridgeCreate**
Purpose	Create Local Bridge Connection
Description	Use this to create a new Local Bridge connection on the VPN Server. By using a Local Bridge, you can configure a Layer 2 bridge connection between a Virtual Hub operating on this VPN server and a physical Ethernet Device (Network Adapter). You can create a tap device (virtual network interface) on the system and connect a bridge between Virtual Hubs (the tap device is only supported by Linux versions). It is possible to establish a bridge to an operating network adapter of your choice for the bridge destination Ethernet device (network adapter), but in high load environments, we recommend you prepare a network adapter dedicated to serve as a bridge. To execute this command, you must have VPN Server administrator privileges.
Command-line	*BridgeCreate [hubname] [/DEVICE:device_name] [/TAP:yes|no]*
Arguments for "BridgeCreate":	

hubname	Specify the Virtual Hub to create bridge. To get a list of Virtual Hubs, you can use the HubList command. It is not essential that you specify a Virtual Hub that is currently operating. If you specify a Virtual Hub name that is not currently operating or that does not exist, the Local Bridge connection will become enabled when the actual operation of that Virtual Hub begins.
/DEVICE	Specify the bridge destination Ethernet device (network adapter) or tap device name. You can get the list of Ethernet device names by using the BridgeDeviceList command.
/TAP	Specify yes if you are using a tap device rather than a network adapter for the bridge destination (only supported for Linux versions). When this is omitted, it will be treated the same as when no is specified.

6.3.39 "BridgeDelete": Delete Local Bridge Connection

Command Name	**BridgeDelete**
Purpose	Delete Local Bridge Connection
Description	Use this to delete an existing Local Bridge connection. To get a list of current Local Bridge connections use the BridgeDeviceList command. To execute this command, you must have VPN Server administrator privileges.
Command-line	*BridgeDelete [hubname] [/DEVICE:device_name]*
Arguments for "BridgeDelete":	
hubname	Specify the Virtual Hub of the Local Bridge to delete.
/DEVICE	Specify the device name (network adapter or tap device name) of the Local Bridge to delete.

6.3.40 "Caps": Get List of Server Functions/Capability

Command Name	**Caps**
Purpose	Get List of Server Functions/Capability
Description	Use this get a list of functions and capability of the VPN Server currently connected and being managed. The function and capability of VPN Servers are different depending on the operating VPN server's edition and version. Sometimes commands may be included in the command line management utility that cannot operate because of the function and capability of the

	destination VPN Server. Using this command, you can find out the capability of the target VPN Server and report it. If the version of the VPN Server is newer than the command line management utility and there are functions that the command line management utility does not recognize, you can display the contents strings (variable names) as they are.
Command-line	*Caps*
Arguments for "Caps":	
No arguments are required.	

6.3.41 "Reboot": Reboot VPN Server Service

Command Name	**Reboot**
Purpose	Reboot VPN Server Service
Description	Use this to restart the VPN Server service. When you restart the VPN Server, all currently connected sessions and TCP connections will be disconnected and no new connections will be accepted until the restart process has completed. By using this command, only the VPN Server service program will be restarted and the physical computer that VPN Server is operating on does not restart. This management session will also be disconnected, so you will need to reconnect to continue management. Also, by specifying the /RESTCONFIG:yes parameter, the contents of the configuration file (.config) held by the current VPN Server will be initialized. To execute this command, you must have VPN Server administrator privileges.
Command-line	*Reboot [/RESETCONFIG:yes|no]*
Arguments for "Reboot":	
/RESETCONFIG	By specifying yes, the contents of the configuration file (.config) held by the current VPN Server will be initialized. Please carefully consider the implications when setting this parameter.

6.3.42 "ConfigGet": Get the current configuration of the VPN Server

Command Name	**ConfigGet**
Purpose	Get the current configuration of the VPN Server
Description	Use this to get a text file (.config file) that contains the current configuration contents of the VPN server. You can get the status on

	the VPN Server at the instant this command is executed. When part of the contents of the configuration file does not specify a parameter, it will be displayed on screen as it is. By specifying a save destination file name by parameter, the contents will be saved by that file name. You can edit the configuration file by using a regular text editor. To write an edited configuration to the VPN Server, use the ConfigSet command. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ConfigGet [path]*
Arguments for "ConfigGet":	
path	When you want to save the contents of the configuration file to a file, use this to specify the file name. If left unspecified, the configuration contents will be displayed on screen. If the configuration file contains multiple-byte characters, the encoding must be saved as Unicode (UTF-8).

6.3.43 "ConfigSet": Write Configuration File to VPN Server

Command Name	**ConfigSet**
Purpose	Write Configuration File to VPN Server
Description	Use this to write the configuration file to the VPN Server. By executing this command, the contents of the specified configuration file will be applied to the VPN Server and the VPN Server program will automatically restart and upon restart, operate according to the new configuration contents. Because it is difficult for an administrator to write all the contents of a configuration file, we recommend you use the ConfigGet command to get the current contents of the VPN Server configuration and save it to file. You can then edit these contents in a regular text editor and then use the ConfigSet command to rewrite the contents to the VPN Server. This command is for people with a detailed knowledge of the VPN Server and if an incorrectly configured configuration file is written to the VPN Server, it not only could cause errors, it could also result in the lost of the current setting data. Take special care when carrying out this action. To execute this command, you must have VPN Server administrator privileges.
Command-line	*ConfigSet [path]*

Arguments for "ConfigSet":	
path	Specify the file name of the write destination configuration file. If the write destination file contains multiple-byte characters, the encoding must be Unicode (UTF-8).

6.3.44 "RouterList": Get List of Virtual Layer 3 Switches

Command Name	RouterList
Purpose	Get List of Virtual Layer 3 Switches
Description	Use this to get the list of Virtual Layer 3 Switches defined on the VPN Server. You can get the following information on the Virtual Layer 3 Switches: Switch Name, Operating Status, Number of Interfaces, and Number of Routing Tables. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge.
Command-line	*RouterList*
Arguments for "RouterList":	
No arguments are required.	

6.3.45 "RouterAdd": Define New Virtual Layer 3 Switch

Command Name	RouterAdd
Purpose	Define New Virtual Layer 3 Switch
Description	Use this to define a new Virtual Layer 3 Switch on the VPN Server. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge. [Explanation on Virtual Layer 3 Switch Function] You can define Virtual Layer 3 Switches between multiple Virtual Hubs operating on this VPN Server and configure routing between different IP networks. [Caution about the Virtual Layer 3 Switch Function] The Virtual Layer 3 Switch functions are provided for network administrators and other people who know a lot about networks and IP routing. If you are using the regular VPN functions, you do not need to use the Virtual Layer 3 Switch functions.

	If the Virtual Layer 3 Switch functions are to be used, the person who configures them must have sufficient knowledge of IP routing and be perfectly capable of not impacting the network.
Command-line	*RouterAdd [name]*
Arguments for "RouterAdd":	
name	Use this to specify the name of the newly created Virtual Layer 3 Switch name. You cannot add a name that is identical to an existing Virtual Layer 3 Switch.

6.3.46 "RouterDelete": Delete Virtual Layer 3 Switch

Command Name	**RouterDelete**
Purpose	Delete Virtual Layer 3 Switch
Description	Use this to delete an existing Virtual Layer 3 Switch that is defined on the VPN Server. When the specified Virtual Layer 3 Switch is operating, it will be automatically deleted after operation stops. To get a list of existing Virtual Layer 3 Switches, use the RouterList command. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge.
Command-line	*RouterDelete [name]*
Arguments for "RouterDelete":	
name	Use this to specify the name of the Virtual Layer 3 Switch to be deleted.

6.3.47 "RouterStart": Start Virtual Layer 3 Switch Operation

Command Name	**RouterStart**
Purpose	Start Virtual Layer 3 Switch Operation
Description	Use this to start the operation of an existing Virtual Layer 3 Switch defined on the VPN Server whose operation is currently stopped. To get a list of existing Virtual Layer 3 Switches, use the RouterList command. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge. [Explanation on Virtual Layer 3 Switch Function]

	You can define Virtual Layer 3 Switches between multiple Virtual Hubs operating on this VPN Server and configure routing between different IP networks. [Caution about the Virtual Layer 3 Switch Function] The Virtual Layer 3 Switch functions are provided for network administrators and other people who know a lot about networks and IP routing. If you are using the regular VPN functions, you do not need to use the Virtual Layer 3 Switch functions. If the Virtual Layer 3 Switch functions are to be used, the person who configures them must have sufficient knowledge of IP routing and be perfectly capable of not impacting the network.
Command-line	*RouterStart [name]*
Arguments for "RouterStart":	
name	Use this to specify the name of the Virtual Layer 3 Switch to start.

6.3.48 "RouterStop": Stop Virtual Layer 3 Switch Operation

Command Name	**RouterStop**
Purpose	Stop Virtual Layer 3 Switch Operation
Description	Use this to stop the operation of an existing Virtual Layer 3 Switch defined on the VPN Server whose operation is currently operating. To get a list of existing Virtual Layer 3 Switches, use the RouterList command. To execute this command, you must have VPN Server administrator privileges.
Command-line	*RouterStop [name]*
Arguments for "RouterStop":	
name	Use this to specify the name of the Virtual Layer 3 Switch to stop.

6.3.49 "RouterIfList": Get List of Interfaces Registered on the Virtual Layer 3 Switch

Command Name	**RouterIfList**
Purpose	Get List of Interfaces Registered on the Virtual Layer 3 Switch
Description	Use this to get a list of virtual interfaces when virtual interfaces have been defined on a specified Virtual Layer 3 Switch. You can define multiple virtual interfaces and routing tables for a single Virtual Layer 3 Switch.

	A virtual interface is associated to a virtual Hub and operates as a single IP host on the Virtual Hub when that Virtual Hub is operating. When multiple virtual interfaces that respectively belong to a different IP network of a different Virtual Hub are defined, IP routing will be automatically performed between these interfaces. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge.
Command-line	*RouterIfList [name]*
Arguments for "RouterIfList":	
name	Use this to specify the name of the Virtual Layer 3 Switch.

6.3.50 "RouterIfAdd": Add Virtual Interface to Virtual Layer 3 Switch

Command Name	**RouterIfAdd**
Purpose	Add Virtual Interface to Virtual Layer 3 Switch
Description	Use this to add to a specified Virtual Layer 3 Switch, a virtual interface that connects to a Virtual Hub operating on the same VPN Server. You can define multiple virtual interfaces and routing tables for a single Virtual Layer 3 Switch. A virtual interface is associated to a virtual Hub and operates as a single IP host on the Virtual Hub when that Virtual Hub is operating. When multiple virtual interfaces that respectively belong to a different IP network of a different Virtual Hub are defined, IP routing will be automatically performed between these interfaces. You must define the IP network space that the virtual interface belongs to and the IP address of the interface itself. Also, you must specify the name of the Virtual Hub that the interface will connect to. You can specify a Virtual Hub that currently doesn't exist for the Virtual Hub name. The virtual interface must have one IP address in the Virtual Hub. You also must specify the subnet mask of an IP network that the IP address belongs to. Routing via the Virtual Layer 3 Switches of IP spaces of multiple virtual Hubs operates based on the IP address specified here. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge. To execute this command, the target Virtual Layer 3 Switch must be

	stopped. If it is not stopped, first use the RouterStop command to stop it and then execute this command.
Command-line	*RouterIfAdd [name] [/HUB:hub] [/IP:ip/mask]*
Arguments for "RouterIfAdd":	
name	Use this to specify the name of the Virtual Layer 3 Switch.
/HUB	Use this to specify the name of the Virtual Hub to be the connection destination of the virtual interface to be newly added. To get a list of Virtual Hubs, you can use the HubList command. It is not essential that you specify a Virtual Hub that is currently operating. If you specify a Virtual Hub name that is not currently operating or that does not exist, the Virtual Layer 3 Switch will become enabled when the actual operation of that Virtual Hub begins.
/IP	Using the format: "IP address/subnet mask", specify the IP address and subnet mask held by the virtual interface to be newly added. Specify the IP address by separating the decimal values using dots such as 192.168.0.1 For the subnet mask, either specify decimal values separated by dots such as 255.255.255.0, or you can specify the bit length from the header using a decimal value such as 24.

6.3.51 "RouterIfDel": Delete Virtual Interface of Virtual Layer 3 Switch

Command Name	**RouterIfDel**
Purpose	Delete Virtual Interface of Virtual Layer 3 Switch
Description	Use this to delete a virtual interface already defined in the specified Virtual Layer 3 Switch. You can get a list of the virtual interfaces currently defined, by using the RouterIfList command. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge. To execute this command, the target Virtual Layer 3 Switch must be stopped. If it is not stopped, first use the RouterStop command to stop it and then execute this command.
Command-line	*RouterIfDel [name] [/HUB:hub]*
Arguments for "RouterIfDel":	
name	Use this to specify the name of the Virtual Layer 3 Switch.
/HUB	Use this to specify the name of the Virtual Hub to be the connection destination of the virtual interface to be deleted.

6.3.52 "RouterTableList": Get List of Routing Tables of Virtual Layer 3 Switch

Command Name	**RouterTableList**
Purpose	Get List of Routing Tables of Virtual Layer 3 Switch
Description	Use this to get a list of routing tables when routing tables have been defined on a specified Virtual Layer 3 Switch. If the destination IP address of the IP packet does not belong to any IP network that belongs to a virtual interface, the IP routing engine of the Virtual Layer 3 Switch will reference this routing table and execute routing. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge.
Command-line	*RouterTableList [name]*
Arguments for "RouterTableList":	
name	Use this to specify the name of the Virtual Layer 3 Switch.

6.3.53 "RouterTableAdd": Add Routing Table Entry for Virtual Layer 3 Switch

Command Name	**RouterTableAdd**
Purpose	Add Routing Table Entry for Virtual Layer 3 Switch
Description	Here you can add a new routing table entry to the routing table of the specified Virtual Layer 3 Switch. If the destination IP address of the IP packet does not belong to any IP network that belongs to a virtual interface, the IP routing engine of the Virtual Layer 3 Switch will reference the routing table and execute routing. You must specify the contents of the routing table entry to be added to the Virtual Layer 3 Switch. You must specify any IP address that belongs to the same IP network in the virtual interface of this Virtual Layer 3 Switch as the gateway address. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge. To execute this command, the target Virtual Layer 3 Switch must be stopped. If it is not stopped, first use the RouterStop command to stop it and then execute this command.
Command-line	*RouterTableAdd [name] [/NETWORK:ip/mask] [/GATEWAY:gwip] [/METRIC:metric]*

Arguments for "RouterTableAdd":	
name	Use this to specify the name of the Virtual Layer 3 Switch.
/NETWORK	Using the format: "IP address/subnet mask", specify the network address and subnet mask of the routing table entry to be newly added. Specify the network address by separating the decimal values using dots such as "192.168.0.1". For the subnet mask, either specify decimal values separated by dots such as 255.255.255.0, or you can specify the bit length from the header using a decimal value such as 24. If you specify 0.0.0.0/0.0.0.0, the default route will be used.
/GATEWAY	Specify the gateway IP address.
/METRIC	Specify a metric value. Specify an integer (1 or higher).

6.3.54 "RouterTableDel": Delete Routing Table Entry of Virtual Layer 3 Switch

Command Name	**RouterTableDel**
Purpose	Delete Routing Table Entry of Virtual Layer 3 Switch
Description	Use this to delete a routing table entry that is defined in the specified Virtual Layer 3 Switch. You can get a list of the already defined routing table entries by using the RouterTableList command. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Bridge. To execute this command, the target Virtual Layer 3 Switch must be stopped. If it is not stopped, first use the RouterStop command to stop it and then execute this command.
Command-line	*RouterTableDel [name] [/NETWORK:ip/mask] [/GATEWAY:gwip] [/METRIC:metric]*
Arguments for "RouterTableDel":	
name	Use this to specify the name of the Virtual Layer 3 Switch.
/NETWORK	Using the format: "IP address/subnet mask", specify the network address of the routing table entry to be deleted.
/GATEWAY	Specify the gateway IP address.
/METRIC	Specify a metric value. Specify an integer (1 or higher).

6.3.55 "LogFileList": Get List of Log Files

Command Name	**LogFileList**
Purpose	Get List of Log Files
Description	Use this to display a list of log files outputted by the VPN Server that have been saved on the VPN Server computer. By specifying a log file file name displayed here and calling it using the LogFileGet command you can download the contents of the log file. If you are connected to the VPN Server in server admin mode, you can display or download the packet logs and security logs of all Virtual Hubs and the server log of the VPN Server. When connected in Virtual Hub Admin Mode, you are able to view or download only the packet log and security log of the Virtual Hub that is the target of management.
Command-line	*LogFileList*
Arguments for "LogFileList":	
No arguments are required.	

6.3.56 "LogFileGet": Download Log file

Command Name	**LogFileGet**
Purpose	Download Log file
Description	Use this to download the log file that is saved on the VPN Server computer. To download the log file first display the list of log files using the LogFileList command and then download the log file using the LogFileGet command. If you are connected to the VPN Server in server admin mode, you can display or download the packet logs and security logs of all Virtual Hubs and the server log of the VPN Server. When connected in Virtual Hub Admin Mode, you are able to view or download only the packet log and security log of the Virtual Hub that is the target of management. If you have specified the file name as a parameter, the downloaded log file will be saved to the file of that file name. If the destination file is not specified, the log file will be displayed onscreen. The size of the log file can get very big, so pay careful attention to this issue.
Command-line	*LogFileGet [name] [/SERVER:server] [/SAVEPATH:savepath]*
Arguments for "LogFileGet":	
name	Specify the name of the log file to be downloaded. To get a list of downloadable log files, use the LogFileList command.

/SERVER	Use this to specify the server name when making a download request to a cluster controller. Specify the server that will be displayed by the LogFileGet command.
/SAVEPATH	Use this to specify the destination file name for when saving the downloaded log file. When this is left unspecified, the file will be displayed onscreen.

6.3.57 "HubCreate": Create New Virtual Hub

Command Name	**HubCreate**
Purpose	Create New Virtual Hub
Description	Use this to create a new Virtual Hub on the VPN Server. The created Virtual Hub will begin operation immediately. When the VPN Server is operating on a cluster, this command is only valid for the cluster controller. Also, the new Virtual Hub will operate as a dynamic Virtual Hub. You can change it to a static Virtual Hub by using the HubSetStatic command. To get a list of Virtual Hubs that are already on the VPN Server, use the HubList command. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Servers that are operating as a VPN Bridge or cluster member. When issuing the command to a cluster controller on a cluster to create a Virtual Hub, use either the HubCreateStatic command or the HubCreateDynamic command (issuing the HubCreate command to a cluster controller has the same operational effect as issuing the HubCreateDynamic command).
Command-line	*HubCreate [name] [/PASSWORD:password]*
Arguments for "HubCreate":	
name	Specify the name of the Virtual Hub to create.
/PASSWORD	Specify an administrator password when the administrator password is going to be set for the Virtual Hub to be created. If this is not specified, a prompt will appear to input the password.

6.3.58 "HubCreateDynamic": Create New Dynamic Virtual Hub (For Clustering)

Command Name	**HubCreateDynamic**

Purpose	Create New Dynamic Virtual Hub (For Clustering)
Description	Use this to create a new dynamic Virtual Hub on the VPN Server. The created Virtual Hub will begin operation immediately. When the VPN Server is operating on a cluster, this command is only valid for the cluster controller. Also, the new Virtual Hub will operate as a dynamic Virtual Hub. You can change it to a static Virtual Hub by using the HubSetStatic command. To get a list of Virtual Hubs that are already on the VPN Server, use the HubList command. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Servers that are operating as a VPN Bridge, cluster member or standalone server.
Command-line	*HubCreateDynamic [name] [/PASSWORD:password]*
Arguments for "HubCreateDynamic":	
name	Specify the name of the Virtual Hub to create.
/PASSWORD	Specify an administrator password when the administrator password is going to be set for the Virtual Hub to be created. If this is not specified, a prompt will appear to input the password.

6.3.59 "HubCreateStatic": Create New Static Virtual Hub (For Clustering)

Command Name	**HubCreateStatic**
Purpose	Create New Static Virtual Hub (For Clustering)
Description	Use this to create a new static Virtual Hub on the VPN Server. The created Virtual Hub will begin operation immediately. When the VPN Server is operating on a cluster, this command is only valid for the cluster controller. Also, the new Virtual Hub will operate as a dynamic Virtual Hub. You can change it to a static Virtual Hub by using the HubSetStatic command. To get a list of Virtual Hubs that are already on the VPN Server, use the HubList command. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Servers that are operating as a VPN Bridge, cluster member or standalone server.
Command-line	*HubCreateStatic [name] [/PASSWORD:password]*
Arguments for "HubCreateStatic":	
name	Specify the name of the Virtual Hub to create.

/PASSWORD	Specify an administrator password when the administrator password is going to be set for the Virtual Hub to be created. If this is not specified, a prompt will appear to input the password.

6.3.60 "HubDelete": Delete Virtual Hub

Command Name	**HubDelete**
Purpose	Delete Virtual Hub
Description	Use this to delete an existing Virtual Hub on the VPN Server. If you delete the Virtual Hub, all sessions that are currently connected to the Virtual Hub will be disconnected and new sessions will be unable to connect to the Virtual Hub. Also, this will also delete all the Hub settings, user objects, group objects, certificates and Cascade Connections. Once you delete the Virtual Hub, it cannot be recovered. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Servers that are operating as a VPN Bridge or cluster member.
Command-line	*HubDelete [name]*
Arguments for "HubDelete":	
name	Specify the name of the Virtual Hub to delete.

6.3.61 "HubSetStatic": Change Virtual Hub Type to Static Virtual Hub

Command Name	**HubSetStatic**
Purpose	Change Virtual Hub Type to Static Virtual Hub
Description	Use this when a VPN Server is operating on a cluster and you want to change the type of the Virtual Hub to a static Virtual Hub. When the type of the Virtual Hub is changed, all sessions that are currently connected to the Virtual Hub will be disconnected. When there is a Virtual Hub operating as a static virtual Hub, a Virtual Hub with that name will be created on all the cluster member servers. A user who attempts to connect this Virtual Hub will be connected to one of the cluster members hosting this Virtual Hub as determined by an algorithm based on each server's load status. A static Virtual Hub, for example, could be used for a remote access VPN that allows thousands or tens of thousands of users to connect at the same time for the purpose of remotely accessing an internal

	company LAN from the Internet for business. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Servers that are operating as a VPN Bridge, cluster member or standalone server. This command cannot be used for VPN Servers that are newer than Build 5190.
Command-line	*HubSetStatic [name]*
Arguments for "HubSetStatic":	
name	Specify the name of the Virtual Hub to be set as the static Virtual Hub.

6.3.62 "HubSetDynamic": Change Virtual Hub Type to Dynamic Virtual Hub

Command Name	**HubSetDynamic**
Purpose	Change Virtual Hub Type to Dynamic Virtual Hub
Description	Use this when a VPN Server is operating on a cluster and you want to change the type of the Virtual Hub to a dynamic Virtual Hub. When the type of the Virtual Hub is changed, all sessions that are currently connected to the Virtual Hub will be disconnected. When there is not even one client connected to a dynamic Virtual Hub defined on the cluster, then that Virtual Hub does not exist on any cluster member. When the first client to attempt to connect to the dynamic Virtual Hub does so, the server with the lowest load on the cluster starts hosting that Virtual Hub. When the second and subsequent clients attempt to connect to the same virtual Hub, they are automatically connected to the server hosting the Virtual Hub. When all the clients are disconnected from a particular dynamic Virtual Hub, the Virtual Hub will return to the original state of not existing on any of the servers. There is a broad range of applications for dynamic Virtual Hubs, such as a Virtual Hub defined for each business section within a company so that employees can connect to the Virtual Hub of their own department to do their work in a centralized management environment that is deployed on a single cluster. To execute this command, you must have VPN Server administrator privileges. Also, this command does not operate on VPN Servers that are operating as a VPN Bridge, cluster member or standalone server.

	This command cannot be used for VPN Servers that are newer than Build 5190.
Command-line	*HubSetDynamic [name]*
Arguments for "HubSetDynamic":	
name	Specify the name of the Virtual Hub to be set as the dynamic Virtual Hub.

6.3.63 "HubList": Get List of Virtual Hubs

Command Name	**HubList**
Purpose	Get List of Virtual Hubs
Description	Use this to get a list of existing Virtual Hubs on the VPN Server. For each Virtual Hub, you can get the following information: Virtual Hub Name, Status, Type, Number of Users, Number of Groups, Number of Sessions, Number of MAC Tables, Number of IP Tables, Number of Logins, Last Login, and Last Communication. Note that when connecting in Virtual Hub Admin Mode, if in the options of a Virtual Hub that you do not have administrator privileges for, the option Don't Enumerate this Virtual Hub for Anonymous Users is enabled then that Virtual Hub will not be enumerated. If you are connected in Server Admin Mode, then the list of all Virtual Hubs will be displayed. When connecting to and managing a non-cluster-controller cluster member of a clustering environment, only the Virtual Hub currently being hosted by that VPN Server will be displayed. When connecting to a cluster controller for administration purposes, all the Virtual Hubs will be displayed.
Command-line	*HubList*
Arguments for "HubList":	
No arguments are required.	

6.3.64 "Hub": Select Virtual Hub to Manage

Command Name	**Hub**
Purpose	Select Virtual Hub to Manage
Description	Use this to select the Virtual Hub to be the target of administration. For an administration utility with the status of being connected to a VPN Server, before executing a command to set or manage a Virtual Hub, you must use the Hub command to select the Virtual Hub to

	manage. When in the status of being connected to a VPN Server in Virtual Hub Admin Mode, you can select a single Virtual Hub to be the target of administration but you cannot select other Virtual Hubs. When having the status of being connected to the VPN Server in Server Admin Mode, you can make all Virtual Hubs the target of administration. To get a list of Virtual Hubs that currently exist on the VPN Server, use the HubList command. For the VPN Bridge, you can only select the Virtual Hub that has the name "BRIDGE".
Command-line	*Hub [name]*
Arguments for "Hub":	
name	Specify the name of the Virtual Hub to manage. If this parameter is left unspecified, the Select Virtual Hub to Manage will be cancelled.

6.3.65 "MakeCert": Create New X.509 Certificate and Private Key

Command Name	**MakeCert**
Purpose	Create New X.509 Certificate and Private Key
Description	Use this to create a new X.509 certificate and private key and save it as a file. The algorithm used to create the public key and private key of the certificate is RSA 1024 bit. You can choose to create a root certificate (self-signed certificate) or a certificate signed by another certificate. To create a certificate that is signed by another certificate, you require a private key file (base 64 encoded) that is compatible with the certificate that uses the signature (X.509 format file). When creating a certificate, you can specify the following: Name (CN), Organization (O), Organization Unit (OU), Country (C), State (ST), Locale (L), Serial Number, and Expiration Date. The created certificate will be saved as an X.509 format file and the private key file will be saved in a Base 64 encoded RSA 1024 bit format file. The MakeCert command is a tool that provides the most rudimentary function for creating certificates. If you want to create a more substantial certificate, we recommend that you use either free software such as OpenSSL, or commercial CA (certificate authority) software.

	Note: This command can be called from the SoftEther VPN Command Line Management Utility. You can also execute this command while connected to the current VPN Server or VPN Client in Administration Mode but, what actually performs the RSA computation, generates the certificate data and saves it to file is the computer on which the command is running, and all this is executed in a context that has absolutely no relationship to the computer that is the destination of the Administration Mode connection.
Command-line	*MakeCert [/CN:cn] [/O:o] [/OU:ou] [/C:c] [/ST:st] [/L:l] [/SERIAL:serial] [/EXPIRES:expires] [/SIGNCERT:signcert] [/SIGNKEY:signkey] [/SAVECERT:savecert] [/SAVEKEY:savekey]*
Arguments for "MakeCert":	
/CN	Specify the Name (CN) item of the certificate to create. You can specify "none".
/O	Specify the Organization (O) item of the certificate to create. You can specify "none".
/OU	Specify the Organization Unit (OU) item of the certificate to create. You can specify "none".
/C	Specify the Country (C) item of the certificate to create. You can specify "none".
/ST	Specify the State (ST) item of the certificate to create. You can specify "none".
/L	Specify the Locale (L) item of the certificate to create. You can specify "none".
/SERIAL	Specify the Serial Number item of the certificate to create. Specify using hexadecimal values. You can specify "none".
/EXPIRES	Specify the Expiration Date item of the certificate to create. If you specify "none" or "0", 3650 days (approx. 10 years) will be used. You can specify a maximum of 10950 days (about 30 years).
/SIGNCERT	For cases when the certificate to be created is signed by an existing certificate, specify the X.509 format certificate file name to be used to sign the signature. When this parameter is omitted, such signature signing is not performed and the new certificate is created as a root certificate.
/SIGNKEY	Specify a private key (RSA, base-64 encoded) that is compatible with the certificate specified by /SIGNCERT.
/SAVECERT	Specify the file name to save the certificate you created. The certificate is saved as an X.509 file that includes a public key that is RSA format 1024 bit.
/SAVEKEY	Specify the file name to save private key that is compatible with the certificate you created. The private key will be saved as an RSA-format 1024-bit private key file.

6.3.66 "TrafficClient": Run Network Traffic Speed Test Tool in Client Mode

Command Name	TrafficClient
Purpose	Run Network Traffic Speed Test Tool in Client Mode
Description	Use this to execute the communication throughput measurement tool's client program. Two commands, TrafficClient and TrafficServer, are used for the communication throughput measurement tool to enable the measurement of communication throughput that can be transferred between two computers connected by IP network. The TrafficServer command is used first on another computer which puts the communication throughput measurement tool server in a listening condition. Then the TrafficClient command is used to connect to that server by specifying its host name or IP address and port number, which makes it possible to measure the communication speed. Measurement of the communication speed is carried out by concurrently establishing multiple TCP connections and calculating the actual number of bits of data that can be transferred within a specified time based on the respective results of transferring the maximum stream data on each connection and then using that to calculate the average value (bps) of communication throughput. Normally when there is one TCP connection, it is common to only be able to achieve communication speeds slower than the actual net throughput because of limitations related to the TCP algorithm. We therefore recommend the establishment of multiple concurrent TCP connections when measuring communication results. Because the throughput that is measured using this measurement method is calculated from the bit length of the data that arrives on the receiver side as a stream by TCP, the packet loss that occurs during transfer and the packets with corrupted data are not included in the packets that actually arrive, which means it is possible to calculate a genuine value that is close to the maximum possible communication bandwidth of the network. Using the measurement results, i.e. the stream size transferred by TCP, the approximate value of data volume that actually passed through the network is calculated and this is divided by time to calculate the bits per sec (bps). The calculation assumes the type of the physical network is Ethernet (IEEE802.3) and the MAC frame payload size is 1,500 bytes (TCP MSS is 1,460 bytes). By specifying the /RAW option, the calculation will not make corrections for the TCP/IP header and MAC header data volume.

	Note: This command can be called from the SoftEther VPN Command Line Management Utility. You can also execute this command while connected to the current VPN Server or VPN Client in Administration Mode but, what actually conducts communication and measures the throughput is the computer on which the command is running, and all this is executed in a context that has absolutely no relationship to the computer that is the destination of the Administration Mode connection.
Command-line	*TrafficClient [host:port] [/NUMTCP:numtcp] [/TYPE:download|upload|full] [/SPAN:span] [/DOUBLE:yes|no] [/RAW:yes|no]*
Arguments for "TrafficClient":	
host:port	Specify the host name or IP address and port number that the communication throughput measurement tool server (TrafficServer) is listening for. If the port number is omitted, 9821 will be used.
/NUMTCP	Specify the number of TCP connections to be concurrently established between the client and the server for data transfer. If omitted, 32 will be used.
/TYPE	Specify the direction of data flow when throughput measurement is performed. Specify one of the following options: "download", "upload" or "full". By specifying "download" the data will be transmitted from the server side to the client side. By specifying "upload" the data will be transmitted from the client side to the server side. By specifying "full", the data will be transferred in both directions. When "full" is specified, the NUMTCP value must be an even number of two or more (half the number will be used for concurrent TCP connections in the download direction and the other half will be used in the upload direction). If this parameter is omitted, "full" will be used.
/SPAN	Specify, using seconds, the time span to conduct data transfer for the measurement of throughput. If this parameter is omitted, "15" will be used.
/DOUBLE	When "yes" is specified, the throughput of the measured result will be doubled and then displayed. This option is used for cases when a network device etc. is somewhere on the data route and the total throughput capability that is input and output by this network device is being measured.
/RAW	By specifying "yes", the calculation will not make corrections for the TCP/IP header and MAC header data volume.

6.3.67 "TrafficServer": Run Network Traffic Speed Test Tool in Server Mode

Command Name	**TrafficServer**
Purpose	Run Network Traffic Speed Test Tool in Server Mode
Description	Use this to execute the communication throughput measurement tool's server program. Two commands, TrafficClient and TrafficServer, are used for the communication throughput measurement tool to enable the measurement of communication throughput that can be transferred between two computers connected by IP network. To set the TCP port of this computer to the Listen status to listen for the connection from the TrafficClient of another computer, specify the port number and start the server program using the TrafficServer command. You can display more detailed information on the communication throughput measurement tool by inputting "TrafficClient /?". Note: This command can be called from the SoftEther VPN Command Line Management Utility. You can also execute this command while connected to the current VPN Server or VPN Client in Administration Mode but, what actually conducts communication and measures the throughput is the computer on which the command is running, and all this is executed in a context that has absolutely no relationship to the computer that is the destination of the Administration Mode connection.
Command-line	*TrafficServer [port]*
Arguments for "TrafficServer":	
port	Specify, using an integer, the port number at which to listen for the connection. If the specified port is already being used by another program, or if the port cannot be opened, an error will occur.

6.3.68 "Check": Check whether SoftEther VPN Operation is Possible

Command Name	**Check**
Purpose	Check whether SoftEther VPN Operation is Possible
Description	Use this to check if the current computer that is running vpncmd is a suitable operation platform for SoftEther VPN Server / Bridge. If this check passes on a system, it is highly likely that SoftEther VPN software will operate correctly on that system. Also, if this check does not pass on a system, then this indicates that

	some type of trouble may arise if SoftEther VPN software is used on that system.
Command-line	*Check*
Arguments for "Check":	
No arguments are required.	

6.3.69 "IPsecEnable": Enable or Disable IPsec VPN Server Function

Command Name	**IPsecEnable**
Purpose	Enable or Disable IPsec VPN Server Function
Description	Enable or Disable IPsec VPN Server Function on SoftEther VPN Server. If you enable this function, Virtual Hubs on the VPN Server will be able to accept Remote-Access VPN connections from L2TP-compatible PCs, Mac OS X and Smartphones, and also can accept EtherIP Site-to-Site VPN Connection. VPN Connections from Smartphones suchlike iPhone, iPad and Android, and also from native VPN Clients on Mac OS X and Windows can be accepted. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*IPsecEnable [/L2TP:yes|no] [/L2TPRAW:yes|no] [/ETHERIP:yes|no] [/PSK:pre-shared-key] [/DEFAULTHUB:default_hub]*
Arguments for "IPsecEnable":	
/L2TP	Enable or Disable the L2TP over IPsec Server Function. To accept VPN connections from iPhone, iPad, Android, Windows or Mac OS X, enable this option.
/L2TPRAW	Enable or Disable the L2TP Server Function (Raw L2TP with No Encryptions). To accept special VPN clients, enable this option.
/ETHERIP	Enable or Disable the EtherIP / L2TPv3 over IPsec Server Function (for site-to-site VPN Server function). Router Products which are compatible with EtherIP over IPsec can connect to Virtual Hubs on the VPN Server and establish Layer-2 (Ethernet) Bridging.
/PSK	Specify the IPsec Pre-Shared Key. An IPsec Pre-Shared Key is also called as "PSK" or "secret". Specify it equal or less than 8 letters, and distribute it to every users who will connect to the VPN Server. Please note: Google Android 4.0 has a bug which a Pre-Shared Key

	with 10 or more letters causes a unexpected behavior. For that reason, the letters of a Pre-Shared Key should be 9 or less characters.
/DEFAULTHUB	Specify the default Virtual HUB in a case of omitting the name of HUB on the Username. Users should specify their username such as "Username@Target Virtual HUB Name" to connect this L2TP Server. If the designation of the Virtual Hub is omitted, the above HUB will be used as the target.

6.3.70 "IPsecGet": Get the Current IPsec VPN Server Settings

Command Name	**IPsecGet**
Purpose	Get the Current IPsec VPN Server Settings
Description	Get and view the current IPsec VPN Server settings on the SoftEther VPN Server. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*IPsecGet*
Arguments for "IPsecGet":	
No arguments are required.	

6.3.71 "EtherIpClientAdd": Add New EtherIP / L2TPv3 over IPsec Client Setting to Accept EthreIP / L2TPv3 Client Devices

Command Name	**EtherIpClientAdd**
Purpose	Add New EtherIP / L2TPv3 over IPsec Client Setting to Accept EthreIP / L2TPv3 Client Devices
Description	Add a new setting entry to enable the EtherIP / L2TPv3 over IPsec Server Function to accept client devices. In order to accept connections from routers by the EtherIP / L2TPv3 over IPsec Server Function, you have to define the relation table between an IPsec Phase 1 string which is presented by client devices of EtherIP / L2TPv3 over IPsec compatible router, and the designation of the destination Virtual Hub. After you add a definition entry by EtherIpClientAdd command, the defined connection setting to the Virtual Hub will be applied on the

	login-attepting session from an EtherIP / L2TPv3 over IPsec client device. The username and password in an entry must be registered on the Virtual Hub. An EtherIP / L2TPv3 client will be regarded as it connected the Virtual HUB with the identification of the above user information. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*EtherIpClientAdd [ID] [/HUB:hubname] [/USERNAME:username] [/PASSWORD:password]*
Arguments for "EtherIpClientAdd":	
ID	Specify an ISAKMP Phase 1 ID. The ID must be exactly same as a ID in the configuration of the EtherIP / L2TPv3 Client. You can specify IP address as well as characters as ID, if the EtherIP Client uses IP address as Phase 1 ID. If you specify '*' (asterisk), it will be a wildcard to match any clients which doesn't match other explicit rules.
/HUB	Specify the name of the Virtual Hub to connect.
/USERNAME	Specify the username to login to the destination Virtual Hub.
/PASSWORD	Specify the password to login to the destination Virtual Hub.

6.3.72 "EtherIpClientDelete": Delete an EtherIP / L2TPv3 over IPsec Client Setting

Command Name	**EtherIpClientDelete**
Purpose	Delete an EtherIP / L2TPv3 over IPsec Client Setting
Description	This command deletes an entry to accept VPN clients by EtherIP / L2TPv3 over IPsec Function. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*EtherIpClientDelete [ID]*
Arguments for "EtherIpClientDelete":	
ID	Specify the ISAKMP Phase 1 ID to delete.

6.3.73 "EtherIpClientList": Get the Current List of EtherIP / L2TPv3 Client Device Entry Definitions

Command Name	**EtherIpClientList**
Purpose	Get the Current List of EtherIP / L2TPv3 Client Device Entry Definitions
Description	This command gets and shows the list of entries to accept VPN clients by EtherIP / L2TPv3 over IPsec Function. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*EtherIpClientList*
Arguments for "EtherIpClientList":	
No arguments are required.	

6.3.74 "OpenVpnEnable": Enable / Disable OpenVPN Clone Server Function

Command Name	**OpenVpnEnable**
Purpose	Enable / Disable OpenVPN Clone Server Function
Description	This VPN Server has the clone functions of OpenVPN software products by OpenVPN Technologies, Inc. Any OpenVPN Clients can connect to this VPN Server. The manner to specify a username to connect to the Virtual Hub, and the selection rule of default Hub by using this clone server functions are same to the IPsec Server functions. For details, please see the help of the IPsecEnable command. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*OpenVpnEnable [yes|no] [/PORTS:udp_port_list]*
Arguments for "OpenVpnEnable":	

yes|no	Specify yes to enable the OpenVPN Clone Server Function. Specify no to disable.
/PORTS	Specify UDP ports to listen for OpenVPN. Multiple UDP ports can be specified with splitting by space or comma letters, for example: "1194, 2001, 2010, 2012". The default port for OpenVPN is UDP 1194. You can specify any other UDP ports.

6.3.75 "OpenVpnGet": Get the Current Settings of OpenVPN Clone Server Function

Command Name	**OpenVpnGet**
Purpose	Get the Current Settings of OpenVPN Clone Server Function
Description	Get and show the current settings of OpenVPN Clone Server Function. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*OpenVpnGet*
Arguments for "OpenVpnGet":	
No arguments are required.	

6.3.76 "OpenVpnMakeConfig": Generate a Sample Setting File for OpenVPN Client

Command Name	**OpenVpnMakeConfig**
Purpose	Generate a Sample Setting File for OpenVPN Client
Description	Originally, the OpenVPN Client requires a user to write a very difficult configuration file manually. This tool helps you to make a useful configuration sample. What you need to generate the configuration file for the OpenVPN Client is to run this command. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.

Command-line	*OpenVpnMakeConfig [ZIP_FileName]*
Arguments for "OpenVpnMakeConfig":	
ZIP_FileName	Specify the output setting files to be saved as ZIP compression format. If no file extension is specified, the ".zip" extension will be appended to the filename.

6.3.77 "SstpEnable": Enable / Disable Microsoft SSTP VPN Clone Server Function

Command Name	**SstpEnable**
Purpose	Enable / Disable Microsoft SSTP VPN Clone Server Function
Description	This VPN Server has the clone functions of MS-SSTP VPN Server which is on Windows Server 2008 / 2012 by Microsoft Corporation. Standard MS-SSTP Clients in Windows Vista / 7 / 8 / RT can connect to this VPN Server. [Caution] The value of CN (Common Name) on the SSL certificate of VPN Server must match to the hostname specified on the client, and that certificate must be in the trusted list on the SSTP VPN client. For details refer the Microsoft's documents. You can use the ServerCertRegenerate command to replace the current certificate on the VPN Server to a new self-signed certificate which has the CN (Common Name) value in the fields. In that case, you have to register such a new self-signed certificate on the SSTP VPN Client as a trusted root certificate. If you do not want to do such a bother tasks, please consider to purchase a SSL certificate provided by commercial authority such as VeriSign or GlobalSign. The manner to specify a username to connect to the Virtual Hub, and the selection rule of default Hub by using this clone server functions are same to the IPsec Server functions. For details, please see the help of the IPsecEnable command. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*SstpEnable [yes|no]*
Arguments for "SstpEnable":	

yes|*no*	Specify yes to enable the Microsoft SSTP VPN Clone Server Function. Specify no to disable.

6.3.78 "SstpGet": Get the Current Settings of Microsoft SSTP VPN Clone Server Function

Command Name	**SstpGet**
Purpose	Get the Current Settings of Microsoft SSTP VPN Clone Server Function
Description	Get and show the current settings of Microsoft SSTP VPN Clone Server Function. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*SstpGet*
Arguments for "SstpGet":	
No arguments are required.	

6.3.79 "ServerCertRegenerate": Generate New Self-Signed Certificate with Specified CN (Common Name) and Register on VPN Server

Command Name	**ServerCertRegenerate**
Purpose	Generate New Self-Signed Certificate with Specified CN (Common Name) and Register on VPN Server
Description	You can use this command to replace the current certificate on the VPN Server to a new self-signed certificate which has the CN (Common Name) value in the fields. This command is convenient if you are planning to use Microsoft SSTP VPN Clone Server Function. Because the value of CN (Common Name) on the SSL certificate of VPN Server must match to the hostname specified on the SSTP VPN client. For details please see the help of SstpEnable command. This command will delete the existing SSL certificate of the VPN Server. It is recommended to backup the current SSL certificate and

	private key by using the ServerKeyGet command beforehand. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*ServerCertRegenerate [CN]*
Arguments for "ServerCertRegenerate":	
CN	Specify a Common Name (CN) which the new certificate will have.

6.3.80 "VpnOverIcmpDnsEnable": Enable / Disable the VPN over ICMP / VPN over DNS Server Function

Command Name	**VpnOverIcmpDnsEnable**
Purpose	Enable / Disable the VPN over ICMP / VPN over DNS Server Function
Description	You can establish a VPN only with ICMP or DNS packets even if there is a firewall or routers which blocks TCP/IP communications. You have to enable the following functions beforehand. Warning: Use this function for emergency only. It is helpful when a firewall or router is misconfigured to blocks TCP/IP, but either ICMP or DNS is not blocked. It is not for long-term stable using. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge.
Command-line	*VpnOverIcmpDnsEnable [/ICMP:yes|no] [/DNS:yes|no]*
Arguments for "VpnOverIcmpDnsEnable":	
/ICMP	Specify yes to enable the VPN over ICMP Server. Specify no to disable.
/DNS	Specify yes to enable the VPN over DNS Server. Specify no to disable.

6.3.81 "VpnOverIcmpDnsGet": Get Current Setting of the VPN over ICMP / VPN over DNS Function

Command Name	**VpnOverIcmpDnsGet**

Purpose	Get Current Setting of the VPN over ICMP / VPN over DNS Function
Description	Get and show the current VPN over ICMP / VPN over DNS Function status. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge.
Command-line	*VpnOverIcmpDnsGet*
Arguments for "VpnOverIcmpDnsGet":	
No arguments are required.	

6.3.82 "DynamicDnsGetStatus": Show the Current Status of Dynamic DNS Function

Command Name	**DynamicDnsGetStatus**
Purpose	Show the Current Status of Dynamic DNS Function
Description	Get and show the current status of the Dynamic DNS function. The Dynamic DNS assigns a unique and permanent DNS hostname for this VPN Server. You can use that hostname to specify this VPN Server on the settings for VPN Client and VPN Bridge. You need not to register and keep a domain name. Also, if your ISP assignes you a dynamic (not-fixed) IP address, the corresponding IP address of your Dynamic DNS hostname will be automatically changed. It enables you to keep running the VPN Server by using only a dynamic IP address. Therefore, you need not any longer to keep static global IP addresses with expenses monthly costs. [Caution] To disable the Dynamic DNS Function, modify the configuration file of VPN Server. The "declare root" directive has the "declare DDnsClient" directive. In this directive, you can switch "bool Disable" from false to true, and reboot the VPN Server, then the Dynamic DNS Function will be disabled. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge.
Command-line	*DynamicDnsGetStatus*

Arguments for "DynamicDnsGetStatus":
No arguments are required.

6.3.83 "DynamicDnsSetHostname": Set the Dynamic DNS Hostname

Command Name	DynamicDnsSetHostname
Purpose	Set the Dynamic DNS Hostname
Description	You can use this command to change the hostname assigned by the Dynamic DNS function. The currently assigned hostname can be shown by the DynamicDnsGetStatus command. The Dynamic DNS assigns a unique and permanent DNS hostname for this VPN Server. You can use that hostname to specify this VPN Server on the settings for VPN Client and VPN Bridge. You need not to register and keep a domain name. Also, if your ISP assignes you a dynamic (not-fixed) IP address, the corresponding IP address of your Dynamic DNS hostname will be automatically changed. It enables you to keep running the VPN Server by using only a dynamic IP address. Therefore, you need not any longer to keep static global IP addresses with expenses monthly costs. [Caution] To disable the Dynamic DNS Function, modify the configuration file of VPN Server. The "declare root" directive has the "declare DDnsClient" directive. In this directive, you can switch "bool Disable" from false to true, and reboot the VPN Server, then the Dynamic DNS Function will be disabled. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge.
Command-line	*DynamicDnsSetHostname [hostname]*
Arguments for "DynamicDnsSetHostname":	
hostname	Specify the new hostname with 3 letters at least and 31 letters at most. Only alphabets and numerics can be used.

6.3.84 "VpnAzureGetStatus": Show the current status of VPN Azure function

Command Name	**VpnAzureGetStatus**
Purpose	Show the current status of VPN Azure function
Description	Get and show the current status of the VPN Azure function. VPN Azure makes it easier to establish a VPN Session from your home PC to your office PC. While a VPN connection is established, you can access to any other servers on the private network of your company. You don't need a global IP address on the office PC (VPN Server). It can work behind firewalls or NATs. No network administrator's configuration required. You can use the built-in SSTP-VPN Client of Windows in your home PC. VPN Azure is a cloud VPN service operated by SoftEther Corporation. VPN Azure is free of charge and available to anyone. Visit http://www.vpnazure.net/ to see details and how-to-use instructions. The VPN Azure hostname is same to the hostname of the Dynamic DNS setting, but altering the domain suffix to "vpnazure.net". To change the hostname use the DynamicDnsSetHostname command. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*VpnAzureGetStatus*
Arguments for "VpnAzureGetStatus":	
No arguments are required.	

6.3.85 "VpnAzureSetEnable": Enable / Disable VPN Azure Function

Command Name	**VpnAzureSetEnable**
Purpose	Enable / Disable VPN Azure Function
Description	Enable or disable the VPN Azure function. VPN Azure makes it easier to establish a VPN Session from your home PC to your office PC. While a VPN connection is established,

	you can access to any other servers on the private network of your company. You don't need a global IP address on the office PC (VPN Server). It can work behind firewalls or NATs. No network administrator's configuration required. You can use the built-in SSTP-VPN Client of Windows in your home PC. VPN Azure is a cloud VPN service operated by SoftEther Corporation. VPN Azure is free of charge and available to anyone. Visit http://www.vpnazure.net/ to see details and how-to-use instructions. The VPN Azure hostname is same to the hostname of the Dynamic DNS setting, but altering the domain suffix to "vpnazure.net". To change the hostname use the DynamicDnsSetHostname command. To execute this command, you must have VPN Server administrator privileges. This command cannot be run on VPN Bridge. You cannot execute this command for Virtual Hubs of VPN Servers operating as a cluster.
Command-line	*VpnAzureSetEnable [yes|no]*
Arguments for "VpnAzureSetEnable":	
yes|no	Specify 'yes' to enable VPN Azure. 'no' to disable it.