MultiNet 5.6 Administrator's Reference

November 2020

This guide provides information to configure and manage MultiNet for the experienced system manager. Before using this guide, install and start MultiNet as described in the *MultiNet Installation and Administrator's Guide*.

Operating System/Version: OpenVMS VAX V5.5-2 or later

OpenVMS Alpha V6.2 or later

OpenVMS Itanium V8.2 or later

Software Version: MultiNet 5.6

Process Software
Framingham, Massachusetts
USA

Preface

Obtaining Technical Support

Process Software provides technical support if you have a current Maintenance Service Agreement. If you obtained MultiNet from an authorized distributor or partner, you receive your technical support directly from them.

You can contact Technical Support by sending electronic mail or calling the Technical Support center.

Before Contacting Technical Support

Before you call or send e-mail please verify that your Maintenance Service Agreement is current, and have the following information available:

- Your name
- Your company name
- Your e-mail address
- Your telephone number
- Your Maintenance Agreement Number
- OpenVMS architecture and version
- MultiNet version

Have complete information about your configuration, error messages that appeared, and problem specifics.

Be prepared to let an engineer connect to your system either with TELNET or SSH. Be prepared to give the engineer access to a privileged account to diagnose your problem.

You can obtain information about your OpenVMS architecture, OpenVMS version, and MultiNet version with the `MULTINET SHOW /LICENSE` command. For example:

```
$ MULTINET SHOW /LICENSE
Process Software MultiNet V5.6, VAXstation 4000-90, OpenVMS VAX V7.1
```

In this example:

- The machine or system architecture is VAX.
- The OpenVMS version is V7.1.

- The MultiNet version is V5.6

Sending Electronic Mail

For most questions, electronic mail is the preferred communications method. Technical support via electronic mail is available to customers with a current support contract. Send electronic mail to support@process.com

At the beginning of your mail message, include the information listed in the section *Before Contacting Technical Support*. Continue with the description of your situation and problem specifics. Include all relevant information to help your Technical Support Specialist process and track your electronic support request.

Electronic mail is answered Monday through Friday from 9:00 a.m. to 5:00 p.m. United States Eastern Time.

Calling Technical Support

For regular support issues, call 800-394-8700 or 508-628-5074 for support Monday through Friday from 9:00 a.m. to 5:00 p.m. United States Eastern Time.

For our customers in North America with *critical* problems, an option for support 7 days per week, 24 hours per day is available at an additional charge. Please contact your account representative for further details.

Before calling, have available the information described in the section *Before Contacting Technical Support*. When you call, you will be connected to a Technical Support Specialist.

Be prepared to discuss problem specifics with your Technical Support Specialist and to let that person connect to your system.

If a Specialist is not immediately available, your call will be returned as soon as possible.

Obtaining Online Help

Extensive information about MultiNet is provided in the MultiNet help library. For more information, use the following command:

```
$ HELP MULTINET
```

MultiNet Frequently Asked Questions List

You can obtain an updated list of frequently asked questions (FAQs) and answers about MultiNet products from the Process Software home page located at http://www.process.com/

Accessing the MultiNet Public Mailing List

Process Software maintains two public mailing lists for MultiNet customers.

The Info-MultiNet@lists.process.com mailing list is a forum for discussion among MultiNet system managers and programmers. Questions and problems regarding MultiNet can be posted for a response by any of the subscribers. To subscribe to Info-MultiNet, send a mail message with the word `SUBSCRIBE` in the body to Info-MultiNet@lists.process.com.

The MultiNet-Announce@lists.process.com mailing list is a one-way communication (from Process Software to you) used for the posting of announcements relating to MultiNet (patch releases, product releases, etc.). To subscribe to MultiNet-Announce, send a mail message with the word `SUBSCRIBE` in the body to MultiNet-Announce@lists.process.com.

Obtaining Software Patches Over the Internet

Process Software provides software patches in save set and ZIP format on its anonymous FTP server, ftp.multinet.process.com. For the location of software patches, read the `.WELCOME` file in the top-level anonymous directory. This file refers you to the directories containing software patches.

To retrieve a software patch, enter the following commands:

```
$ MULTINET FTP /USERNAME-ANONYMOUS /PASSWORD=email
FTP.MULTINET.PROCESS.COM
```

A message welcoming you to the Process Software FTP directory appears next followed by the FTP prompt. Enter the following at the prompts:

```
FTP>CD [PATCHES.MULTINETnnn]
FTP>GET update_filename
```

- *emailaddress* is your e-mail address in the standard *user@host* format.
- *nnn* is the version of MultiNet you want to transfer.
- *update_filename* is the name of the file you want to transfer.

To transfer files from Process Software directly to an OpenVMS system, you can use the `GET` command without any other FTP commands. However, if you need to transfer a software patch through an intermediate non-OpenVMS system, use `BINARY` mode to transfer the files to and from that system.

In addition, if you are fetching the software patch in save set format, make sure the save set record size is 2048 bytes when you transfer the file from the intermediate system to your OpenVMS system:

- If you use the `GET` command to download the file from the intermediate system, use the FTP `RECORD-SIZE 2048` command *before* transferring the file.
- If you use the `PUT` command to upload the file to your OpenVMS system, log into the intermediate system and use the FTP `quote site rms recsize 2048` command *before* transferring the file.

The following example shows how to use the `UNZIP` utility, assuming you have copied the appropriate version of `UNZIP.EXE` to your current default directory.

```
$ UNZIP := $SYS$DISK:[]UNZIP.EXE
$ UNZIP filename.ZIP
```

Use `VMSINSTALL` to upgrade your MultiNet system with the software patch.

Conventions Used

Examples in this guide use the following conventions:

Convention	Meaning
host	Any computer system on the network. The local host is your computer. A remote host is any other computer.
`monospaced type`	System output or user input. User input is in **`reversed bold`** `type`. Example: `Is this configuration correct?` **`YES`** Monospaced type also indicates user input where the case of the entry should be preserved.
italic type	Variable value in commands and examples. For example, *username* indicates that you must substitute your actual username. Italic text also identifies documentation references.
[*directory*]	Directory name in an OpenVMS file specification. Include the brackets in the specification.
[*optional-text*]	(Italicized text and square brackets) Enclosed information is optional. Do not include the brackets when entering the information. Example: `START/IP` *`line address`* [*`info`*] This command indicates that the *`info`* parameter is optional.
{value | value}	Denotes that you should use only one of the given values. Do not include the braces or vertical bars when entering the value.

Note	Information that follows is particularly noteworthy.
Caution	Information that follows is critical in preventing a system interruption or security breach.
`key`	Press the specified key on your keyboard.
`Ctrl+key`	Press the control key and the other specified key simultaneously.
`Return`	Press the Return or Enter key on your keyboard.

1. MultiNet DCL Command Reference

This chapter describes the MultiNet administration commands you can run from the DCL prompt.

Command Summary

MultiNet Command Summary describes the MultiNet administrative commands available at the DCL prompt.

Command	Description
`MULTINET ACCOUNTING`	Processes the accounting file that FTP and SMTP can write.
`MULTINET CHECK`	Tests the MultiNet configuration.

`MULTINET CONFIGURE /CONFIGURATION_FILE`
 	
`/DECNET`
 	
`/MAIL`
 	
`/NETWORK`
 	
`/PRINTERS`
 	
`/SERVERS`
	Specifies the configuration file read by the DECNET-, PRINTER-, or SERVER-CONFIG utility.
	
	Invokes the DECnet Circuit Configuration Utility (`DECNET-CONFIG`).
	
	Invokes the Electronic Mail Configuration Utility (`MAIL-CONFIG`).
	
	Invokes the Network Interface Configuration Utility (`NET-CONFIG`).
	
	Invokes the Printer Configuration Utility (`PRINTER-CONFIG`).
	
	Invokes the Server Configuration Utility (`SERVER-CONFIG`).

/SERVER_IMAGE	Specifies the master server image associated with the server configuration file.
MULTINET DIG	Tests the domain name service (DNS) system.
MULTINET DNSKEYGEN	Generates and maintains keys for DNS Security (DNSSEC) within the DNS.
MULTINET DNSSIGNER	Signs zone files for DNS Security (DNSSEC) within the DNS (Domain Name System).
MULTINET FONT COMPILE	Compiles an ASCII BDF (bitmap distribution format) font file into a binary PCF (portable compiled format) file.
MULTINET FONT INFO	Displays font server information.
MULTINET FONT LIST	Lists font names and font information.
MULTINET FONT MKFONTDIR	Creates a DECW$FONT_DIRECTORY.DAT file when adding fonts.
MULTINET FONT SHOW	Displays font data.
MULTINET FONT UNCOMPILE	Uncompiles a PCF file into an ASCII BDF file.
MULTINET GATED/CHECK	Checks the syntax of a GateD configuration file.
MULTINET GATED/DUMP	Tells GateD to dump internal state into a text file.
MULTINET GATED/LOAD	Loads new configuration file.
MULTINET GATED/SET/TRACE	Controls tracing in GateD.
MULTINET GATED/SHOW/OSPF	Queries OSPF routers.
MULTINET GATED/SHOW/RIP	Request all routes known by a RIP gateway.

MULTINET GATED/SHOW/TRACE	Queries tracing in GateD.
MULTINET GATED/STOP	Tells the GateD process to halt in an orderly manner.
MULTINET GATED /TOGGLE_TRACING	Toggles GateD tracing on and off.
MULTINET GATED/UPDATE_ INTERFACES	Tells the GateD process to rescan the network interfaces.
MULTINET HOST_TABLE GET	Retrieves a HOSTS.TXT file.
MULTINET HOST_TABLE INSTALL	Installs host tables as global sections.
MULTINET IPP SHOW	Allows a user to learn the capabilities supported by an IPP server.
MULTINET KERBEROS DATABASE DUMP	Stores the contents of the Kerberos database in an ASCII text file.
MULTINET KERBEROS DATABASE EDIT	Adds principal information to the database.
MULTINET KERBEROS DATABASE INITIALIZE	Initializes the Kerberos database.
MULTINET KERBEROS DATABASE LOAD	Loads the database from an ASCII text file produced by the MULTINET KERBEROS DATABASE DUMP utility.
MULTINET KERBEROS DATABASE NEW_MASTER_KEY	Permits the Kerberos master key to be changed.
MULTINET KERBEROS DATABASE SRVTAB	Creates an encrypted server key file for a remote system.
MULTINET KERBEROS DATABASE STASH	Saves the Kerberos master key in a protected file for the KDC.
MULTINET LOAD	Loads and invokes the network image.

MULTINET NETCONTROL	Sends commands to MULTINET_SERVER internal services.
MULTINET NFSDISMOUNT	Dismounts a locally-mounted remote NFS file system.
MULTINET NFSMOUNT	Mounts a remote NFS file system so it can be used locally.
MULTINET NSLOOKUP	Sends a test query to DNS.
MULTINET NSUPDATE	Performs dynamic updates to the domain name service (DNS) server.
MULTINET PING	Tests connections by sending ICMP echo requests.
MULTINET PING6	Tests connections by sending ICMPv6 echo requests.
MULTINET RDATE	Queries the remote system for time and sets the local clock accordingly.
MULTINET RMTALLOC	Allocates a remote tape drive or CD-ROM for access by a single process.
MULTINET RWALL	Sends a message to all system users.
MULTINET SET /ARP	Changes ARP tables.
MULTINET SET /DECNET	Configures DECnet devices to run DECnet-over-UDP circuits.
MULTINET SET /INTERFACE	Sets parameters for network devices
MULTINET SET /ROUTE	Specifies static IP routing.
MULTINET SET /TIMEZONE	Specifies the local time zone name.
MULTINET TCPDUMP	Decodes network packets selected by a Boolean expression.
MULTINET TCPVIEW	Traces packets and interprets the results.

`MULTINET TRACEROUTE`	Determines the route to the specified host.
`MULTINET TRACEROUTE6`	Determines the route to the specified host for IPv6.
`MULTINET X11DEBUG`	Performs tests on the most common causes of problems when running X11 clients over MultiNet.

MULTINET ACCOUNTING

Processes the accounting file that session accounting writes for SMTP and for FTP. It extracts the selected records from it and either displays it on the user's terminal or sends it to the specified output file.

Format

`$ MULTINET ACCOUNTING /QUALIFIERS`

Qualifiers

/BEFORE=*latest_date_to_include*
/SINCE=*first_date_to_include*
Specify VMS-formatted dates to limit the records displayed based on date.

/CSV
Makes the output file a Comma Separated Values file that can be imported into an Excel-type document for processing.

/INPUT=*accounting_file_name*
/OUTPUT=*output_file_name*

/PROTOCOL=*(MAIL,SMTP,FTP)*
These are the protocols to include.

MULTINET CHECK

Invokes the MultiNet configuration test utility to perform one or more checks for common MultiNet configuration problems. Requires CMKRNL, SYSPRV, and WORLD privileges.

Format

`MULTINET CHECK [test,...]`

Parameter

test

Specifies the name of a test to be performed. Valid test names are ARP, BROADCASTS, DATABASES, HOST_NAME, HOST_TABLE, INTERFACES, LICENSE, MISCELLANEOUS, PARAMETERS, PROTOCOL_ERRORS, ROOT_NAMESERVERS, ROUTES, and VERSION. You can specify multiple tests by separating the names with commas. If you do not specify a test parameter, all tests are performed.

The host name check verifies that the address associated with the local host name matches one of the interface addresses.

Qualifiers

/IGNORE_ERRORS
/NOIGNORE_ERRORS

MULTINET CHECK usually stops when it encounters an error. Specify this qualifier to force MULTINET CHECK to continue testing even after an error is encountered.

/OUTPUT=file-spec
/NOOUTPUT

MULTINET CHECK usually displays all output on the standard error output device. Specify this qualifier to either redirect output to the specified file or turn output off altogether.

/VERBOSE
/NOVERBOSE

Causes MULTINET CHECK to display more information about the tests it performs. By default, it only displays a message when it encounters an error or if all tests pass.

MULTINET CONFIGURE

Invokes one of the MultiNet configuration utilities which are interactive programs that maintain network configuration information. If you do not specify a configuration utility with a qualifier, the network interface configuration utility (NET-CONFIG) is invoked.

Format

MULTINET CONFIGURE *[/qualifier(s)]*

Qualifiers

/CONFIGURATION_FILE=*config_file*

Used with the /DECNET, /PRINTERS, or /SERVERS qualifier, specifies the configuration file read by the corresponding utility.

/DECNET

Invokes the DECnet Configuration Utility (DECNET-CONFIG) that lets you view and alter the configuration of DECnet-over-IP services. If used with the /CONFIGURATION_FILE qualifier, DECNET-CONFIG reads the specified configuration file (by default, MULTINET:DECNET-CIRCUITS.COM).

/INTERFACES

Invokes the Network Interface Configuration Utility (NET-CONFIG) that lets you view and alter the configuration of network interfaces, routing, and host name lookup. If used with the /CONFIGURATION_FILE qualifier, NET-CONFIG reads the specified configuration file (by default, MULTINET:NETWORK_DEVICES.CONFIGURATION).

/MAIL

Invokes the Electronic Mail Configuration Utility (MAIL-CONFIG) that lets you view and alter SMTP configuration. If used with the /CONFIGURATION_FILE qualifier, MAIL-CONFIG reads the specified configuration file (by default, MULTINET_COMMON_ROOT:[MULTINET]START_SMTP.COM).

/NETWORK

Invokes the Network Interface Configuration Utility (NET-CONFIG) that lets you view and alter the configuration of network interfaces, routing, and host name lookup. If used with the /CONFIGURATION_FILE qualifier, NET-CONFIG reads the specified configuration file (by default, MULTINET:NETWORK_DEVICES.CONFIGURATION).

MULTINET CONFGURE /NETWORK now has the command:

SET SNMP-AGENTX TRUE to enable SNMP Agent X service.
SET SNMP-AGENTX FALSE to disable SNMP Agent X service.

A line displays in the output of the SHOW command if SNMP Agent X subagents are enabled.

/NFS
Invokes the NFS server configuration utility (NFS-CONFIG) for the NFS server option.

If used with the /CONFIGURATION_FILE qualifier, NFS-CONFIG reads the specified configuration file (by default, MULTINET:NFS.CONFIGURATION).

/NOT
Invokes the NOT configuration utility NOT-CONFIG for DECnet applications services (formerly known as Phase/IP). DECnet application services allow you to run applications designed to use DECnet using TCP/IP instead. DECnet application services provide the DECnet API (Application Programming Interface) across TCP seamlessly, without DECnet protocols or software, and without the additional overhead of running both protocol stacks.

/PRINTERS
Invokes the MultiNet printer configuration utility (PRINTER-CONFIG) that lets you view and alter the configuration of MultiNet-based print services. If used with the /CONFIGURATION_FILE qualifier, PRINTER-CONFIG reads the specified configuration file (by default, MULTINET:REMOTE-PRINTER-QUEUES.COM).

/SERVERS
Invokes the MultiNet service configuration utility (SERVER-CONFIG) that lets you view and alter the configuration of MultiNet services. If used with the /CONFIGURATION_FILE qualifier, SERVER-CONFIG reads the specified configuration file (by default, MULTINET:SERVICES.MASTER_SERVER).

/SERVER_IMAGE=*server_image_file*

Used with the /SERVERS qualifier, *server_image_file* specifies the MultiNet master server image associated with the server configuration file. This file is used by SERVER-CONFIG to determine which network services are available. If not specified, SERVER-CONFIG uses MULTINET:SERVER.EXE.

MULTINET DIG

Similar to NSLOOKUP, DIG tests the domain name service (DNS) system. It uses the DNS resolver to send queries to the DNS server and prints out the response. DIG executes a single command or reads commands from a file (in "batch mode").

DIG can be used with the UNIX-style syntax by defining it as a foreign command:

```
$ DIG:==$MULTINET:DIG.EXE
```

Both the UNIX-style options and the OpenVMS qualifiers are listed below.

Format

MULTINET DIG [*name* [*type* [*class*]]]

Parameters

name
Specifies a host or domain name.

> **Note:** You must specify fully-qualified names. DIG will not append any domain names.

type
Specifies which TYPE resource records are asked for. The default is A (address records).
Valid values are the same as for the NSLOOKUP /TYPE qualifier.

class
Specifies which CLASS resource records are asked for. The default is IN (internet records).
Valid values are ANY, IN, CHAOS, and HESIOD.

Qualifiers

+[no]addit
/ADDITIONAL
/NOADDITIONAL

Tells the resolver to print the additional section of the reply. /ADDITIONAL is the default.

-x *ip-address*
/ADDRESS=*ip-address*
Convenient form to specify an inverse address mapping query. For example, MULTINET DIG/ADDRESS=10.5.64.1 is equivalent to MULTINET DIG 1.64.5.10.IN-ADDR.ARPA ANY.

+[no]answer
/ANSWER
/NOANSWER
Tells the resolver to print the answer section of the reply. /ANSWER is the default.

+[no]author
/AUTHORITY
/NOAUTHORITY
Tells the resolver to print the authority section of the reply. /AUTHORITY is the default.

-c *recordclass*
/CLASS=*recordclass*
Specifies which CLASS resource records are asked for. Alternative to specifying the class parameter. The *recordclass* value may be either the integer value of the class or the name of the class (ANY, IN, CHAOS, HESIOD). The default is IN (internet records).

+[no]cmd
/CMD
/NOCMD
Tells DIG to echo parsed arguments from the command. /CMD is the default.

+[no]debug
/DEBUG
/NODEBUG
Causes the resolver to print debugging information. /NODEBUG is the default.

+[no]d2
/DEBUG2
/NODEBUG2
Causes the resolver to print additional, less useful debugging information. Off by default.

-envsav
/ENVSAVE
Specifies that the DIG environment (defaults, print options, etc.), after all of the arguments are parsed, should be saved to a file to become the default environment. This is useful if you do not like the standard set of defaults and do not desire to include a large number of options each time DIG is used. The environment consists of resolver state variable flags, timeout, and retries as well as the flags detailing DIG output. If the logical name LOCALDEF is set to the name of a file, this is where the default DIG environment is saved. If not, the file DIG.ENV is created in the current default directory.

Each time DIG is executed, it looks for DIG.ENV or the file specified by LOCALDEF. If such a file exists, then the environment is restored from this file before any arguments are parsed.

-envset
/ENVSET
This qualifier only affects batch query runs. When -envset is specified on a line in a DIG batch file, the DIG environment after the arguments are parsed becomes the default environment for the duration of the batch file, or until the next line which specifies -envset. Remember that commands in the DIG batch file must be in UNIX-style syntax.

-f *filename*
/FILE=*filename*
Causes DIG to run in batch mode, executing the commands in the specified file. The commands in this file must be in the UNIX-style syntax.

+[no]Header
/HEADER
/NOHEADER
Tells the resolver to print basic header information. On by default.

+[no]header
/HFLAGS
/NOHFLAGS

Tells the resolver to print header flags. On by default.

+[no]ignore
/IGNORE
/NOIGNORE
Tells the resolver to ignore truncation in responses. Off by default.

+[no]ko
/KEEPOPEN
/NOKEEPOPEN
If using virtual circuits (TCP), keeps the connection open. Off by default.

-k *keydir+keyname*
/KEY=(KEYNAME=*key*[,KEYDIR=*directory*])

Specifies a TSIG key for DIG to use to sign its queries. The default value for `KEYDIR` is the current default directory.

> **Note:** On UNIX systems, the syntax is *`keydir:keyname`*. On OpenVMS, the colon is replaced by a plus sign (+). The *`keyname`* must be specified to match the key and private filenames, with periods instead of dollar signs. This may not match the domain name if `DNSKEYGEN` had to abbreviate it to fit into an OpenVMS file name.

+pfand=*number*
/PFAND=*number*
Causes DIG to do a bitwise-AND of the print flags with the specified value.

+pfdef
/PFDEF
/NOPFDEF
Sets the print flags to the default. `/PFDEF` is the default.

+pfmin
/PFMIN
/NOPFMIN
Sets the print flags to the minimum. /NOPFMIN is the default.

+pfor=*number*
/PFOR=*number*
Causes DIG to do a bitwise-OR of the print flags with the specified value.

+pfset=*number*
/PFSET=*number*
Sets the print flags to the specified value.

"-P" *ping-command*
/PING[=*ping-command*]
Causes DIG to execute a ping command to the queried name server after the query returns, for response time comparison. If the optional *ping-command* is present, it is used as the ping command. The default ping command is MULTINET PING.

-p *port*
/PORT=*port*
Specifies a port other than the standard name server port of 53.

+[no]qr
/QUERY
/NOQUERY
Tells the resolver to print the outgoing query. /NOQUERY is the default.

+[no]ques
/QUESTION
/NOQUESTION
Tells the resolver to print the question section of the reply. /QUESTION is the default.

+[no]recurse
/RECURSE
/NORECURSE
Requests that the name server use recursion to answer the query. /RECURSE is the default.

+[no]reply
/REPLY
/NOREPLY
Tells the resolver to print the reply. /REPLY is the default.

+retry=*retrycount*
/RETRY=*retrycount*
Specifies the number of retries the resolver makes when querying a name server via UDP. The default is 4.

@*server*
/SERVER=*server*
Specifies the name server to query. May be specified as either a domain name or a dot-notation internet address. If a domain name is specified, DIG looks up the name using the default name server. If /SERVER is not specified, the default is to use the system's default name server.

+[no]stats
/STATS
/NOSTATS
Tells the resolver to print query statistics. /STATS is the default.

-[no]stick
/STICKY
/NOSTICKY
This qualifier only affects batch query runs. -stick specifies that the DIG environment (as read initially or set by -envset switch) is to be restored before each query (line) in a DIG batch file. The default /NOSTICKY means that the DIG environment does not stick, hence options specified on a single line in a DIG batch file will remain in effect for subsequent lines (i.e., they are not restored to the "sticky" default). Remember that commands in the DIG batch file must be in UNIX-style syntax.

+time=*seconds*
/TIMEOUT=*seconds*
Specifies a different period to wait for responses. The default is 4 seconds.

"-T" *seconds*
/TIMEWAIT=*seconds*
Causes DIG to wait the specified number of seconds between the start of successive queries when running in batch mode. Can be used to keep two or more batch DIG commands running roughly in sync. The default is 0.

-t *recordtype*
/TYPE=*recordtype*
Specifies which TYPE resource records are asked for. Alternative to specifying the type parameter. The *recordtype* value may be either the integer value of the type or the name of the type (see NETCONTROL TELNET Commands). The default is A (address records).

+[no]vc
/VC
/NOVC
Specifies that the resolver use virtual circuits (TCP) instead of datagram (UDP) queries. /NOVC is the default.

Example

The following is an example of the default DIG output:

```
$ MULTINET DIG WWW.EXAMPLE.COM

; <<>> DiG 8.3 <<>> WWW.EXAMPLE.COM
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 2
;; QUERY SECTION:
;;      WWW.EXAMPLE.COM, type = A, class = IN

;; ANSWER SECTION:
WWW.EXAMPLE.COM.        2H IN CNAME     example.com.
example.com.            2H IN A         10.0.0.83

;; AUTHORITY SECTION:
```

```
example.com.            2H IN NS        ns1.example.net.
example.com.            2H IN NS        ns2.example.net.

;; ADDITIONAL SECTION:
ns1.example.net.        2D IN A         10.1.0.11
ns2.example.net.        2D IN A         10.1.0.11

;; Total query time: 14289 msec
;; FROM: bos.example.com to SERVER: default -- 127.0.0.1
;; WHEN: Thu Jun  1 14:52:49 2002
;; MSG SIZE  sent: 29  rcvd: 141
```

MULTINET DNSKEYGEN

DNSKEYGEN (DNS Key Generator) is a tool to generate and maintain keys for DNS Security (DNSSEC) within the DNS (Domain Name System). DNSKEYGEN can generate public and private keys to authenticate zone data, and shared secret keys to be used for Request/Transaction Signatures.

DNSKEYGEN can be used with the UNIX-style syntax by defining it as a foreign command:

```
$ DNSKEYGEN :== $MULTINET:DNSKEYGEN.EXE
```

Both the UNIX-style options and the OpenVMS qualifiers are listed below.

DNSKEYGEN stores each key in two files: *Kname.alg-footprint*-private and *Kname.alg-footprint*-key. *name* is the domain name with the periods replaced by dollar signs. The first file contains the private key in a portable format. The second file contains the public key in the DNS zone file format:

```
name IN KEY flags protocol algorithm exponent|module
```

If the domain name is too long for an OpenVMS filename, it is truncated to fit and the last six characters are replaced by unique digits. The full domain name can be found inside the key file.

Format

MULTINET DNSKEYGEN *name*

Parameters

-n name
name
Specifies the domain name to generate the key for.

Qualifiers

"-D" ***size***
/DSA_DSS=*size*
"-H" ***size***
/HMAC_MD5=*size*
"-R" ***size***
/RSA=*size*
These flags specify the type of key to generate. **You must specify one and only one of these**.

If /DSA_DSS is specified, DNSKEYGEN generates a DSA/DSS key. *size* must be one of: 512, 576, 640, 704, 768, 832, 896, 960, or 1024.

If /HMAC_MD5 is specified, DNSKEYGEN generates an HMAC-MD5 key. *size* must be between 128 and 504.

If /RSA is specified, DNSKEYGEN generates an RSA key. *size* must be between 512 and 4096.

"-F"
/LARGE_EXPONENT
Used for RSA only. If specified, DNSKEYGEN uses a large exponent for key generation.

-z
/ZONE_KEY
-h
/HOST_KEY
-u
/USER_KEY
These flags define the type of key being generated. **You must specify one and only one of these.**

- Zone (DNS validation) key
- Host (host or service) key
- User (e.g., email) key

-a
/NOAUTHENTICATION
Indicates that the key CANNOT be used for authentication.

-c
/NOENCRYPTION
Indicates that the key CANNOT be used for encryption.

-p *num*
/PROTOCOL=*num*
Sets the key's protocol field to *num*. If /ZONE_KEY (-z) or /HOST_KEY (-h) is specified, the default is 3 (DNSSEC); otherwise, the default is 2 (EMAIL). Other accepted values are 1 (TLS), 4 (IPSEC), and 255 (ANY).

-s *num*
/STRENGTH=*num*
Sets the key's strength field to *num*; the default is 0.

Example

The following example generates an RSA key.

```
$ MULTINET DNSKEYGEN/RSA=512/ZONE_KEY zone.example
** Adding dot to the name to make it fully qualified domain name**
Generating 512 bit RSA Key for ZONE.EXAMPLE.

Generated 512 bit Key for ZONE.EXAMPLE. id=49663 alg=1 flags=257
```

DNSKEYGEN generates the following (for example):

File KZONE$EXAMPLE$.001-49663-KEY:

```
ZONE.EXAMPLE. IN KEY 257 3 1
AQOojr81q9PfmQXCUAJOoMu3CYaS78RZnhiV/uAfSbzZusWYLSeVF47OwZlmgwclswZoaM5NSuzFX3w5RDIEwf9c
```

File KZONE$EXAMPLE$.001-49663-PRIVATE:

```
Private-key-format: v1.2
Algorithm: 1 (RSA)
Modulus:
qI6/NbPT35kGwlACTqDLtwmGku/EWZ4Ylf7gH0m82arFmC0nlReOjsGJZoMHJbMGaGjOTUrsxV98OUQyAMH/Ww==
PublicExponent: Aw==
PrivateExponent:
cF8qI8036mZD1uABjcCHz1uvDJ/YO767Dqmqv4Z95ntuhY7uIMmn8zy0Ur9kj/7P5Dvpu7ZG91ZtuQ1YhWAMyw==
Prime1: 2IQQP2+DvU/G0038OCoji00NDQHA0az8lDV1fh8Qf9k=
Prime2: x0vGgXRlWVIfp5xnuCORP0UB4rK3sKVhQ246rx2hbFM=
Exponent1: kFgK1PQCfjUvN4lS0BwXtN6Is1aBNnNTDXj4/r9gVTs=
Exponent2: hN0vABhDjja/xRLv0Be2Kl4BQcv6dcOWLPQnH2kWSDc=
Coefficient: YQGEh81Y720mRfAV/tEs3eWKd11Mm10b5R4lFjVwtAU=
```

MULTINET DNSSIGNER

DNSSIGNER is a tool to sign zone files for DNS Security (DNSSEC) within the DNS (Domain Name System). DNSSIGNER's job is to read the data of one zone of DNS data, and perform the necessary work to produce the data for a secured zone.

DNSSIGNER can be used with the UNIX-style syntax by defining it as a foreign command:

```
$ DNSSIGNER :== $MULTINET:DNSSIGNER.EXE
```

Both the UNIX-style options and the OpenVMS qualifiers are listed below.

You can get help on the UNIX-style options using:

`$ dnssigner -h`	! for short help
`$ dnssigner -help`	! for long help

Signing is done on a zone-by-zone basis, regardless of the relationship of zones to name servers. DNSSIGNER is designed to operate in a dynamic environment, including those in which secret keys are not available to all of those covering a zone, and where information may be arriving after the beginning of the signing process. DNSSIGNER makes an effort to retain valid signatures instead of computing new signatures.

Using traditional BIND DNS zone master files, there are two things necessary as input to use DNSSIGNER to sign a zone. One is the names of the input files and the other is the names of the keys to use. There are two kinds of data files used as input to the signing process. The standard zone master file, and a master file introduced by DNSSEC called the parent file. A parent file contains output from the signing of the parent zone, most importantly the signature by the parent of the zone's keys.

Input/Output Details

The default input zone is START-ZONE. A different zone input file can be specified with /ZONE=(INPUT=*filename*) (-zi). There is no default input parent file. A parent file can be specified with /PARENT=(INPUT=*filename*) (-pi).

The default output files are FINISH-ZONE. and FINISH-PARENT. /ZONE=(OUTPUT=*filename*) (-zo) changes the name of the zone output file, and /PARENT=(OUTPUT=*filename*) (-po) changes the name of the parent file generated by the zone.

Parent File Handling

There are two forms of parent file generation. One form is to place all of the parent files in one file (good for zones with many delegations), the other is to make a separate file for each delegation. Since it is easier to erase one file than potentially thousands, `DNSSIGNER` defaults to the single signer file.

`/PARENT=NOBULK` (`-no-p1`) turns single parent file generation off, `/PARENT=BULK` (`-p1`) turns it on. As mentioned earlier `/PARENT=OUTPUT=`*`filename`* (`-po`) sets the name of the single parent file (default `FINISH-PARENT.`).

`/PARENT=INDIVIDUAL` (`-ps`) turns on individual parent files, `/PARENT=NOINDIVIDUAL` (`-no-ps`) turns it off. `/PARENT=DIRECTORY=`*`spec`* (`-pd`) sets the directory into which the individual files are put (default is the current working directory).

NXT Details

`/NONXT` (`-no-n`) turns off RFC 2065 NXT processing.
`/NXT` (`-n`) (default) turns on RFC 2065 NXT processing.

Key Details

Use the `-k1` flag (the `/SIG=(KEY=( )`) qualifier) or the `-ks` flag (no OpenVMS-style equivalent) to specify a key. `-k1` is followed by a domain name owner of a key, the algorithm, and the key id. `-ks` is followed by a sequence of names, algorithms, and key ids until the end of the command line.

SIG Expiration Details

There are two time durations that are important to the handling of signatures. One is the duration until a newly generated signature is set to expire. The other is the duration in which existing signatures will be considered to be expired.

`/SIG=DURATION=`*`ttl`* (`-dur`) sets the duration for which a signature is valid.

The time included in the SIG RR expiration field is the current absolute time plus the duration. Wrapping around 32 bits is not a problem, as time is considered to be "circular."

/SIG=PURGE_PERIOD=*ttl* (-pt) sets the period into the future in which SIGs expiring then are considered to have expired. Any signature that has an expiry time in the past of the current time is thrown out, as well as signature whose expiry time falls into the span between now and the purge period duration. The past is considered to be the time from now back to 2 to the 31st seconds ago; the rest is the future.

Format

MULTINET DNSSIGNER

Qualifiers

This section describes the syntax of all flags. The meanings can be found in RFC 2065 and the drafts associated with the DNSSEC working group.

-[no-]bind
/[NO]BIND
/BIND (-bind) instructs DNSSIGNER to use BIND's extended TTLs and KEY flags when writing files. This is the default. Use /NOBIND (-no-bind) to turn this feature off. In this case TTLs and flags are written as numeric values.

-l *option*
/DEBUG=*option*
Specifies the level of output (debug) messages that DNSSIGNER should print. Specify one of the following levels: (UNIX-syntax equivalents are also shown)

-l 7 -l deb DEBUGGER	Print source code locations, errors, and warnings.
-l 10 -l dev DEVELOPER	Print source code locations and cryptography messages.

-l 1 -l m MINIMAL	Print just errors.
-l 4 -l u USER	Print errors and warnings. This is the default.

-[no-]n
/[NO]NXT
/NXT (-n) (default) instructs DNSSIGNER to generate NXT RRs for the zone, signing them with the keys that sign the SOA record. (If none sign the SOA, no NXT's are signed.). Use /NONXT (-no-n) to turn this feature off.

-or *domain*
/ORIGIN=*domain*
This is equivalent to the $ORIGIN *domain* directive in the zone file, except that the terminating period is not needed in the domain name. Specifying an origin is only mandatory for the root zones and other zones using relative names in the zone files. It is recommended that the $ORIGIN *domain* directive be put in the data file. By default, this is unspecified.

/PARENT=(*keyword*[,...])
Specifies options related to parent zone files. Possible keywords (and their UNIX-syntax equivalents) are as follows.

There are two ways in which parent files are made: individual and bulk. The two methods use independent keywords. Both can be used, neither can be used, or just one. By default, the bulk approach is used.

-[no-]p1 [NO]BULK	BULK (-p1) (default) tells DNSSIGNER to place all of the generated parent data for the zone's delegation points into one file. Separating lines are added to identify the start and end of the information destined for individual zones. Use NOBULK (-no-p1) to turn this feature off.

-pd directory DIRECTORY=directory	Specifies the directory to put individual parent files into. The default is the current default directory.
-[no-]ps [NO]INDIVIDUAL	INDIVIDUAL (-ps) tells DNSSIGNER to place the generated parent data into individual files, named *zone*.PARENT. For large delegated zones, there will be many files. The default is NOINDIVIDUAL (-no-ps).
-pi file INPUT=file	Specifies the parent file received from the parent zone to be used as input to this zone. If specified, all records that would conflict with it (apex upper NXT, KEYs, and SIGs for these) are dropped. If the UP policy is specified, then the parent's KEY, NS, and glue are also dropped. The default is to have no parent file.
-pa domain NAME=domain	Specifies the apex's parent zone. If the keys for this zone are known and the UP policy is used, the apex zone keys sign the key. If UP is used and this is not specified, then DNSSIGNER acts as if it does not otherwise know the parent's identity. This is equivalent to the $PARENT directive in the zone file, except that relative domain names are treated as absolute names. By default, the parent's domain name is unspecified.
-po file OUTPUT=file	Specifies the name of the file to hold the bulk generated parent data. The default is FINISH-PARENT.

/POLICY=*option*

Specifies what policy to use when signing the zone. Specify one of the following options: (UNIX-syntax equivalents are also shown)

-dn DOWN	DNSSIGNER signs according to the DOWN policy. That is, the apex does not sign the parent's keys. The parent's keys and glue data are not expected from nor written to the parent files. This is the default.
-up UP	DNSSIGNER signs according to the UP policy. That is, the apex signs the parent's keys. The parent's keys and glue data are expected from and written to the parent files. This policy is not recommended.

-[no-]ess
/[NO]SELF_SIGN
/SELF_SIGN (-ess) instructs DNSSIGNER to make sure each key in the file is signed by its corresponding private key. This is done by implicitly adding $SIGNER directives to the zone file around each key set, adding those keys for just the set. If no private key is available, the $SIGNER directive remains in the output file.

The intent of this feature is to insert proof into DNS that the public key's corresponding private key is held by the owner (or at least the entity signing the zone).

The default is /NOSELF_SIGN (-no-ess).

/SIG=(*keyword*[,...])
Specifies options related to the generation of SIG RRs. Possible keywords (and their UNIX-style equivalents) are as follows.

For DURATION and PURGE_PERIOD, *ttl* format is taken from the BIND definition of TTL. Numeric seconds is accepted, as well as:

numberW	weeks
numberD	days
numberH	hours
numberM	minutes (not months!)
numberS	seconds

The "end of the future" and "beginning of the past" are points in time which have the same time representation (one second apart) in a 32-bit roll-over specification of time. The end of the future is 2 to the 31st power seconds from the current time.

-dur ttl DURATION=ttl	All SIG records generated are set to expire at the current time + duration. The default is 31 days.

`-ks domain algorithm keyid [...] (to end of line)`	There is no OpenVMS syntax equivalent. This adds the specified keys (key owner, algorithm, and key id) to the list of keys with which to sign. Equivalent to `$SIGNER ADD <><>` directives in the zone file. This switch is interpreted as the last switch of the command line. Any number of keys can be specified. See the description for `-k1` for the *`domain`* syntax.
`-k1 domain algorithm keyid KEY=(DOMAIN=domain, ALGORITHM=algorithm, KEY_ID=keyid)`	This adds the specified key (key owner, algorithm, and key id) to the list of keys with which to sign. Equivalent to a `$SIGNER ADD <><>` directive in the zone file. This switch may appear anywhere in the run command. It adds just one key. The default is that keys are specified by `$SIGNER` directives in the data files. A zone may elect not to use any keys. *`domain`* must be specified to match the key and private file names, with periods instead of dollar signs. This may not match the domain name if `DNSKEYGEN` had to abbreviate it to fit into an OpenVMS file name.
`-pt ttl PURGE_PERIOD=ttl`	Specifies that all SIG records with expiration times between the beginning of past up through (current time + the purge period) are treated as expired. SIG records with expiration times from (current + purge period) to the end of the future are retained if they are not proved invalid. The default is 1 week.

-[no-]st
/[NO]STATISTICS
/STATISTICS (-st) instructs DNSSIGNER to print summary statistics at the end of the run. The default is /NOSTATISTICS (-no-st).

/ZONE=(*keyword*[,...])
Specifies options related to zone files. Possible keywords (and their UNIX-syntax equivalents) are as follows:

-zi *file* INPUT=*file*	Specifies the zone data input file. The first RR must be an SOA. The first record may be found in an $INCLUDEd file. The default is START-ZONE.
-zo *file* OUTPUT=*file*	Specifies the file where signed zone data is left. The default is FINISH-ZONE.

Examples

1. Assuming that the zone data is in f.zone and the parent file is in f.parent, to run the files through DNSSIGNER, do the following:

```
$ multinet dnssigner/zone=(input=f.zone)/parent=(input=f.parent)
```

or

```
$ dnssigner :== $multinet:dnssigner.exe
$ dnssigner -zi f.zone -pi f.parent
```

The outputs default to FINISH-ZONE. and FINISH-PARENT. This does no signing, but merges the files, removes duplicates, generates NXT resource records, and makes signing instructions for them (if the zone is judged to be signed).

2. To sign the above zone with the key of test. and a key id of 27782:

```
$ multinet dnssigner/zone=(input=f.zone)/parent=(input=f.parent) -
  /sig=(key=(domain=test.,alg=dsa,key_id=27782)
```

or

```
$ dnssigner -zi f.zone -pi f.parent -k1 test. dsa 27782
```

3. To sign with both keys 27782 and 3696:

```
$ dnssigner -zi f.zone -pi f.parent -ks test. dsa 27782 test. dsa 3696
```

MULTINET FONT COMPILE

Compiles an ASCII BDF (bitmap distribution format) font file into a binary PCF (portable compiled format) file.

Format

`MULTINET FONT COMPILE [`*`qualifiers`*`] [`*`bdf_font_file`*`]`

Qualifiers

/BIT_ORDER=*bit_order*
Specifies the order in which bits in each glyph are placed. Accepted values are `MSBFIRST` (most significant bit) or `LSBFIRST` (least significant bit).

The default is `LSBFIRST` on both the OpenVMS VAX and AXP architectures.

/BYTE_ORDER=*byte_order*
Specifies the order in which multibyte data in the file is written. Multibyte data includes metrics and bitmaps. Accepted values are `MSBFIRST` (most significant bit) or `LSBFIRST` (least significant bit).

The default is `LSBFIRST` on both the OpenVMS VAX and AXP architectures.

/OUTPUT=*file_name*
Specifies an output file name in which the results are written.

/PADDING=*font_glyph_padding*
Sets the font glyph padding. Each glyph in the font has each scanline padding into the specified size. Accepted values are `BYTE`, `WORD`, `LONGWORD`, or `QUADWORD`. On an OpenVMS VAX system, the default is `BYTE`; on an AXP, the default is `LONGWORD`.

/SCANLINE=*data_size*
Specifies the unit of data swapped when the font bit order differs from the font byte order. Accepted values are `BYTE`, `WORD`, and `LONGWORD`. On an OpenVMS VAX system, the default is `BYTE`; on an Alpha or Itanium, the default is `LONGWORD`.

/SERVER=*host*:*port*

The /SERVER qualifier specifies the server from which the font is read. The default value is LOCALHOST:7000.

MULTINET FONT INFO

Displays X font information useful for determining the capabilities and defined values of a font server.

Format

`MULTINET FONT INFO` *`[qualifiers]`*

Qualifiers

`/OUTPUT=`*`file_name`*
Specifies an output file name in which the results are written.

`/SERVER=`*`host:port`*
Specifies the server from which the font is read (by default, `LOCALHOST:7000`).

MULTINET FONT LIST

Lists the font names that match a specified pattern.

Format

`MULTINET FONT LIST [`*`qualifiers`*`] [`*`pattern`*`]`

Parameter

`pattern`
Specifies the pattern to match in font names. Wildcards are permitted in the patterns. If you do not specify a pattern, an asterisk (`*`) is assumed.

Qualifiers

`/BOUNDS`
Indicates long listings should display the minimum and maximum bounds of each font.

`/COLUMNS`
Indicates listings should display in multiple columns.

`/LISTING_TYPE=`*`size`*
Specifies the relative length of a font listing. Accepted values are `SMALL`, `MEDIUM`, `LONG`, and `VERYLONG`.

Consider using `/NOSORT` if you want `LONG` or `VERYLONG` listings faster; otherwise, these types of listings can take a long time to generate. You can also use `/OUTPUT` to write the results to a file.

`/NOSORT`
Indicates the listing is not sorted. Using this qualifier decreases the time required to produce a listing.

`/OUTPUT=`*`file_name`*
Specifies an output file name in which the results are written.

/SERVER=*host:port*
Specifies the server from which the font is read (by default, `LOCALHOST:7000`).

/WIDTH=*display_column_width*
Specifies the width of the columns (by default, 79).

MULTINET FONT MKFONTDIR

Creates a DECW$FONT_DIRECTORY.DAT file in each specified directory.

MKFONTDIR reads all font files in each specified directory. The order in which font files are read is *.PCF files, *.SNF files, then *.BDF files. For scalable fonts, you must edit the created DECW$FONT_DIRECTORY.DAT file to insert the X font name. If you edit this file, back up your changes so they are not lost when MKFONTDIR is run again.

The command fails if you don't have the necessary privileges to write into the directory you specify.

Format

MULTINET FONT MKFONTDIR *[directory_names]*

Parameter

directory_names
Specifies the list of directories in which MKFONTDIR creates a DECW$FONT_DIRECTORY.DAT file.

MULTINET FONT SHOW

Displays font information from files that match the specified pattern.

Format

`MULTINET FONT SHOW [*qualifiers*] [*pattern*]`

Parameter

pattern

Specifies the pattern to match in font names. Wildcards are permitted in the patterns. If you do not specify a pattern, an asterisk (`*`) is assumed.

Qualifiers

/BITMAP_PADDING=*bitmap_size*

Specifies how a character bitmap is padded. Accepted values are `MINIMUM`, `MAXIMUM`, and `MAXWIDTH`.

/BIT_ORDER=*bit_order*

Specifies the order in which bits in each glyph are placed. Accepted values are `MSBFIRST` (most significant bit) or `LSBFIRST` (least significant bit). The default is `LSBFIRST` on all VMS architectures.

/BYTE_ORDER=*byte_order*

Specifies the order in which multibyte data (including metrics and bitmaps) in the file is written. Accepted values are `MSBFIRST` (most significant bit) or `LSBFIRST` (least significant bit). The default is `LSBFIRST` on all VMS architectures.

/END=*decimal_character_value*

Specifies the ending character number (in decimal) about which you want font information listed. Use `/END` with the `/START` qualifier to specify character ranges. If you do not specify `/END`, all characters

from the starting value to the end of the character set are listed. Possible values range from 0 to 255 for normal character sets, and from 0 to 65535 for X double-wide character sets.

/EXTENTS
Indicates that only the extents for a font are displayed.

/OUTPUT=*file_name*
Specifies an output file name in which the results are written.

/PADDING=*font_glyph_padding*
Sets the font glyph padding. Each glyph in the font has each scanline padding into the specified size. Accepted values are BYTE, WORD, LONGWORD, or QUADWORD. On VAX systems the default is BYTE; on Alpha and Itanium, the default is LONGWORD.

/SCANLINE=*data_size*
Specifies the unit of data swapped when the font bit order differs from the font byte order. Accepted values are BYTE, WORD, and LONGWORD. On VAX systems the default is BYTE; on Alpha and Itanium, the default is LONGWORD.

/SERVER=*host*:*port*
Specifies the server from which the font is read (by default, LOCALHOST:7000).

/START=*decimal_character_value*
Specifies the starting character number (in decimal) about which you want font information listed. Use /START with the /END qualifier to specify character ranges. If you do not specify /END, all characters from the starting value to the end of the character set are listed. Possible values range from 0 to 255 for normal character sets, and from 0 to 65535 for X double-wide character sets.

MULTINET FONT UNCOMPILE

Converts a binary PCF-format font file to an ASCII BDF-format file.

Format

`MULTINET FONT UNCOMPILE [`*`qualifiers`*`] [`*`pcf_font_file`*`]`

Qualifiers

`/OUTPUT=`*`file_name`*
Specifies the output file name into which the results are written.

`/SERVER=`*`host:port`*
Specifies the server from which the font is read (by default, `LOCALHOST:7000`).

Example

```
$ MULTINET FONT UNCOMPILE -
_$ -Adobe-Helvetica-Medium-R-Normal--25-180-100-100-P-130-ISO8859-1
```

MULTINET GATED/CHECK

Checks the syntax of a GateD configuration file. If no input file is specified, MultiNet checks the default configuration file, `MULTINET:GATED.CONF`. This command does not affect a running GateD process.

Format

```
MULTINET GATED/CHECK
```

Parameters

filename
Name of the configuration file to check. If omitted, defaults to `MULTINET:GATED.CONF`.

Example

Checks the syntax of a GateD configuration file called `TEST.CONF` located in the user's current working directory.

```
$ MULTINET GATED/CHECK TEST.CONF
```

MULTINET GATED/DUMP

Tells GateD to dump its internal state into a text file. If you omit the filename, the default is MULTINET:GATED.DUMP.

Format

MULTINET GATED/DUMP [*log*]

Parameters

log
Contains log statements generated by GateD. If omitted, defaults to MULTINET:GATED.DUMP.

MULTINET GATED/LOAD

Tells the GateD process to load a configuration file. If no file is specified, the default file `MULTINET:GATED.CONF` is loaded.

> **CAUTION!** If the GateD process detects an error in the configuration file being loaded, it stops running.

Format

`MULTINET GATED/LOAD [file]`

Parameters

file
Name of the configuration file to load. If omitted, defaults to `MULTINET:GATED.CONF`.

Example

This example tells the GateD process to load a new configuration file called `TEST_CONFIG.CONF` from the system manager's current working directory.

```
$ MULTINET GATED/LOAD TEST_CONFIG.CONF
```

MULTINET GATED/SET/TRACE

Tells the GateD process to turn on or off various tracing flags. This controls what is placed in the MULTINET:GATED.LOG file. By default, minimal tracing is done.

Format

MULTINET GATED/SET/TRACE *qualifier*

Qualifiers

/ALL
Turns on all tracing.

/DETAILS
/NODETAILS
Sets tracing of all send and receive information.

/RECV_DETAILS
/NORECV_DETAILS
Sets tracing of receive information.

/SEND_DETAILS
/NOSEND_DETAILS
Sets tracing of send information.

/EVENTS
/NOEVENTS
Sets tracing of normal events.

/NONE
Turns off all tracing.

/PACKETS
/NOPACKETS
Sets tracing of packet sends and receives.

/RECV_PACKETS
/NORECV_PACKETS
Sets tracing of packet receives.

/SEND_PACKETS
/NOSEND_PACKETS
Sets tracing of packet sends.

/PARSING
/NOPARSING
Sets tracing of configuration file parsing.

/POLICY
/NOPOLICY
Sets tracing of policy decisions.

/ROUTING
/NOROUTING
Sets tracing of routing table changes.

/STATES
/NOSTATES
Sets tracing of state machine transitions.

/SYMBOLS
/NOSYMBOLS
Sets tracing of kernel symbols.

/TASKS
/NOTASKS

Sets tracing of task and job functions.

/TIMER
/NOTIMER
Sets tracing of timer functions.

Example

This example tells the GateD process to turn on tracing of policy decisions and turn off tracing of state machine transitions.

```
$ MULTINET GATED/SET/TRACE/POLICY/NOSTATES
```

MULTINET GATED/SHOW/OSPF

Queries OSPF routers. You can obtain a wide variety of detailed information from these routers using these commands.

All of the `SHOW OSPF` commands use a file called `MULTINET:OSPF_DESTS.DAT`. This is a file of OSPF destination records. Each record is a single line entry listing the destination IP address, the destination host name, and an optional OSPF authentication key (if the destination activates authentication).

> **CAUTION!** Since the `OSPF_DESTS.DAT` file may contain authentication information, you should restrict access to it.

> **NOTE:** To stop the output of this command, enter a **Ctrl+C** at the command line.

Format

`MULTINET GATED/SHOW/OSPF` *option*

Options

/ADVERTISE ***area-id***
type
ls-id
adv-router
index
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*

Displays link state advertisements. The parameters and qualifiers for `MULTINET SHOW OSPF ADVERTISE` are as follows:

Parameter and Qualifier	Description
area-id	OSPF area for which the query is directed.
type	The available types are /INTERFACES — Requests the router links advertisements. Describes the collected states of the router's interfaces. For this request, the *ls-id* field should be set to the originating router's Router ID. /ROUTERS — Requests the network links advertisements. Describes the set of routers attached to the network. For this request, the *ls-id* field should be set to the IP interface address of the network's Designated Router. /NETWORK_ROUTES — Requests the summary link advertisements describing routes to networks. Describes the inter-area routes and enables the condensing of routing information at area borders. For this request, the *ls-id* field should be set to the destination network's IP address. /BOUNDARY_ROUTES — Requests the summary link advertisements describing routes to AS boundary routers. Describes the inter-area routes and enables the condensing of routing information at area borders. For this request, the *ls-id* field should be set to the Router ID of the described AS boundary router. /EXTERNAL_ROUTES — Requests the AS external link advertisements. Describes routes to destinations external to the AS. For this request, the *ls-id* field should be set to the destination network's IP address.
ls-id	See the type parameter.

adv-route	Router ID of the router that originated this link state advertisement.
index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.

/AS *index*
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*

Shows the Autonomous System (AS) external database entries. This table reports the advertising router, forwarding address, age, length, sequence number, and metric for each AS external route. The parameters and qualifiers for MULTINET GATED /SHOW /OSPF /AS are as follows:

index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.

DESTINATIONS/OUTPUT=*file*
/FILE=*file*

This command displays the list of destinations and their indices described in an OSPF destination records file. The parameters and qualifiers for MULTINET GATED /SHOW /OSPF /DESTINATIONS are as follows:

/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.

/ERRORS ***index***
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*

Shows the error log. This reports the different error conditions that can happen between OSPF routing neighbors and shows the number of occurrences for each. The parameters and qualifiers for `MULTINET GATED /SHOW /OSPF /ERRORS` are as follows:

index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.

/HOPS ***index***
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*

Shows the set of next hops for the OSPF router being queried. The parameters and qualifiers for `MULTINET GATED /SHOW /OSPF /HOPS` are as follows:

index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.

/INTERFACES ***index***
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*

Displays all interfaces. This shows all the interfaces configured for OSPF. The information includes the area, interface IP address, interface type, interface state, cost, priority and the IP address of the DR and BDR of the network. The parameters and qualifiers for `MULTINET GATED /SHOW /OSPF /INTERFACES` are as follows:

index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.

/LOG ***index***
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*

Shows the cumulative log. This log includes input and output statistics for monitor requests, hellos, database descriptions, link state updates, and link state ACK packets. Area statistics are provided that describe the total number of routing neighbors and number of active OSPF interfaces. Routing table statistics are summarized and reported as the number of intra-area routes, inter-area routes, and AS external database entries.

The parameters and qualifiers for MULTINET GATED /SHOW /OSPF /LOG are as follows:

index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.

/NEIGHBORS ***index***
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*
/RETRANSMIT

This command shows all OSPF routing neighbors. The information shown includes the area, local interface address, router ID, neighbor IP address, state and mode. The parameters and qualifiers for MULTINET GATED /SHOW /OSPF /NEIGHBORS are as follows:

index	Indexes into a file of OSPF destination records.

/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.
/RETRANSMIT	Displays the retransmit list of neighbors.

/ROUTING ***index***
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*

Shows the OSPF routing table. This table reports the AS border routes, area border routes, summary AS border routes, and the networks managed using OSPF. The parameters and qualifiers for MULTINET GATED/SHOW/OSPF/ROUTING are as follows:

index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.
/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.

/STATE ***index***
/OUTPUT=*file*
/FILE=*file*
/TIMEOUT=*seconds*
/RETRANSMIT

Shows the link state database (except for ASEs). This describes the routers and networks making up the AS. The parameters and qualifiers for MULTINET GATED/SHOW/OSPF/STATE are as follows:

index	Indexes into a file of OSPF destination records.
/OUTPUT=*file*	Name of an output file to write the results to.
/FILE=*file*	Alternate file of OSPF destination records to use.

/TIMEOUT=*seconds*	Interval to wait for a response. Default is 20 seconds.
/RETRANSMIT	Displays the retransmit link state database.

Examples

1. Displays the OSPF cumulative log for index 1 in the OSPF_DESTS.DAT file.

```
$ MULTINET GATED/SHOW/OSPF/LOG 1
          Source <<192.168.5.31     izar.example.com>>
IO stats
        Input  Output   Type
            2       0  Monitor request
            0       0  Hello
            0       0  DB Description
            0       0  Link-State Req
            0       0  Link-State Update
            0       0  Link-State Ack
       ASE:  0 checksum sum 0

       LSAs originated: 39   received: 0
                Router: 39

       Area 0.0.0.0:
                Neighbors: 0    Interfaces: 0
                Spf: 1 Checksum sum CE9D
                DB: rtr: 1 net: 0 sumasb: 0 sumnet: 0

Routing Table:
         Intra Area: 0   Inter Area: 0     ASE: 0
```

2. Displays the OSPF interface log for index 1 in the OSPF_DESTS.DAT file.

```
$ MULTINET GATED/SHOW/OSPF/INTERFACE 1
          Source <<192.168.5.31      izar.example.com>>
IO stats
        Input  Output   Type
            6       0  Monitor request
            0       0  Hello
            0       0  DB Description
            0       0  Link-State Req
            0       0  Link-State Update
            0       0  Link-State Ack
       ASE: 0 checksum sum 0
```

```
        LSAs originated: 39   received: 0
                Router: 39

        Area 0.0.0.0:
                Neighbors: 0    Interfaces: 0
                Spf: 1  Checksum sum CE9D
                DB: rtr: 1 net: 0 sumasb: 0  sumnet: 0

Routing Table:
        Intra Area: 0   Inter Area: 0   ASE: 0
```

3. Displays the OSPF destination records in the `OSPF_DESTS.DAT` file.

```
$ MULTINET GATED/SHOW/OSPF/DESTINATIONS
1: 192.168.5.31    izar.example.com
```

4. Displays the OSPF link state database log for index 1 in the `OSPF_DESTS.DAT` file.

```
$ MULTINET GATED/SHOW/OSPF/STATE 1
          Source <<192.168.5.31    izar.example.com>>
LS Data Base:
Area: 0.0.0.0
Type LinkState ID   AdvRouter    Age  Len Sequence Metric Where
-----------------------------------------------------------------
Rtr  192.168.5.31   192.168.5.31 986  24  80000027 0      SpfTree
```

5. Displays the OSPF next hops log for index 1 in the `OSPF_DESTS.DAT` file.

```
$ MULTINET GATED/SHOW/OSPF/HOPS 1
          Source <<192.168.5.31    izar.example.com>>
Next hops:

Address         Type      Refcount  Interface
-----------------------------------------------------
192.168.5.31    Direct    1         192.168.5.31    SVA-0
```

6. Displays the OSPF error log for index 1 in the `OSPF_DESTS.DAT` file.

```
$ MULTINET GATED/SHOW/OSPF/ERRORS 1
        Source <<192.168.5.31  izar.example.com>>
Packets Received:
   3: Monitor request              0: Hello
   0: DB Description               0: Link-State Req
```

```
   0: Link-State Update             0: Link-State Ack

Packets Sent:
   0: Monitor response              0: Hello
   0: DB Description                0: Link-State Req
   0: Link-State Update             0: Link-State Ack

Errors:
   0: IP: bad destination           0: IP: bad protocol
   0: IP: received my own packet    0: OSPF: bad packet type
   0: OSPF: bad version             0: OSPF: bad checksum
   0: OSPF: bad area id             0: OSPF: area mismatch
   0: OSPF: bad virtual link        0: OSPF: bad authentication type
   0: OSPF: bad authentication key 0: OSPF: packet too small
   0: OSPF:packet size > ip length 0: OSPF: transmit error
   0: OSPF: interface down          0: OSPF: unknown neighbor
   0: HELLO: netmask mismatch       0: HELLO: hello timer mismatch
   0: HELLO: dead timer mismatch    0: HELLO: extern option mismatch
   0: HELLO: router id confusion    0: HELLO: virtual neighbor unknown
   0: HELLO: NBMA neighbor unknown 0: DD: neighbor state low
   0: DD: router id confusion       0: DD: externoption mismatch
   0: DD: unknown LSA type          0: LS ACK: neighbor state low
   0: LS ACK: bad ack               0: LS ACK: duplicate ack
   0: LS ACK: Unknown LSA type      0: LS REQ: neighbor state low
   0: LS REQ: empty request         0: LS REQ: bad request
   0: LS UPD: neighbor state low    0: LS UPD: newer self-gen LSA
   0: LS UPD: LSA checksum bad      0: LS UPD:received less recent LSA
   0: LS UPD: unknown LSA type
```

MULTINET GATED/SHOW/RIP

Used to request all routes known by a RIP gateway. The routing information in any routing packets returned is displayed numerically and symbolically. This command is intended to be used as a tool for debugging gateways, not for network management.

> **Note:** To stop the output of this command, enter a **Ctrl+C** at the command line.

Format

`MULTINET GATED /SHOW /RIP` *`gateway-ia`*

Parameters

gateway-ia
Internet address or name of the gateway to be queried.

Qualifiers

/AUTHENTICATION=*authkey*
Authentication password to use for queries. If specified, an authentication type of `SIMPLE` is used. The default authentication type is `NONE`.

/NONAME
Prevents the responding host's address from being looked up to determine the symbolic name.

/POLL
Requests information from the gateway's routing table. This is the default. If there is no response to the `/POLL` qualifier, the `/REQUEST` qualifier is tried.

/REQUEST

Requests information from the gateway's routing table. Unlike the /POLL qualifier, all gateways should support this command. If there is no response, the /POLL qualifier is tried.

/TIMEOUT=*seconds*
Number of seconds to wait for the initial response from a gateway. Default is 5 seconds.

/TRACE
Traces the RIP packets being sent and received by this command.

/V1
Sends the query as a RIP version 1 packet.

/V2
Sends the query as a RIP version 2 packet.

Example

Shows the routers known by RIP gateway 192.168.10.2.

```
$ MULTINET GATED/SHOW/RIP 192.168.10.2
24 bytes from omega1.example.com(192.168.10.2):
             net/mask              router        metric   tag
       192.168.5.0/255.255.255.0    192.168.10.1   2        0000
```

MULTINET GATED/SHOW/TRACE

Queries tracing in GateD.

Format

```
MULTINET GATED /SHOW /TRACE
```

Example

```
$ multinet gated/show/trace
     Summary of GateD tracing
-------------------------------------------------
     State Machine Transitions Logging  is :  'OFF'
     Internal Events Logging            is :  'OFF'
     Policy Decision Logging            is :  'OFF'
     Task Information Logging           is :  'OFF'
     Timer Logging                      is :  'OFF'
     Routing Information Logging        is :  'OFF'
     General Send and Receive Logging   is :  'OFF'
     General Receive Logging            is :  'OFF'
     General Send Logging               is :  'OFF'
     Packet Send and Receive Logging    is :  'OFF'
     Packet Receive Logging             is :  'OFF'
     Packet Send Logging                is :  'OFF'
     Configuration File Parsing Logging is :  'OFF'
     Route Advertisement Logging        is :  'OFF'
     Kernel Symbols Logging             is :  'OFF'
     Network Interface Logging          is :  'OFF'
```

MULTINET GATED /STOP

Tells the GateD process to halt in an orderly manner.

Format

```
MULTINET GATED /STOP
```

MULTINET GATED /TOGGLE_TRACING

Toggles GateD tracing on and off. This command opens and closes the GateD log file `MULTINET:GATED.LOG` as needed.

Format

```
MULTINET GATED /TOGGLE_TRACING
```

MULTINET GATED/UPDATE_INTERFACES

Tells the GateD process to rescan the network interfaces.

Format

```
MULTINET GATED /UPDATE_INTERFACES
```

MULTINET HOST_TABLE COMPILE

The MultiNet host table compiler generates binary host tables from the ASCII host table files. After modifying a MultiNet host table, use this command to compile it into its binary form.

After recompiling your host tables, reinstall the host tables by rebooting, or by invoking the `@MULTINET:INSTALL_DATABASES` command. Then make the host table usable to the `MULTINET_SERVER` process servers by restarting this process with `the @MULTINET:START_SERVER` command. MultiNet uses the compiled host tables for fast lookups of host names, and for translation of host, network, protocol, and service names to numbers.

Format

`MULTINET HOST_TABLE COMPILE` *`[files]`*

Parameters

`files`
Contains a comma-separated list of one or more input files to be compiled. These files must be in the format described in RFC 952 *DoD Internet Host Table Specification*. If not specified, the input files default to `MULTINET:HOSTS.SERVICES`, `MULTINET:HOSTS.LOCAL`, and `MULTINET:HOSTS.TXT`.

Qualifiers

`/HOST_TABLE_FILE=`*`file`*
Specifies the file to which the compiler writes the binary host table (by default, `MULTINET:NETWORK_DATABASE`).

`/SILENTLY`
`/NOSILENTY`
Determines whether the compilation proceeds quietly. The default, `/NOSILENTLY`, can take some time to process.

`/STARTING_HASH_VALUE=`*`value`*

Specifies the initial hash size for the host table hash. Starting at this value, the host table compiler searches for an acceptable hashing function. The default for this qualifier is the "best value," which is computed from the size of the data as the utility attempts to create 512-byte units.

When you run `HOST TABLE COMPILE`, the hash value is listed in the displayed messages. To select a value for this qualifier, choose a number from the displayed range of values.

/TBLUK_FILE=*file*
Specifies the file to which the compiler writes the "host-completion" database, used by programs that allow for escape-completion of partially typed host names. The default is `MULTINET:HOSTTBLUK.DAT`.

/UNIX_HOST_FILE=*file*
Specifies the file to which the compiler writes a UNIX-style hosts file that can be used on most UNIX systems and with many other vendors' TCP implementations. The default, `/NOUNIX_HOST_FILE`, inhibits the creation of a UNIX-style hosts file.

MULTINET HOST_TABLE GET

Connects to the HOSTNAME port of NIC.DDN.MIL and uses the HOSTNAME protocol to retrieve the `HOSTS.TXT` file. After retrieving a new MultiNet host table, compile it into binary form with the `MULTINET HOST_TABLE COMPILE` command so the host table can be accessed.

> **Caution!** The `HOSTS.TXT` file located on NIC.DDN.MIL is no longer maintained by the DDN NIC (or anyone else). This file contains out-of-date information and should be used with caution. If your host is connected to the Internet, DNS is a desirable alternative to using host tables.

Qualifiers

`/HOST=`*`host`*
Specifies a host other than NIC.DDN.MIL. If you specify the host name instead of the address, the host name must exist in your existing host tables.

`/OUTPUT_FILE=`*`file`*
Specifies a different output file (by default, `MULTINET:HOSTS.TXT`).

`/SILENTLY`
`/NOSILENTLY`
Specifies that various debugging information is written to `SYS$ERROR` as the program executes. The default is `/NOSILENTLY`.

`/QUERY`
Specifies an arbitrary HOSTNAME protocol request to the host of interest as follows:

- If the `/QUERY` qualifier is present, use its value
- Otherwise, if the `/VERSION` qualifier is present, use `VERSION`
- Otherwise, use `ALL`

`/VERSION`
Retrieves only the `HOSTS.TXT` version number.

The HOSTNAME protocol supports simple text query requests of the form:

```
command_key argument(s) [options]
```

command_key is a keyword indicating the nature of the request and square brackets ([ ]) indicate an optional field. The defined keys are described in the following table:

Keyword	**Response**
HELP	The information in this table.
VERSION	"VERSION: *string*" where *string* is different for each version of the host table.
HNAME *hostname*	One or more matching host table entries.
HADDR *hostaddr*	One or more matching host table entries.
ALL	The entire host table.
ALL-OLD	The entire host table without domain-style names.
DOMAINS	The entire top-level domain table (domains only).
ALL-DOM	Both the entire domain table and the host table.
ALL-INGWAY	All known gateways in TENEX/TOPS-20 INTERNET.GATEWAYS format.

MULTINET HOST_TABLE INSTALL

Installs the binary host tables as global sections. Do not run `HOST_TABLE INSTALL` directly. Instead, use the `MULTINET:INSTALL_DATABASES.COM` command procedure.

Format

```
MULTINET HOST_TABLE INSTALL
```

MULTINET IPP SHOW

The MULTINET IPP SHOW utility allows a user to learn the capabilities supported by an IPP server. This utility queries the server and displays the supported attributes. The program can be used to see what a given server supports, by a program to gather information about a number of printers, or by a DCL or other program to check the capabilities of a given server before submitting a print job to a queue. The command syntax is:

```
$ MULTINET IPP SHOW server_URI /qualifiers...
```

Qualifiers

/ATTRIBUTE=attribute

Puts the program into a mode suitable for use from a DCL command procedure. Not compatible with the /FORMAT or /OUTPUT qualifiers or those associated with them. It causes the program to return the value of a single attribute as a character string in a DCL symbol. The symbol may be specified with the /SYMBOL qualifier if the default of IPP_SHOW_RESULT is not desired. This is intended for use in a procedure to check to see if, for example, a given server supports color printing before submitting a job to a queue that requires color output. Allowable values for *attribute* are:

```
Charset_Configured
Charset_Supported
Color_Supported
Compression_Supported
Copies_Default
Copies_Supported
Document_Format_Default
Document_Format_Supported
Finishings_Default
Finishings_Supported
Gen_Natural_Language_Supported
Job_Hold_Until_Default
Job_Hold_Until_Supported
Job_Impressions_Supported
Job_K_Octets_Supported
Job_Media_Sheets_Supported
Job_Priority_Default
Job_Priority_Supported
Job_Sheets_Default
Job_Sheets_Supported
Media_Default
Media_Supported
Multiple_Doc_Handling_Default
Multiple_Doc_Handling_Supported
Multiple_Operation_Timeout
Orientation_Requested_Default
Orientation_Requested_Supported
Page_Ranges_Default
Page_Ranges_Supported
PDL_Override_Supported
Print_Quality_Default
Print_Quality_Supported
Printer_Current_Time
Printer_Driver_Installer
Printer_Info
Printer_Is_Accepting_Jobs
Printer_Location
Printer_Make_and_Model
Printer_Message_From_Operator
Printer_More_Info
Printer_More_Info_Manufacturer
Printer_Name
Printer_Resolution_Default
Printer_Resolution_Supported
Printer_State
Printer_State_Message
Printer_State_Reasons
Printer_Uptime
Printer_URI_Supported
Queued_Job_Count
```

```
Natural_Language_Configured          Reference_URI_Schemes_Supported
Number_Up_Default                    Sides_Default
Number_Up_Supported                  Sides_Supported
Operations_Supported                 URI_Security_Supported
```

/[NO]APPEND
Specifies that output should be appended to an existing output file if possible. /NOAPPEND is the default.

/FORMAT=*style*
Specifies what print style to use. *style* is either

- SCREEN (default) which writes in a human-friendly screen-formatted mode or
- LIST which writes an easy to parse, *name=value* format, one name/value pair per line.

/[NO]FULL
Causes all IPP attributes to be included in the display, whether the server supports them or not. Those not supported are marked as such. /NOFULL is the default.

/[NO]GLOBAL
Specifies whether the named symbol should be created as a DCL global symbol. Used only with /ATTRIBUTE. If specified as /NOGLOBAL, the symbol will be local to the calling procedure level. /GLOBAL is the default.

/OUTPUT=*file*
Specifies a file to write output to. SYS$OUTPUT: is the default.

/SYMBOL=*symbolname*
Specifies a DCL symbol name that should be set to the value of the specified attribute. Used only with /ATTRIBUTE. The default is IPP_SHOW_RESULT if /SYMBOL is not specified.

Examples

1. Basic operation with all defaults:

```
$ MULTINET IPP SHOW LILLIES.EXAMPLE.COM
LILLIES.EXAMPLE.COM as of Tue Mar 9 16:08:43 2020

CURRENT INFO:
    Printer State:       Idle
    State Reasons:       none
    Accepting Jobs?:     Yes
    Queued Job Count:    0

PRINTER INFO:
    Name:                Lexmark Optra T610
    Make & Model:        Lexmark Optra T610



DEFAULTS:
    Document Format:     application/octet-stream
    Orientation:         Portrait
    Number-Up:           1
    Copies:              1
    Job Media Sheets:    none
    Character Set:       utf-8
    Natural Language:    en-us

SUPPORTED FEATURES AND ALLOWED VALUES:
    Color?:              No
    Orientation:         Portrait, Landscape
    Document Formats:    application/octet-stream, application/postscript,
                         application/vnd.hp-PCL, text/plain
    Job Sheets:          none, standard
    Number-Up:           1:16
    Copies:              1:999
    PDL Override:        not-attempted
    Character Sets:      utf-8, us-ascii
    Natural Languages:   en-us
    Operations:          Print_Job, Validate-Job, Cancel-Job,
                         Get-Job_Attributes, Get-Jobs,
                         Get-Printer_Atrributes, Unknown: 18

URIs Supported and associated security options:
        URI:      http://192.168.50.2/
        Security: none

        URI:      http://192.168.50.2:631/
       Security: none
```

2. Operation with `/FULL` and output to a file (note that the "/" character in the URI requires use of quotes around the server URI parameter):

```
$ MULTINET IPP SHOW "LILLIES.EXAMPLE.COM/IPP" /FULL /OUTPUT=FOO.BAR
FOO.BAR contains:
LILLIES.EXAMPLE.COM/IPP as of Tue Mar 9 16:11:54 2020
CURRENT INFO:
    Printer State:       Idle
    State Reasons:       none
    State Message:       <not supported>
    Accepting Jobs?:     Yes
    Queued Job Count:    <not supported>
    Uptime (seconds):    <not supported>
    Printer Time:        <not supported>

 PRINTER INFO:
    Name:                LILLIES
    Printer Location:    <not supported>
    Printer Info:        MANUFACTURER:Hewlett-Packard;COMMAND SET:PJL,ML -
                           C,PCL,PCLXL,POSTSCRIPT;MODEL:HP LaserJet 2100  -
                           Series;CLASS:PRINTER;DESCRIPTION:H
    URL for more info:   <not supported>
    URL for driver:      <not supported>
    Make & Model:        <not supported>
    URL for Maker:       <not supported>

DEFAULTS:
    Document Format:     application/octet-stream
    Orientation:         <not supported>
    Number-Up:           <not supported>
    Sides:               <not supported>
    Copies:              <not supported>
    Mult. Doc. Handling: <not supported>
    Media:               <not supported>
    Job Media Sheets:    <not supported>
    Finishings:          <not supported>
    Job Priority:        <not supported>
    Job Hold Until:      <not supported>
    Print Quality:       <not supported>
    Printer Resolution:  <not supported>
    Character Set:       us-ascii
    Natural Language:    en-us
    Mult. Op. Timout:    <not supported>

SUPPORTED FEATURES AND ALLOWED VALUES:
    Color?:              <not supported>
    Orientation:         <not supported>
    Document Formats:    text/plain, text/plain; charset=US-ASCII,
                           application/postscript, application/vnd.hp-PCL,
                           application/octet-stream
    Job Sheets:          <not supported>
```

```
    Number-Up:           <not supported>
    Sides:               <not supported>
    Copies:              <not supported>
    Mult. Doc. Handling: <not supported>
    Media Names:         <not supported>
    Job Media Sheets:    <not supported>
    Finishings:          <not supported>
    Job Priority:        <not supported>
    Job Hold Until:      <not supported>
    Page Ranges?:        <not supported>
    Print Qualities:     <not supported>
    Resolutions:         <not supported>
    Compression Modes:   <not supported>
    Job K-octets:        <not supported>
    Job Impressions:     <not supported>
    PDL Override:        not-attempted
    Character Sets:      us-ascii, utf-8
    Natural Languages:   en-us
    URI Schemes:         <not supported>
    Operations:          Print_Job, Validate-Job, Cancel-Job,
                           Get-Job_Attributes, Get-Jobs,
                           Get-Printer_Atrributes

URIs Supported and associated security options:

        URI:      /ipp
        Security: none

        URI:      /ipp/port1
        Security: none

MESSAGE FROM OPERATOR:

    <no Message>
```

3. Operation with `/ATTRIBUTE` and `/SYMBOL` and `/GLOBAL` to get a single attribute into a DCL symbol:

```
$ MULT IPP SHOW LEXIM /ATTRIB=NUMBER_UP_SUPPORTED /SYMBOL=NUMUP
/GLOBAL
$ SHO SYM NUMUP
  NUMUP == "1:16"
$
```

MULTINET LOAD

Invokes the MultiNet network `LOADER`. This program loads a network image into the VMS kernel and starts the network.

> **Note:** This utility is invoked automatically by the network startup command file generated by the Network Configuration Utility and should not be invoked by a user.

Format

```
MULTINET LOAD
```

MULTINET NETCONTROL

Sends commands to services internal to the MULTINET_SERVER process. NETCONTROL can select any server provided in the MultiNet configuration or those previously added with the Server Configuration Utility. This command affects only the currently running configuration.

Format

MULTINET NETCONTROL [*service*] [*command*]

Parameters

service
Connects to the specified service (by default, NETCONTROL).

command
Sends a specified command string to the server. If you do not specify a command string, NETCONTROL enters interactive mode.

Qualifiers

/HOST=*host*
Connects to the NETCONTROL service on the specified host (by default, the NETCONTROL service on the local host).

/VERBOSE
Displays the entire NETCONTROL protocol conversation. This qualifier is useful only for debugging purposes.

Description

The NETCONTROL program sends commands to services internal to the MULTINET_SERVER process. NETCONTROL currently provides access to the following MultiNet services:

ACCESS	BOOTP	BWNFSD	CLUSTERALIAS
DHCLIENT	DHCP	DOMAINNAME	EKLOGIN
FONTSERVER	GATED	IPXRIP	KERBEROS
KLOGIN	KSHELL	NETCONTROL	NFS
NFSV3	NOT	NTP	PCNFSD
RACOON	RARP	"R" Server	RDISC
REXEC	RLOGIN	RPCBOOTPARAMS	RPCLOCKMGR
RPCMOUNT	RPCPORTMAP	RPCQUOTAD	RPCSTATUS
RSHELL	SAP	SNMP	SSH
SYSLOG	TELNET	TFTP	UCXQIO
VIADECNET	VIAPSI	XDM	

For loadable services (those with an INIT setting of Merge_image), you can use the SERVER-CONFIG SET PROCESS *process_name* command to have the service run in an auxiliary master server process with the specified name, rather than in the main master server process (which has a process name of MULTINET_SERVER).

Unlike earlier versions of MultiNet:

- The auxiliary server no longer has to be manually started.
- The service(s) running in the auxiliary process can be controlled with NETCONTROL.

Use the SERVER-CONFIG SET FLAGS START_AUX_SERVER command to have the main master server start the auxiliary server process automatically.

Services running in auxiliary master server processes can be controlled with NETCONTROL, independent of whether the START_AUX_SERVER flag is set, or whether the auxiliary process was started manually.

Auxiliary server processes are most useful for services which may require very large quantities of process quotas such as virtual memory. They can also be useful when there are problems with a loadable

service that cause the master server to terminate abnormally; the service can be isolated in a separate process to prevent other services from being interrupted when the abnormal termination occurs.

The following command shuts down all services, including those run in auxiliary master server processes. (In earlier versions, services running in auxiliary master server processes were not affected by this command.)

```
$ MULTINET NETCONTROL NETCONTROL SHUTDOWN ALL
```

In addition, specifying a SET PROCESS command on the NETCONTROL service (in SERVER-CONFIG) has no effect; the NETCONTROL server always runs in the main master server process.

Invoke NETCONTROL with MULTINET NETCONTROL, or with NETCONTROL commands from inside the Server Configuration Utility or NFS Server Configuration Utility.

NETCONTROL connects to the NETCONTROL server on the local host, or on a remote host if one is specified. The following example demonstrates two ways of connecting to the RLOGIN service. Note: the NETCONTROL prompt indicates the name of the service to which you are connected.

```
$ MULTINET NETCONTROL RLOGIN
Connected to NETCONTROL server on "LOCALHOST"
< SP1.EXAMPLE.COM Network Control 5.6 (nnn) at Mon 15-Mar-2020 7:42am-EST
RLOGIN>
$ MULTINET NETCONTROL
Connected to NETCONTROL server on "LOCALHOST"
< SP1.EXAMPLE.COM Network Control 5.6 (nnn) at Mon 15-Mar-2020 7:42am-EST
NETCONTROL>SELECT RLOGIN
RLOGIN>
```

The following example shows how to specify a NETCONTROL command from the command line. When used this way, NETCONTROL exits to DCL upon completion.

```
$ MULTINET NETCONTROL RLOGIN SHOW
Connected to NETCONTROL server on "LOCALHOST"
< SP1.EXAMPLE.COM Network Control 5.6 (nnn) at Mon 15-Mar-2020 7:42am-EST <
File Cache:
< MULTINET:HOSTS.EQUIV  (Expires in 59 minutes)
< USERS:[MIGUEL].rhosts  (Expired)
< Authorization Cache:
< EDUARDA  (Expires in 59 minutes)
< MIGUEL   (Expired)
$
```

You can also use NETCONTROL to control the MULTINET_SERVER on a remote system, subject to the restrictions set on that system's NETCONTROL server.

The following example shows how to invoke NETCONTROL on a remote system.

```
$ MULTINET NETCONTROL/HOST=SP1.EXAMPLE.COM RLOGIN
Connected to NETCONTROL server on "SP1.EXAMPLE.COM"
< SP1.EXAMPLE.COM Network Control 5.6 (nnn) at Mon 15-Mar-2020 7:42am-EST
RLOGIN>
```

To change the ACCOUNTING and DEBUG parameters with the MULTINET NETCONTROL command, use the following commands. Use the DEBUG parameter to dynamically set the MultiNet server debugging level to the specified value *n*. By default, additional information is provided in the accounting record by the MultiNet server. You can disable this feature by setting *n* to 0. When set to 1, the remote name and service name are added to the ACCOUNTING record.

```
$ MULTINET NETCONTROL NETCONTROL DEBUG n
$ MULTINET NETCONTROL NETCONTROL ACCOUNTING n
```

The below table shows the NETCONTROL commands you can use at any time.

Command	Description
LIST	Prints a list of the active services that support NETCONTROL.
NOOP	Does nothing; provided for testing the NETCONTROL server.
QUIT	Exits NETCONTROL.
QUOTE *string*	Sends a string verbatim to the NETCONTROL server.
SELECT *service*	Selects a service on which to operate. Use the question mark (?) character to get a list of services.
SERVER-VERSION	Prints the version number of the MULTINET_SERVER process.
STATISTICS	Prints server usage statistics.
TIMERS	Prints debugging information about the various scheduler events in the MULTINET_SERVER process.
VERBOSE	Turns on verbose printing of NETCONTROL commands and responses.
VERSION	Prints the version number of the selected server.

The below table shows the NETCONTROL commands you can use with the ACCESS server.

Command	Description
DEBUG *n*	Sets the ACCESS debugging level. The larger the number, the more verbose the output.
RELOAD *n*	Rereads your access configuration file MULTINET:START_ACCESS.COM if it is not redefined with the configuration file parameter.

The below table shows the NETCONTROL ACCOUNTING commands you can use.

Command	Description
ACC-CONTROL-VERSION	Shows the version of the accounting control logs.
FILE <*file_specification*>	Starts a new accounting control image.
NOOP	Does nothing.
RELOAD	Restarts the accounting server.
SHUTDOWN	Stops the accounting server.
START	Starts the accounting server.
VERSION	Displays the version of the accounting server control image.

The below table shows the NETCONTROL commands you can use with the BOOTP server.

Command	Description
DEBUG *n*	Sets the debugging log level.
DUMP	Dumps the BOOTP database.

RELOAD	Reloads the BOOTP database.

The below table shows the NETCONTROL command you can use with the BWNFSD server.

Command	Description
DEBUG *n*	Sets the debugging log level.

The below table shows the NETCONTROL commands you can use with the CLUSTERALIAS server.

Command	Description
DEBUG *n*	Sets the CLUSTERALIAS debugging level. The larger the number, the more verbose the output.
RELEASE *ip address*	Releases the system's lock of the cluster alias address.
SHOW *n*	Displays the state of the cluster alias.

The below table shows the NETCONTROL commands you can use with the DHCP client.

Command	Description
DEBUG *value*	Specifies a decimal integer that is a bitmask of debugging levels used to select messages to pass to OPCOM and the debug log file specified in the DEBUG-FILE parameter. The debugging levels are (in decimal): 1 Fatal Errors 3 Errors and Warnings 7 Informationals 15 Debug Messages 31 Dump Packets (Formatted) 63 Dump Packets (Hex)

	By default, Fatal Errors and Warnings are logged.
VERSION	Prints the version number of the DHCP client control component.
SHUTDOWN	Causes the DHCP client to shut down. **Note:** This command does not delete the service from the Master Server. It is still registered with NETCONTROL.
START	Starts a DHCP client that has been down by the DHCLIENT SHUTDOWN command.

The below table shows the NETCONTROL commands you can use with the DHCP server.

Command	**Description**
DEBUG *value*	Specifies a decimal integer that is a bitmask of debugging levels used to select messages to pass to OPCOM and the debug log file specified in the DEBUG-FILE parameter. The debugging levels are (in decimal): 1 Fatal Errors 3 Errors and Warnings 7 Informationals 15 Debug Messages 31 Dump Packets (Formatted) 63 Dump Packets (Hex) By default, Fatal Errors, Errors, and Warnings are logged.
DHCP-CONTROL-VERSION	Prints the version number of the DHCP Control component.

DUMP *file*	Writes the in-memory configuration of the DHCP server to the file specified or to the DUMPFILE *parameter* value if *file* is not specified.
NEWLOG	Starts a new debug log file, if one is in use.
PARTNERDOWN	For Safe-Failover DHCP: Causes the DHCP server to transition into Partner Down state, which indicates that its safe-failover DHCP partner is down.
RELEASE *ip-address*	Forces the DHCP server to act as if it heard a DHCP release from the client. This applies to dynamically assigned IP addresses only. **Note:** the DHCP protocol has no way to tell the client that the address is released, so this command must be used with caution.
RESTART or RELOAD	Causes the server to restart, at which time it rereads the configuration file.
SHOW SHOW ALL SHOW CID *client-identifier* SHOW CLIENT *ip-address* SHOW HADDR *hardware-address*	Prints a variety of information, depending on the arguments used. Takes no arguments. Displays on your computer screen the SHOW SUBNET output for all subnets followed by information about all static assignments in the DHCP server configuration. Shows all lease binding and static assignment details for the specified client identifier. Shows lease binding details for the specified IP address. The IP address must be in a dynamic address pool.

SHOW ISKNOWN HOST *hardware-address-or-client-identifier* SHOW ISKNOWN SUBCLASS *class-name subclass-data* SHOW SUBNET *ip-address* SHOW LEASES SHOW POOLS	Shows all lease binding and static assignment details for the specified hardware address. Shows whether the given hardware address or client identifier is "known", that is if there is a *host* declaration for that hardware address or client address. Shows whether the given subclass data exists as a subclass within the given class. Shows the DHCP address pools for the shared network that *ip-address* is in. Lists each subnet that is on the shared network and each IP address in each pool. Shows all IP addresses with leases (pending, active, or expired) for each shared network. Uses the same format as the SHOW ALL command. For each address pool, shows the total number of IP addresses, number of abandoned IP addresses, number reserved for secondary, and number available to be leased. Uses a table format.
SHUTDOWN	Stops the server. This command does not delete the service from the master server. It is still registered with NETCONTROL.
START	Starts a server that has been shut down by the DHCP SHUTDOWN command.
STATISTICS	This command is supplied only for backward compatibility with previous versions of MultiNet. It has been superseded by the SHOW POOLS command.

UPDATE [(*file*)]	Causes the server to execute the commands in the specified file, if any, or the file specified in the UPDATEFILE parameter, if any, or the default file MULTINET:DHCPD.UPDATES.

The below table shows the NETCONTROL commands you can use with the DOMAINNAME server.

Command	**Description**
DEBUG *n*	Sets the debug level of the domain name server (the default is no debugging). The larger the number, the more verbose the output. A value of 0 turns off debugging.
DUMP	Dumps the Domain Nameserver cache to the file MULTINET:DOMAIN-NAME-SERVICE.DB. Use to diagnose database problems.
MAXIMUM-TTL *ttl*	Changes the maximum time-to-live (TTL) that resource records are cached from the default of 604800 seconds (1 week) to the specified value.
MINIMUM-TTL *ttl*	Changes the minimum time-to-live (TTL) that resource records are cached from the default of zero (0) seconds to the specified value. It is recommended you use this command only if there is a specific need. This could cause problems in that you may be caching resource records for longer than the authoritative administrator intended.
QUERYLOG	Toggles query logging ON and OFF. Query logging shows an informational message every time a query is received by the server. Query logging can be directed to OPCOM or a file in the MULTINET:NAMED.CONF file using the logging category *queries*.
RELOAD	Causes the domain name server to re-read the configuration file, and subsequently re-read any zone files that have changed.
RESTART	Instructs the name server to shutdown if it exists, then instructs the master server to start a new name server process.

REWRITE-TTL *ttl*	Sets the time-to-live (TTL) that load balanced resource records are cached from the default of 300 seconds (5 minutes) to the specified value.
SHOW	Shows the nodename, address, and rating of any cluster server names.
START	Instructs the master server to start the name server process.
STATISTICS	Appends domain name server server statistics to the file MULTINET:DOMAIN-NAME-SERVICE.STATS and memory statistics to the file MULTINET:DOMAIN-NAME-SERVICE.MEMSTATS.
STOP or SHUTDOWN	Stops the server. This command does not delete the service from the master server. It is still registered with NETCONTROL.
VERSION	Prints the current DNS server version number. (This is the version of BIND from which the MultiNet DNS server is derived.)

The below table shows the NETCONTROL commands you can use with the EKLOGIN server:

Command	**Description**
DEBUG *n*	Sets the debugging log level.
FLUSH-CACHE	Flushes the “KR” services authentication cache.

The below table shows the NETCONTROL commands you can use with the FONTSERVER server.

Command	**Description**
DEBUG	Sets the FONTSERVER debugging level. The larger the number, the more verbose the output.

FLUSH	Removes the fonts loaded into the font server's cache.
FS-CONTROL-VERSION	Prints the version number of the FONTSERVER control component.
RELOAD	Reloads the font server configuration file MULTINET:FONT_SERVER.CONFIGURATION.
RESET	Resets the server and closes down all connections to the client.
RESTART	Restarts the server.
START	Starts the server.
SHUTDOWN	Stops the server.

The below table shows the NETCONTROL commands you can use with the IPXRIP server.

Command	**Description**
ADVERTISE true | false	Advertises non-local routes. The default is true.
DEBUG *n*	Sets the debugging log level.
FLUSH	Flushes the non-local routes and updates the interface configuration.
SEND	Sends the IPX RIP packets.
SHOW	Displays the IPX RIP routing table.

The below table shows the NETCONTROL command you can use with the KERBEROS V4 server.

Command	**Description**
DEBUG *n*	Sets the debugging log level.

The below table shows the NETCONTROL commands you can use with the KLOGIN server.

Command	Description
DEBUG *n*	Sets the debugging log level.
FLUSH-CACHE	Flushes the “KR” services authentication cache.

The below table shows the NETCONTROL commands you can use with the KSHELL server.

Command	Description
DEBUG *n*	Sets the debugging log level.
FLUSH-CACHE	Flushes the “KR” services authentication cache.

The below table shows the NETCONTROL commands you can use with the NETCONTROL server.

Command	Description
ACCOUNTING *n*	Disables accounting with an integer value of 0. A positive value enables accounting. The accounting value is checked whenever the master server starts up a service. The accounting provided is PID, host name, node name, and service name.
DEBUG *n*	Sets the NETCONTROL debugging level. The larger the number, the more verbose the output.
SHUTDOWN *all* or service name	Shuts down all services or shuts down a specific services.

The below table shows the NETCONTROL commands you can use with the NFS v2 Server.

Command	Description
ADD MOUNT-RESTRICTION	Exports a mount point for *read-only access*. Any attempt to write to the disk specified by this mount point fails. This restriction affects any NFS group associated with that particular mount point.
AVERAGE-RESPONSE-TIMES	Computes average response times. Do not use without first contacting Process Software Technical Support.
DUMP	Dumps the NFS Server cache into the file MULTINET:NFS_SERVER.DUMP for debugging; DUMP is ignored unless the NFS Server is running in user mode.
FILECACHE-DEBUG *n*	Sets the debug level of the file system cache portion of the server.
NFS-CONTROL-VERSION	Prints the version number of the NFS NETCONTROL module.
NFSDEBUG *n*	Sets the debug level of the NFS protocol portion of the server.
RECORD-RESPONSE-TIMES	Records response time data. Do not use without first contacting Process Software Technical Support.
RELOAD	Reloads the UID/GID table in the NFS Server. This command does not reload the client; that is done with NFSMOUNT/RELOAD. If you want to reload both, use MULTINET CONFIGURE /NFS when you modify the configuration file.
RESTART	Restarts the NFS Server process.
RPCDEBUG *n*	Sets the debug level of the RPC protocol portion of the server.
SHOW-RESPONSE-TIMES	Displays response time data. Do not use without first contacting Process Software Technical Support.
SHUTDOWN	Stops the server.
START	Starts the server.

The below table shows the NETCONTROL commands you can use with the NFS v3 Server.

Command	Description
RESTART	Restarts the NFS v3 Server process.
SHUTDOWN	Stops the server.
START	Starts the server.

The below table shows the NETCONTROL commands you can use with the NOT server.

Command	Description
DEBUG	Sets the NOT debugging level. The larger the number, the more verbose the output.
RELOAD	Reloads the NOT.CONFIGURATION or the configuration file the parameter for the NOT service is set to.

The below table shows the NETCONTROL commands you can use with the NTP server.

Command	Description
DEBUG *n*	Sets the debug level of the NTP server. The larger the number, the more verbose the output.
NOOP	Does nothing but verify that the server is running.
NTP-CONTROL-VERSION	Displays the version information for the NTP NETCONTROL interface.
PANIC *seconds*	Sets the largest value (in seconds) that will be corrected. The default is 4000 seconds, or just over 66 minutes.
RELOAD	Restarts the NTPD server process. Equivalent to SHUTDOWN followed by START

SHOW	Shows the current state of all server and peer connections. Also displays the current value of WAYTOOBIG/PANIC.
SHUTDOWN	Causes the NTPD server process to shut down and exit.
START	Starts the NTPD server process.
VERSION	Displays the version of the NTP server in use.
WAYTOOBIG	A synonym for PANIC. Retained for historical reasons.

The below table shows the NETCONTROL command you can use with the PCNFSD server.

Command	Description
DEBUG *n*	Sets the debugging log level.

The below table shows the NETCONTROL commands you can use with the RACOON server.

Command	Description
DEBUG *n*	Sets the debugging level
DELETE	Delete an established key exchange session.
ESTABLISH *remote-ip-address* [*local-ip-address*]	Initiate key exchange protocol communication between the *remote-ip-address* and the *local-ip-address*. If *local-ip-address* is not specified then the value of MULTINET_HOST_NAME is used. This does not install security associations, but does the initial negotiation necessary to allow security associations to be established when necessary. It is not necessary to manually establish the negotiation information – RACOON will do it automatically when necessary.
FLUSH	Flush existing key exchange sessions.

NOOP	No operation
SHOW	Shows the current state of key negotiation between IP addresses
SHUTDOWN	Shutdown Racoon
START	Start Racoon
STOP	Stop Racoon (equivalent to SHUTDOWN)
VERSION	Version of the control interface

The below table shows the NETCONTROL commands you can use with the RARP server.

Command	Description
DEBUG *n*	Sets the RARP debugging level. The larger the number, the more verbose the output.
RELOAD	Reloads the RARP database.

The below table shows the NETCONTROL commands you can use with NETCONTROL REXEC, NETCONTROL RLOGIN, or NETCONTROL RSHELL.

Command	Description
DEBUG	Sets the R services debugging level. The larger the number, the more verbose the output.
FLUSH-CACHE	Flushes the R services authentication caches.
SHOW-CACHE	Prints the contents of the R services authentication caches.

The below table shows the NETCONTROL commands you can use with the RPCBOOTPARAMS server.

Command	Description
`DEBUG n`	Sets the debugging log level.
`RELOAD`	Reloads the RPC boot parameters for diskless hosts.

The below table shows the `NETCONTROL` commands you can use with the RPCLOCKMGR server.

Command	Description
`DEBUG n`	Sets the debugging log level.
`NOOP`	Does nothing; provided for testing the NETCONTROL server.
`RPCLOCKMGR-CONTROL-VERSION n`	Displays the version number of the RPC Lock Manager Control Server.
`SHOW n`	Shows the locks associated with this server.
`START`	Starts the RPC Lock Manager.
`STOP`	Stops the RPC Lock Manager.
`TIMERS n`	Displays the timers.

The below table shows the `NETCONTROL` commands you can use with the RPCMOUNT server.

Command	Description
`CLEAR`	Clears the database of clients that have file systems mounted.
`DEBUG n`	Sets the RPCMOUNT debugging level. The larger the number, the more verbose the output.

DUMP	Forces RPCMOUNT to write the current mount database to the on-disk cache; it is normally only written every few minutes.
RELOAD	Reloads export and restriction databases from the NFS.CONFIGURATION file.
SHOW	Prints the current mount database.

The below table shows the NETCONTROL commands you can use with the RPC Portmapper server.

Command	Description
DEBUG *n*	Sets the RPCPORTMAP debugging level. The larger the number, the more verbose the output.
SHOW	Prints the current portmap database.

The below table shows the NETCONTROL command you can use with the RPCQUOTAD server.

Command	Description
DEBUG *n*	Sets the debugging log level.

The below table shows the NETCONTROL commands you can use with the RPCSTATUS server.

Command	Description
DEBUG *n*	Sets the RPCSTATUS debugging level. The larger the number, the more verbose the output.
RELOAD	Reloads the RPCSTATUS database.
SHOW	Shows the parameters governing RPCSTATUS.
SIMULATE-CRASH	Causes the server to notify all monitoring clients that the system has crashed.

	Do not use this command without first contacting Process Software Technical Support.

The below table shows the NETCONTROL commands you can use with the SNMP server.

Command	Description
RELOAD	Causes the SNMP Agent to reread the configuration file.
SHUTDOWN	Stops the SNMP Agent. This command does not delete the service from the master server. It is still registered with NETCONTROL.
SNMP-CONTROL-VERSION	Prints the version number of the SNMP Agent component.
START	Starts a SNMP Agent that has been shut down by the SNMP SHUTDOWN command.

The below table shows the NETCONTROL commands you can use with the SSH server.

Command	Description
DEBUG	Toggles debugging on/off in all SSHD daemon processes running on the server.
MASTER_RESTART	Stops and restarts only the SSHD Master process. All other SSH processes and users are not affected.
RESTART	Stops and restarts the server. This stops not only the SSHD_MASTER process but also all SSHD processes running on the server, which has the effect of logging out all SSH sessions currently active on the server.

SHOW	Displays information on all executing daemon processes. For example: BIGBOOTE_$ **mu netcontrol ssh show** Connected to NETCONTROL server on "LOCALHOST" < bigboote.example.com Network Control V5.6(10) at Tue 11-May-2019 11:41AM-EDT < SSHD Master PID = 20800099 < Process "SSHD 0000" (pid 20800156) < User = ALICE < From system 217.225.329.75 port 1064 < Started: 05/09/2018 17:41:43 < Bytes in: 23 out: 425 (from child process: 425) < PTD Device = _FTA5: < Current child process = "ALICE_@FTA5" (pid 20800157) < End of Show SSH BIGBOOTE_$
SHUTDOWN	Stops the server. This stops not only the SSHD_MASTER process but also all SSHD processes running on the server, which has the effect of logging out all SSH sessions currently active on the server.
START	Starts the server.

The below table shows the NETCONTROL commands you can use with the SYSLOG server.

Command	**Description**
DEBUG *n*	Sets the debugging log level.
RELOAD	Reloads the configuration file.

The below table shows the NETCONTROL commands you can use with the TELNET server.

Command	**Description**

DEBUG	Sets the TELNET debugging level. The larger the number, the more verbose the output.
TRACE *n*	Sets the trace level.

The below table shows the NETCONTROL commands you can use with the TFTP server.

Command	Description
DEBUG *n*	Sets the TFTP debugging level. The larger the number, the more verbose the output.
RELOAD	Reloads the TFTP server parameters (the default TFTP directory and MULTINET:TFTP.FILENAME-TRANSLATIONS).
SHOW	Prints the current status of the TFTP server.
SHOW-TRANSLATION	Shows the TFTP file name translation table.

The below table shows the NETCONTROL command you can use with the UCXQIO server.

Command	Description
DEBUG *n*	Sets the UCXQIO debugging level. The larger the number, the more verbose the output.

The below table shows the NETCONTROL commands you can use with the VIADECNET server.

Command	Description
DEBUG *n*	Sets the VIADECNET debugging level. The larger the number, the more verbose the output.
RELOAD	Reloads the configuration from the network kernel.
SHUTDOWN	Stops the server.

The below table shows the NETCONTROL commands you can use with the VIAPSI server.

Command	Description
DEBUG *n*	Sets the VIAPSI debugging level. The larger the number, the more verbose the output.
DISCONNECT *interface-or-DTE*	Clears the X.25 connection for the given interface (for example, psi0) or peer DTE.
IDLE *seconds*	Sets the default connection-idle interval.
RELOAD	Checks the PSI devices for the configuration, and if changes were made, reinitializes the configuration.
SHUTDOWN	Stops the server; not yet implemented.

The below table shows the NETCONTROL commands you can use with the XDM server.

Command	Description
DEBUG	Sets the XDM debugging level. The larger the number, the more verbose the output.
RELOAD	Causes the XDM server to reload its configuration file.
RESTART	Restarts the XDM server.
SHOW	Shows the status of all managed displays.
START	Starts the server.
SHUTDOWN	Stops the server.
XDM-CONTROL-VERSION	Prints the version number of the XDM Control component.

MULTINET NFSDISMOUNT

Dismounts a remotely mounted NFS file system.

Format

`MULTINET NFSDISMOUNT` *`mount_device`*

Parameters

`mount_device`
Specifies an NFS*x*: device associated with a remotely mounted file system.

Qualifiers

`/ALL [`*`host_name`*`]`
Specifies that `NFSDISMOUNT` notifies remote systems that no file systems are currently mounted (this is usually used as part of the reboot procedure).

> **Note:** `NFSDISMOUNT/ALL` does not dismount file systems, but rather notifies an NFS server that the local system does not have any mounted. If you do not specify *`host_name`*, `NFSDISMOUNT` broadcasts the request to the local network.

`/LOG`
Specifies that `NFSDISMOUNT` displays information when a dismount occurs.

Example

This example shows how to dismount a remotely mounted file system attached to the local mount device `NFS3:`.

```
$ MULTINET NFSDISMOUNT NFS3:
```

MULTINET NFSMOUNT

Mounts a remote NFS file system so it can be used locally. NFSMOUNT requires CMKRNL, SETPRV, SYSPRV, SYSNAM, ALTPRI, DETACH, ACNT, and SYSLCK privileges.

Format

MULTINET NFSMOUNT *node*::*mount_point logical_name*

Parameters

node
Specifies the name of the computer serving the file system to the network.

mount_point
Specifies the portion of an NFS file system to be mounted. The format of the specified mount point depends on the server. Enclose *mount_point* in quotes if it contains special or lowercase characters.

logical_name
Specifies an OpenVMS logical name to assign to the mount device. OpenVMS users can access remote files using this logical name.

Qualifiers

/FID_CACHE=*size*
Specifies the size (in bytes) of the File Identifier (FID) cache. Values range from 5 to 5000000. The default is 10000.

/LOCKING=([local,] [network])
Specifies the type of file locking used on the NFS-mounted file system.

- LOCAL - Specifies that OpenVMS file-locking operations will only be consistent on the local system.
- NETWORK - Uses the NFS Network Lock Manager to obtain consistent file locking between all NFS clients. Use of the /LOCKING=NETWORK qualifier adds protocol overhead.

If you specify both LOCAL and NETWORK, network locking is attempted; if the remote lock manager cannot be contacted, local locking is used.

/PAGEFILE=*pages*
Specifies the page file quota for the NFS_CLIENT_ACP process. This process is created when the first NFS file system is mounted. This qualifier is ignored on subsequent mounts. The default is 65535 pages.

/PORT=*port_number*
Specifies the remote port to connect to for NFS service.

/PRIORITY=*priority*
Specifies the base process priority of the NFS_CLIENT_ACP process created when the first NFS file system is mounted. The qualifier is ignored on subsequent mounts. The default is 7.

/PROCESSOR={UNIQUE | SAME}
Determines whether a separate ACP process is created for every NFS device. This mechanism allows NFS devices to function in parallel so that one NFS device does not have to wait for an NFS operation on another NFS device to complete. Multiple ACPs allow for multiple outstanding I/O, and operations happen in parallel.

A setting of UNIQUE creates a separate NFS_CLIENT_*n* process for each mount, where *n* is the number of the NFS device (such as NFS_CLIENT_2, which corresponds with the device NFS2).

A setting of /PROCESSOR=SAME=*nfs_device* assigns the mount to the same ACP process as the specified *nfs_device*. For example, /PROCESSOR=SAME=NFS3 assigns this mount to the NFS_CLIENT_3 ACP process.

Any mounts specified without the /PROCESSOR qualifier use a single, default process.

It is recommended that you use the /PROCESSOR qualifier to group mounts based on the remote server. That way, if the server goes down, it does not cause access to other servers to hang. (You can use the /SOFT qualifier to permit NFS operations to time out instead of hanging indefinitely.)

/READ_SIZE=*read_size*
Specifies the maximum size of the read operations the NFS client performs. The default, 8192, is correct for most servers.

/RELOAD
Instructs the NFS client software to reload its UID translation table from the NFS configuration file.

```
                        [ advisory_close, ]
                        [ case_insensitive_filenames, ]
                        [ nofdl_files, ]
                        [ nolinks, ]
                        [ nostream_conversion, ]
                        [ nounique_fileno, ]
/SEMANTICS=(            [ noversions, ]
                        [ novms_access_checking, ]
                        [ preserve_dates, ]
                        [ upper_case_default, ]
                        [ vms_filenames, ]
                        [ vms_server, ]
```

Specifies the capabilities and characteristics of the NFS server that control the behavior of the MultiNet NFS client, as described in the following table.

Attribute	Description
ADVISORY_CLOSE	Sends a VMS server a command to close the file when there are no more references to it on the client.
CASE_INSENSITIVE_FILENAMES	Specifies that UNIX files accessed by an OpenVMS system not have their file names converted using the conversion characters (see HELP MULTINET File_Name_Character_Map for a list of these characters). Use this option when an NFS server treats all file names as case-insensitive. When this option is set, all file names accessed through NFS are converted to lowercase. When returned to the server, they are handled in lowercase. The NFS server must be able to accept lowercase file names. This option is disabled by default.
NOFDL_FILES	Disables the use of .fdl files by the MultiNet NFS client to store RMS attributes. This option must be used if the NFS server

	doesn't allow these file names. Its use severely limits the ability of the NFS client to store record attributes.
NOLINKS	Disables the automatic creation of hard links to the latest version of a file. The NFS client normally uses a hard link operation to link the top version of a file name `foo.bar;12` to the unversioned name `foo.bar` for more convenient access from the NFS server side. This option may be used either to reduce the overhead in creating it or if the NFS server does not support hard links.
NOSTREAM_CONVERSION	Disables the automatic conversion of text files to STREAM format. The NFS client normally converts requests to create Variable Length Record Carriage Return Carriage Control files into requests to create Stream files. This option disables this conversion.
NOUNIQUE_FILENO	Specifies whether or not the NFS server is to generate unique file numbers for each file (most NFS servers do). If the client knows that file numbers are unique, it uses a faster algorithm to refresh stale directory entries in the cache. Use of this qualifier disables the faster refresh algorithm, and is equivalent to the `/NOUNIQUE_FILENO` qualifier.
NOVERSIONS	Disables support for multiple file versions. The NFS client normally stores multiple versions of OpenVMS files by using the semicolon character in the file name on the NFS server side. You must use this option to disable the ability to create multiple versions of files if the NFS server does not support file names with the semicolon character.
NOVMS_ACCESS_CHECKING	Specifies that the client does not perform a full OpenVMS access check, including a check for ACLs and security alarms. If this option is not specified, the NFS client considers ACLs and security alarms when granting or denying access.
PRESERVE_DATES	Allows you to store VMS-style dates and times for files.

UPPER_CASE_DEFAULT	Assumes file names are in uppercase on the server until it sees the $ character used to toggle case.
VMS_FILENAMES	Specifies that the NFS client should not perform the usual mapping between OpenVMS and UNIX-style file names. This option can be used to permit all OpenVMS file names to be stored using the NFS client; however, its use prevents the NFS client from being used to access files which do not conform to the OpenVMS file name conventions.
VMS_SERVER	Specifies that the NFS server is a MultiNet NFS server of revision V3.0 or later and supports OpenVMS-specific extensions to the NFS protocol to store file attributes. If the NFS server does not support these extensions, the mount will fail. This option is equivalent to the /VMS_SERVER qualifier and overrides any other semantics specified.

/SOFT
Specifies that, if the NFS client is unable to reach the NFS server after the time period specified by /TIMEOUT, an error is returned to the user (SS$_UNREACHABLE). If the file system is mounted without the /SOFT qualifier, the NFS client retries the operation forever.

/TIMEOUT=*timeout*
Specifies the total time, in tenths of a second, that it takes for an RPC request to timeout. Retries are attempted via UDP for an interval of one-fifth the value specified for /TIMEOUT. The minimum value allowed for this setting is 30 tenths of a second. /TIMEOUT does not affect TCP timeouts.

/TRANSPORT=([tcp | udp])
Specifies the underlying transport used for the NFS requests. (The default is UDP if /TRANSPORT is not specified.) The TCP transport can be used with servers that support it. If you specify both transports, TCP is tried first; if it fails, the mount uses UDP.

/UNIQUE_FILENO
/NOUNIQUE_FILENO

Specifies whether or not the NFS server is to generate unique file numbers for each file (most NFS servers do). If the NFS client knows that file numbers are unique, it uses a faster algorithm to refresh stale directory entries in the cache. The default /NOUNIQUE_FILENO qualifier is equivalent to /SEMANTICS=NOUNIQUE_FILENO.

/VMS_SERVER
Specifies that the NFS server is a MultiNet NFS server of revision V3.0 or later and supports OpenVMS-specific extensions to the NFS protocol to store file attributes. If the NFS server does not support these extensions, the mount fails. This qualifier is equivalent to /SEMANTICS=VMS_SERVER and overrides any other semantics specified.

/VOLUME=*volume_name*
Specifies the display name of the mounted volume (which appears via SHOW DEVICE). (The default is the remote *mount_point* name.)

/WRITE
/NOWRITE
Specifies whether or not the file system is to be mounted for both read and write access. /NOWRITE prevents users from modifying the file system. /WRITE is the default.

/WRITE_SIZE=*write_size*
Specifies the maximum size of packets written by the NFS client. The default, 8192, is correct for most servers.

/WSEXTENT=*pages*
Specifies the working set extent for the NFS_CLIENT_ACP process. This process is created when the first NFS file system is mounted. The qualifier is ignored on subsequent mounts. The default is 20000 pages.

/WSQUOTA=*pages*
Specifies the working set quota for the NFS_CLIENT_ACP process. This process is created when the first NFS file system is mounted. The qualifier is ignored on subsequent mounts. The default is 2000 pages.

Examples

This example shows how to mount the remote file system `/usr` on the server named "sunset" on the local mount device `NFS3:`.

```
$ MULTINET NFSMOUNT SUNSET::"/usr" disk$sunset
%NFSMOUNT-I-MOUNTED, SUN::/ufs NFS mounted on _NFS3:
$
```

This example illustrates the use of `/PROCESSOR=UNIQUE`, creating four ACP processes - one for each device.

```
$ MULTINET NFSMOUNT/VMS/PROCESSOR=UNIQUE SCOOBY::USERS: SCOOBY1
$ MULTINET NFSMOUNT/VMS/PROCESSOR=UNIQUE SCOOBY::USERS2: SCOOBY2
$ MULTINET NFSMOUNT/VMS/PROCESSOR=UNIQUE SHAGGY::USERS:  SHAGGY1
$ MULTINET NFSMOUNT/VMS/PROCESSOR=UNIQUE SHAGGY::USERS2: SHAGGY2
```

This example illustrates the use of `/PROCESSOR=SAME`. In this example, all access to the server named SCOOBY goes through one ACP process, and all access to SHAGGY goes through another process.

```
$ MULTINET NFSMOUNT/VMS SCOOBY::USERS: SCOOBY1
$ MULTINET NFSMOUNT/VMS/PROCESSOR=SAME=SCOOBY1 SCOOBY::USERS2: SCOOBY2
$ MULTINET NFSMOUNT/VMS SHAGGY::USERS: SHAGGY1
$ MULTINET NFSMOUNT/VMS/PROCESSOR=SAME=SHAGGY1 SHAGGY::USERS2: SHAGGY2
```

MULTINET NSLOOKUP

Performs test queries on the domain name service (DNS) system. When invoked with no parameters, `MULTINET NSLOOKUP` allows commands to be run interactively. The below table lists the commands that can be run in interactive mode.

Command	Description
name	Prints information about *name* using the default server.
name server	Prints information about *name* using *server*.
exit	Exits NSLOOKUP.
finger [*user*]	Finger the optional *user* at the current default host.
help or ?	Prints help information.
set all	Prints the current status of all options.
set class=*class*	Sets the query class to one of these: IN, CHAOS, HESIOD, or ANY.
set [no]debug	Prints debugging information.
set [no]d2	Prints exhaustive debugging information.
set [no]defname	Appends the domain name to each query.
set [no]recurse	Asks for a recursive answer to a query.
set [no]vc	Always uses a virtual circuit.
set domain=*name*	Sets the default domain name to *name*.
set port=*port*	Sets the port number on which to send a query.

`set root=name`	Sets the root name server to *name*.
`set retry=n`	Sets the number of retries to *n*.
`set srchlist=name1 [/name2/.../name6]`	Sets the domain to *name1* and the search list to *name1* through *name6*.
`set timeout=n`	Sets the timeout interval to *n*.

`set query-type=type`
 	
or
 	
`set type=type`
	Sets the resource record (RR) type to query for.

`server name`	Sets the default server to *name*, using the current default server.
`lserver name`	Sets the default server to *name*, using the original default server.
`root`	Sets the current default server to the root.
`ls [option] name [>file]`	Lists the domain *name*, with output optionally going to *file*. *option* is one of the following: `-a` List fully-qualified names and aliases `-h` List HINFO (CPU type and operating system) `-s` List well-known services `-d` List all records `-t type` List records of the given type (such as A, CNAME, and MX)

Format

`MULTINET NSLOOKUP [name] [nameserver]`

Parameters

name
Specifies a host or domain name.

nameserver
Specifies the name server to query.

Qualifiers

/CLASS=*recordclass*
Specifies which class records are asked for. Valid classes are `ANY`, `IN`, `CHAOS`, and `HESIOD`. (The default is `/CLASS=IN`, Internet records.)

/DEBUG
/NODEBUG
Causes the resolver to print debugging information, including formatted responses. `/NODEBUG` is the default.

/DEBUG2
/NODEBUG2
Causes the resolver to print formatted queries, and additional, less useful debugging information. The default is `/NODEBUG2`.

/DEFNAMES
/NODEFNAMES
Specifies that the resolver adds this system's domain name to any name not explicitly terminated with a period. `/DEFNAMES` is the default.

/DNSRCH
/NODNSRCH
Specifies that the resolver searches up the domain tree from this system's name for any name not explicitly terminated with a period. /DNSRCH is the default.

/DOMAIN=*domainname*

Specifies a default domain other than the domain of this host.

/IGNTC
/NOIGNTC
Tells the resolver to ignore truncation in responses. /NOIGNTC is the default.

/PORT=*port*
Specifies a port other than the standard nameserver port of 53.

/RECURSE
/NORECURSE
Requests that the name server use recursion to answer the query. Recursion is on by default.

/RETRY=*retrycount*
Specifies the number of retries the resolver makes when querying a name server via UDP (by default, 4).

/ROOT_SERVER=*rootservername*
Specifies a root name server other than A.ROOT-SERVERS.NET.

/TIMEOUT=*seconds*
Specifies a different period to wait for responses. The default is 4 seconds.

/TYPE=*recordtype*
Specifies which type resource records are asked for. The default is /TYPE=A (address records).

All standard DNS record types are supported. The table below gives a partial list of valid values for the /TYPE qualifier.

Resource Record	Description	Resource Record	Description
A	Address records	MR	Mail rename domain name

ANY	Any	MX	Mail exchanger
AXFR	Zone transfer	NS	Authoritative name server
CNAME	Fully-qualified name for an alias	PTR	Domain name pointer
GID	Group ID	SOA	Start of a zone of authority
HINFO	Host information	TXT	Arbitrary text
MAILB	Mailbox for a user	UID	User ID
MB	Mailbox domain name	UINFO	Arbitrary user information
MG	Mail group member	WKS	Well-known service description
MINFO	Mailbox or mail list information		

/VC
/NOVC
Specifies that the resolver uses virtual circuits instead of datagram queries. `/NOVC` is the default.

MULTINET NSUPDATE

Performs dynamic updates to the domain name service (DNS) server. NSUPDATE can read commands from a specified file or from the terminal.

NSUPDATE can be used with the UNIX-style syntax by defining it as a foreign command:

```
$ NSUPDATE :== $MULTINET:NSUPDATE
```

Both the UNIX-style options and the OpenVMS qualifiers are listed below.

NSUPDATE reads input records, one per line, each line contributing a resource record to an update request. All domain names used in a single update request must belong to the same DNS zone. A blank line causes the accumulated records to be formatted into a single update request and transmitted to the zone's authoritative name servers. Additional records may follow, which are formed into additional, completely independent, update requests. For the last request to be transmitted, a blank line must end the input.

Records take one of two general forms:

- *Prerequisite* records specify conditions that must be satisfied before the request will be processed.
- *Update* records specify changes to be made to the DNS database.

An update request consists of zero or more prerequisites and one or more updates. Each update request is processed atomically, that is, all prerequisites must be satisfied before all updates will be performed.

NSUPDATE understands the input record formats listed in the below table:

Command	**Description**
prereq nxdomain *name*	Requires that no RR of any type exists with name *name*.
prereq nxrrset *name* [*class*] *type*	Requires that no RR exists of the specified *type* and *name*.
prereq yxdomain *name*	Requires that at least one RR named *name* must exist.

prereq yxrrset *name* [*class*] *type [data...]*	Requires that a RR exists of the specified *type* and *name*. If ***data*** is specified, it must match exactly.
update add *name ttl* [*class*] *type data...*	Adds a new RR with specified *ttl*, *type*, and *data*.
update delete *name* [*class*] [*type* [*data...*]]	Deletes RRs named *name*. If *type* (and possibly *data)* is specified, only matching records will be deleted.

Format

MULTINET NSUPDATE [*filename*]

Parameters

filename
Specifies a file containing NSUPDATE commands to be executed.

Qualifiers

/DEBUG
/NODEBUG
Causes the resolver to print debugging information. Debugging is off by default.

-k *keydir+keyname*
/KEY=(KEYNAME=*key*[,KEYDIR=*directory*])
Specifies a TSIG key for NSUPDATE to use to sign its updates. The default value for KEYDIR is the current default directory.

> **Note:** On Unix, the syntax is *keydir:keyname*. On OpenVMS, the colon is replaced by a plus sign (+). The *keyname* must be specified to match the key and private filenames, with periods

instead of dollar signs. This may not match the domain name if DNSKEYGEN had to abbreviate it to fit into an OpenVMS file name.

-v
/VC
/NOVC
Specifies that the resolver uses virtual circuits (TCP) instead of datagram (UDP) messages. /NOVC is the default.

Examples

The following example illustrates the interactive use of NSUPDATE to change an IP address by deleting any existing A records for a domain name and then inserting a new one. Since no prerequisites are specified, the new record will be added even if there were no existing records to delete.

Note: The trailing blank line is required to process the request.

```
$ multinet nsupdate
> update delete test.example.com A
> update add test.example.com 3600 A 10.1.1.1
>
```

In this example, a CNAME alias is added to the database only if there are no existing A or CNAME records for the domain name.

```
$ multinet nsupdate
> prereq nxrrset www.example.com A
> prereq nxrrset www.example.com CNAME
> update add www.example.com 3600 CNAME test.example.com
>
```

MULTINET PING

Sends ICMP Echo Request packets to the specified host to measure network packet loss and latency. `MULTINET PING` returns the following status codes:

Status Code	Description
`SS$_NORMAL`	Successful PING.
`SS$_IVBUFLEN`	An invalid length was specified on the `/DATA_LENGTH` qualifier. The maximum value is 65468.
`SS$_NOSUCHNODE`	Failed attempt to PING an unknown host.
`SS$_PROTOCOL`	Remote system is not configured to support ICMP.
`SS$_NOPRIV`	Access to PING denied by the system manager.
`SS$_DATALOST`	Some PING responses were received, but some were lost; that is, a PING success rate of less than 100%.
`SS$_UNREACHABLE`	No responses were received.

`MULTNET PING6` performs the same function for IPv6 networks, sending ICMP6 packets over IPv6.

Format

`MULTINET PING` *`host`*

Parameter

`host`
Specifies the host to ping.

Qualifiers

/ADDRESSES
Sends a node information query packet instead of an ICMP6 echo request to request the addresses which the host responds to. Not all systems support node information query packets. This qualifier is only valid for `MULTINET PING6`.

/DATA_LENGTH=*number-of-bytes*
Specifies the number of bytes of data to attach to ICMP Echo Request packets. If not specified, a reasonable default value is supplied. Increase the data length to check for gateways that do not fragment IP packets correctly.

/DEBUG
/NODEBUG
Enables socket-level debugging in the MultiNet kernel. This qualifier is usually only useful for debugging the MultiNet kernel. Debugging is off by default.

/IPV6
Specifies that an IPv6 ping is desired. (The default is IPv4.)

/FLOOD
Indicates that `MULTINET PING` is used to flood the network with ICMP Echo packets. `MULTINET PING /FLOOD` transmits these packets 100 times per second or whenever a response is received. Requires `SYSPRV` privilege.

/NUMBER_OF_PACKETS=*number_of_packets_to_send*
Specifies the number of ICMP echo responses received before terminating. If not specified, `MULTINET PING` runs until you press **Ctrl+C**.

/PRELOAD=*number_of_packets_to_send*
Specifies the number of packets sent in rapid succession before entering the normal mode of operation.

/QUIET
/NOQUIET

Causes MULTINET PING to not display information when packets are received. By default, the information is displayed (/NOQUIET).

/RECORD_ROUTE
Displays a list of IP routers that the ICMP Echo Request packets traverse. This qualifier uses the IP record route option to display a list of IP routers that the ICMP echo request packet traverses. Not all implementations of IP handle this option correctly, so the use of /RECORD_ROUTE may result in a garbled response.

/ROUTE
/NOROUTE
Disables IP routing of ICMP packets. The default, /ROUTE, allows IP routing to get the packet to destinations separated by gateways.

/VERBOSE
/NOVERBOSE
Displays extra information as ICMP packets are sent or received. /NOVERBOSE is the default.

Examples

This example shows using PING to test the round-trip delay to a distant host.

```
$ MULTINET PING WWW.EXAMPLE.COM
PING WWW.EXAMPLE.COM (192.168.64.3) : 56 data bytes
64 bytes from 192.168.64.3: icmp_seq=1 time=670 ms
64 bytes from 192.168.64.3: icmp_seq=2 time=670 ms
64 bytes from 192.168.64.3: icmp_seq=3 time=670 ms
64 bytes from 192.168.64.3: icmp_seq=4 time=650 ms Ctrl+C
----WWW.EXAMPLE.COM  PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round trip (ms)   min/avg/max = 650/663/670
```

MULTINET RDATE

Uses the TCP TIME service to query a remote system for the current time and sets the clock on the local system to that time.

Format

`MULTINET RDATE` *host*

Parameters

host
Specifies the name or Internet address of the host to query.

Qualifiers

/DELTA
/NODELTA
Displays the time difference between the local and remote hosts, expressed in standard VMS time format. The default is `/NODELTA`.

/LOG
/NOLOG
Displays a message indicating that the time has been set. The `/NOLOG` qualifier suppresses this message.

/SET
/NOSET
RDATE usually sets the VMS time; `/NOSET` retrieves, but does not set, the current time.

MULTINET RMTALLOC

Provides local access to a remote tape or CD-ROM device on the specified remote host. RMTALLOC does not actually read from or write to the magnetic tape, or read from the CD-ROM; other programs supplied with the VMS Operating System provide this support. Optionally, RMTALLOC can associate a logical name with the device.

Format

`MULTINET RMTALLOC` *`host.domain`*`[::][["]`*`device-name`*`["]] [`*`logical-name`*`[:]]`

Parameters

`host.domain`

Specifies the remote host name on which the tape or CD-ROM device is allocated (the domain name - either just the domain name or the fully qualified domain name - of the remote host).

`device-name`

Specifies an optional device name entered with single or double colons. If the device name contains special characters, such as a UNIX-style device name (`/dev/rst8`), enclose the name in double quotes (`"/dev/rst8"`).

`logical-name`

Specifies the name associated with the device. Use a name you created or one designated by your system manager. The string is from 1 to 255 alphanumeric characters. If the string contains spaces, enclose the string in single quotes. (Do not use trailing colons.) The logical name you specify becomes a process name, with the device name as the equivalence name. The logical name remains defined until it is explicitly deleted or until your process terminates.

Qualifiers

`/CD`
`/NOCD`

Specifies that the remote device is a CD-ROM rather than a tape device. When `/CD` is specified, the local device takes the `RCD`*`xxx`*`:` name. When `/NOCD` (the default) is specified, the local device takes

the RMT*xxx*: name. The remote system is tested to ensure that the specified device type exists; if not, an error displays and RMTALLOC fails.

> **Note:** /CD cannot be used with either the /SEMANTICS or /WRITE qualifiers.

/LOG
/NOLOG
Displays a message indicating the name of the local device allocated, and the official host name of the remote host and device name. /NOLOG is the default.

/PASSWORD[=*password*]
Specifies the password to use to access the remote host. You may optionally specify the password as the qualifier value (which is not recommended). Specifying /PASSWORD without the value causes the password to be prompted for and read without echoing it (if the current input device supports it). If present, this qualifier causes RMTALLOC to use the REXEC server on the remote host rather than the RSHELL server.

```
                    [ blocksize=blocksize ]
                    [ comment="comment" ]
                    [ density=density ]
/SEMANTICS=         [( [ label="label" ] )]
                    [ [no]mount ]
                    [ [no]rewind ]
                    [ [no]unload ]
```

Specifies attributes for a magnetic tape device. Do not use with the /CD qualifier. Enter keywords separated by commas and enclosed in parentheses. These values pass information to the system operator at the remote system. For example, the values in *LABEL* and *COMMENT* display on the remote system console and request that the tape name indicated by *LABEL* be mounted.

Attribute	Description
BLOCKSIZE	Specifies the remote tape blocksize.
COMMENT	Specified as a string enclosed in double quotes; the information is displayed in the remote OPCOM message, either appended to or replacing the default text, depending

	on whether the resulting length is less than the maximum of 78 characters. Supplying the COMMENT value is the only way you can send a tape-specific message to the remote operator.
	
	The OPCOM message from the DCL MOUNT/COMMENT command is not passed to the remote RMT server; this message is only sent to OPCOM for a local operation. The default RMTALLOC command causes the remote tape to be mounted foreign, causing an OPCOM message to be generated if the tape drive is offline.
	
	The default RMTALLOC command is equivalent to the RMTALLOC /SEMANTICS=MOUNT command, which causes RMTALLOC to not complete until a tape has been physically loaded and the drive is online. Therefore, use the COMMENT value to ensure that the operator is informed of your request. Override the RMTALLOC default with the RMTALLOC /SEMANTICS=NOMOUNT command, which allocates the tape unit but does not wait for completion.
	
	Without the comment, RMTALLOC provides user, node, and device information, as shown in this example:
	
	`%%% OPCOM 25-MAR-2018 11;24:35.46 %%%`
	`FROM NODE WHORFIN AT 25-MAR-2020 11:24:35.44`
	`REQUEST 87, FROM USER HOLMES ON WHORFIN`
	`Please mount device _WHORFIN$ mka500 :`
	`RMT tape service request from WHORFIN.EXAMPLE.COM`

DENSITY	Specifies the density in bits per inch (BPI).
LABEL	Indicates the name by which the tape is known to the remote system. This could be your name, a site-specific numbering scheme, and so on.
[NO]MOUNT	Indicates whether the tape needs to be mounted. This option does not replace the MOUNT command; it only means that RMTALLOC should continue until the remote tape is mounted.

[NO]REWIND	Indicates whether the tape must be rewound before or after use.
[NO]UNLOAD	Indicates whether the tape must be unloaded from the drive after use.

Note: The remote tape drive must be able to write variable length blocks to permit VMS BACKUP to work correctly. Sun QIC tapes cannot do this and do not work with the VMS BACKUP utility.

The RMTALLLOC /SEMANTICS=NOMOUNT command does not work correctly with multivolume BACKUP save sets.

When using RMTALLOC to allocate a remote VMS TMSCP tape drive, the VMS COPY utility cannot copy files from a tape if the TMSCP tape drive is served from a different node than the one specified in the RMTALLOC command.

/TRUNCATE_USERNAME
/NOTRUNCATE_USERNAME
Truncates VMS user names to eight characters or less. Under UNIX, the remote user name has a maximum of eight characters. If a longer user name is supplied to such a system, a "remuser too long" error results and RMTALLOC fails. /NOTRUNCATE_USERNAME is the default.

/UNIX_SERVER=*value*
Specifies that RMTALLOC provide special handling for systems with problematic tape devices. Accepted values are:

Value	Description
BROKEN	Enables one OpenVMS BACKUP save set to be written to a remote UNIX tape. Use this value for SunOS 4.1 and SunOS 4.1.2. May also be useful on other UNIX-incompatible tapes and servers.
UNIX	Enables full OpenVMS tape functionality on an ULTRIX tape drive.

/USERNAME=*remote-username*
Specifies the remote user name to which you want to log in. If not specified, the default is the user name associated with your process.

/VMS_ATTRIBUTES
/NOVMS_ATTRIBUTES
Verifies whether the remote RMT server is also running MultiNet. If it is, RMT uses an improved RMT protocol to transfer VMS device attributes and I/O completion status values between your system and the remote host. Because this negotiation is compatible with UNIX implementations of RMT (including BSD and SunOS), it is enabled by default, but may be disabled if compatibility problems arise.

/WRITE
/NOWRITE
Specifies that the tape is not write-protected; if `/NOWRITE` is specified, the tape is write-protected. `/WRITE` cannot be specified with `/CD`. `/WRITE` is the default for mag tapes, while `/NOWRITE` is the default for CD-ROMs.

Examples

This example illustrates the use of the VMS TAR utility. (VMS TAR is a public domain program available from CETS.) First the tape is allocated with RMTALLOC, then the drive is mounted. Next, a file is written to the tape, the tape contents are listed, and the file is extracted back from the tape. Finally, the tape is dismounted and deallocated.

```
$ RMTALLOC CONE.EXAMPLE.COM::MUA0: MYTAPE
%RMT-I-ALLOC, _MYSYS$RMT1: allocated (CONE.EXAMPLE.COM::MUA0:)

$ MOUNT /FOREIGN /RECORD_SIZE=512 /BLOCK_SIZE=10240 MYTAPE
%MOUNT-I-MOUNTED,  MYTAPE mounted on _MYSYS$RMT1:

$ TAR /ARCHIVE=MYTAPE WRITE AFILE.TXT
%TAR-S-WRITTEN, written USERS:[ME]AFILE.TXT;1 (13495 bytes)
%TAR-S-TOTWRITE,  total of 1 file written

$ TAR LIST /ARCHIVE=MYTAPE
Listing of archive _MYSYS$RMT2:
-rw------    0/      0     13495  24 Apr 2019 14:31  afile.txt
Total of 1 files listed, 1 files in archive.
```

```
$ TAR /ARCHIVE=MYTAPE EXTRACT AFILE.TXT
%TAR-S-TOTCREAT,  total of 0 files created, 1 file scanned

$ DISMOUNT _MYSYS$RMT1:

$ DEALLOCATE _MYSYS$RMT1:
```

This example illustrates how to allocate access to a UNIX tape.

```
$ RMTALLOC FOO::"/deV/rst42" UNIXTAPE
%RMT-I-ALLOC, _MIURA$RMT7: allocated (FOO.EXAMPLE.COM::/dev/rst8)
$
```

This example allocates remote UNIX operating system tape device `/dev/rst42` on host FOO.EXAMPLE.COM and associates `UNIXTAPE` with the `_MIURA$RMT7` local pseudo-device.

```
$ RMTALLOC/CD/NOWRITE CONTROL::DISK$CD: DISK$CONTROL_CD/USER=SYSTEM
%RMT-I-ALLOC _GRUB$RCD3: allocated (CONTROL.EXAMPLE.COM::DISK$CD:)
$ MOUNT/OVER=ID DISK$CONTROL_CD:
%MOUNT-I-WRITELOCK, volume is write locked
%MOUNT-I-MOUNTED, VMS055LST1  mounted on _GRUB$RCD3:
$ DISMOUNT DISK$CONTROL_CD:
$ DEALLOCATE DISK$CONTROL_CD
$
```

This example allocates a CD-ROM for access between two VMS systems. The drive is allocated, mounted, dismounted, and deallocated.

The next example allocates a CD-ROM drive on a remote machine running UNIX.

```
$ RMTALLOC /CD/NOWRITE SYS1:: DISK$SYS1_CD/USER=ROOT
%RMT-I-ALLOC,  _GRUB$RCD3:  allocated (SYS1.EXAMPLE.COM::/dev/rsr0)
$ MOUNT /OVER=ID DISK$MEL_CD:
%MOUNT-I-WRITELOCK, volume is write locked
%MOUNT-I-MOUNTED,  VMS055LST2  mounted on _GRUB$RCD3:
$ DISMOUNT DISK$MEL_CD:
$ DEALLOCATE DISK$MEL_CD:
$
```

This example allocates a UNIX CD drive. The device name defaults to `/dev/rsr0`. You could specify another device name, using the same example with the `SYS1::"/dev/rsr42"` value in the RMTALLOC command. After the device is allocated in the previous example, it is mounted, dismounted, and finally deallocated.

The next example allocates a tape and then invokes BACKUP to write to it.

```
$ REPLY /ENABLE
```

```
$ RMTALLOC COMMENT="PLEASE MOUNT TAPE #A1234" WHORFIN::MKA500: TAPE
%%%%%%%%%% OPCOM  25-MAR-2020 11:24:35.46  %%%%%%%%%%
(FROM NODE WHORFIN AT 25-MAR-2018 11:24:35.44)
REQUEST 87, FROM USER HOLMES ON WHORFIN
Please mount device _WHORFIN$mka500:
RMT tape  service request from WHORFIN.EXAMPLE.COM
Please mount tape #A1234
 %%%%%% OPCOM  25-MAR-2020 11:25:29.12  %%%%%%%%%%
(FROM NODE   HOLMES
25-MAR-2020 11:25:29.12)
REQUEST 87 WAS SATISFIED.
%RMT-I-ALLOC, _HOLMES$RMT2: ALLOCATED (WHORFIN.EXAMPLE.COM::MKA500:)

$ INIT TAPE: FOO

$ BACKUP/LOG/INGORE=LABEL/VERIFY USERS:[ATMA.TEST]*.EXE;0
TAPE:EXES.BCK/SAVE
%MOUNT-I-MOUNTED, FOO MOUNTED ON _HOLMES$RMT2:
  . .

$ BACKUP/LOG/IGNORE=LABEL/VERIFY USERS:[ATMA.TEST]*.H;0
TAPE:H.BCK/SAVE
  . .

$ BACKUP/LOG/INGORE=LABEL/VERIFY USERS:[ATMA.TEST]*.C;0
TAPE:C.BCK/SAVE
  . .

$ DISMOUNT/NOUNLOAD TAPE:

$ MOUNT/OVER=ID TAPE:
%MOUNT-I-MOUNTED, FOO MOUNTED ON _HOLMES$RMT2:

$ DIR TAPE:
DIRECTORY  _HOLMES$RMT2:[]
EXES.BCK;1     H.BCK;1       C.BCK;1
TOTAL OF  3 FILES.

$ DISMOUNT TAPE:

$ DEALL TAPE:
$
```

This example allocates access to a tape, then writes to it.

When issuing a RMTALLOC to a remote MultiNet system, the remote tape drive must be online with the tape physically loaded. Otherwise, RMTALLOC fails with the error, `%SYSTEM-F-MEDOFL, medium is offline.`

You can override this default with the /SEMANTICS=MOUNT qualifier. RMTALLOC does not complete until a tape has physically been loaded and the tape drive is online. Use the /SEMANTICS=COMMENT keyword to specify a mount message to send to the operator via OPCOM.

MULTINET RWALL

Uses Remote Procedure Calls (RPCs) to send a network broadcast message to all users on the specified host. If you specify the host as an asterisk (*), the message is broadcast to all hosts on Ethernets to which the local host is attached.

Format

`MULTINET RWALL [qualifier1] [qualifier2 . . . ] ["message_text"]`

Restriction

RWALL messages are only received on hosts that support RWALL service.

Parameters

message_text
Contains the message to broadcast.

Qualifiers

/HEADER[="*header_text*"]
/NOHEADER
Adds header text to the specified message. If you use the /NOHEADER qualifier, RWALL does not preface any header text to the specified message. By default, the header is prefaced with "Broadcast message from *username*@*hostname*:", although you may specify any header text as the value of this qualifier.

/HOST=[*hostname*]
Specifies the host on which the message is displayed. The default is /HOST=LOCALHOST, which prints the message on the host from which the RWALL command was invoked. If you specify the qualifier as /HOST=*, the network broadcast displays on all directly reachable hosts on all connected networks that support broadcasting. /HOST=* is most appropriate for network-wide system shutdown messages.

Example

This example shows how to broadcast a shutdown message to users on the local host.

```
$ MULTINET RWALL "Node ROMEO is shutting down"
RWALL MESSAGE:
Broadcast message from HOLMES@ROMEO: Node ROMEO is shutting down
```

MULTINET SET /ARP

Modifies Address Resolution Protocol (ARP) tables. These tables are normally modified dynamically by the ARP protocol. Use with `MULTINET SHOW /ARP` to view the contents of the ARP table.

Qualifiers

/ADD=(PROTOCOL=*protocol*,HOST_ADDRESS=*host_addr*, ETHER_ADDRESS=*ether_addr*)

Adds a specified host-to-Ethernet address translation to the ARP tables. The `PROTOCOL` specification identifies which protocol (IP, for example) is being described. The `HOST_ADDRESS` specification gives the host address in IP form. The `ETHER_ADDRESS` specification gives the hardware Ethernet address in the form `aa:bb:cc:dd:ee:ff`, where `aa` through `ff` are specified in hexadecimal. If not specified, the default is `PROTOCOL=IP`.

/COMMUNITY_NAME=*string*

Overrides the default community string (private) for remote `SNMP SET` requests. The `/SNMP_HOST` qualifier must be present if the `/COMMUNITY_NAME` qualifier is specified.

/DELETE=*host*

Deletes the specified host-to-Ethernet address translation from the ARP tables.

/FLUSH

Flushes the current ARP table. By default only temporary entries are flushed. If the qualifier `/PERMANENT` is specified, all entries are flushed.

/PERMANENT
/TEMPORARY

Indicates that the translation to be added is kept (or deleted) permanently (used with the `/ADD` or `/FLUSH` qualifiers). The default (`/TEMPORARY`) indicates that this entry is considered for normal ARP table purging of old entries.

/PROXY

Used with the `/ADD` qualifier, indicates that the translation to the local host's Ethernet address is published on behalf of another host.

/PUBLISH
Indicates that the translation to be added is published on behalf of another host (that is, this host should answer with the specified translation on behalf of the other host). This qualifier is used with the /ADD qualifier.

/SNMP_HOST=*hostname*
Specifies the host affected by the `MULTINET SET /ARP` command. The SNMP agent on the remote host must support read-write access to elements of the MIB-II variable `ipNetToMedia`.

Examples

This example displays the contents of the ARP table. Note: if the host name and IP address are longer than the "Host Network Address" field, they are truncated to fit.

```
$ MULTINET SHOW /ARP /SYMBOLIC=NAMESERVER
Multinet ARP table:
Host Network Address                    Ethernet Address    Arp Flags
----------------------                  ----------------    ---------
FSGATE.CC.EXAMPLE.COM (IP 128.0.33.123 AA:00:04:00:79:4C   Temporary
EXPLORER.ME.EXAMPLE.COM (IP 128.0.41.1 08:00:11:00:90:B0   Temporary
GOOFY.CC.EXAMPLE.COM (IP 128.0.83.122) 08:00:20:01:27:6D   Temporary
BEGWS2.BEG.EXAMPLE.COM (IP 128.0.30.23 AA:00:04:00:65:4C   Temporary
ARPAGATEWAY.EXAMPLE.COM (IP 128.0.11.2 AA:00:04:00:0F:4C   Temporary
PORTAL1.CC.EXAMPLE.COM (IP 128.0.19.10 08:00:4C:00:23:CE   Temporary
WILMA.CC.EXAMPLE.COM (IP 128.0.7.125)  AA:00:04:00:64:4C   Temporary
FS4.CC.EXAMPLE.COM (IP 128.0.19.251)   AA:00:04:00:12:4C   Temporary
```

This example is often used to solve a problem that occurs in environments with a mixture of UNIX 4.2 BSD and 4.3 BSD systems. 4.2 BSD systems use zero-filled (*nn.mm*.0.0) IP broadcast addresses, while 4.3 BSD systems use ones-filled (*nn.mm*.255.255) broadcast addresses. To prevent 4.2 BSD systems from creating Ethernet "broadcast storms" when they issue ARP requests for the 4.3 BSD broadcast address, the above command publishes an ARP translation for the ones-filled broadcast address.

```
$ MULTINET SET /ARP /ADD=(HOST=128.0.255.255,-
  PROTOCOL=IP, ETHER=0:0:D:E:A:D) /PUBLISH
$
```

This example flushes all temporary ARP table entries.

```
$ MULTINET SET /ARP /FLUSH
$
```

MULTINET SET /DECNET

Configures the DECnet TCPA*x*: devices for running DECnet-over-UDP circuits.

Qualifiers

/BUFFERS=*buffercount*
Specifies the number of buffers the driver preallocates for this device (by default, 6).

/CLOSE
Shuts down and deletes a socket created with the `socket()` routine. After issuing a `CLOSE` command, the socket cannot be used again until the `MULTINET SET/DECNET` command is reissued.

/CONNECT
Issues a `connect()` call to bind the remote address of the socket to the address specified in `/REMOTE_ADDRESS`.

/DEVICE=*device*
Specifies the DECnet device name (by default, `TCPA0:`).

/FILTER_OUT_OF_ORDER=AUTOMATIC
/FILTER_OUT_OF_ORDER=OFF
/FILTER_OUT_OF_ORDER=ON
Controls the handling of out-of-order DECnet packets arriving via IP. Prior to VMS V4.7, DECnet could not handle packets arriving out-of-order and would drop the line if it received them. If you have any VMS V4.6 or earlier systems in your DECnet network with which you are communicating, you must use the `/FILTER_OUT_OF_ORDER=ON` qualifier. The default action `/FILTER_OUT_OF_ORDER=AUTOMATIC`, selects the correct filtering based on the VMS version of the current system only.

/LOGDATA
Specifies that `send()` and `recv()` log a sample of the data passed through them to OPCOM. Use this qualifier only for debugging network problems.

/LOGERRORS
Specifies that `send()` and `recv()` errors are logged to OPCOM. The default is to log all errors except these.

/PORT=*UDP-port-number*
Specifies the UDP port number to use for communication (by default, 700).

/REMOTE_ADDRESS=*ip-address*
Specifies the peer's IP address.

/TCP=mode
/TCP=CONNECT
/TCP=LISTEN
Specifies that DECnet is encapsulated in TCP instead of UDP. This mode is not supported by the normal configuration utility, but is of use over high-loss lines. `LISTEN` specifies that this end of the connection listens on the specified port; `CONNECT` specifies that this end attempts to connect to the listener on the specified port.

MULTINET SET /IPS

Controls the FILTER_SERVER process of the MultiNet Intrusion Detection and Prevention subsystem.

Qualifiers

/DEBUG=*level*

Specifies the level of debug for the filter server. Zero indicates no debug should be written to the log file, while increasing numbers indicate increasing amounts of debug will be written. This parameter should normally never be set above 4 without explicit instruction by Process Software.

/CLEAR_FILTERS

Causes the FILTER_SERVER process to remove all filters set by IPS on all interfaces configured for IPS. This may be used with SET /IPS /START and SET /IPS/RESTART, or may be used by itself with SET /IPS/CLEAR_FILTERS. When used by itself this causes a running IPS subsystem to remove the IPS filters and reset the event count information for the source address associated with each filter being removed.

/RELOAD

Causes the filter server to re-read and parse the configuration files. Note that this will not wipe out existing event and rule information; it will simply update it so no potential filter information will be lost.

/RESTART

Stop and restart the filter server. All existing event and rule information will be lost and reloaded from the configuration files.

/START

Start the filter server if it's not already running.

/STOP

Stop the filter server. All existing event and rule information will be lost.

MULTINET SET /INTERFACE

Sets parameters for the specified network device. This command is invoked automatically by the network startup command file generated by the NET-CONFIG utility.

Format

MULTINET SET/INTERFACE *interface*

Parameters

interface
Specifies the name of the interface to change; for example, se0.

Qualifiers

/ADDRESS=*network_address*
Specifies a network address to assign to the network interface. The address format is dependent on the protocol specified with the /PROTOCOL specifier:

- IP-address is of the form *AA.BB.CC.DD*
- IPX-address is a hexadecimal value
- IPv6-address is of the form *XXXX*:*XXXX*:*XXXX*:*XXXX*:*XXXX*:*XXXX*:*XXXX*:*XXXX*

/ARP
/NOARP
/NOARP disables the Address Resolution Protocol on the specified interface (supported only on Ethernet interfaces). /ARP is the default.

/COMMON_LINK=*line-ids*
The /COMMON_LINK qualifier works for systems that have multiple interfaces on a common Ethernet, FDDI, or Token Ring cable. The system manager configures this support using the following qualifier:

```
$ MULTINET SET /INTERFACE xxx/COMMON_LINK=(yyy[,zzz...])
```

xxx is the hardware device that the pseudo device that has the actual IP address of the machine is tied to (see the *MultiNet Installation and Administrator's Guide* for an example on how to set up a pseudo

device). *yyy* and *zzz* are device names like `se0`, `se1`, and `se2`. With this qualifier, MultiNet links the interfaces together. A performance benefit of this linking occurs if data is to be transmitted on an interface that happens to be busy, MultiNet assigns the data to the least busy linked interface for transmission.

This linking also provides a level of redundancy. If a linked interface is shut down using `MULTINET SET/INTERFACE/DOWN` or if a fatal error is detected with the interface and an automatic restart cannot be attempted, then any routing table entries or pseudo devices associated with the shutdown interface will be failed over to one of the common link interfaces.

Restrictions:

- The joined interfaces must be connected to the same cable.
- The joined interfaces must have the same MTU.

Example Configuration:

The actual IP address for SYSA.EXAMPLE.COM is 192.168.0.1; this address is used for a pseudo device (`pd0`), which uses `se0`.

```
$ multinet configure/network
MultiNet Network Configuration Utility V5.6
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]

NET-CONFIG>show
Interface                                Adapter     CSR Address    Flags/Vector
---------                                -------     -----------    ------------
se0      (Shared VMS Ethernet/FDDI)       -NONE-        -NONE-       -NONE-
          [TCP/IP: 192.168.1.1, IP-SubNet: 255.255.255.0]
          [VMS Device: EWA0, Link Level: Ethernet]
se1      (Shared VMS Ethernet/FDDI)       -NONE-        -NONE-       -NONE-
          [TCP/IP: 192.168.1.2, IP-SubNet: 255.255.255.0]
          [VMS Device: EWB0, Link Level: Ethernet]
pd0      (Secondary Ethernet Address)     -NONE-        -NONE-       -NONE-
          [TCP/IP: 198.168.0.1, IP-SubNet: 255.255.255.0]
          [Hardware-Device: se0]

Official Host Name:             sysa.example.com
Domain Nameserver:              127.0.0.1
Timezone:                       EST
Timezone Rules:                 US/EASTERN
Load UCX $QIO driver:           TRUE
Load PWIP (Pathworks) driver:   TRUE
SNMP Agent X subagents are enabled
NET-CONFIG>
```

If DECnet is being used, then MULTINET:SE1_CONFIGURE.COM will need to be created (see MULTINET_ROOT:[MULTINET.EXAMPLES]SE0_CONFIGURE.COM) to configure SE1 without the DECnet Ethernet address.

The command

```
$ MULTINET SET /INTERFACE SE0/COMMON_LINK=(SE1)
```

can be added to the SE1_CONFIGURE.COM, or put in MULTINET:LOCAL_INITIALIZATION.COM so that it will be executed each time MultiNet is started.

/COMMUNITY_NAME=*string*
Overrides the default community string (private) for remote SNMP SET requests. The /SNMP_HOST qualifier must be present if the /COMMUNITY_NAME qualifier is specified.

/CREATE
Requests that a dynamic interface (e.g. gif1) be created

/DELETE
Requests that a dynamic interface (e.g. gif) be deleted.

/D1
/NOD1
Enables or disables (the default) the device-dependent IFF_D1 flag.

/D2
/NOD2
Enables or disables (the default) the device-dependent IFF_D2 flag.

/D3
/NOD3
Enables or disables (the default) the device-dependent IFF_D3 flag.

/DEBUG
/NODEBUG
Enables interface-specific debugging. Some interfaces have debugging code and send debugging information to the users with OPCOM OPERATOR messages enabled. /NODEBUG is the default.

/DECNET_ETHERNET_ADDRESS
/NODECNET_ETHERNET_ADDRESS
Initializes a DECnet shared Ethernet interface to determine what Ethernet address to use. If other protocols are currently using the device, the Ethernet address cannot be changed and this qualifier is ignored.

The default behavior, /DECNET_ETHERNET_ADDRESS, is used by MULTINET SET /INTERFACE to look at the SCSSYSTEMID SYSGEN parameter and set the Ethernet address to match. If SCSSYSTEMID is not set, the address on the Ethernet card's PROM is used.

If /NODECNET_ETHERNET_ADDRESS is specified, MultiNet uses the PROM address.

/DOWN
Marks the network interface as not UP and packets are no longer accepted or transmitted. See /UP for more information.

/DYNAMIC
/NODYNAMIC
Reverts the terminal line to a normal VMS terminal line if a modem hangup occurs. Use /DYNAMIC to create dynamic-dialup SLIP links with the /LINK_LEVEL=SLIP qualifier. When creating a dynamic SLIP link, CMKRNL, LOG_IO, and SYSPRV privileges are required. /NODYNAMIC is the default.

/EXTRACT_FILTERS=*file*
Reads all non-expired filters from the specified interface and writes them to the specified filename in the same text format that would be used as input to the MU SET/INTERFACE/FILTER command.

/FFI_BUFFERS=*number_of_buffers*
Initializes a shared VMS Ethernet or FDDI interface, and specifies the number of packet buffers to allocate to each protocol port of the VMS device driver (by default, 4).

/FILTER=filter_file
/NOFILTER
Associates a file containing a packet filter list with a particular network interface. The contents of this file are parsed and the individual filters are loaded for the interface. If the file MULTINET:FILTER-*interface*.DAT exists when MultiNet is started, the MultiNet startup procedure will automatically load these filters for the specific interface. /NOFILTER is the default.

/FORMAT=[NORMAL | COMMA]
Log events in the specified format. If NORMAL, then the formatting used by MULTINET SHOW/INTERFACE/FILTER is used. If COMMA, then a comma-delimited line is output to the file. This can then be loaded into, for example, a spreadsheet for analysis. If the log destination is OPCOM, use of the /FORMAT qualifier is illegal.

/HARDWARE_DEVICE=*primary_interface*
Specifies the name of the real interface for a secondary IP address device, and connects the interface to the specified primary interface.

/INTERVAL=*seconds*
Reporting interval in seconds. The minimum reporting interval is seconds, so that a flood of filter events doesn't adversely impact the system. The minimum interval that can be specified is 5 seconds. If no interval has been specified when logging is enabled (see the /LOG qualifier in this section), an interval of 5 seconds will be used.

/IP_BROADCAST=*ip_address*
Specifies a non-standard IP broadcast address. The default IP broadcast address has all bits in the host part of an IP address set to 1 (the standard format under 4.3 BSD). Some sites may still use the 4.2 BSD standard of IP broadcasts with the host part of an IP address set to 0.

/IP_SUBNET_MASK=*ip_address*
Specifies the network portion of the interface IP address. *ip_address* is an IP address in which each bit corresponding to a bit in the network portion is set to 1. All interfaces on the same subnet must have the same subnet mask.

By default, MultiNet uses the subnet mask implied by the interface's IP address. Do not use the default subnet mask if your site has subnets. For example, the default subnet mask of an interface with the

address 161.44.128.15 is 255.255.0.0. (255.255.255.0 would be a suitable subnet mask if that interface is on a subnet, and there are fewer than 256 subnets, and the total number of hosts is less than 256.)

/IP6_SUBNET_MASK=*length*
Specifies the length of the IPv6 subnet mask. The range of this is from 1 to 128 bits. The default length is 128.

```
              [ 802 ]
              [ ethernet ]
              [ extended_8022 ]
              [ ppp ]
/LINK_LEVEL=([ proteon ])
              [ slip ]
              [ standard_8022 ]
              [ raw_8023 ]
```

Specifies the type of device being initialized. Use /LINK_LEVEL with the /VMS_DEVICE qualifier. This qualifier supersedes the former /SLIP_DEVICE and /PROTEON_DEVICE qualifiers.

- Specify 802 or STANDARD_8022 for IEEE 802.2 encapsulation.
- Specify EXTENDED_8022 for IEEE 802.2 with SNAP (System Network Access Protocol) extensions.
- Specify PPP for Point-to-Point Protocol devices.
- Specify SLIP for Serial Line Internet Protocol (SLIP) devices.
- Specify RAW_8023 for 802.3 encapsulation.
- Specify ETHERNET for ETHERNET_II encapsulation.

/LOCAL=*node_name*
Specifies the name of the local node on this side of an IP interface; may be used with DECnet and PSI links.

/LOG=[filename | OPCOM]
Used to turn logging on or off for those filters that contain the LOG qualifier in their definition. The logging may be to OPCOM or the specified file. Turn logging off using /NOLOG.

/MTU=*mtu*

Specifies the Maximum Transmission Units - the size of IP packets over a given interface. Not all devices support the use of `/MTU`, and there may be additional, device-dependent restrictions dictating when it can be used.

/MULTICAST=ALL
Enables reception of all multicast packets. Use this qualifier only for OpenVMS VAX V5.5-2 and later. Reception is enabled automatically in OpenVMS VAX V6.1 and OpenVMS AXP versions.

/PEER=*peer_name*
Specifies the name of the node on the other side of an IP interface; used with DECnet and PSI links.

/POINT_TO_POINT_DESTINATION=*ip_address*
Specifies the IP address of the node on the other side of a point-to-point interface.

/PPP_NOICMP
Prevents ICMP packets from being passed to IP via the PPP interface.

/PPP_OPTIONS=*options_list*
Specifies values for the PPP options included in a comma-separated *option_list*. The following options may be enabled:

ACCM=*mark*	MRU=*size*
AUTHENTICATION=*method*	NOICMP
COMPRESS_PROTOCOL	TCP_COMPRESSION
COMPRESS_ADDRESS_AND_CONTROL	TERMINATION_RETRIES=*count*
CONFIGURATION_RETRIES=*count*	TIMEOUT=*seconds*
IDLE=*seconds*	

/PROTOCOL=*protocol_name*
Specifies the protocol to which the `/ADDRESS` qualifier refers (by default, IP). For IPv6 use `I6`.

/PREFIX=*ipv6_prefix*

Specifies the IPv6 prefix for an interface to use to generate a global IPv6 address. The default prefix length is 64, or a different value can be specified with the IP6_SUBNET_MASK qualifier.

/RARP
/NORARP

Initializes the VMS Ethernet device to receive RARP packets. The /RARP qualifier is used with the /VMS_DEVICE qualifier. The RARP packet type is disabled by default and must be enabled to use the RARP service on VMS Ethernet devices.

/SEND_QUEUE_LENGTH=*number*

Specifies the maximum queue length for packets waiting to be sent from the interface. The minimum value is 10, default values are interface specific. If an interface has a heavy transmit load and is showing dropped packets, then specifying a larger number here may help.

/SNMP_HOST

Specifies the host affected by the MULTINET SET /INTERFACE command. The SNMP agent on the remote host must support read-write access to the MIB-II variable ifAdminStatus.

/SNMP_HOST can only be used with the /UP or /DOWN qualifiers.

The device specified with the /SNMP_HOST qualifier may be either the full text string of the remote interface name or the numeric index of the interface to be set. You can display a list of remote interface names with the MULTINET SHOW/INTERFACE/SNMP_HOST command.

/TRAILERS
/NOTRAILERS

Enables IP trailer encapsulation for the specified interface (only supported on Ethernet and FDDI interfaces). If trailers are enabled, the use of IP trailer encapsulation is negotiated between hosts as a byproduct of IP-to-Ethernet address resolution using Address Resolution Protocol (ARP). On an HP Ethernet controller, /TRAILERS must be used with /VMS to initialize the trailer protocol ports. The default is /NOTRAILERS.

/TUNNEL=(DESTINATION_ADDRESS=*ip_address*, GATEWAY_ADDRESS=*ip_address*)

Set up a tunnel with a gif interface. Specifies the local (gateway) and remote (destination) public addresses when setting tunnel addresses. Tunnels also need a local address set with /ADDRESS and a remote address set with /POINT_TO_POINT_DESTINATION. For more detail see chapter 11 in the *Administrator's Guide.*

/UP
/DOWN
/UP (the default) marks the network interface as "up" and ready to accept or transmit packets. /DOWN marks the network interface "down" and packets are no longer accepted or transmitted.

/VMS_DEVICE=*vms_device*
Initializes an interface that has an associated VMS device, telling the MultiNet kernel which VMS device to associate with the IP device. If /VMS_DEVICE is used with /DOWN, the specified VMS device is disconnected from the IP device and made available to other VMS applications.

Examples

This example disables the se0 interface.

```
$ MULTINET SET/INTERFACE SE0 /DOWN
```

This example enables the se0 interface with the address 192.0.0.1.

```
$ MULTINET SET/INTERFACE SE0 /UP/ADDRESS=192.0.0.1
```

This example enables a dynamic SLIP line.

```
$ MULTINET SET/INTERFACE SL1 /DYNAMIC/LINK_LEVEL=SLIP/VMS_DEVICE
```

Enter the following command at MultiNet startup:

```
$ MULTINET SET/INTERFACE PD0/COMMON_LINK=(SE0,SE1)
```

The PD0 has the real IP address, the SE*n* devices have something else (like 10.*n.n.n*).

```
$ MULTINET SET/INTERFACE SE0 /LOG=OPCOM/INTERVAL=10
```

enables logging to OPCOM, with a reporting interval of 10 seconds.

```
$ MULTINET SET/INTERFACE SE0 /LOG=FOO.DAT/FORMAT=COMMA
```

enables logging to the file `FOO.DAT` in comma-delimited format, and a reporting interval of 5 seconds (the default).

```
$ MULTINET SET /INTERFACE SE0 /NOLOG
```

This disables all logging for the interface, closing all open log files.

MULTINET SET /ROUTE

Specifies static IP routing, including the default route. This command is invoked automatically by the network startup command file generated by the network configuration utility (NET-CONFIG). Before making changes with SET /ROUTE, use MULTINET SHOW /ROUTE to view the routing information.

Qualifiers

/ADD=(DESTINATION=*ip-address*,GATEWAY=*ip-address* [,NETMASK=*network-mask*][,INTERFACE][,MASK_LENGTH=*integer*])

Adds a static IP route to the MultiNet kernel routing tables.

- The DESTINATION specification gives the network or host for which the routing information is valid.
- The GATEWAY specification gives the next hop for the packet to take on its way to the destination.
- The optional INTERFACE keyword forces the routing to be for a locally connected interface, and is normally not used.
- The optional NETMASK specification dictates which bits of the DESTINATION *ip-address* comprise the network portion of an ip-address. If not specified, the DESTINATION address is given a class-based network mask.
- The optional MASK_LENGTH specifies the length in bits of the mask to apply to the DESTINATION address. Either NETMASK or MASK_LENGTH can be specified, not both.

/COMMUNITY_NAME=*string*

Overrides the default community string (private) for remote SNMP SET requests. The /SNMP_HOST qualifier must be present if the /COMMUNITY_NAME qualifier is specified.

/DELETE=(DESTINATION=*ip-address*, GATEWAY=*ip-address* ,NETMASK=*network-mask*][,INTERFACE])

Deletes an IP route from the MultiNet kernel routing tables.

- The DESTINATION specification gives the network or host for which the routing information is valid.
- The GATEWAY specification gives the next hop for the packet to take on its way to the DESTINATION.

- The optional INTERFACE keyword forces the routing to be for a locally connected interface, and is normally not used.
- The optional NETMASK specification dictates which bits of the DESTINATION *ip-address* comprise the network portion of an *ip-address*. If not specified, the DESTINATION address is given a class-based network mask.

/FLUSH
Deletes all IP routes in the MultiNet kernel.

/FORCE_HOST
Interprets the DESTINATION as a host address when used with the /ADD or the /DELETE qualifiers.

/FORCE_NETWORK
Interprets the DESTINATION as a network address when used with the /ADD or the /DELETE qualifiers.

/NETWORK_IMAGE=*file-spec*
Specifies the network image associated with the running MultiNet kernel. This is used to read IP routing information in the MultiNet kernel. If not specified, the image currently loaded is used.

/PROTOCOL=*protocol_name*
Specifies the protocol that the route applies to. The default is IP, use I6 for IPv6.

/SNMP_HOST=*hostname*
Specifies an IP host. The SNMP agent on the remote host must support read-write access to elements of the MIB-II variable ipRouteTable.

Examples

This example displays the current state of the MultiNet routing tables. /NOSYMBOLIC forces MULTINET SHOW/ROUTE to display the information numerically.

```
$ MULTINET SHOW /ROUTE /NOSYMBOLIC
MultiNet IP Routing tables:
Destination   Gateway     Flags        Refcnt   Use    Interface
-----------   -------     ------       ------   ---    ---------
127.0.0.1     127.0.0.1   Up,Host      2        2529   lo0
192.0.0.1     192.0.0.2   Up,Host      3        10521  sl0
0.0.0         192.0.0.1   Up,Gateway   3        6105   sl0
192.0.0.64    192.0.0.65  Up           2        2372   se0
```

This example deletes the default route to EXAMPLE.COM.

```
$ MULTINET SET/ROUTE/DELETE=(DEST=DEFAULT,GATE=192.0.0.1)
Delete Route DEFAULT, Gateway EXAMPLE.COM
$
```

MULTINET SET /TIMEZONE

Specifies the local time zone name that was either previously compiled into MultiNet or is a name from a selected time zone in the time zone database files.

Format

`MULTINET SET /TIMEZONE  localzone`

Parameters

localzone
The name of the local time zone; for example, `PST`.

Qualifiers

/LOG
/NOLOG
Displays a list of the time zones that are loaded, and a list of the compiled-in zones that were selected but not loaded because they were compiled in. `/NOLOG` is the default.

/SELECT=(*rule1* [,*rule2* [...]])
Specifies a list of countries or time zones to load. Specifying a country loads all time zones in that country.

/FILES=(*file1* [,*file2* [...]])
Specifies a list of files from which to load the time zone data. The default is `MULTINET:TIMEZONES.DAT`. Locally-written rules are normally added to `MULTINET:TIMEZONES.LOCAL`.

Examples

This example sets the local timezone to PST.

```
$ MULTINET SET /TIMEZONE PST
```

This example sets the local time zone to MST and loads Arizona time zone rules.

```
$ MULTINET SET /TIMEZONE MST/SELECT="US/ARIZONA"
```

MULTINET SETKEY

Manually manipulates the IPsec SA/SP database. In order to use `SETKEY`, a foreign command needs to be defined.

```
$ SETKEY :== $MULTINET:SETKEY.EXE
```

Note that only UNIX-style options can be used. For more details, please refer to Chapter 31 in the *MultiNet Installation and Administrator's Guide*.

Synopsis

```
setkey [-v] -c
setkey [-v] -f filename
setkey [-aPv] -D
setkey [-Pv] -F
setkey [-h] -x
```

Description

`SETKEY` adds, updates, dumps, or flushes Security Association Database (SAD) entries, as well as Security Policy Database (SPD) entries in the kernel.

`SETKEY` takes a series of operations from the file named `MULTINET:IPSEC.CONF` (when invoked with `-f filename`).

Arguments

> **Note:** Since `SETKEY` supports both uppercase and lowercase command options, these have to be enclosed within quotation marks (e.g, `setkey "-F"`).

`-a`	Also displays the SAD (Security Association Database) entries. A SAD entry is when it has expired, but it may still be referenced by SPD (Security Policy Database) entries.

`-D`	Dumps the SAD entries. If used with `-P`, the SPD entries are dumped.
`-F`	Flushes the SAD entries. If used with `-P`, the SPD entries are flushed.
`-xx`	Makes each timestamp unformatted.
`-h`	Adds hexadecimal dump on `-x` mode.
`-l`	Loops forever with short output on `-D`.
`-P`	Dumps (when specified with `-D`) or flush (with `-F`) the SPD entries.
`-v`	Verbose. The program will dump messages exchanged on PF_KEY socket, including messages sent from other processes to the kernel.
`-x`	Loops forever and dumps all the messages transmitted to the PF_KEY socket.
`-f` *filename*	File that contains the operations to be performed. For more information about the operations, see the *Header Operations* section below.

Header Operations

Header operations have the following grammar. Note that lines starting with hashmarks (#) are treated as comment lines.

Adds a SAD entry:

```
$ add src dst protocol spi [extensions] algorithm... ;
```

Shows a SAD entry:

```
$ get src dst protocol spi ;
```

Removes a SAD entry:

```
$ delete src dst protocol spi ;
```

Removes all SAD entries that match the specification:

```
$ delete all src dst protocol ;
```

Clears all SAD entries matched by the protocol:

```
$ flush [protocol] ;
```

Dumps all SAD entries matched by the protocol:

```
$ dump [protocol] ;
```

Adds an SPD entry:

```
$ spdadd src_range dst_range upperspec policy ;
```

Deletes an SPD entry:

```
$ spddelete src_range dst_range upperspec -P direction ;
```

Clears all SPD entries:

```
$ spdflush ;
```

Dumps all SPD entries:

```
$ spddump ;
```

Meta-Arguments

Meta-arguments used in the header operations are as follows:

src
dst

Source/destination of the secure communication is specified as an IPv4 address. `setkey` does not consult hostname-to-address for arguments *`src`* and *`dst`*. They must be in numeric form.

protocol
`protocol` is one of following:

- `esp` - ESP based on rfc2405
- `ah` - AH based on rfc2402

spi
Security Parameter Index (SPI) for the SAD and the SPD. It must be decimal number or hexadecimal number (with 0x attached). You cannot use the set of SPI values in the range 0 through 255.

extensions

`-m` *`mode`*	Specifies a security protocol mode for use. *`mode`* is one of following: `transport`, `tunnel` or `any`. The default value is `any`.
`-E` *`ealgo key`*	Specifies an encryption algorithm
`-A` *`aalgo key`*	Specifies an authentication algorithm. If `-A` is used with *`protocol`* `esp`, it will be treated as ESP payload authentication algorithm.

`protocol` `esp` accepts `-E` and `-A`. *`protocol`* accepts `-E` only. *`protocol`* `ah` accepts `-A` only.

`key` must be double-quoted character string or series of hexadecimal digits. Possible values for *`ealgo`*, *`aalgo`* and *`calgo`* are specified in a separate section.

src_range
dst_range
These are selections of the secure communication specified as IPv4/v6 address or IPv4/v6 address range, and it may accompany TCP/UDP port specification. This takes the following form:

`address`
`address/prefixlen`

address[*port*]
address/*prefixlen*[*port*]

prefixlen and *port* must be decimal number. The square bracket around *port* is really necessary – it's not a documentation convention.

setkey does not consult hostname-to-address for arguments *src* and *dst*. They must be in numeric form.

upperspec
Upper-layer protocol to be used. icmp and any can be specified. any stands for "any protocol". You can also use the protocol number.

> **Note:** *upperspec* does not work against forwarding case at this moment, as it requires extra reassembly at forwarding node (not implemented at this moment). There are many protocols in /etc/protocols, but protocols other than TCP, UDP, and ICMP may not be suitable to use with IPSec.

policy
policy is the one of following:

-P *direction* discard
-P *direction* none
-P *direction* ipsec *protocol*/*mode*/*src-dst*/*level*

You must specify the policy's *direction* as either out or in.

discard means the packet matching indexes will be discarded.

none means that IPsec operations will not take place onto the packet.

ipsec means that IPSEC operation will take place onto the packet.

ah, esp, or ipcomp must be set as *protocol*.

mode is either transport or tunnel.

If *mode* is tunnel, you must specify the end-point addresses of the SA as *src* and *dst* with - between these addresses, which is used to specify the SA. If *mode* is transport, both *src* and *dst* can be omitted.

level is to be one of the following: default, use, require, or unique. If the SA is not available in every level, the kernel will request getting the SA to the key exchange daemon.

default means the kernel consults to the system wide default against protocol you specified, e.g. esp_trans_deflev sysctl variable, when the kernel processes the packet.

use means that the kernel uses an SA if it's available, otherwise the kernel keeps normal operation.

require means an SA is required whenever the kernel sends a packet matched with the policy.

unique is the same as require, except that unique allows the policy to bind with the unique outbound SA. If you use the SA by manual keying, you can put the decimal number as the policy identifier after unique, provided it is separated by a colon similar to this example: unique:*number*. *number* must be between 1 and 32767. It corresponds to *extensions* -u.

Algorithms

The following list shows the supported algorithms. Following is a list of authentication algorithms that can be used as *aalgo* in -A of the *protocol* parameter:

Algorithm	Key Len (bits)	Comment
hmac-md5	128	ah: rfc2403
	128	ah-old: rfc2085
hmac-sha1	160	ah: rfc2404
	160	ah-old: 128bit ICV (no document)
keyed-md5	128	ah: 96bit ICV (no document)
	128	ah-old: rfc1828
keyed-sha1	160	ah: 96bit ICV (no document)

	160	ah-old: 128bit ICV (no document)
`null`	0 to 2048	for debugging
`hmac-sha2-256`	256	ah: 96bit ICV (no document)
	256	ah-old: 128bit ICV (no document)
`hmac-sha2-384`	384	ah-old: 128bit ICV (no document)
	384	ah-old: 128bit ICV (no document)
`hmac-sha2-512`	512	ah: 96bit ICV (no document)
	512	ah-old: 128bit ICV (no document)

Following is a list of encryption algorithms that can be used as *`ealgo`* in `-E` *`ealgo`* of *`protocol`* parameter:

Algorithm	**Key Length (Bits)**	**Comment**
`des-cbc`	64	esp-old: rfc1829, esp: rfc2405
`3des-cbc`	192	rfc2451
`blowfish-cbc`	40 to 448	rfc2451
`cast128-cbc`	40 to 128	rfc2451

SETKEY File Example

```
add    10.0.11.41 10.0.11.33 esp 0x110010
         -E des-cbc “ESP with”
         -A hmac-md5 “authentication!!” ;
flush ;
dump esp ;
spdadd  10.0.11.41/32[21] 10.0.11.33/32[any] any
         -P out ipsec esp/transport/192.168.0.1-192.168.1.2/require ;
```

MULTINET SHOW

Displays MultiNet network information.

Format

`MULTINET SHOW`

Qualifiers

`/ALL`
Displays information provided by all other `MULTINET SHOW` qualifiers.

`/ARP`
Displays the Address Resolution Protocol (ARP) tables.

`/BUFFERS`
Displays MultiNet kernel memory usage statistics.

`/COMMUNITY_NAME=`*`community`*
Overrides the default community string (public) for remote SNMP requests. The `/SNMP_HOST` qualifier must be present if the `/COMMUNITY_NAME` qualifier is specified. A value must be passed to this qualifier.

`/CONFIGURATION`
Displays network interface configuration information.

```
                [ all, ]
                [ pid, ]
/CONNECTIONS[=( [ process_names ] )]
                [ nokernel ]
```

Displays network connections.

- If you specify `MULTINET SHOW` with no qualifiers, `/CONNECTIONS` is the default.

- If you specify the ALL keyword, sockets associated with active listeners also display.
- If you specify the PID keyword, the process ID (PID) displays.
- If you specify the PROCESS_NAMES keyword, the name of the process that owns each socket displays. Sockets not associated with a process (for example, an inbound TELNET session) display with a process name of kernel.
- If you specify PID or PROCESS_NAMES and the NOKERNEL keyword, connections not associated with processes do not display.

Warning! Line information is truncated if the display width is too small. As a consequence, IP addresses may appear incomplete. To display more complete information, increase the display width with the SET TERM /WIDTH=*value* or MULTINET SHOW /CONNECTIONS /WIDTH=*value* at the command prompt.

/CONTINUOUS
Updates the display continuously with information about the network by using the VMS Screen Management Graphics (SMG) library routines. If used with more than one other qualifier, MULTINET SHOW cycles between the different displays.

/IPS
When used with the /CONFIG=*filename* qualifier, writes the current stats of the filter server to the specified filename.

/FULL
Displays more information about a queue. Use /FULL only with /QUEUE. (See /QUEUE for more information.)

/HOST
Displays addresses and names for the host name specified.

/INTERFACE

Displays information about a specific interface. Use the MULTINET SHOW /STATISTICS command to display the available interfaces, then use SHOW /INTERFACE to display additional information on each interface.

/IP
Shows network connections. (/IP is the same as /CONNECTION.)

/MIB_VAR=*mibIIvalue*
Displays the value of SNMP MIB variables; used with the /SNMP_HOST qualifier. This value can be any MIB II variable described in RFC-1213.

```
            [ all, ]
            [ multinet, ]
/LICENSE[=( [nfs_server, ] ])
            [nfs_client ]
```

Displays the status of MultiNet software product licenses. Without a keyword, this qualifier displays license information including the authorization for MultiNet products. The ALL keyword is the default. All other values display license status for the specified product.

```
             [ all, ]
/NFSMOUNT[=( [ directory, ] )]
             [ exports ]
```

Indicates which hosts are mounted on your system, and what mount points are exported by the server.

- ALL displays all remote mounts.
- DIRECTORY displays directories that have been remotely mounted by clients.
- EXPORTS displays a list of exported file systems.

Use /NFSMOUNT with /REMOTE to display information about a remote host.

/OUTPUT=*file_spec*
Specifies a filename to which the command output is written. The default is SYS$OUTPUT.

```
            [ all, ]
            [ internet, ]
```

```
            [ ip,  ]
            [ ipx, ]
/PROTOCOLS= [ ns, ]
            [ spx, ]
            [ tcp ]
```

Specifies the protocols about which information is displayed. The default, `/PROTOCOLS=ALL`, displays information about all active protocols. Use `/PROTOCOLS` with other qualifiers. The quantity of information displayed varies by queue hardware; for example, UNIX shows more than just queues handled by other independent vendor's queue controllers.

/QUEUE=queue_name [/full]

Displays the contents of the specified local VMS and corresponding remote LPD protocol queues. Use the TCP LPD service to access the contents of the remote queue for display. If `/FULL` is specified, the queue is displayed in long form. If the remote system is also running MultiNet, the long form is identical to the short form. MultiNet queues configured with the STREAM protocol cannot be displayed with this command.

/REMOTE_HOST=*host*

Displays network status and configuration information about a remote host by using the NETSTAT service. The host specification can be either a host name or address. The remote host must support the NETSTAT service for this command to work.

If the remote host is also a MultiNet system, this command is the same as running `MULTINET SHOW /ALL` on the remote host.

```
        [ /destinations=(dest1[,dest2, . . . ])         ]
/ROUTE  [ /gateways=(gateway1[,gateway2, . . . ])       ]
        [ /interfaces=(interface1[,interface2, . . . ])]
```

Displays routing information for the IP, IPX, NS, and SPX protocols.

- `/DESTINATIONS` displays only routes to these destination addresses; this qualifier is only valid for IP routes.
- `/GATEWAYS` displays only routes through these gateways; this qualifier is only valid for IP routes.
- `/INTERFACES` displays only routes through these interfaces.

You can use all other `MULTINET SHOW` qualifiers with `MULTINET SHOW /ROUTE`.

> **Note:** The /ROUTE qualifier must precede all other qualifiers.

/RPC_PORTMAP
Displays the currently registered RPC protocols by contacting the RPC portmapper.

```
              [ interface ]
/STATISTICS[= [ protocol] ]
              [ all]
```

Displays network interface statistics, protocol statistics, or both. If /STATISTICS is specified with no value, interface statistics are displayed.

/SNMP_HOST=*hostname*
Used with the following MULTINET SHOW qualifiers to obtain information from a remote SNMP agent. You can override the default community name (public) using the /COMMUNITY_NAME qualifier.

/COMMUNITY_NAME
/CONNECTIONS[=(all)]
/ARP
/MIB_VAR
/ROUTE (note: /ROUTE must precede /SNMP_HOST on the command line)
/STATISTICS

```
                          [ host_table ]
/SYMBOLIC_ADDRESSES    [= [ nameserver ] ]
/NOSYMBOLIC_ADDRESSES     [ nameserver_first ]
```

Determines how certain fields in the output are formatted before being displayed to the user. These qualifiers are used with the other MULTINET SHOW qualifiers.

- /SYMBOLIC_ADDRESSES=HOST_TABLE specifies that the static host tables are used to translate IP addresses to host names, network numbers to network names, and port numbers to service names.
- /SYMBOLIC_ADDRESSES=NAMESERVER specifies that the Domain Name System (DNS) is queried to translate IP addresses into host names if the normal host table lookup fails. This operation can generate many queries to DNS domain servers (and can, therefore, be quite slow).

- /SYMBOLIC_ADDRESSES=NAMESERVER_FIRST specifies that the DNS is queried first to translate IP addresses into host names, falling back to the host tables if the query should fail.
- /NOSYMBOLIC_ADDRESSES specifies that "raw" protocol addresses and port number are displayed in the output, rather than determining the host, network, and service names that correspond to the addresses and numbers.

/TCP
Shows network connections. (/TCP is the same as /CONNECTION.)

/VERSION
Displays the MultiNet version and the version of the VMS Operating System.

/WIDTH=width
Specifies the width of displayed output when used with the /ARP, /CONNECTIONS, /ROUTE, and /STATISTICS qualifiers. The width must be greater than 80.

Examples

This example shows how to use the /OUTPUT qualifier to direct the output of a MULTINET SHOW command to the file MULTINET.ALL.

```
$ MULTINET SHOW /ALL /OUTPUT=MULTINET.ALL
$
$ MULTINET SHOW
MultiNet Active Connections:
Proto  Rcv-Q  Snd-Q Local Address (Port) Foreign Address  State
-----  -----  ----- -------------------- ---------------  -----
TCP        0      0 LOCALHOST(790)       LOCALHOST(RPC)   TIME_WAIT
TCP        0      0 LOCALHOST(1033)      LOCALHOST(SMTP)  TIME_WAIT
TCP        0      0 EXAMPLE(NETSTAT)     WARBUCKS(3335)   FIN_WAIT_2
TCP        0      0 EXAMPLE(FTP)         WARBUCKS(3334)   ESTABLISHED
TCP        0      0 EXAMPLE(1031)        WARBUCKS(TELNET) ESTABLISHED
UDP        0      0 EXAMPLE(NAMESERV)    *(*)
UDP        0      0 LOCALHOST(NAMESERV)  *(*)
UDP        0      0 EXAMPLE(DECNET)      IU(DECNET)

$ MULTINET SHOW /CONFIGURATION
** Configuration for file "MULTINET:NETWORK_DEVICES.CONFIGURATION" **
```

```
Device                              Adapter   CSR Address  Flags/Vector
------                              -------   -----------  ------------
se0  (Shared VAX/VMS Ethernet)      -NONE-    -NONE-       -NONE-
s10  (Serial Line IP)               -NONE-    -NONE-       -NONE-
dn0  (IP over DECNet link)          -NONE-    -NONE-       -NONE-

$ MULTINET SHOW /STATISTICS=INTERFACE
MultiNet  Network Interface statistics:
Name  Mtu   Network       Address       Ipkts  Ierrs Opkts Oerrs Collis
----  ---   -------       -------       -----  ----- ----- ----- ------
se0   1500  EXAMPLE-NET   EXAMPLE.COM   150    0     116   0     0
s10   1006  EXAMPLE-NET   EXAMPLE.COM   597    0     697   0     0
pd0   1500  EXAMPLE-NET   192.0.0.1     0      0     0     0     0
dno*  1500  EXAMPLE-NET   EXAMPLE.COM   0      0     0     0     0
lo0   1536  LOOPBACK-NET  LOCALHOST     53     0     53    0     0
$
```

This example displays the status of MultiNet licenses.

```
$ MULTINET SHOW /LICENSE
Process Software MultiNet 5.6 Rev A, HP rx2600 (1.30GHz/3.0MB), OpenVMS I64
V8.4-L1
Product         License    Authorization
---------       -------    -------------
MULTINET        Yes        A-2336-15873
NFS-SERVER      Yes        A-2336-15879
NFS-CLIENT      Yes        A-2336-15882
$
```

In this example, user ROSE on host EXAMPLE.COM has issued a print request to print the file `PROGRAMMERS.PS` on the `REMOTE_PS` local queue. The `REMOTE_PS` queue, however, is a MultiNet VMS remote print queue that uses the LPD protocol to send the print request to the print queue `SYS$PS` on host 192.0.0.89.

The `MULTINET SHOW /QUEUE` command is then used to display the contents of both queues; the remote queue first (`SYS$PS` on EXAMPLE) then the local queue (`REMOTE_PS`).

```
$ PRINT /QUEUE=REMOTE_PS PROGRAMMERS.PS
Job PROGRAMMERS (queue REMOTE_PS, entry 972) started on REMOTE_PS
$ MULTINET SHOW /QUEUE=REMOTE_PS
Jobname       Username    Entry   Blocks   Status
-------       --------    -----   ------   ------
MANAGE        DAISY       111     988      Printing
INSTALL       DAISY       115     238      Pending

Printer queue REMOTE_PS, on EXAMPLE::NLP0:"192.0.0.89/SYS$PS"
Jobname       Username    Entry   Blocks   Status
-------       --------    -----   ------   ------
```

```
PROGRAMMERS   ROSE        972     1112     Printing at block 370
$
```

This example displays the routing table on the local host without doing IP address-to-name translation.

```
$ MULTINET SHOW /ROUTE /NOSYMB
MultiNet IP Routing tables:
Destination     Gateway        Flags          Refcnt Use   Interface
-----------     -------        -----------    ------ ---   ---------
10.41.228.129   127.0.0.1      Up,Gateway,H 0      0     lo0
127.0.0.1       127.0.0.1      Up,Host      2      53    lo0
10.41.228.130   10.41.228.129  Up,Host      3      340   sl0
10.41.228.131   10.41.228.129  Up,Host      0      0     dn0
0.0.0           10.41.228.130  Up,Gateway   0      353   sl0
10.41.228.64    10.41.228.65   Up           2      112   se0
10.41.228       10.41.228.1    Up           0      0     pd0
$
```

This example displays local host information.

```
$ MULTINET SHOW/ROUTE/DESTINATIONS=127.0.0.1
MultiNet IP Routing tables:
Destination   Gateway     Flags    Refcnt   Use   Interface
-----------   -------     -----    ------   ---   ---------
LOCALHOST     LOCALHOST   Up,Host    1      464   lo0
$
```

This example displays the interface SE1 along with its associated packet filters.

```
$ MULTINET SHOW /INTERFACE SE1 /FILTERS
Device se1: flags=8863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST,D2>
            VMS Device = EWB0
            IP Address = 192.168.0.16
            No common links defined

MultiNet Packet Filter List for se1:

Logging is disabled

                          Source Address / Port
Action   Proto    Hits    Destination Address / Port
------   -----    -----   ------------------------------------------
deny      tcp        0    192.168.0.11/32
                          192.168.0.0/24  eq 22
                          LOG
```

```
               START: 16-MAR-2020 10:33:19  END: 16-MAR-2020 10:38:19

permit     ip    13484    0.0.0.0/0
                          0.0.0.0/0
                          FLTSVR

            Average 0 bytes out, 0 bytes in per second
            Average 0 packets out, 0 packets in per second
```

This example displays the current version of MultiNet and the VMS Operating System.

```
$ MULTINET SHOW /VERSION
Process Software MultiNet 5.6 Rev A, HP rx2600 (1.30GHz/3.0MB), OpenVMS I64
V8.4-H1
```

MULTINET TCPDUMP

Displays the contents of Ethernet packet headers that match the specified Boolean expression. To stop the dump, press **Ctrl+C**.

Format

`MULTINET TCPDUMP` *`[expression]`*

Restrictions

The following restrictions apply to the use of `MULTINET TCPDUMP`.

- Although the TCPDUMP expression grammar allows the use of the exclamation point (!) character as the NOT operator and as part of the NOT-EQUAL comparator, DCL interprets it as a comment character. Therefore, use NOT instead.

  For example, to print the start and end packets (the SYN and FIN packets) of each TCP conversation that involves a remote host:

  ```
  $ MULTINET TCPDUMP NOT (TCP[13] & 3 = 0) AND NOT SRC -
  _$ AND DST NET LOCALNET
  ```

- PHY_IO, LOG_IO, and SYSPRV or BYPASS privileges are required to use TCPDUMP.
- The packet filter code is not very efficient and adds significant overhead to your VMS system when monitoring a busy network. In addition, if you are using DNS and a problem occurs with name server access, TCPDUMP can appear to hang while waiting for a response from the network.
- IP options are ignored and not displayed.
- Understands PPP frames and does not treat all data as IP datagrams.
- No attempt is made to reassemble IP fragments or at least compute the right length for the higher level protocol.
- Name server inverse queries are not dumped correctly. An empty question section is printed rather than the real query in the answer section.
- Though TCPDUMP recognizes IPsec packets, it does not decrypt encrypted packets.

Parameters

expression

Selects which packets are dumped. If an expression is not given, all packets on the net are dumped. Otherwise, only packets for which the expression is "true" are dumped. Enter `HELP MULTINET TCPDUMP EXPRESSION` for a list of expression values.

Qualifiers

/AFTER=*time*
Selects packets dated after the specified time. The time value can be any valid OpenVMS time specification (absolute, delta, or a combination of the two).

/BEFORE=*time*
Selects packets dated prior to the specified time. The time value can be any valid OpenVMS time specification (absolute, delta, or a combination of the two).

/COUNT=*number_of_packets*
Exits TCPDUMP after the specified number of packets is received. The default is 0, or no limit.

/DEBUG

Displays debugging information.

/DEVICE=*devicename*
Specifies the VMS device name of the Ethernet device to use. By default, TCPDUMP searches for ECA0, EIA0, EWA0, EZA0, EXA0, EFA0, ETA0, ERA0, ESA0, ICA0, IRA0, LLA0, XEA0, and XQA0 devices.

/DOMAINS
/NODOMAINS
Displays host names with the domain information; `/NODOMAINS` strips the domain names.

/EBCDIC
Modifies the behavior of the `/HEXADECIMAL` qualifier by adding the EBCDIC translation of the data in addition to the ASCII translation to the TCPDUMP output.

/ETHERNET_HEADER
Displays the Ethernet header (source, destination, protocol, and length) on each dump line.

/FILE_FORMAT=SNIFFER
Use in conjunction with /READ_BINARY or /WRITE_BINARY to read or generate output automatically formatted for display on version 2.0 Network General sniffers.

/FOREIGN_NUMERICALLY
Displays "foreign" Internet addresses numerically rather than symbolically.

/HEXADECIMAL_DUMP
Displays each packet (less its 14-byte Ethernet header) in hexadecimal format. Up to 64 bytes of the packet are printed.

/INTERFACE=*device*
Specifies the device to trace. Valid devices are those for Ethernet/FDDI (se), the loopback connection (lo0), SLIP lines (sl), PPP lines (ppp), PSI connections (psi), and IP-over-DECNET connections (dn). This qualifier cannot be used with the /DEVICE qualifier.

/NUMERICALLY
Specifies that host addresses and port numbers are not converted to names on output.

/OUTPUT=*filename*
Redirects TCPDUMP output to a file.

/QUIET
Specifies that less protocol information is displayed, making output lines shorter.

/READ_BINARY=*binary_file*

Reads in a file previously written using the /WRITE_BINARY qualifier. (Refer to /WRITE_BINARY for more information.)

This file is written in libpcap format. When the interface specified is an Ethernet device the data in the file can be analyzed with Ethereal and similar tools.

You can use /READ_BINARY with /FILE_FORMAT=SNIFFER to read output formatted automatically for display on version 2.0 Network General sniffers. This feature permits sites to analyze Network General analyzer, rather than only examining the TCPDUMP packets.

/RPC
Interprets RPC calls in the output.

/SNAPSHOT_SIZE=*snaplen*
Indicates the specified number of bytes of data to capture from each packet rather than the default of 54 bytes (which is adequate for most applications). 96 bytes is adequate for IP, ICMP, TCP, and UDP, but may truncate protocol information from name server and NFS packets.

/TIMESTAMPS=*value*
/NOTIMESTAMPS
Causes TCPDUMP to display a timestamp on each output line. Accepted values are DEFAULT, UNIX, DELTA, and RELATIVE. The default /NOTIMESTAMPS qualifier disables the TCPDUMP timestamp on each output line.

/VERBOSE
Provides additional information in the output listing.

/WRITE_BINARY=*binary_file*
Stores the output of TCPDUMP in a file. Use this qualifier to "record" the TCPDUMP information until you press **Ctrl+Y**. After recording the output of a TCPDUMP session, use /READ_BINARY to read in the binary file for examination.

You can use /WRITE_BINARY with /FILE_FORMAT=SNIFFER to generate output automatically formatted for display on version 2.0 Network General sniffers. This feature permits sites to analyze Network General analyzer, rather than only examining the TCPDUMP packets.

Examples

This example displays all traffic addressed to or transmitted from host OL.EXAMPLE.COM.

```
$ MULTINET TCPDUMP HOST OL.EXAMPLE.COM
18:56:24.25 BIG.EXAMPLE.COM.x11 > OL.EXAMPLE.COM.1030:.ack 2152730 win 4096.
```

This example displays all traffic between local hosts and hosts at the network IRIS-ETHER.

```
$ MULTINET TCPDUMP NET IRIS-ETHER
. . .
```

This example displays all FTP traffic being sent to host BETTY.EXAMPLE.EDU.

```
$ MULTINET TCPDUMP -
_$ DST HOST BETTY.EXAMPLE.EDU AND (PORT FTP OR PORT FTP-DATA)
```

This example displays IP traffic not sent from or destined for the network IRIS-ETHER. If IRIS-ETHER is the local network, only transient traffic displays.

```
$ MULTINET TCPDUMP IP AND NOT NET IRIS-ETHER
```

MULTINET TRACEROUTE

Attempts to trace the route that an IP packet follows to another Internet host.

Format

`MULTINET TRACEROUTE` *`host`* [*`data_length`*]

Description

`TRACEROUTE` finds the intermediate hops by sending probe packets with a small TTL (time-to-live), then listening for an ICMP "time exceeded" reply from a gateway. It starts probing with a TTL of one, then increases by one in each successive probe until an ICMP "port unreachable" reply is received (indicating that a probe reached the host) or the TTL exceeded 30 (the default maximum).

By default, three probes are sent at each TTL setting, and a line is printed showing the TTL, the gateway address, and round-trip time of each probe. If the probe answers come from different gateways, the address of each responding system is printed. If there is no response within a five-second timeout interval, a `*` is printed for that probe. `TRACEROUTE` prints a `!` after the time if the TTL is less than or equal to one. The following table shows other possible annotations:

Annotation	Meaning
`!H`	Host unreachable
`!N`	Network unreachable
`!P`	Protocol unreachable
`!S`	Source route failed
`!F`	Fragmentation needed

The `!S` and `!F` annotations are rare and indicate that the associated gateway is not working properly. If most of the probes result in "unreachable" annotations, `TRACEROUTE` stops running and exits.

Parameters

host
Specifies the target host to which you want to determine the route.

data_length
Specifies the amount of data sent in each ICMP Echo Request packet.

Qualifiers

/DEBUG
Enables socket-level debugging in the MultiNet kernel. This qualifier is used only for debugging the MultiNet kernel.

/IPV6
Specifies that an IPv6 trace is desired. (The default is IPv4.)

/MAXIMUM_TTL=*maximum_ttl*
Specifies the maximum TTL (time-to-live) to explore looking for ICMP Time Exceeded responses. If not specified, the default of 30 hops is used.

/MINIMUM_TTL=*minimum_ttl*
Specifies the minimum TTL to explore looking for ICMP Time Exceeded responses. If not specified, the default of 1 hop is used.

/NUMBER_OF_PROBES=*n*
Specifies the number of probe packets sent to each hop (by default, 3).

/OUTPUT=*filename*
Redirects TRACEROUTE output to a file.

/PORT=*udp_port*

Specifies a non-standard port number. TRACEROUTE sends data to an unused port and expects an error message. If the default port of 33434 is in use, use /PORT to specify another.

/ROUTE
/NOROUTE
Disables any IP routing of the ICMP packets. The default, /ROUTE, allows IP routing to send the packet to destinations separated by gateways.

/SOURCE=*ip_address*
Specifies the local IP address from which packets are sent.

/SYMBOLIC_ADDRESSES
/NOSYMBOLIC_ADDRESSES
Specifies that IP addresses are displayed numerically instead of being converted into host names. /SYMBOLIC_ADDRESSES is the default.

/TYPE_OF_SERVICE=*tos*
Specifies the Type-Of-Service (TOS) field of the IP packet. The default TOS is 0 (no specific type of service).

/VERBOSE
Displays extra information as ICMP packets are sent or received.

/WAIT_TIME=*seconds*
Specifies how long TRACEROUTE waits for responses (by default, 5 seconds).

Examples

This example shows tracing a route to an NSFnet gateway. Note: lines 2 and 3 are the same. This is because the gateway lilac-dmc.Berkeley.Edu has a kernel bug that causes the system to forward packets with a TTL of zero.

```
$ MULTINET TRACEROUTE NIS.NSF.NET
traceroute to nis.nsf.net (35.1.1.48), 30 hops max, 38 byte packet
```

```
1  FLOWERS.BARRNET.NET (192.41.228.71)        0 ms   0 ms   0 ms
2  UCSC.BARRNET.NET (131.119.46.7)           10 ms   0 ms  20 ms
3  SU1.BARRNET.NET (131.119.1.5)             10 ms  20 ms  20 ms
4  SU-B.BARRNET.NET (131.119.254.201)        20 ms  20 ms  20 ms
5  E-NSS.BARRNET.NET (192.31.48.244)         50 ms  10 ms  20 ms
6  t3-1.cnss9.t3.nsf.net (140.222.9.2)       20 ms  10 ms  20 ms
7  t3-3.cnss8.t3.nsf.net (140.222.8.4)       20 ms  30 ms  30 ms
8  t3-0.cnss24.t3.nsf.net (140.222.24.1)     70 ms  60 ms  60 ms
9  t3-0.cnss40.t3.nsf.net (140.222.40.1)     70 ms  70 ms  60 ms
10 t3-0.cnss41.t3.nsf.net (140.222.41.1)     70 ms  70 ms  60 ms
11 t3-0.enss131.t3.nsf.net (140.222.131.1)   70 ms  80 ms  80 ms
12 nis.nsf.net (35.1.1.48)                   80 ms  80 ms  70 ms
$
```

MULTINET X11DEBUG

The X11DEBUG utility performs four tests that check the most common causes of problems encountered when running X11 clients over MultiNet:

- Checks for the UCX driver.
- Verifies that a display has been defined with the SET DISPLAY command.
- Checks TCP/IP connections.
- Verifies that the X11 client can access the server.

If any of these tests fail, X11DEBUG recommends a course of action to resolve the problem. Otherwise, X11DEBUG displays the message, %X11DEBUG-S-PASSEDALL, passed all X11 tests.

Format

```
MULTINET X11DEBUG [/LOG]
```

Qualifier

/LOG

Enables additional debugging information.

2. DECNET-CONFIG Command Reference

This chapter describes the commands you can run from the DECNET-CONFIG command line. DECNET-CONFIG lets you examine, modify, and save configuration files for DECnet-over-IP circuits.

To invoke DECNET-CONFIG:

```
$ MULTINET CONFIGURE /DECNET
```

At any DECNET-CONFIG prompt, type **?** to list the available commands. Use the DECNET-CONFIG HELP command to view online help for each DECNET-CONFIG command.

Changes do not take effect until you do one of the following:

- Restart the DECnet-over-IP driver with the command

  ```
  $ @MULTINET:DECNET-CIRCUITS.COM.
  ```

- Restart your system.

For details on configuring DECnet-over-IP circuits, refer to the *MultiNet Installation and Administrator's Guide.*

Command Summary

The below table lists the commands you can use from the DECNET-CONFIG prompt.

Command	Description
ADD	Adds a DECnet-over-IP circuit to the current configuration.
ATTACH	Attaches to a subordinate process.
CLEAR	Deletes all DECnet-over-IP circuits from the current configuration; same as ERASE.

DELETE	Deletes a single DECnet-over-IP circuit from the current configuration.
ERASE	Deletes all DECnet-over-IP circuits from the current configuration; same as CLEAR.
EXIT	Exits DECNET-CONFIG and saves the current configuration.
GET	Reads in a DECnet-over-IP circuit configuration file; same as USE.
HELP	Displays command information.
MODIFY	Changes the parameters in a DECnet-over-IP circuit configuration file.
PUSH	Accesses the DCL command line and pauses DECNET-CONFIG.
QUIT	Exits DECNET-CONFIG, but prompts to save changes before exiting.
SAVE	Writes out the DECNET-CONFIG current configuration file; same as WRITE.
SHOW	Displays the current DECnet-over-IP circuit configuration.
SPAWN	Executes a single DCL command or starts a subprocess.
STATUS	Displays the status of the DECnet-over-IP circuit configuration.
USE	Reads in a DECnet-over-IP circuit configuration file; same as GET.
VERSION	Displays DECNET-CONFIG version and release information.
WRITE	Writes out the current DECNET-CONFIG configuration file; same as SAVE.

ADD

Adds a new DECnet-over-IP circuit to the current MultiNet configuration, and prompts for circuit configuration parameters.

Format

ADD [*circuit_name*]

Parameter

circuit_name

Specifies the name of the DECnet circuit to add to the configuration. If not specified in the command, the first unused circuit in the configuration is used by default.

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
DECNET-CONFIG>ADD
[Adding new configuration entry for DECnet circuit "TCP-0-0"]
Destination IP Address: [NONE] 192.0.0.6
DECnet circuit cost: [1] 1
DECnet hello timer (in seconds): [300] 300
[TCP-0-0 => 192.0.0.6 (Cost=1, Hello Timer=300)
DECNET-CONFIG>EXIT
[Writing configuration to MULTINET:DECNET-CIRCUITS.COM]
$
```

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the SPAWN SHOW PROCESS /SUBPROCESSES command to list the names of the subprocesses. Use the DCL LOGOUT command to return to the original process. If the MULTINET_DISABLE_SPAWN logical is enabled, ATTACH does not work.

Format

ATTACH *process-name*

Parameter

process-name
Specifies the name of a process to which you want your terminal attached. (Not all subprocesses can be attached; some testing may be required.)

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM
DECNET-CONFIG>SPAWN
$ MM
MM>SPAWN SHOW PROCESS/SUB
...
There are 3 processes in this job:
_TWA42:
PROC_1
PROC_2 (*)
MM>ATTACH _TWA42:
DECNET-CONFIG>ATTACH PROC_1
MM>QUIT
$ LOGOUT
DECNET-CONFIG>
```

This example shows the use and exit of attached subprocesses.

The first command uses SPAWN to create a subprocess. MM is invoked from the DCL command line. Next, the SPAWN SHOW PROCESS/SUB command is used to list all subprocess names. The display

shows that three subprocesses are active. (Process _TWA42: is DECNET-CONFIG, PROC_1 is MM, and PROC_2 is the SPAWN SHOW PROCESS/SUB command.)

In the next command, the MM ATTACH command returns control to the DECNET-CONFIG process. From this utility, ATTACH returns control to MM. To exit, the QUIT command is invoked from MM, and LOGOUT is invoked at the original spawned DCL command line; control returns to DECNET-CONFIG. (If SPAWN SHOW PROCESS/SUB had been entered, only this command and the configuration processes would be active.)

CLEAR

Deletes all DECnet-over-IP circuits from the current MultiNet configuration.

Format

```
CLEAR
```

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>CLEAR
DECNET-CONFIG>EXIT
$
```

DELETE

Deletes the specified DECnet-over-IP circuit from the current MultiNet configuration.

Format

DELETE *circuit_name*

Parameter

circuit_name
Specifies the name of the DECnet-over-IP circuit to delete.

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>DELETE TCP-0-0
DECNET-CONFIG>EXIT
$
```

ERASE

Clears all DECnet-over-IP circuits from the current MultiNet configuration. (Functionally equivalent to `CLEAR`.)

Format

```
ERASE
```

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>ERASE
DECNET-CONFIG>EXIT
$
```

EXIT

Saves the current configuration, if it has been modified, then quits.

Format

```
EXIT
```

Examples

If the configuration has not changed, a message displays indicating that the configuration file is not updated.

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>EXIT
$
```

If the configuration has changed, a message displays indicating that the configuration file has been updated.

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>CLEAR
DECNET-CONFIG>EXIT
[Writing configuration to MULTINET:DECNET-CIRCUITS.COM}
$
```

GET

Reads in a MultiNet DECnet-over-IP circuit configuration file, which defaults to the highest version number of the MULTINET:DECNET-CIRCUITS.COM file. (Functionally equivalent to USE.) After using GET, you can use other DECNET-CONFIG commands to display and modify the new configuration.

Format

GET *config_file*

Parameter

config_file
Specifies the name of the configuration file to read in.

Example

This example reads in the highest version number of the MULTINET_ROOT:[MULTINET]DECNET-CIRCUITS.COM configuration file. In this case, the highest version number is ";7". This number is appended to the file name as the value ".7".

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>GET
[Reading in configuration from MULTINET_ROOT:[MULTINET]DECNET-
CIRCUITS.COM.7]
DECNET-CONFIG>EXIT
$
```

HELP

Invokes the help command.

Format

`HELP [topics]`

Parameter

topics

Contains a space-delimited list of topics that begins with a topic followed by subtopics. The default topic is `HELP`.

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>HELP ?
ADD      ATTACH   CLEAR   DELETE   ERASE    EXIT     GENERAL    GET
MODIFY   PUSH     QUIT    SAVE     SHOW     SPAWN    STATUS     USE
VERSION  WRITE
DECNET-CONFIG>
```

MODIFY

Changes the parameters of the specified DECnet-over-IP Circuit in the MultiNet configuration.

Format

MODIFY *circuit_name*

Parameter

circuit_name
Specifies the name of the DECnet-over-IP circuit whose parameters will be modified.

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
DECNET-CONFIG>MODIFY TCP-0-0
[Adding new configuration entry for DECnet circuit "TCP-0-0"]
Destination IP Address: [10.0.0.6] RETURN
DECnet circuit cost: [1]  5
DECnet hello timer (in seconds): [300] RETURN
[TCP-0-0 => 10.0.0.6 (Cost=5, Hello Timer=300)]
DECNET-CONFIG>EXIT
$
```

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command.

If the MULTINET_DISABLE_SPAWN logical is set, PUSH does not work.

Format

PUSH

Example

In this example, PUSH is used to access the DCL command line to disable broadcasts. The LOGOUT command returns control to DECNET-CONFIG.

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>PUSH
$ SET TERM /NOBROADCAST
$ LOGOUT
DECNET-CONFIG>
```

QUIT

If the configuration file has been edited, QUIT prompts you to save the file before quitting.

Format

```
QUIT
```

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>QUIT
Configuration modified, do you want to save it ? [NO]NO
```

SAVE

Writes out the current configuration file. (Functionally equivalent to `WRITE`; see `WRITE` for additional information.)

SHOW

Displays the current MultiNet DECnet-over-IP circuit configuration.

Format

```
SHOW
```

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>SHOW
Circuit Name        IP Destination       Cost      Hello Timer
------------        --------------       ----      -----------
TCP-0-0             10.1.30.1          1         300
DECNET-CONFIG>QUIT
```

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. If the MULTINET_DISABLE_SPAWN logical is set, SPAWN does not work.

Format

SPAWN [*command*]

Parameter

command
Specifies a command to execute. If you omit *command*, a DCL command line subprocess is created.

QUALIFIERS

/INPUT=*file-spec*
Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES
/NOLOGICAL_NAMES
Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS
/NOSYMBOLS
Specifies that global and local names are not passed to the subprocess.

/WAIT
/NOWAIT
Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=*file-spec*

Specifies a file that retains the output of the command invoked with SPAWN. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after SPAWN or other qualifiers.

Examples

This example displays terminal information, captures the output in a file, and displays the information with the TYPE command.

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>SPAWN/OUTPUT=FOO. SHOW TERM
DECNET-CONFIG>SPAWN TYPE FOO.
...
```

This example invokes a command procedure.

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>SPAWN @COMPROC
...
```

This example displays help information about DECNET-CONFIG. Use the LOGOUT command to return control to DECNET-CONFIG.

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>SPAWN
$ HELP MULTINET CONFIGURE /DECNET ...
$ LOGOUT
DECNET-CONFIG>
```

STATUS

Displays the status of the MultiNet DECnet-over-IP circuit configuration.

Format

```
STATUS
```

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>STATUS
This is the MultiNet DECnet circuit configuration program Version 5.6 (nnn)
There are 1/100 circuits in the current configuration.
The configuration MULTINET:DECNET-CIRCUITS.COM is not modified.
DECNET-CONFIG>QUIT
$
```

USE

Reads in a configuration file. (Functionally equivalent to `GET`; see `GET` for additional information.)

Format

`USE` *`config_file`*

Parameter

`config_file`
Specifies the name of the configuration file to read in.

VERSION

Displays the version and release information of the MultiNet DECnet-over-IP circuit configuration program.

Format

```
VERSION
```

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>VERSION
This is the MultiNet DECnet circuit configuration program Version 5.6 (nnn)
DECNET-CONFIG>QUIT
$
```

WRITE

Writes the current MultiNet DECnet-over-IP circuit configuration to a file. (Functionally equivalent to SAVE.)

Format

WRITE [*config_file*]

Parameter

config_file
Specifies the name of the DECnet-over-IP circuit configuration file to write out (by default, the same file from which the configuration was read).

Example

```
$ MULTINET CONFIGURE /DECNET
MultiNet DECNET Circuit Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:DECNET-CIRCUITS.COM]
DECNET-CONFIG>WRITE
[Writing configuration to MULTINET_ROOT:[MULTINET]DECNET CIRCUITS.COM.2]
DECNET-CONFIG>EXIT
```

3. MAIL-CONFIG Command Reference

This chapter describes the commands you can run from the MAIL-CONFIG command line. MAIL-CONFIG lets you examine, modify, and save configuration files for the MultiNet SMTP mail system.

To invoke MAIL-CONFIG:

```
$ MULTINET CONFIGURE /MAIL
```

At any MAIL-CONFIG prompt, type **?** to list the available commands. Use the MAIL-CONFIG HELP command to view online help for each MAIL-CONFIG command.

Changes do not take effect until you do one of the following:

- Restart the SMTP service with the commands:

  ```
  @MULTINET:START_SMTP.COM
  ```

  or

  ```
  @MULTINET:START_SMTP_LOCAL.COM
  ```

- Restart your system.

For details on configuring electronic mail, refer to the *MultiNet Installation and Administrator's Guide*.

Command Summary

The below table lists the commands you can run from the MAIL-CONFIG prompt.

MAIL-CONFIG Command	Description
ADD GATEWAY	Adds a mail gateway to another domain.
ADD LOCAL-DOMAIN	Adds a domain to a list of domains that the MultiNet SMTP symbiont considers to be local. If users send mail to hosts beyond

	the local domains, MultiNet forwards the mail to the mail hub specified by the FORWARDER parameter. The local domain list affects mail forwarding only when the FORWARD-REMOTE-MAIL parameter is TRUE.
ADD QUEUE-GROUP	Forms a mail queue grouping of nodes in a cluster, or adds new nodes to an existing queue group.
ATTACH	Attaches your terminal to another process.
CLEAR	Erases all information from the current configuration; same as ERASE.
DELETE GATEWAY	Deletes a mail gateway.
DELETE LOCAL-DOMAIN	Deletes a domain from MultiNet's list of local domains.
DELETE QUEUE-GROUP	Deletes a queue group or removes a node from a queue group. When a node is removed from a named queue group, it becomes part of the default queue group.
ERASE	Erases all information from the current configuration; same as CLEAR.
EXIT	Saves the configuration file and exits from MAIL-CONFIG.
GET	Reads in a MultiNet SMTP configuration file. (Functionally equivalent to USE.)
HELP	Invokes MAIL-CONFIG command help.
PUSH	Accesses the DCL command interpreter.
QUIT	Prompts you to save the configuration file if it has been modified, then exits MAIL-CONFIG.
REMOVE GATEWAY	Functionally equivalent to DELETE GATEWAY.

REMOVE QUEUE-GROUP	Functionally equivalent to DELETE QUEUE-GROUP.
SAVE	Saves the current configuration file.
SET ACCOUNTING-HOST	Sets the host that SMTP accounting information should be sent to.
SET ACCOUNTING-PORT	Sets the port that the accounting logger is to listen on.
SET ALIAS-FILE	Identifies the file that holds mail aliases.
SET DECNET-DOMAIN	Sets the domain name for DECnet mail.
SET DELIVERY-RECEIPTS	Specifies whether mail receipts are sent when incoming mail containing Delivery-Receipt-To: or Return-Receipt-To: headers is submitted to the SMTP queue.
SET DISABLE-PSIMAIL	When TRUE, the MultiNet SMTP symbiont looks for messages addressed through PSImail, usually of the form PSI%*address*::*user*, and returns them to the sender marked user unknown.
SET DISALLOW-USER-REPLY-TO	When set to TRUE, prevents VMS MAIL users from setting a Reply-To: header address with the logical name MULTINET_SMTP_REPLY_TO.
SET FORWARDER	Specifies the host that will forward mail messages to other hosts.
SET FORWARD-LOCAL-MAIL	Forwards mail addressed to users on the local host to a central mail hub specified by the FORWARDER parameter.
SET FORWARD-REMOTE-MAIL	Forwards mail addressed to users on non-local hosts to a central mail hub specified by the FORWARDER parameter.
SET HEADER-CONTROL	Specifies which RFC-822 message headers should be included in messages delivered to local VMS MAIL users.

SET HOST-ALIAS-FILE	Specifies a file from which MultiNet obtains a list of host aliases.
SET LOCAL-MAIL-FORWARDER	Forwards local mail to a specific host.
SET POSTMASTER	Identifies the user responsible for mail on the system.
SET QUEUE-COUNT	Specifies the number of mail processing queues that should be created on a particular system.
SET REPLY-CONTROL	Specifies how Internet mail headers should be mapped to the VMS MAIL "From" header.
SET RESENT-HEADERS	When FALSE, the MultiNet SMTP symbiont omits the Resent-From, Resent-To, and Resent-Date headers that are usually included when a message is forwarded using a VMS MAIL forwarding address.
SET RETRY-INTERVAL	Specifies the amount of time that elapses before another attempt is made to send a message after a failed attempt.
SET RETURN-INTERVAL	Specifies the amount of time that a message can remain in the processing queue before it is returned to sender.
SET RFC2789	Enables the startup of the Mail Monitoring MIB (RFC 2789). Requires SNMP AGENTX to be set.
SET SEND-BROADCAST-CLASS	Specifies the broadcast class to use to deliver immediate SEND messages.
SET SMTP-HOST-NAMES	Sets the host name from which all outgoing mail appears to be sent and aliases for which this host accepts incoming mail.
SET START-QUEUE-MANAGER	Determines whether START_SMTP.COM starts the VMS queue manager if it is not already running.
SHOW	Displays the current configuration.

SPAWN	Executes a single DCL command.
STATUS	Indicates whether the SMTP configuration has been modified.
USE	Reads in a non-standard configuration file.
VERSION	Displays the MAIL-CONFIG version and release information.
WRITE	Saves the current configuration file.

ADD GATEWAY

Adds a mail gateway to another domain. Specifies a gateway host to which mail for the specified host or domain will be forwarded.

> **Note:** To define a mail gateway to an IP address (instead of a host name), you must enclose the IP address in square brackets.

Format

`ADD GATEWAY` *`domain_name hostname`*

Parameters

`domain_name`

Specifies the name of the domain for which the new gateway will handle mail. This can be a fully qualified host name (for example, WHORFIN.EXAMPLE.COM) or a domain tag beginning with a dot (for example, .BITNET).

`hostname`

Specifies the name of the host that acts as a gateway for mail addressed to *`domain_name`*.

ADD LOCAL-DOMAIN

Adds a domain to a list of domains that the MultiNet SMTP symbiont considers to be local. If users send mail to hosts beyond the local domains, MultiNet forwards the mail to the mail hub specified by the FORWARDER parameter. The local domain list affects mail forwarding only when the FORWARD-REMOTE-MAIL parameter is TRUE.

Format

ADD LOCAL-DOMAIN *domain_name*

Parameter

domain_name
Specifies the name of a domain (for example, LOT-49.EXAMPLE.COM) that MultiNet considers to be local.

ADD QUEUE-GROUP

Forms a mail queue grouping of nodes in a cluster, or adds new nodes to an existing queue group. The SMTP queues on the nodes in the group you create will share responsibility for handling mail messages generated on nodes within the group. If a node is not placed in a named queue group, it is made part of the default queue group.

Format

`ADD QUEUE-GROUP` *`groupname`* [*`node_name_list`*]

Parameters

groupname
Specifies the name of the queue group to add, or the name of an existing group to which nodes will be added.

node_name_list
Contains a list of names of VMScluster (SCS) nodes to add to the queue group.

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the `SPAWN SHOW PROCESS /SUBPROCESSES` command to list the names of the subprocesses. Use the `DCL LOGOUT` command to return to the original process. If the `MULTINET_DISABLE_SPAWN` logical is enabled, `ATTACH` does not work.

Format

`ATTACH` *process-name*

Parameter

process_name

Specifies the name of a process to which you want your terminal attached. (Not all subprocesses can be attached; some testing may be required.)

CLEAR

Clears all information from the current configuration. (Functionally equivalent to `ERASE`.)

Format

```
CLEAR
```

DELETE GATEWAY

Deletes a mail gateway.

Format

`DELETE GATEWAY` *`domain_name`*

Parameter

`domain_name`
Specifies the name of the domain whose gateway will be deleted.

DELETE LOCAL-DOMAIN

Deletes a domain from MultiNet's list of local domains.

Format

`DELETE LOCAL-DOMAIN` *`domain_name`*

Parameter

`domain_name`
Specifies the name of the domain to delete from the list of local domains.

DELETE QUEUE-GROUP

Deletes a queue group or removes a node from a queue group. When a node is removed from a named queue group, it becomes part of the default queue group.

Format

```
DELETE QUEUE-GROUP group_name [node_names]
```

Parameters

group_name
Specifies the name of the group to delete or the name of the group from which to remove the specified nodes.

node_names
Specifies the VMScluster (SCS) node name to remove from the specified queue group.

ERASE

Erases all information from the current configuration. (Functionally equivalent to `CLEAR`.)

Format

`ERASE`

EXIT

Saves the current configuration, if it has been modified, then quits.

Format

```
EXIT
```

GET

Reads in a MultiNet SMTP configuration file. (Functionally equivalent to `USE`.) After a `GET`, you can use the various configuration commands to modify the SMTP configuration.

Format

`GET` *`config_file`*

Parameter

`config_file`
Specifies the name of the SMTP configuration file to read in.

HELP

Invokes command help.

Format

`HELP [topics]`

Parameter

topics
Contains a space-delimited list of topics that begins with a topic followed by subtopics. The default topic is `HELP`.

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command.

If the MULTINET_DISABLE_SPAWN logical is set, PUSH does not work.

Format

PUSH

QUIT

If the configuration file has been edited, `QUIT` prompts you to save the file before quitting.

Format

```
QUIT
```

REMOVE GATEWAY

Functionally equivalent to `DELETE GATEWAY`.

Format

`REMOVE GATEWAY` *`domain_name`*

Parameter

`domain_name`
Specifies the name of the gateway to remove.

REMOVE QUEUE-GROUP

Functionally equivalent to `DELETE QUEUE-GROUP`.

Format

`REMOVE QUEUE-GROUP` *`group_name`* [*`node_names`*]

Parameters

`group_name`
Specifies the name of the group to remove or the name of the group from which to remove the specified nodes.

`node_names`
Specifies the VMScluster (SCS) node name to remove from the specified queue group.

SAVE

Writes the current MultiNet SMTP configuration to SMTP configuration files. (Functionally equivalent to `WRITE`.)

Format

`SAVE` *`config_file`*

Parameter

`config_file`
Specifies the name of the file to which to write the current MultiNet SMTP configuration (by default, the same file from which it was read).

SET ACCOUNTING-HOST

Sets the host that SMTP accounting information should be sent to.

Format

`SET ACCOUNTING-HOST` *`hostname`*

Parameter

`hostname`

Specifies the name of the host that SMTP accounting information should be sent to. `ACCOUNTING-PORT` must be specified also.

SET ACCOUNTING-PORT

Sets the port that the accounting logger is to listen on.

Format

`SET ACCOUNTING-PORT` *`port_number`*

Parameter

`port_number`
Specifies the port number that the accounting logger is listening on. `ACCOUNTING-HOST` must be specified also.

SET ALIAS-FILE

Identifies the file that holds system-wide mail aliases.

Format

`SET ALIAS-FILE` *`[file-spec]`*

Parameter

`file-spec`
Specifies the name of the file that contains system-wide mail aliases (by default, `MULTINET:SMTP_ALIASES`).

SET DECNET-DOMAIN

Sets the domain name for DECnet mail.

Format

`SET DECNET-DOMAIN` *`domain_name`*

Parameter

`domain_name`
Specifies the domain name for DECnet mail. For information on the SMTP-DECnet gateway, see the *MultiNet Installation and Administrator's Guide*.

SET DELIVERY-RECEIPTS

Specifies whether mail receipts are sent when incoming mail containing Delivery-Receipt-To: or Return-Receipt-To: headers is submitted to the SMTP queue. If `TRUE`, mail receipts are sent.

Format

```
SET DELIVERY-RECEIPTS {TRUE | FALSE}
```

SET DISALLOW-USER-REPLY-TO

When set to `TRUE`, prevents VMS MAIL users from setting a Reply-To: header address with the `MULTINET_SMTP_REPLY_TO` logical name.

Format

```
SET DISALLOW-USER-REPLY-TO {TRUE | FALSE}
```

SET DISABLE-PSIMAIL

When TRUE, the MultiNet SMTP symbiont looks for messages addressed through PSImail, usually of the form PSI%*address*::*user*, and returns them to the sender marked user unknown. The default is FALSE. This parameter does not affect mail delivery to local users who have set up forwarding entries to PSImail addresses with the VMS MAIL SET FORWARD command.

Format

```
SET DISABLE-PSIMAIL {TRUE | FALSE}
```

SET FORWARDER

Specifies the host that will forward mail messages to other hosts.

Format

```
SET FORWARDER [host_name]
```

Parameter

host_name

Specifies the name of the host to which mail is forwarded when attempts by the local system to send mail to a remote system fail because of a host name lookup failure.

If no host name is specified, no forwarder is used, and failed messages are tried repeatedly (based on the `RETRY-INTERVAL` setting) until they are returned to sender (based on the `RETURN-INTERVAL` setting).

SET FORWARD-LOCAL-MAIL

When TRUE, MultiNet forwards mail addressed to users on the local host to a central mail hub specified by the FORWARDER parameter.

Format

```
SET FORWARD-LOCAL-MAIL {TRUE | FALSE}
```

Description

To configure MultiNet to direct mail to a central mail hub, you must specify the IP address of the mail hub with the FORWARDER parameter, and define the scope of addresses that you want the mail hub to handle.

By default, when users on the same MultiNet host send mail to each other, MultiNet does not route the messages through the mail hub. When FORWARD-LOCAL-MAIL is TRUE, MultiNet forwards local mail to the mail hub.

To exclude a specific user from the local mail-forwarding system, add the following type of mail alias to MULTINET:SMTP_ALIASES:

```
username : *;
```

SET FORWARD-REMOTE-MAIL

When TRUE, MultiNet forwards mail addressed to non-local users on a central mail hub specified by the FORWARDER parameter.

Format

```
SET FORWARD-REMOTE-MAIL {TRUE | FALSE}
```

Description

To configure MultiNet to direct mail to a central mail hub, you must specify the IP address of the mail hub with the FORWARDER parameter, and define the scope of addresses that you want the mail hub to handle.

By default, when MultiNet users send mail to users on other hosts, MultiNet does not route the messages through the mail hub. When FORWARD-REMOTE-MAIL is TRUE, MultiNet forwards non-local mail to the mail hub.

By default MultiNet considers all remote hosts non-local. You can add hosts in other domains to the local-domain list with the ADD LOCAL-DOMAIN command.

SET HEADER-CONTROL

Specifies which RFC-822 message headers are included in messages delivered to local VMS MAIL users.

Format

`SET HEADER-CONTROL` *`header_type`*

Parameter

`header_type`
Either `NONE`, `MAJOR`, or `ALL`:

- `NONE` eliminates the RFC-822 message headers from locally delivered VMS MAIL messages.
- `MAJOR` (the default) includes all but Received and Return Path headers.
- `ALL` includes all headers.

SET HOST-ALIAS-FILE

Specifies a file from which MultiNet obtains a list of host aliases. A common use for SMTP host names is when your system is a member of a homogeneous VMScluster, and you want all mail from any cluster member to appear to be from the same host (for example, the cluster alias).

Unlike the `MAIL-CONFIG SET SMTP-HOST-NAMES` command which has a limit of 16 host names, `SET HOST-ALIAS-FILE` lets you specify a host alias file containing as many host aliases as needed.

> **Note:** The host name or alias you specify should be registered in the Domain Name System or in the host tables of any system to which you send mail; otherwise, the recipients of your mail will be unable to reply to it.

If this logical name is not defined, the SMTP software looks for the file `MULTINET:SMTP_HOST_ALIASES` by default.

Format

`SET HOST-ALIAS-FILE` *`file_spec`*

Parameter

`file_spec`
Specifies the file that contains a list of SMTP host names.

SET LOCAL-MAIL-FORWARDER

Forwards failed local mail to a specific host.

Format

```
SET LOCAL-MAIL-FORWARDER hostname
```

Parameter

hostname
Specifies the name of the host to which failed local mail is directed.

SET POSTMASTER

Identifies the user responsible for mail on the system.

Format

`SET POSTMASTER` *`[username]`*

Parameter

`username`

Specifies the name of the user who will receive messages addressed to Postmaster on the local host. If omitted, the user name POSTMASTER is used.

To assign multiple users as the postmaster, enter `POSTMASTER`, then create an alias for postmaster in the alias file. For example, to make both `username1` and `username2` postmasters, enter the following line in the alias file:

```
postmaster: username1, username2;
```

SET QUEUE-COUNT

Specifies the number of mail processing queues that should be created on a system.

Format

`SET QUEUE-COUNT` *`node_name`* [*`count`*]

Parameter

`node_name`

Specifies the VMScluster (SCS) node name of the node whose queue count you want to set, or specifies `DEFAULT` to set the default for all nodes not specifically set. In a non-cluster environment, only the `DEFAULT` setting is used.

`count`

Specifies the number of queues to create on the specified node. If a count is omitted, the queue-count setting for the specified node is removed.

SET REPLY-CONTROL

Specifies how Internet mail headers are mapped to the VMS MAIL "From" header.

Format

`SET REPLY-CONTROL` *`[hdr_types]`*

Parameter

`hdr_types`

Specifies a comma-delimited list of SMTP headers (`ENVELOPE-FROM`, `FROM`, or `REPLY-TO`) that are mapped to the VMS MAIL "From" header. The default is `ENVELOPE-FROM, FROM, REPLY-TO`

SET RESENT-HEADERS

When FALSE, the MultiNet SMTP symbiont omits the Resent-From, Resent-To, and Resent-Date headers that are usually included when a message is forwarded using a VMS MAIL forwarding address. The default is TRUE.

Use this option if mail user agents at your site cannot properly distinguish between normal "From" headers and "Resent-From" headers.

Format

```
SET RESENT-HEADERS {TRUE | FALSE}
```

SET RETRY-INTERVAL

Specifies the amount of time that elapses before another attempt is made to send a message after a failed attempt.

Format

```
SET RETRY-INTERVAL [interval]
```

Parameter

interval
Specifies the interval, in minutes (by default, 30 minutes).

SET RETURN-INTERVAL

Specifies the amount of time that a message can remain in the processing queue before it is returned to the sender.

Format

```
SET RETURN-INTERVAL [interval]
```

Parameter

interval

Specifies the interval, in hours; by default, 96 (four days). A message typically only remains in the processing queue if it cannot be sent over the network to a remote host. When such a message is returned to its sender, the returned message includes the reason why it could not be sent.

SET RFC2789

Enables the startup of the Mail Monitoring MIB (RFC 2789). Requires SNMP AGENTX to be set.

Format

```
SET RFC2789
```

SET SEND-BROADCAST-CLASS

Specifies the broadcast class to use to deliver immediate (SEND) messages.

Format

```
SET SEND-BROADCAST-CLASS [class_number]
```

Parameter

class_number

Specifies the class-number in a range from 1 to 16, corresponding to the VMS USER1 through USER16 broadcast classes (by default, 16).

SET SMTP-HOST-NAMES

Sets the host name from which all outgoing mail appears to be sent and the aliases for which this host accepts incoming mail.

A common use for `SMTP HOST NAME` is when your system is a member of a homogeneous VMScluster, and you want all mail from any cluster member to appear to be from the same host.

Format

`SET SMTP-HOST-NAMES` *`host_names`*

Parameter

`host_names`

Contains a comma-delimited list of host names. The first name in the list specifies the host name from which all outgoing mail appears to be sent. The remaining host names in the list specify the aliases for which this host accepts incoming mail.

> **Note:** The specified host name or alias should be registered in the Domain Name System or in the host tables of any system that you send mail to; otherwise, the recipients of your mail will be unable to reply to it.

SET START-QUEUE-MANAGER

Determines whether `START_SMTP.COM` starts the VMS queue manager if it is not already running. The default is `TRUE`.

Format

```
SET START-QUEUE-MANAGER {TRUE | FALSE}
```

SHOW

Displays the current configuration.

Format

```
SHOW
```

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. If the MULTINET_DISABLE_SPAWN logical is set, SPAWN does not work.

Format

SPAWN [*command*]

Parameter

command
Specifies a command to execute. If you omit command, a DCL command line subprocess is created.

Qualifiers

/INPUT=*file-spec*
Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES
/NOLOGICAL_NAMES
Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS
/NOSYMBOLS
Specifies that global and local names are not passed to the subprocess.

/WAIT
/NOWAIT
Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=*file-spec*

Specifies a file that retains the output of the command invoked with `SPAWN`. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after `SPAWN` or other qualifiers.

STATUS

Indicates whether the SMTP configuration has been modified.

Format

```
STATUS
```

USE

Reads in a MultiNet SMTP configuration file. After a `USE`, you can use the various configuration commands to modify the SMTP configuration. (Functionally equivalent to `GET`.)

Format

`USE` *`config_file`*

Parameter

`config_file`
Specifies the name of the SMTP configuration file to read in.

VERSION

Displays the MAIL-CONFIG version and release information.

Format

```
VERSION
```

WRITE

Writes the current MultiNet SMTP configuration to SMTP configuration files. (Functionally equivalent to SAVE.)

Format

WRITE *config_file*

Parameter

config_file
Specifies the name of the file to which to write the current MultiNet SMTP configuration. By default, the configuration is saved to the same file from which it was read.

4. NET-CONFIG Command Reference

This chapter describes the commands you can run from the NET-CONFIG command line. NET-CONFIG lets you examine, modify, and save configuration files for MultiNet network interfaces.

To invoke NET-CONFIG:

```
$ MULTINET CONFIGURE /INTERFACES
```

At any NET-CONFIG prompt, you can list the available commands by typing **?**. Online help for each NET-CONFIG command is available through the NET-CONFIG HELP command.

Of the SET commands, the SET LOAD-EXOS-DRIVER and SET WINS-COMPATIBILITY commands require that you reboot the system after you use the command; all other SET commands can be executed without rebooting the system.

For details on configuring network interfaces, refer to the *MultiNet Installation and Administrator's Guide*.

NET-CONFIG Command Summary

The below table lists the commands you can run from the NET-CONFIG prompt.

NET-CONFIG Command	Description
ADD	Adds a device to the network configuration.
ATTACH	Attaches a terminal to a process.
CHECK	Performs a check on the current configuration.
CLEAR	Deletes all devices from the network configuration.

CREATE	Create a Six-To-Four interface to carry IPv6 traffic over IPv4.
DELETE	Deletes a single device from the network configuration.
DISABLE	Disables a device and deletes it from the configuration.
ENABLE	Enables a device.
ERASE	Deletes all devices from the network configuration.
EXIT	Exits NET-CONFIG and saves the configuration if it changed.
GET	Reads in a nonstandard configuration file.
HELP	Displays command help information.
MODIFY	Changes a device configuration.
PUSH	Accesses the DCL command interpreter.
QUIT	Exits NET-CONFIG and prompts to save the configuration if it changed.
SAVE	Writes configuration parameters to a file.
SET ANONYMOUS-FTP-ACCESS	Restricts anonymous FTP access.
SET ANONYMOUS-FTP-DIRECTORY	Restricts anonymous FTP directory access.
SET CLUSTER-SERVICE-ADDRESS	Identifies the IP address to use when responding to address queries from hosts specified by CLUSTER-SERVICE-NAMES.
SET CLUSTER-SERVICE-NAMES	Lists the host names to use for load balancing.

SET DEFAULT-RMT-TAPE-DEVICE	Defines the default RMT server tape device.
SET DEFAULT-ROUTE	Defines the default IP route.
SET DOMAIN-NAMESERVERS	Configures the DNS domain resolver.
SET HOST-NAME	Defines the local host name and domain.
SET IP-CLUSTER-ALIASES	Specifies a list of cluster-wide Internet addresses to which this node responds.
SET LOAD-EXOS-DRIVER	Enables automatic loading of the Novell EXOS-compatible $QIO driver when MultiNet starts.
SET LOAD-PWIP-DRIVER	Enables automatic loading of the version 5 PATHWORKS server interface (PWIP) driver when MultiNet starts
SET LOCAL-DOMAIN	Overrides the default local domain derived from previous use of the SET HOST-NAME command.
SET LPD-DEFAULT-USERNAME	Specifies a default user name for print jobs received from a remote machine via the LPD protocol.
SET NAMESERVER-RETRANSMISSION	Specifies a time between and the number of name server requests that are made before the system stops sending name server requests to a nonresponding server.
SET SPOOL-DIRECTORY	Sets the default spool directory name.
SET TFTP-DIRECTORY	Sets the default directory for the TFTP server.
SET TIMEZONE	Sets the local time zone.
SET TIMEZONE-RULES	Specifies the time zone rules file name.
SET WHOIS-DEFAULT-SERVER	Sets the default WHOIS server.

SET WINS-COMPATIBILITY	Defines the logical names used with the WIN/TCP and Pathway products from The Wollongong Group.
SHOW	Displays the current configuration or device names.
SPAWN	Executes a DCL command.
STATUS	Displays the device count in the configuration.
USE	Reads in a nonstandard configuration file.
VERSION	Displays the NET-CONFIG and VMS versions.
WRITE	Writes the current configuration.

ADD

Adds a device to the network configuration.

FORMAT

`ADD interface`

PARAMETERS

interface
Specifies the name of the device interface to add.

Refer to the below table for a list of supported network interfaces and the corresponding interface parameters for which you are prompted. For descriptions of interface parameters, refer to the following table.

<table>
<tr><th>Type</th><th>Description</th></tr>
<tr><td>nsip</td><td>Interface name: nsip0, nsip1, ... nsip9
Device type: IPX-over-IP tunnel

<table><tr><th>Parameter Prompt</th><th>Example Value</th></tr><tr><td>IP address of remote system:</td><td>10.1.10.70</td></tr></table>
The nsip interface provides access to IPX-over-IP for connecting to Novell's IP tunnel feature.</td></tr>
<tr><td>pd</td><td>Interface name: pd0
Device type: Secondary Ethernet Address

<table><tr><th>Parameter Prompt</th><th>Example Value</th></tr><tr><td>TCP/IP</td><td>10.1.128.21</td></tr></table></td></tr>
</table>

	Hardware-Device	se0
	IP SubNet Mask	None
	Non-Standard IP Broadcast Address	None
	pd0 (Secondary Ethernet Address)	Csr=None, Flags=%X0
	IPv6 on this interface [DISABLED]	ENABLE
	IPv6 global address [NONE]	fe80:0db8:ac10:fe01
	IPv6 mask length	16

Careless assignment of a secondary address can cause network problems. In general, you should assign pseudo devices (pd) addresses on the same network or subnet as the se device to which the pd device is linked.

If the pd interface is not in the same IP network as its associated se interface, some TCP/IP packages retransmit broadcast packets for the other IP network back to the network segment from which they were transmitted. This can cause network storms.

> **Note:** GateD will listen to traffic on pseudo devices in MultiNet. Some services listen to traffic on se interfaces only and ignore traffic on pd interfaces. One such service is the RIP listener in GATED.

ppp

Interface name: ppp0, ...
Device type: Any supported PPP terminal interface

Parameter Prompt	Example Value
VMS Device:	TTA0:
Baud Rate:	19200
PPP ACCM Mask:	0

	PPP Authentication Method: None PPP Protocol Compression: OFF PPP Address and Control Field Compression: OFF PPP Retry Count: 0 PPP Idle Timeout: 0 PPP MRU Size: 0 PPP ICMP: ENABLED PPP TCP Compression: OFF PPP Termination Retry Count: 0 PPP Timeout: 0 IP Address: 0.0.0.0 Point-to-Point Device IP Destination Address: 0.0.0.0 IP Subnet Mask: 255.0.0.0
rp	**Interface name:** rp0, rp1, rp2, ... **Device type:** Raw packet **Parameter Prompt** **Example Value** IP Address: 10.41.228.70 IP SubNet Mask: 255.255.255.0 The rp interface allows IP packets that are normally destined for transmission to be directed instead to a user process by way of an AF_RAWPACKET socket.
se	**Interface name:** se0, se1, se2, ... **Device type:** Any HP VMS Ethernet, FDDI, or Token-Ring Alpha controller **Parameter Prompt** **Example Value** VMS Device: XEA0 Link Level Encapsulation Mode: ETHERNET BSD Trailer Encapsulation: DISABLED IP Address: 10.41.228.70 IP SubNet Mask: 255.255.255.0 Non-Standard IP Broadcast Address: 10.41.228.71

	DHCP CLIENT [DISABLED]: DISABLED Jumbo Frames [DISABLED]: ENABLED IPv6 on this interface [DISABLED]: ENABLED IPv6 global address [NONE]: 3FFE:1200:3006::C673:8EBE IPv6 mask length: 48 The se interface uses any Ethernet controller to provide access to a 10/100/1000 Mb/s Ethernet network, any FDDI controller to provide access to a 100 Mb/s FDDI network, and a Token-Ring controller to provide access to a 4 Mb/s or 16 Mb/s Token-Ring. The se interface uses the standard VMS Ethernet driver to allow MultiNet to share the Ethernet devices with other protocols such as LAT, LAVC, and DECnet.
sl	**Interface name:** sl0, sl1, sl2, ... **Device type:** Any VMS-supported terminal interface **Parameter Prompt** — **Example Value** VMS Device: TTA0 Baud Rate: 19200 Header Compression Mode: DISABLED IP Address: 10.41.228.70 Point-To-Point Device IP Destination Address: 10.41.228.71 IP SubNet Mask: 255.255.255.0 The MultiNet software supports SLIP with Van Jacobson's header compression algorithm, reducing the size of the headers and increases the bandwidth available to data. Header compression mode is determined by what both sides can support.

NET-CONFIG Prompts	**Function**

ACCM Mask	Asynchronous Control Character Map Mask. A 32-bit mask that indicates the set of ASCII control characters to be mapped into two-character sequences for transparent transmission over the line. The default is `%x00000000`.
Address and Control Field Compression (ACFC)	When `ON`, PPP eliminates the address and control fields when they are identical over a series of frames. The default is `OFF`.
Authentication Method	Determines the type of remote peer authentication required to allow network-layer protocol packets to be exchanged. Accepted values are `PAP` (Password Authentication Protocol) and `NONE` (the default).
Baud Rate	Indicates the baud rate of transmission. Possible values are: `110`, `300`, `1200`, `2400`, `4800`, `9600`, `19200`, and `UNSPECIFIED`.
BSD Trailer Encapsulation	For 10Mb/sec Ethernet controllers only. Can be used to enable Berkeley Trailer encapsulation of IP packets on those Ethernets. There are two possible settings: `NEGOTIATED` or `DISABLED`. The default, `DISABLED`, prevents the use of trailer encapsulation.
Hardware Device	The name of the real Ethernet device; for example, `se0`.
Header Compression Mode	For PPP and SLIP devices, indicates whether to use Van Jacobson's TCP header compression algorithm. The parameter has three possible settings: `DISABLED`, `ENABLED`, or `NEGOTIATED`. `DISABLED` indicates that headers should never be compressed. `ENABLED` indicates that headers should always be compressed. The default is `DISABLED`. `NEGOTIATED` indicates that headers should not be compressed until a compressed header is received from the other side. At least one side of a link must be `ENABLED` for compression to be used; that is, both sides of a link cannot be set to `NEGOTIATED` for compression to be used.
ICMP	When `ENABLED` (the default), PPP allows ICMP packets over the PPP connection. Administrators may want to disable ICMP packets if they are concerned with "service attacks" from dial-up connections.

Idle Timeout	Determines how long (in seconds) the connection must be idle before PPP attempts to close it with "Terminate-Request" packets. The default is `0`.
IP Address	Indicates the Internet address, in dotted decimal notation, associated with the interface. For PPP interfaces, you can specify `0.0.0.0` to indicate that the local IP address will be specified by the remote peer when a serial connection is established. The default is `0.0.0.0`.
IP Address of remote system	Indicates the Internet address associated with the remote system.
IP Broadcast Address	For devices that support broadcasts, allows the setting of a non-standard IP broadcast address. The parameter defaults to an address whose host portion is all ones.
IP Subnet Mask	The subnet mask of the local interface in dotted decimal format. The default depends on the local interface IP address. For example, a class A address results in a default subnet mask of 255.0.0.0.
IPv6 global address	Indicates the global unique address associated with this interface. The interface may also have a link-local address which will be automatically generated when the interface is started
IPv6 mask length	The length of the mask for the IPv6 address.
Link Level Encapsulation Mode	For Ethernet devices, indicates whether to use the standard Ethernet encapsulation of IP datagrams, or extended 802.2 encapsulation as specified in RFC 1042. Enter either `ETHERNET` or `802.2` as the value.
Maximum Receive Unit (MRU) Size	Determines the maximum number of 8-bit bytes for the PPP Information field, including Padding, but not including the Protocol field. Because opposite ends of a PPP connection may have different MRU values, PPP negotiates a suitable MRU for both systems. The default is `500`.
Point-To-Point Device IP Destination Address	For point-to-point interfaces, indicates the IP address of the peer system on the other side of the connection. The default is `0.0.0.0`.

Protocol Compression	When ON, PPP negotiates with the peer to use one byte instead of two for the Protocol fields to improve transmission efficiency on low-speed lines. Default is OFF.
Retry Count	Determines the number of attempts PPP makes to configure a connection with "Configure-Request" packets. The default is 0.
Termination Retry Count	Determines the number of attempts PPP makes to terminate a connection with "Terminate-Request" packets. The default is 0.
Timeout	Determines the time (in seconds) between successive Configure-Request or Terminate-Request packets. The default is 0.
VMS Device	For devices that use a VMS device driver to interface with the hardware, indicates the name of the VMS device that MultiNet will use. This parameter is used with HP Ethernet, PPP, and SLIP interfaces.

EXAMPLE

```
$ MULTINET CONFIGURE /NETWORK
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG> ADD PPP2
[Adding new configuration entry for device "PPP2"]
VMS Device: [TTA0]RETURN
Baud Rate: [UNSPECIFIED] 9600
PPP ACCM Mask:  [%x0] RETURN
PPP Authentication Method: [NONE] PAP
PPP Protocol Compression: [OFF] ? ON
PPP Address and Control Field Compression: [OFF] ON
PPP Retry Count: [0] 10
PPP Idle Timeout: [0] 10
PPP MRU Size: [0] RETURN
PPP ICMP:  [ENABLED]  RETURN
PPP TCP Compression:  [OFF] RETURN
PPP Termination Retry Count: [0] 10
PPP Timeout:  [0]  10
IP Address: [NONE]  10.45.67.27
Point-to-Point Device IP Destination Address: [NONE] RETURN
```

```
IP Subnet Mask: [NONE] RETURN
NET-CONFIG> EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
```

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the SPAWN SHOW PROCESS /SUBPROCESSES command to list the names of subprocesses. Use the DCL LOGOUT command to return to the original process. If the MULTINET_DISABLE_SPAWN logical is enabled, ATTACH does not work.

FORMAT

ATTACH *process-name*

PARAMETER

process_name
Specifies the name of a process to which you want our terminal attached. (Not all subprocesses can be attached; some testing may be required.)

EXAMPLE

```
$ MULTINET CONFIGURE  /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SPAWN
$ MM
MM>SPAWN SHOW PROCESS /SUB
...
There are 3 processes in this job:
_TWA42:
  PROC_1
   PROC_2 (*)
MM>ATTACH _TWA42:
NET-CONFIG>ATTACH PROC_1
MM>QUIT
$ LOGOUT
NET-CONFIG>
```

This example shows the use and exit of attached subprocesses.

1. The first command uses SPAWN to create a subprocess. MM is invoked from the DCL command line. Next, the SPAWN SHOW PROCESS/SUB command is used to list all the

subprocess names. The display shows that three subprocesses are active. (`Process _TWA42:` is `NET-CONFIG`, `PROC_1` is `MM`, and `PROC_2` is the `SPAWN SHOW PROCESS/SUB` command.)

2. In the next command, the `MM ATTACH` command returns control to the `NET-CONFIG` process. From this utility, `ATTACH` returns control to `MM`. To exit, `QUIT` is invoked from `MM`, and `LOGOUT` is invoked at the original spawned DCL command line; control returns back to NET-CONFIG. (If `SPAWN SHOW PROCESS/SUB` had been entered, only this command and the configuration processes would be active.)

CHECK

Checks the configuration parameters to ensure all required information is provided. If a problem is found, an error message displays; otherwise, if the check is successful, no information displays.

FORMAT

CHECK

DESCRIPTION

CHECK provides a check of the current configuration. The following messages can display:

Message	Description
ERROR: *device_name* can't $ASSIGN to FFI device: *value*	The specified X device does not exist, or a lack of privileges made opening the device impossible.
ERROR: *device_name* can't $ASSIGN to SLIP device: *value*	The specified device does not exist, or a lack of privileges made opening the device impossible.
ERROR: *device_name*: Default route cannot be the local machine: *default_route*	The default route is the gateway that connects this system to the Internet; it cannot be the local system.
ERROR: *device_name*: Default route route must be directly connected	The IP address specified in the SET DEFAULT-ROUTE command must be for a system connected to the Internet.
ERROR: *device_name*: Illegal value for IP Address: *address*	The specified IP address either has an octet above 255, or contains an illegal character such as a space, an alphabetical character, or a control character. Use the MODIFY command to specify a new value.
ERROR: *device_name*: Illegal value for IP Broadcast Address: *address*	The specified IP broadcast address either has an octet above 255, or contains an illegal character such as a space, an

	alphabetical character, or a control character. Use the MODIFY command to specify a new value.
ERROR: *device_name*: Illegal value for IP SubNet Mask: *mask*	The specified IP subnet mask value either has an octet above 255, or contains an illegal character such as a space, an alphabetical character, or a control character. Use the MODIFY command to specify a new value.
ERROR: *device_name* is slave to nonexistent device *hardware_device*	The MultiNet device points to a device that does not exist for the local system.
ERROR: *device_name*'s DECnet peer must be DECnet node name, not address	DECnet node names are alphanumeric strings of six characters or less; specify the correct value.
ERROR: *device_name*'s point-to-point destination is unspecified	A destination address was not specified for a point-to-point device, such as a SLIP line.
ERROR: *device_name*'s PSI peer must be PSI DTE, not node.	The "IP Over PSI Peer Host's DTE" prompt is requesting the DTE destination name; specify the correct value.
ERROR: *device_name'*s PSI local must be PSI DTE, not node.	The "IP Over PSI Local Host's DTE" prompt is requesting the DTE source name; specify the correct value.
ERROR: The host name is not a domain-style host name and domain name service is enabled.	The domain name did not contain dot separators.
ERROR: There is no Host Name specified.	A host name value was not entered either in the configuration file you are creating or in a read-in configuration file.
WARNING: *device_name* has no FFI device specified	The added or modified device requires that an FFI device be specified.
WARNING: *device_name* has no protocol addresses specified	The added or modified device requires that protocol addresses be specified.

WARNING: *device_name*'s DECnet peer is unspecified (link will not come up)	Both ends of a point-to-point DECnet link must be specified.
WARNING: *device_name*'s hardware device is unspecified	A device name must be entered for this device.
WARNING: *device_name*'s PSI peer is unspecified (link will not come up)	Both ends of a point-to-point PSI link must be specified.
WARNING: *device_name*'s PSI local DTE is unspecified (link will not come up)	Both ends of a point-to-point PSI link must be specified.

EXAMPLE

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>ADD PPP1
[Adding new configuration entry for device "ppp1"]
VMS Device: [TTA0]  TTA1
Baud Rate:  [UNSPECIFIED] 9600
PPP ACCM Mask:  [%x0] RETURN
PPP Authentication Method: [NONE] PAP
PPP Protocol Compression: [OFF] ON
PPP Address and Control Field Compression: [OFF] ON
PPP Retry Count: [0] 10
PPP Idle Timeout: [0] 10
PPP MRU Size: [0] 10
PPP ICMP:  [ENABLED] RETURN
PPP TCP Compression: [OFF] RETURN
PPP Termination Retry Count: [0] RETURN
PPP Timeout: [0] RETURN
IP Address: [NONE] RETURN
Point-to-Point Device IP Destination Address: [NONE] 10.45.24.13
IP Subnet Mask: [NONE] RETURN
[ppp] (Point-to-Point Protocol): Csr=NONE, Flags=%X0]
NET-CONFIG>CHECK
WARNING: ppp1 has no protocol addresses specified
```

```
NET-CONFIG>EXIT
$
```

CLEAR

Deletes all devices from the current network configuration.

FORMAT

```
CLEAR
```

EXAMPLE

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>CLEAR
```

CREATE

Create a Six-to-Four IPv6 interface.

FORMAT

```
CREATE
```

EXAMPLE

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>CREATE SIX-TO-FOUR
IPv4 address to use [none]: 192.43.1.1
Mask Length [48]:
```

Note that RFC 1597 private address (10.*.*.*, 172.16.*.*, 192.168.*.*) and RFC 3927 IPv4 link-local addresses (169.254.*.*) are not allowed for the IPv4 address.

DELETE

Removes a single device from the network configuration.

FORMAT

DELETE *interface*

PARAMETER

interface
Specifies the name of the interface for the device being removed.

EXAMPLE

This example deletes a SLIP device, SL0, from the current configuration.

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>DELETE SL0
```

DISABLE

Disables a device. A disabled device is deleted from the configuration and is not configured at network boot.

FORMAT

DISABLE *interface*

PARAMETER

interface
Specifies the name of the interface for the device being disabled.

EXAMPLE

This example disables a SLIP device, SL0.

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>DISABLE SL0
```

ENABLE

Enables a device.

FORMAT

ENABLE *interface*

PARAMETER

interface
Specifies the name of the interface for the device being enabled.

EXAMPLE

This example enables a SLIP device, SL0.

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>ENABLE SL0
```

ERASE

Erases all configured devices from the current configuration. (Functionally equivalent to `CLEAR`.)

EXIT

Saves the current configuration, if it has been modified, and exits the configuration program. (Use the `STATUS` command to display whether the configuration was modified.)

FORMAT

`EXIT`

EXAMPLES

When the configuration has not been changed, a message displays indicating that the configuration file is not updated.

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>EXIT
[Configuration not modified, so no update needed]
$
```

When the configuration has changed, a message displays indicating that the configuration file has been updated.

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>CLEAR
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$
```

GET

Reads in a nonstandard configuration file.

FORMAT

`GET` *`filename`*

PARAMETER

`filename`
Specifies the name of the configuration file to read; by default, `NETWORK_DEVICES.CONFIGURATION` in the current working directory.

EXAMPLE

This example retrieves the configuration file `MULTINET:TEST.CONFIGURATION` into the `NET-CONFIG` workspace.

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>GET MULTINET:TEST.CONFIGURATION
```

HELP

Displays help information by listing either the command names or information about specific commands. You can also display help information by adding a question mark to other NET-CONFIG commands.

FORMAT

HELP *command*

PARAMETER

command

Specifies the command for which to list help information. Use a question mark (?) to list all command names. Entering a question mark on the NET-CONFIG command line has the same effect as using the HELP ? command.

EXAMPLE

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>HELP ?
ADD     ATTACH   CHECK   CLEAR    DELETE  DISABLE  ENABLE   ERASE
EXIT    GET      HELP    MODIFY   PUSH    QUIT     SAVE     SET
SHOW    SPAWN    STATUS  USE      ERSION  WRITE
NET-CONFIG>
```

MODIFY

Modifies an existing device configuration. If the device has not already been configured, an error message displays.

Use MODIFY to modify configuration parameters for an existing network interface. MODIFY produces the same sequence of prompts for network interface parameter values as the ADD command, but uses the current settings for default values.

FORMAT

MODIFY *interface*

PARAMETER

interface
Specifies the name of the interface for the device being modified.

EXAMPLE

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>MODIFY SE0
[Modifying configuration entry for device "se0"]
VMS Device: [EZA0] RETURN
Link Level Encapsulation Mode: [ETHERNET]  ? EXTENDED-802.2
BSD Trailer Encapsulation: [DISABLED] RETURN
IP Address: [10.45.678.90] RETURN
IP Subnet Mask: [255.255.255.0] RETURN
Non-Standard IP Broadcast Address: [NONE] RETURN
se0 (Shared VMS Ethernet/FDDI): Csr=NONE, flags=%x0)
NET-CONFIG>
```

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command.

If the MULTINET_DISABLE_SPAWN logical is set, PUSH does not work.

FORMAT

PUSH

QUIT

Prompts you to save the current configuration if it was modified, and then exits.

FORMAT

```
QUIT
```

EXAMPLE

```
$ MULTINET CONFIGURE /NETWORK_DEVICES
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>DISABLE SL0
NET-CONFIG>QUIT
Configuration modified, do you want to save it?  [NO] RETURN
```

SAVE

Saves the configuration parameters.

FORMAT

`SAVE [/STARTUP] [`*`filename`*`]`

PARAMETER

filename

Specifies the file name for the configuration file or the startup command procedure. The default for filename when saving the configuration data is the file from which the configuration was read. The default for filename when saving the MultiNet startup command procedure is `MULTINET:START_MULTINET.COM`.

QUALIFIER

/STARTUP

Specifies that `NET-CONFIG` saves the MultiNet startup commands in filename.

SET ANONYMOUS-FTP-ACCESS

Defines file access rights for the ANONYMOUS FTP user to read, write, delete, and list files in addition to spawning a new process.

If your configuration permits users to make anonymous file transfers via the VMS ANONYMOUS account, use the ANONYMOUS-FTP-ACCESS parameter to restrict file access for that account.

You can change the value of ANONYMOUS-FTP-ACCESS without rebooting by also defining or redefining the system-wide MULTINET_ANONYMOUS_FTP_ACCESS logical name.

FORMAT

SET ANONYMOUS-FTP-ACCESS (*access* [*access*])

PARAMETERS

access

NOLIST - Disables the listing of files.

NOWRITE - Disables the storing of files.

NOSPAWN - Disables the SPAWN command.

NOREAD - Disables reading of files.

NODELETE - Disables the deleting/renaming of files.

EXAMPLE

The following example sets the ANONYMOUS-FTP-ACCESS parameter to NOWRITE, and then sets the system-wide MULTINET_ANONYMOUS_FTP_ACCESS logical name. By setting this logical name, ANONYMOUS_FTP_ACCESS can be set without rebooting the system. Setting the NOWRITE parameter will prevent anonymous users from storing files.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility V5.6(104)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET ANONYMOUS-FTP-ACCESS NOWRITE
NET-CONFIG>EXIT
```

```
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next MultiNet reload]
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_ANONYMOUS_FTP_ACCESS "NOWRITE"
```

SET ANONYMOUS-FTP-DIRECTORY

Restricts directory tree access of an anonymous FTP account.

If your configuration permits users to make anonymous file transfers via the VMS ANONYMOUS account, use the ANONYMOUS-FTP-DIRECTORY parameter to restrict access for that account to a specific directory tree.

You can change the value of ANONYMOUS-FTP-DIRECTORY without rebooting by also defining or redefining the system-wide MULTINET_ANONYMOUS_FTP_DIRECTORY logical name.

FORMAT

SET ANONYMOUS-FTP-DIRECTORY *[directory]*

PARAMETER

directory
Specifies the name of the directory tree to which the account is restricted. To cancel the restriction, enter the command without a directory name.

EXAMPLE

This example sets the ANONYMOUS-FTP-DIRECTORY parameter to USERS:[ANONYMOUS], then sets the system-wide MULTINET_ANONYMOUS_FTP_DIRECTORY logical name. By setting the system-wide logical name, ANONYMOUS-FTP-DIRECTORY can be set without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET ANONYMOUS-FTP-DIRECTORY USERS:[ANONYMOUS]
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE -
_$ MULTINET_ANONYMOUS_FTP_DIRECTORY "USERS:[ANONYMOUS]"
$
```

SET CLUSTER-SERVICE-ADDRESS

Identifies the IP address to be used when responding to address queries for hosts specified by CLUSTER-SERVICE-NAMES.

If your host is configured with more than one interface, the DOMAINNAME service is enabled, and you are using cluster load balancing, use the CLUSTER-SERVICE-ADDRESS parameter to identify the IP address that this host will return in response to address queries for any host specified by CLUSTER-SERVICE-NAMES.

If you do not specify an address, an interface is chosen at random.

FORMAT

SET CLUSTER-SERVICE-ADDRESS *[ip_address]*

PARAMETER

ip_address
Specifies the IP address returned when responding to address queries from hosts identified by CLUSTER-SERVICE-NAMES.

EXAMPLE

This example sets the CLUSTER-SERVICE-ADDRESS parameter to 10.43.154.10 and then sets the system-wide MULTINET_CLUSTER_SERVICE_ADDRESS logical name. By setting the system-wide logical name, the CLUSTER-SERVICE-ADDRESS can be set without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET CLUSTER-SERVICE-ADDRESS 10.43.154.10
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE -
_$ MULTINET_CLUSTER_SERVICE_ADDRESS "10.43.154.10"
$ @MULTINET:START_SERVER
$
```

SET CLUSTER-SERVICE-NAMES

Lists host names to which cluster load balancing applies, and allows you to configure hosts so that TCP-based connections are directed to the host with the lightest load at the time of the request. The DOMAINNAME service must be enabled on each host. To establish this environment:

1. Configure the primary name server for the parent domain so that it delegates authority for each host specified with the CLUSTER-SERVICE-NAMES parameter to this host.
2. Use the CLUSTER-SERVICE-NAMES parameter to identify the hosts for which this host will accept connections.

For example, if authority for CLUSTER.EXAMPLE.COM is delegated to NODE1, NODE2, and NODE3 on the primary name servers, each node must include CLUSTER.EXAMPLE.COM in its CLUSTER-SERVICE-NAMES list. When a TCP service request is made to CLUSTER.EXAMPLE.COM, DNS returns the IP addresses of the nodes NODE1, NODE2, or NODE3, ordered by load rating.

If the host is configured with more than one interface, specify the desired interface with the CLUSTER-SERVICE-ADDRESS parameter.

FORMAT

SET CLUSTER-SERVICE-NAMES *hostlist*

PARAMETER

hostlist
Specifies a comma-separated list of fully qualified host names for which cluster load balancing is implemented.

EXAMPLE

This example sets the CLUSTER-SERVICE-NAMES parameter to CLUSTER.EXAMPLE.COM, then sets the system-wide MULTINET_CLUSTER_SERVICE_NAMES logical. By setting the system-wide logical, CLUSTER-SERVICE-NAMES can be set without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
```

```
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET CLUSTER-SERVICE-NAMES CLUSTER.EXAMPLE.COM
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE/SYSTEM/EXECUTIVE -
_$ MULTINET_CLUSTER_SERVICE_NAMES "CLUSTER.EXAMPLE.COM"
$ @MULTINET:START_SERVER
$
```

SET DEFAULT-RMT-TAPE-DEVICE

Defines the default VMS tape drive that the RMT server uses. (This drive is the VMS equivalent of the UNIX `/dev/rmt0` interface, and is the drive used by the `rdump` and `rrestore` programs.) If the parameter is not set, the server searches for interface types in the following order and uses the first interface it finds: `MU`, `MK`, `MF`, `MT`, and `MS`.

You can change the value of `DEFAULT-RMT-TAPE-DEVICE` without rebooting by also defining or redefining the system-wide logical name `MULTINET_RMT_TAPE_DEVICE`.

FORMAT

`SET DEFAULT-RMT-TAPE-DEVICE` ***drive_name***

PARAMETER

drive_name
Specifies the name of the default drive.

EXAMPLE

This example sets `DEFAULT-RMT-TAPE-DEVICE` to `HSC001$MUA1:`, then sets the `MULTINET_RMT_TAPE_DEVICE` system-wide logical name. By setting this logical name, the tape device can be given a new value without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET DEFAULT-RMT-TAPE-DEVICE HSC001$MUA1:
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_RMT_TAPE_DEVICE "HSC001$MUA1:"
$
```

SET DEFAULT-ROUTE

Defines the default IP route-that is, the IP address of the gateway that MultiNet uses for all packets sent from the local network.

You can specify more complex routing information by either:

- Creating a `MULTINET:LOCAL_ROUTES.COM` file that contains a list of `MULTINET SET/ROUTE` commands that set up the routing tables on a per-network basis.
- Running a dynamic-routing protocol such as RIP, HELLO, EGP, or BGP by configuring the GATED service.

> **Note:** Any GATED configuration overrides a default IP route set with `NET-CONFIG`. Once started, GATED takes complete control of your routing.

You can change the default route without rebooting by using the `MULTINET SET/ROUTE` command to delete the old default route and add the new default route.

FORMAT

`SET DEFAULT-ROUTE` *`ip_address`*

PARAMETER

`ip_address`
Specifies the IP address of the gateway.

EXAMPLE

This example sets the gateway `DEFAULT-ROUTE` to 10.41.228.100, then performs the commands that change `DEFAULT-ROUTE` without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET DEFAULT-ROUTE 10.41.228.100
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ MULTINET SET/ROUTE -
_$ /DELETE=(DESTINATION=0.0.0.0, GATEWAY=10.41.228.100)
$ MULTINET SET/ROUTE -
_$ /ADD=(DESTINATION=0.0.0.0, GATEWAY=10.41.228.100)
```

SET DOMAIN-NAMESERVERS

Configures the DNS domain resolver, the portion of MultiNet called when a host name must be translated into an Internet address. If you are using DNS, use this parameter to define the Internet addresses of the Domain Name Servers to which the resolver will send requests.

The usual setting for the parameter is the loopback address (127.0.0.1), which directs the resolver to send inquiries to the server on the local system.

To disable DNS and use the host tables instead of the service, enter the `SET DOMAIN-NAMESERVERS` command with no options.

You can change the value of `DOMAIN-NAMESERVERS` without rebooting by also defining or redefining the system-wide logical name `MULTINET_NAMESERVERS` and restarting the `MULTINET_SERVER` (`@MULTINET:START_SERVER`) and `SMTP_SYMBIONT` (`@MULTINET:START_SMTP`) processes.

FORMAT

`SET DOMAIN-NAMESERVERS [`*`address`* `[,`*`address`* `...]]`

PARAMETER

address

Specifies the IP address of a name server. When you list multiple addresses, the resolver successively attempts to send a packet to the addresses, in the listed order, until it receives a response.

RESTRICTION

The resolver nameserver list can only include three IP addresses. This list is controlled by the `MULTINET_NAMESERVERS` logical (which is controlled by the `DOMAIN-NAMESERVERS` parameter).

EXAMPLES

This example tells the resolver to try only the local nameserver. The subsequent DCL commands change domain nameservers without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
```

```
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET DOMAIN-NAMESERVERS 127.0.0.1
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXEC MULTINET_NAMESERVERS "127.0.0.1"
$ DEFINE /SYSTEM /EXEC UCX$BIND_SERVER000 "127.0.0.1"
$ DEFINE /SYSTEM /EXEC UCX$BIND_DOMAIN "EXAMPLE.COM"
$ DEFINE /SYSTEM /EXEC TCPIP$BIND_SERVER000 "127.0.0.1"
$ DEFINE /SYSTEM /EXEC TCPIP$BIND_DOMAIN "EXAMPLE.COM"
$ @MULTINET:START_SERVER
$ @MULTINET:START_SMTP
$
```

This example configures the resolver to try the local name server and, if it fails, to try 10.0.0.1.

```
NET-CONFIG>SET DOMAIN-NAMESERVERS 127.0.0.1,10.0.0.1
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$
$ DEFINE /SYSTEM /EXEC MULTINET_NAMESERVERS "127.0.0.1", "10.0.0.1"
$ DEFINE /SYSTEM /EXEC UCX$BIND_SERVER000 "127.0.0.1"
$ DEFINE /SYSTEM /EXEC UCX$BIND_SERVER001 "10.0.0.1"
$ DEFINE /SYSTEM /EXEC UCX$BIND_DOMAIN "EXAMPLE.COM"
$ DEFINE /SYSTEM /EXEC TCPIP$BIND_SERVER000 "127.0.0.1"
$ DEFINE /SYSTEM /EXEC TCPIP$BIND_SERVER001 "10.0.0.1"
$ DEFINE /SYSTEM /EXEC TCPIP$BIND_DOMAIN "EXAMPLE.COM"
$ @MULTINET:START_SERVER
$ @MULTINET:START_SMTP
$
```

The `SET DOMAIN-NAMESERVERS` disables the local DNS domain server; only host tables are used to translate names and addresses.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET DOMAIN-NAMESERVERS
NET-CONFIG>EXIT
$
$ DEASSIGN /SYSTEM /EXEC MULTINET_NAMESERVER
```

```
$ DEASSIGN /SYSTEM /EXEC UCX$BIND_SERVER000
$ DEASSIGN /SYSTEM /EXEC UCX$BIND_SERVER001
$ DEASSIGN /SYSTEM /EXEC UCX$BIND_DOMAIN
$ DEASSIGN /SYSTEM /EXEC TCPIP$BIND_SERVER000
$ DEASSIGN /SYSTEM /EXEC TCPIP$BIND_SERVER001
$ DEASSIGN /SYSTEM /EXEC TCPIP$BIND_DOMAIN
$ @MULTINET:START_SERVER
$ @MULTINET:START_SMTP
$
```

SET HOST-NAME

Specifies the local computer's host name and defines the default local domain. If your configuration includes Domain Name Service (DNS), you must specify the host name in dotted format, for example, HAMLET.CALTECH.EDU or JETSON.SPROCKETS.COM.

The default local domain is derived from the HOST-NAME parameter. For example, with a host name of HAMLET.CALTECH.EDU, the default local domain is .CALTECH.EDU. MultiNet uses the default local domain to complete abbreviated host names. For example, with the default of CALTECH.EDU, if you entered the command:

```
$ TELNET ROMEO
```

TELNET would attempt to connect to a ROMEO.CALTECH.EDU host.

You can change the value of HOST-NAME without rebooting by also defining or redefining the system-wide logical name MULTINET_HOST_NAME and restarting the MULTINET_SERVER (@MULTINET:START_SERVER) and SMTP_SYMBIONT (@MULTINET_START_STMP) processes.

FORMAT

SET HOST-NAME *host*

PARAMETER

host
Specifies the name of your host.

EXAMPLE

This example sets the HOST-NAME to HAMLET.CALTECH.EDU, then issues the commands that change the parameter without rebooting the system.

```
$ MULTNET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET HOST-NAME HAMLET.CALTECH.EDU
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
```

```
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXEC MULTINET_HOST_NAME "HAMLET.CALTECH.EDU"
$ DEFINE /SYSTEM /EXEC ARPANET_HOST_NAME "HAMLET.CALTECH.EDU"
$ DEFINE /SYSTEM /EXEC UCX$INET_HOST "HAMLET.CALTECH.EDU"
$ DEFINE /SYSTEM /EXEC UCX$BIND_DOMAIN "FLOWERS.COM"
$ DEFINE /SYSTEM /EXEC TCPIP$INET_HOST "HAMLET.CALTECH.EDU"
$ DEFINE /SYSTEM /EXEC TCPIP$BIND_DOMAIN "FLOWERS.COM"
$ @MULTINET:START_SERVER
$ @MULTINET:START_SMTP
$
```

SET IP-CLUSTER-ALIASES

Specifies a list of cluster-wide Internet addresses to which this node should respond.

In a VMScluster, the nodes coordinate among themselves so that only one node responds to the cluster-wide addresses at any one time. Should that node fail or be shut down, another node immediately takes over this task.

You can use this parameter with connectionless protocols (such as NFS) for automatic failover. IP-CLUSTER-ALIASES specifies a list of IP addresses to which this node should respond in addition to the address(es) that are configured for the interfaces you define.

Disable IP-CLUSTER-ALIASES by entering the value without specifying an address.

You can change the value of IP-CLUSTER-ALIASES without rebooting by also defining or redefining the system-wide logical name MULTINET_IP_CLUSTER_ALIASES and restarting the MULTINET_SERVER (@MULTINET:START_SERVER) process.

FORMAT

SET IP-CLUSTER-ALIASES [*address* [,*address* ...]]

PARAMETER

address
Specifies the Internet address to which to respond.

EXAMPLE

This example sets IP-CLUSTER-ALIASES to 10.1.1.2, then executes the commands that change this parameter without rebooting the system.

```
$ MULTNET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET IP-CLUSTER-ALIASES 10.1.1.2
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
```

```
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_IP_CLUSTER_ALIASES "10.1.1.2"
$ @MULTINET:START_SERVER
```

SET LOAD-EXOS-DRIVER

Enables automatic loading of the Novell EXOS-compatible $QIO driver when MultiNet starts. The driver allows MultiNet to support Novell's EXOS programming interface.

Reboot your system after setting `LOAD-EXOS-DRIVER` so that the change is recognized by the operating system.

FORMAT

```
SET LOAD-EXOS-DRIVER {TRUE | FALSE}
```

PARAMETERS

TRUE
Enables the EXOS $QIO programming interface.

FALSE
Disables the EXOS $QIO programming interface (the default setting).

EXAMPLE

```
$ MULTNET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET LOAD-EXOS-DRIVER TRUE
```

SET LOAD-PWIP-DRIVER

Enables automatic loading of the version 5 PATHWORKS server interface (PWIP) driver when MultiNet starts. You must enable the PWIP driver to use version 5 PATHWORKS.

> **Note:** Earlier versions of PATHWORKS use the UCX interface.

Reboot your system after setting `LOAD-PWIP-DRIVER` so that the change is recognized by the operating system.

FORMAT

```
SET LOAD-PWIP-DRIVER {TRUE | FALSE}
```

PARAMETERS

TRUE
Enables the PWIP driver.

FALSE
Disables the PWIP driver (the default setting).

SET LOCAL-DOMAIN

Overrides the default local domain derived from a previous SET HOST-NAME command.

If your configuration includes a Domain Name System (DNS) domain server, you can use the LOCAL-DOMAIN parameter to override the default local domain derived from the HOST-NAME parameter.

You can change the value of LOCAL-DOMAIN without rebooting by also defining or redefining the system-wide logical name MULTINET_LOCALDOMAIN and restarting the MULTINET_SERVER (@MULTINET:START_SERVER) and SMTP_SYMBIONT (@MULTINET_START_SMTP) processes.

FORMAT

SET LOCAL-DOMAIN *domain*

PARAMETER

domain
Specifies the default domain name.

EXAMPLE

This example sets LOCAL-DOMAIN to EXAMPLE.COM, then invokes the commands that change this parameter without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from ULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET LOCAL-DOMAIN EXAMPLE.COM
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_LOCALDOMAIN "EXAMPLE.COM"
$ DEFINE /SYSTEM /EXECUTIVE UCX$BIND_DOMAIN "EXAMPLE.COM"
$ DEFINE /SYSTEM /EXECUTIVE TCPIP$BIND_DOMAIN "EXAMPLE.COM"
$ @MULTINET:START_SERVER
$ @MULTINET:START_SMTP
$
```

SET LPD-DEFAULT-USERNAME

Specifies a default user name for print jobs received from a remote system via the LPD protocol.

The LPD-DEFAULT-USERNAME parameter allows all print jobs requested by remote users to be processed, regardless of whether each user has a local VMS user name. The parameter defines a default user name for users without local user names.

You can change the value of LPD-DEFAULT-USERNAME without rebooting by also defining or redefining the system-wide logical name MULTINET_LPD_DEFAULT_USERNAME.

FORMAT

SET LPD-DEFAULT-USERNAME *user*

PARAMETER

user
Specifies the default user name, which must be an existing name in the OpenVMS system.

EXAMPLE

This example sets LPD-DEFAULT-USERNAME to PYWACKET, then executes the commands that change this parameter without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET LPDD-DEFAULT-USERNAME PYWACKET
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_FPD_DEFAULT_USERNAME "PYWACKET"
```

SET NAMESERVER-RETRANSMISSION

Controls the time between requests and the number of name server requests made before the system stops sending requests to a nonresponding server. The first argument is the time in seconds; the optional second argument is the number of tries to make.

You can change the value of NAMESERVER-RETRANSMISSION without rebooting by also defining or redefining the system-wide logical names MULTINET_NAMESERVER_RETRANS and MULTINET_NAMESERVER_RETRY and restarting the MULTINET_SERVER (@MULTINET:START_SERVER) and SMTP_SYMBIONT (@MULTINET_START_SMTP) processes.

FORMAT

SET NAMESERVER-RETRANSMISSION *seconds retries*

PARAMETERS

seconds
Specifies the time between retransmissions, in seconds (by default, 4 seconds).

retries
Specifies the number of retransmissions to make before giving up (by default, 4 tries).

EXAMPLE

This example specifies that the name server waits eight seconds between retries, and makes up to four retries before the system stops sending requests to the nonresponding server. The commands after setting this parameter allow the parameter to take effect immediately without rebooting the system.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET NAMESERVER-RETRANSMISSION 8 4
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_NAMESERVER_RETRANS 8
```

```
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_NAMESERVER_RETRY 4
$ @MULTINET:START_SERVER
$ @MULTINET:START_SMTP
```

SET SNMP-MAX-CONNECTIONS

Defines the maximum number of connections that the SNMP agent will report on.

Format

`SET SNMP-MAX-CONNECTIONS` *number*

Parameter

number

Specifies the maximum number of connections that he SNMP agent will report on.

Description

Use the `SNMP-MAX-CONNECTIONS` parameter to increase the maximum number of connections that the SNMP agent will report on. A number that is too low will prevent the SNMP agent from reporting on any connections. If this parameter is not set, the SNMP agent uses the default of 256 connections. The maximum value is 3276.

SET SNMP-MAX-ROUTES

Defines the maximum number of routes that the SNMP agent will report on.

Format

`SET SNMP-MAX-ROUTES` *number*

Parameter

number

Specifies the maximum number of routes that the SNMP agent will report on.

Description

Use the `SNMP-MAX-ROUTES` parameter to increase the maximum number of routes that the SNMP agent will report on. A number that is too low will prevent the SNMP agent from reporting on any routes. If this parameter is not set, the SNMP agent uses the default of 256 connections. The maximum value is 2978.

SET SPOOL-DIRECTORY

Changes the MultiNet spool directory from its default of MULTINET_COMMON_ROOT:[MULTINET.SPOOL]. The spool directory is used to store transient mail and print files.

You can change the value of SPOOL-DIRECTORY without rebooting by also defining or redefining the system-wide logical name MULTINET_SPOOL.

FORMAT

SET SPOOL-DIRECTORY *directory*

PARAMETER

directory
Specifies the new spool directory.

EXAMPLE

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET SPOOL-DIRECTORY SYS$SYSROOT:[TMP]
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE MULTINET_SPOOL "SYS$SYSROOT:[TMP]"
```

SET TFTP-DIRECTORY

Defines the TFTP server's default directory.

FORMAT

`SET TFTP-DIRECTORY` *directory*

PARAMETER

directory
Specifies the name of the new default directory.

DESCRIPTION

Use the `TFTP-DIRECTORY` parameter to define the TFTP server's default directory. When the parameter is not set, there is no default directory.

You can change the value of `TFTP-DIRECTORY` without rebooting by also defining or redefining the system-wide logical name `MULTINET_TFTP_DEFAULT_DIRECTORY`, then issuing the `MULTINET NETCONTROL TFTP RELOAD` command.

EXAMPLE

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET TFTP-DIRECTORY USERS:[TFTP-FILES]
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE -
_$ MULTINET_TFTP_DEFAULT_DIRECTORY "USERS:[TFTP-FILES]"
$ MULTINET NETCONTROL TFTP RELOAD
$
```

SET TIMEZONE

Although VMS does not keep track of time zones, MultiNet requires this information. The TIMEZONE parameter sets the time zone for your system and indirectly specifies the offset from GMT (and UTC) at which the local VMS clock is run. MultiNet automatically adjusts for Daylight Savings Time (DST) if appropriate for the rules in effect.

If your VMS clock time and your local time differ, set the TIMEZONE parameter to correspond to the VMS clock.

You can change the value of TIMEZONE without rebooting by using the MULTINET SET TIMEZONE command.

FORMAT

SET TIMEZONE *timezone*

PARAMETER

timezone
Specifies the abbreviation for the time zone, for example, PST.

EXAMPLE

This example sets the time zone to PST (Pacific Standard Time), then uses SET TIMEZONE to assign the same value to the system-wide variable.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET TIMEZONE PST
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ MULTINET SET/TIMEZONE PST
$
```

SET TIMEZONE-RULES

Sets the time zone rules in effect for your system.

The normal time zone rules in MultiNet include zones for the United States and a number of other countries around the world. Certain countries, such as Canada, have their own Daylight Savings Time rules for time zones such as PST, Pacific Standard Time. The TIMEZONE-RULES parameter is used to override the default rules.

You can change the value of TIMEZONE-RULES without rebooting by using the MULTINET SET TIMEZONE command.

FORMAT

SET TIMEZONE-RULES *zonelist*

PARAMETER

zonelist
Specifies a comma-separated list of countries and time zones for which to load rules. These can be found in MULTINET:TIMEZONES.DAT.

EXAMPLE

This example sets the TIMEZONE-RULES to US/ARIZONA, then sets the system-wide variable to the same value so that the system does not have to be rebooted for the new rules to take effect.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET TIMEZONE-RULES US/ARIZONA
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ MULTINET SET/TIMEZONE MST/SELECT="US/ARIZONA"
$
```

SET WHOIS-DEFAULT-SERVER

Sets the default server for the WHOIS command. The WHOIS command displays host information obtained from the RS.INTERNIC.NET host server.

By default, WHOIS connects to the server on the host RS.INTERNIC.NET, but can be overridden using the /HOST qualifier. Use this command to change the default server.

You can change the value of WHOIS-DEFAULT-SERVER without rebooting by also defining or redefining the system-wide logical name MULTINET_WHOIS_DEFAULT_SERVER.

FORMAT

SET WHOIS-DEFAULT-SERVER *host*

PARAMETER

host
Specifies the name of the server to which to connect. If not specified, the host defaults to RS.INTERNIC.NET.

EXAMPLE

This example sets WHOIS-DEFAULT-SERVER to FNORD.EXAMPLE.COM, then sets the system-wide logical name MULTINET_WHOIS_DEFAULT_SERVER to the same value to avoid rebooting the system after changing the WHOIS default server parameter.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET WHOIS-DEFAULT-SERVER FNORD.EXAMPLE.COM
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$ DEFINE /SYSTEM /EXECUTIVE -
_$ MULTINET_WHOIS_DEFAULT_SERVER "FNORD.EXAMPLE.COM"
```

SET WINS-COMPATIBILITY

Enables automatic definition of the logical names required for compatibility with applications developed for the WIN/TCP and Pathway for OpenVMS products from The Wollongong Group. The logical names allow MultiNet to support applications that run under those products. You must also generate a UNIX-format host table using the `MULTINET HOST_TABLE COMPILE/UNIX` command.

FORMAT

```
SET WINS-COMPATIBILITY {TRUE | FALSE}
```

PARAMETERS

TRUE
Enables the definition of the WIN/TCP and Pathway for OpenVMS compatibility logical names.

FALSE
Disables the definition of the WIN/TCP and Pathway for OpenVMS compatibility logical names.

EXAMPLES

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SET LOCAL-DOMAIN EXAMPLE.COM
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
```

Use the following DCL commands to enable WINS compatibility mode without a reboot:

```
$ DIRECTORY = F$TRNLNM("MULTINET_SPECIFIC_ROOT")  - "]"
$ DEFINE/SYSTEM/EXEC/TRANSLATION_ATTRIBUTES=CONCEAL TWG$ETC -
$_ 'DIRECTORY'MULTINET.], 'DIRECTORY'SYSCOMMON.MULTINET.]
$ DEFINE/SYSTEM/EXEC/TRNASLATION_ATTRIBUTES=CONCEAL TWG$TCP
$ DEFINE/SYSTEM/EXEC INET_DOMAIN_NAME 'EXAMPLE.COM"
```

```
$ DEFINE/SYSTEM/EXEC INET_NAMESERVER_LIST "127.0.0.1"
$ DEFINE/SYSTEM/EXEC WINS_MAILSHR MULTINET:SMTP_MAILSHR
$ DEFINE/SYSTEM/EXEC DECW$TRANSPORT_WINTCP DECW$TRANSPORT_TCIP
```

Whether you choose to reboot or not, use the following commands to create a UNIX-format host table:

```
$ SET DEFAULT MULTINET_COMMON_ROOT:[MULTINET]
$ MULTINET HOST_TABLE COMPILE /UNIX
$
```

SHOW

Displays the current MultiNet device configuration.

FORMAT

SHOW *command*

PARAMETER

command

Specifies the type of display. Accepted values are CURRENT (the default) or MAXIMUM. If a command is not entered, the default is CURRENT.

EXAMPLE

This example lists the current configuration.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading in configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SHOW
```

Interface	Adapter	CSR Address	Flags/Vector
se0 (Shared VMS Ethernet) [TCP/IP: 10.41.228.78] [VMS Device: ESA0]	-NONE-	-NONE-	-NONE-
s10 (Serial Line IP) [TCP/IP: 192.41.228.78==>10.41.228.80] [VMS Terminal: TTA2, 9600 Baud] [Header Compression DISABLED]	-NONE-	-NONE-	-NONE-
*s11 (Serial Line IP) [VMS Terminal: TXA0] [Header Compression DISABLED]	-NONE-	-NONE-	-NONE-

```
Official Host Name:         BANANA.EXAMPLE.COM
Default IP Route:           10.41.228.71
Domain Nameservers:         127.0.0.1
Local Domain:               example.com
Timezone:                   PST
SMTP Host Name:             EXAMPLE.COM
Default RMT Tape:           MKB100:
Default TFTP Directory:     MULTINET_ROOT:[MULTINET.TFTP]
Anonymous FTP Directory:    USERS:[ANONYMOUS]
Load EXOS $QIO driver:      TRUE
Load UCX $QIO driver:       TRUE
WINS Compatibility:         TRUE
NET-CONFIG>QUIT
$
```

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. If the MULTINET_DISABLE_SPAWN logical is set, SPAWN does not work.

FORMAT

SPAWN [*command*]

PARAMETERS

command
Specifies a command to execute. If you omit command, a DCL command line subprocess is created.

QUALIFIERS

/INPUT=*file-spec*
Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES
/NOLOGICAL_NAMES
Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS
/NOSYMBOLS
Specifies that global and local names are not passed to the subprocess.

/WAIT
/NOWAIT
Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=*file-spec*

Specifies a file that retains the output of the command invoked with SPAWN. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after SPAWN or other qualifiers.

EXAMPLES

This example displays terminal information, captures the output in a file, then displays the information with the TYPE command.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SPAWN/OUTPUT=FOO. SHOW TERM
NET-CONFIG>SPAWN TYPE FOO.
...
```

This example invokes a command procedure.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SPAWN @COMPROC
...
```

This example displays help information about the NET-CONFIG utility. Use the LOGOUT command to return control to NET-CONFIG.

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>SPAWN RETURN
$ HELP MULTINET CONFIGURE /NETWORK_DEVICES
...
$ LOGOUT
NET-CONFIG>
```

STATUS

Displays the status of the current configuration.

FORMAT

```
STATUS
```

EXAMPLE

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>STATUS
There is the MultiNet network configuration program Version 5.6 (nnn)
There are 3/1024 devices in the current configuration.
There are 190/1024 devices in the MAXIMUM configuration.
The configuration MULTINET:NETWORK_DEVICES.CONFIGURATION is not modified.
The startup file MULTINET:START_MULTINET.COM is not modified.
NET-CONFIG>QUIT
$
```

USE

Reads in a configuration file. (Functionally equivalent to `GET`.)

FORMAT

`USE` *`config_file`*

PARAMETER

`config_file`
Specifies the name of the configuration file to read in.

VERSION

Displays the NET-CONFIG version and release information.

FORMAT

VERSION

EXAMPLE

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>VERSION
This is the MultiNet network configuration program Version 5.6 (nnn)
NET-CONFIG>QUIT
$
```

WRITE

Writes the current configuration to a file. (Functionally equivalent to SAVE.)

FORMAT

WRITE [*config_file*]

PARAMETER

config_file
Specifies the name of the configuration file to write out (by default, the same file from which the configuration was read).

EXAMPLE

```
$ MULTINET CONFIGURE /INTERFACE
MultiNet Network Configuration Utility 5.6 (nnn)
[Reading in MAXIMUM configuration from MULTINET:MULTINET.EXE]
[Reading configuration from MULTINET:NETWORK_DEVICES.CONFIGURATION]
NET-CONFIG>WRITE
[Writing configuration to
MULTINET_ROOT:[MULTINET]NETWORK_DEVICES.CONFIGURATION.7]
NET-CONFIG>EXIT
[Writing configuration to MULTINET:NETWORK_DEVICES.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$
```

5. NFS-CONFIG Command Reference

This chapter describes the commands you can run from the NFS-CONFIG command line. NFS-CONFIG lets you examine, modify, and save configuration files for the MultiNet NFS system.

To invoke NFS-CONFIG, enter:

```
$ MULTINET CONFIGURE /NFS
```

At any NFS-CONFIG prompt, you can list the available commands by typing **?**. Online help for each NFS-CONFIG command is available through the NFS-CONFIG HELP command.

Changes do not take effect until you reload the configuration with the NFS-CONFIG RELOAD command. You can also reload the MultiNet NFS server configuration with the MULTINET NETCONTROL NFS RELOAD and MULTINET NETCONTROL RPCMOUNT RELOAD commands, or by restarting the server with the RESTART command.

For details on configuring MultiNet NFS client and MultiNet NFS server, refer to the *MultiNet Installation and Administrator's Guide*.

Command Summary

The below table lists the commands you can run from the NFS-CONFIG prompt.

NFS-CONFIG Command	**Description**
ADD DECSTATION-MOUNT-POINT	Adds a particular directory or file to the list of DECstation mount points.
ADD EXPORT	Adds a device or directory to the list of exported mount points.

ADD MOUNT-RESTRICTION	Adds to the list of clients authorized to mount a file system.
ADD NFS-GROUP	Adds systems to a particular NFS group.
ADD NFS-PASSWD-FILE	Specifies the name of a UNIX-style /etc/password file that provides username-to-UID translation information.
ADD UID-TRANSLATION	Creates an individual mapping between an OpenVMS user name and a UID/GID pair.
APPEND	Appends configuration changes to a configuration file.
ATTACH	Detaches the terminal from the calling process and reattaches it to another process.
CURRENT	Selects the mount point to be acted on by the ADD and DELETE commands.
DELETE DECSTATION-MOUNT-POINT	Removes a file or directory from the list of DECstation mount points.
DELETE EXPORTED-FILE-SYSTEM	Removes a device or directory from the list of exported mount points.
DELETE MOUNT-RESTRICTION	Removes a client from the mount restrictions list.
DELETE NFS-GROUP	Removes systems from a particular NFS group.
DELETE NFS-PASSWD-FILE	Deletes an NFS password file entry.
DELETE UID-TRANSLATION	Deletes an OpenVMS username-to-UID/GID translation.
EXIT	Exits the configuration program after saving the current configuration, if it has been modified.

GET	Reads in a configuration file.
HELP	Invokes command help.
NETCONTROL	Transfers control to an NFS configuration manager subsystem that contacts the NETCONTROL server at local or remote sites.
PUSH	Creates a subprocess or attaches to a parent process.
QUIT	Prompts you to save the file before quitting.
RELOAD	Reloads the NFS client, NFS server, and RPCMOUNT server databases.
RESTART	Restarts the NFS and RPCMOUNT servers. (Same as WRITE.)
SAV	Saves the configuration parameters.
SELECT	Selects a mount point.
SET APPROXIMATE-TEXT-SIZE-THRESHOLD	Permits UNIX **ls** commands to execute faster.
SET DIRECTORY-INFO-FLUSH-AGE	Controls how long cached headers and data buffers for a directory can remain in the cache.
SET DIRECTORY-INFO-IDLE-FLUSH-AGE	Controls how long cached headers and data buffers for a directory which is not being accessed by any client can remain in the cache.
SET FILE-CACHE-TIMER-INTERVAL	Determines how often the MultiNet NFS server scans the cache, checking other parameters to see if their timers have expired, and processes those that have.
SET FILE-INFO-FLUSH-AGE	Controls how long cached headers and data buffers for a file can remain in the cache.

SET FILE-INFO-IDLE-FLUSH-AGE	Controls how long cached headers and data buffers for a file which is not being accessed by any client can remain in the cache.
SET MAXIMUM-CACHE-BUFFERS	Determines the maximum number of cached file headers and data buffers allowed simultaneously for the cache as a whole.
SET MAXIMUM-CACHE-FILES	Determines the maximum number of cached file headers and data buffers allowed simultaneously for the cache as a whole.
SET MAXIMUM-DIRTY-BUFFERS	Specifies the maximum number of modified buffers in the writeback cache.
SET MAXIMUM-FILESYSTEM-BUFFERS	Determines the maximum number of cached file headers and data buffers allowed simultaneously for single file systems on a per-mount-point basis.
SET MAXIMUM-FILESYSTEM-CHANNELS	Determines the maximum number of open channels allowed simultaneously for single file systems on a per-mount-point basis.
SET MAXIMUM-FILESYSTEM-FILES	Determines the maximum number of cached file headers and data buffers allowed simultaneously for single file systems on a per-mount-point basis.
SET MAXIMUM-OPEN-CHANNELS	Determines the maximum number of open channels allowed simultaneously for the cache as a whole.
SET MAXIMUM-QUEUED-REMOVES	Limits the number of remove operations that may be queued in the delete-behind cache.
SET MAXIMUM-WRITE-JOBS	Limits the number of simultaneous write operations that can occur when the writeback cache is enabled.
SET NUMBER-OF-DUPLICATE-REQUESTS-CACHED	Limits the number of requests that can be retained in the NFS server's duplicate-request detection cache.

SET NUMBER-OF-RPC-TRANSPORTS	Limits the number of simultaneous requests that the NFS server can process.
SET READ-ONLY-FLUSH-AGE	Limits how long idle channels can remain assigned to a file.
SET READ-WRITE-FLUSH-AGE	Limits how long idle channels can remain assigned to a file.
SET SECONDS-BEFORE-WRITEBACK	Sets the length of time that write operations are deferred before the data is written to disk.
SET USE-DIRECTORY-BLOCKING-ASTS	Determines whether the server flushes the cache whenever an OpenVMS user attempts to access a directory from which cached information came.
SET USE-FILE-BLOCKING-ASTS	Determines whether the server flushes the cache whenever an OpenVMS user attempts to access a file from which cached information came.
SHOW	Displays information about the configuration of the NFS server and NFS client.
SPAWN	Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH.
STATUS	Displays the status of the current configuration.
USE	Reads in a configuration file.
VERSION	Displays the NFS-CONFIG version and release information.
WRITE	Writes the current configuration to a file. (Same as SAVE.)

ADD DECSTATION-MOUNT-POINT

Adds a directory or file to the list of DECstations mount points using special DECstation-style file handles. You must do this for the root directory and swap or page files used to boot diskless DECstation systems.

FORMAT

`ADD DECSTATION-MOUNT-POINT` *filename*

PARAMETER

filename
Specifies the file name of the DECstation mount point.

EXAMPLE

This example shows adding a `DECSTATION-MOUNT-POINT`.

```
NFS-CONFIG>ADD DECSTATION-MOUNT-POINT
DECSTATION:[000000]MAXWELL_ROOT.DIR
[Added new DECstation Mount Point "DECSTATION:[000000]MAXWELL_ROOT.DIR;1"]
NFS-CONFIG>
```

ADD EXPORT

Adds a device or directory to the list of exported mount points.

FORMAT

`ADD EXPORT` *`mount_point_name`*

PARAMETER

`mount_point_name`
Specifies the name of the file system's mount point.

EXAMPLE

This example shows how to add `SYS$SYSDEVICE:` to the list of exported file systems.

```
NFS-CONFIG>ADD EXPORT SYS$SYSDEVICE:
[Added new Exported file system "SYS$SYSDEVICE:"]
[Current Exported File System set to "SYS$SYSDEVICE:"]
NFS-CONFIG>
```

ADD MOUNT-RESTRICTION

Adds a client to the list of authorized clients permitted to mount the specified mount point. If no clients are specified, anyone can access the mount point.

FORMAT

`ADD MOUNT-RESTRICTION` *`mount_point_name client_name`*

PARAMETERS

`mount_point_name`
Specifies the name of the mount point to which to add the restriction.

`client_name`
Specifies the name of the client, in full domain form (for example, SALES.EXAMPLE.COM), in abbreviated form (for example, SALES), in IP address form (for example, 192.168.32.4), or NFS group.

> **Note:** Some clients use the UNIX automount facility, and may require you to use the non-domain form of the host name. Before requesting access to a file system, these clients check the mount restriction list and refuse to attempt the mount request if it is not listed in the recognized format (as opposed to trying the request and letting it fail or succeed).

QUALIFIER

`/ro`
Use this qualifier to prevent writing to the disk specified by the mount point. This restriction affects any NFS group associated with that particular mount point.

EXAMPLE

This example shows how to add the client system "sales" to the list of clients authorized to mount `SYS$SYSDEVICE:`.

```
NFS-CONFIG>ADD MOUNT-RESTRICTION SYS$SYSDEVICE: SALES
[Added Mount restriction to "SYS$SYSDEVICE:" allowing host "SALES"]
NFS-CONFIG>
```

ADD NFS-GROUP

Adds NFS client systems to a particular NFS group. NFS groups can be used to group together client systems that share common UID/GID spaces. When you specify a UID/GID-to-OpenVMS username mapping with the `ADD UID-TRANSLATION` command, you can optionally specify an NFS group to associate the translation. This allows you to have different OpenVMS username-to-UID/GID translations for different client hosts (or groups of hosts).

FORMAT

`ADD NFS-GROUP` *`group_name`* `[`*`host_list`*`]`

PARAMETERS

`group_name`
Specifies the NFS group name.

`host_list`
Contains a comma-delimited list of names of hosts to add to the specified NFS group.

> **Note:** Some clients utilize the UNIX automount facility and may require you to use the non-domain form of the host name. Before requesting access to a file system, these clients check the mount restriction list, and refuse to attempt the mount request if it is not listed in the recognized format (as opposed to trying the request and letting it fail or succeed).

EXAMPLE

This example shows how to add two client systems to the NFS group `ENGINEERING`.

```
NFS-CONFIG>ADD NFS-GROUP ENGINEERING KAOS.EXAMPLE.COM,
FANG.EXAMPLE.COM
[Current NFS Group set to "ENGINEERING"]
NFS-CONFIG>
```

ADD NFS-PASSWD-FILE

Specifies the name of a UNIX-style `/etc/password` file that provides username-to-UID translation information.

To create a multi-user mapping, use FTP (or another file transfer utility) to copy each applicable `/etc/password` file from the UNIX system to the OpenVMS system running the server. Then, run the `NFS-CONFIG` utility, and use the `ADD NFS-PASSWD-FILE` command to create the mapping.

FORMAT

`ADD NFS-PASSWD-FILE` *`file_name`* [*`group_name`*]

PARAMETERS

`file_name`
Specifies the OpenVMS name of the copied `/etc/password` file.

`group_name`
Specifies the NFS group name with which the translations specified by the password file will be associated. If not specified, it defaults to the default NFS group (all translations that are not explicitly in a group).

EXAMPLES

This example shows how to add the file `MULTINET:NFS.PASSWD` to the configuration for the default NFS group.

```
NFS-CONFIG>ADD NFS-PASSWD-FILE MULTINET:NFS.PASSWD
[Added new NFS Password File "MULTINET:NFS.PASSWD"]
NFS-CONFIG>
```

This example shows how to add the file `MULTINET:NFS.PASSWD` to the configuration for the NFS group `ENGINEERING`.

```
NFS-CONFIG>ADD NFS-PASSWD-FILE MULTINET:NFS.PASSWD ENGINEERING
[Added new NFS Password File "ENGINEERING/MULTINET:NFS.PASSWD"]
NFS-CONFIG>
```

ADD UID-TRANSLATION

Creates an individual mapping between an OpenVMS user name and a UID/GID pair.

FORMAT

`ADD UID-TRANSLATION` *`username uid gid`* [*`group_name`*]

PARAMETERS

username
Specifies the name of the user's OpenVMS account.

uid
Specifies the user's UID.

gid
Specifies the user's GID.

group_name
Specifies the NFS group name into which this translation is placed (by default, the default NFS group-all translations not explicitly in a group).

EXAMPLES

This example shows how to add a translation between the OpenVMS user JOHN and the UNIX UID 10 and GID 15.

```
NFS-CONFIG>ADD UID-TRANSLATION JOHN 10 15
[Added UID Translation "JOHN" = 10, 15]
NFS-CONFIG>
```

This example shows how to add a translation between the OpenVMS user JETSON and the UNIX UID 101 and GID 20 in the NFS group ENGINEERING.

```
NFS-CONFIG>ADD UID-TRANSLATION JETSON 101 20 ENGINEERING
[Added UID Translation "ENGINEERING/JETSON" = 101, 20]
NFS-CONFIG>
```

APPEND

Adds to or changes the current configuration parameters. The information you want to append is in a text file whose name you specify as an argument. The appended parameters are in effect as long as the current configuration parameters are in memory. When you RELOAD or RESTART NFS, the parameters are reset to those specified in the NFS.CONFIGURATION file.

For the appended changes to become permanent, use the WRITE or EXIT commands, or enter **YES** when prompted by the RELOAD or RESTART command to save the current configuration.

APPEND is similar to GET, except that it does not reset any unspecified parameters.

FORMAT

APPEND *filename*

PARAMETER

filename
Specifies the name of the file you want to append.

EXAMPLE

This example shows how to append the contents of MULTINET:NFS_ALT.CONFIGURATION to the configuration information in the MULTINET:NFS.CONFIGURATION file, then save the enhanced configuration in the MULTINET:NFS.CONFIGURATION file.

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in NFS file server configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>APPEND MULTINET:NFS_ALT.CONFIGURATION
[Reading in NFS file server configuration from
MULTINET:NFS_ALT.CONFIGURATION]
NFS-CONFIG>SAVE
[Writing configuration to MULTINET:NFS.CONFIGURATION]
$ QUIT
```

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the `SPAWN SHOW PROCESS /SUBPROCESSES` command to list the names of subprocesses. Use the DCL `LOGOUT` command to return to the original process. If the `MULTINET_DISABLE_SPAWN` logical is enabled, `ATTACH` does not work.

FORMAT

`ATTACH` *process-name*

PARAMETER

process-name
Specifies the name of a process to which you want your terminal attached. (Not all subprocesses can be attached; some testing may be required.)

EXAMPLE

```
$ MULTINET CONFIGURE  /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>SPAWN
$ MM
MM>SPAWN SHOW PROCESS/SUB
...
There are 3 processes in this job:
_TWA42:
 PROC_1
  PROC_2 (*)
MM>ATTACH PROC_1
MM>QUIT
$ LOGOUT
NFS-CONFIG>
```

This example shows the use and exit of attached subprocesses.

1. The first command uses `SPAWN` to create a subprocess. `MM` is invoked from the DCL command line. Next, the `SPAWN SHOW PROCESS/SUB` command is used to list all the subprocess names. The display shows that three subprocesses are active. (Process

_TWA42: is NFS-CONFIG, PROC_1 is MM, and PROC_2 is the SPAWN SHOW PROCESS/SUB command.)

2. In the next command, the MM ATTACH command returns control to the NFS-CONFIG process. From this utility, ATTACH returns control to MM. To exit, the QUIT command is invoked from MM, and LOGOUT is invoked at the original spawned DCL command line; control returns to NFS-CONFIG. (If SPAWN SHOW PROCESS/SUB had been entered, only this command and the configuration processes would be active.)

CURRENT

Selects the mount point to be acted on by the ADD and DELETE commands if the mount point is not explicitly specified. (Functionally equivalent to SELECT.)

FORMAT

CURRENT *mount_point_name* | none

PARAMETER

mount_point_name
Specifies the name of a mount point. A value of NONE unselects the current mountpoint.

EXAMPLE

This example shows how to select the mount point SYS$SYSDEVICE:

```
NFS-CONFIG>CURRENT SYS$SYSDEVICE:
[Current Exported File System set to "SYS$SYDEVICE:]
NFS-CONFIG>
```

DELETE DECSTATION-MOUNT-POINT

Removes a file or directory from the list of DECstation mount points.

FORMAT

`DELETE DECSTATION-MOUNT-POINT` *filename*

PARAMETER

filename

Specifies the name of the file or directory to delete from the DECstation mount point list.

EXAMPLE

This example shows how to delete a DECstation mount point.

```
NFS-CONFIG>DELETE DECSTATION-MOUNT-POINT RETURN
Name: DECSTATION:[000000]MAXWELL_ROOT.DIR
NFS-CONFIG>
```

DELETE EXPORTED-FILE-SYSTEM

Removes a device or directory from the list of exported mount points.

FORMAT

`DELETE EXPORT` *`mount_point_name`*

PARAMETER

`mount_point_name`
Specifies the name of the file system's mount point.

EXAMPLE

This example shows how to remove `SYS$SYSDEVICE:` from the list of exported file systems.

```
NFS-CONFIG>DELETE EXPORT SYS$SYSDEVICE:
NFS-CONFIG>
```

DELETE MOUNT-RESTRICTION

Removes systems from the mount restrictions list.

FORMAT

`DELETE MOUNT-RESTRICTION` *`mount_point_name name`*

PARAMETERS

`mount_point_name`
Specifies the name of the file system's mount point.

`name`
Specifies the name of the client or NFS group to remove from the restriction list.

EXAMPLE

This example shows how to delete the client "sales" from the mount restriction list for `SYS$SYSDEVICE:`.

```
NFS-CONFIG>DELETE MOUNT-RESTRICTION SYS$SYSDEVICE: SALES
[Deleted Mount restriction "SALES"]
NFS-CONFIG>
```

DELETE NFS-GROUP

Removes NFS systems from a particular NFS group. When the last client in a group is deleted, the group itself is also deleted from the configuration.

FORMAT

`DELETE NFS-GROUP group_name [host_names]`

PARAMETERS

group_name
Specifies the NFS group name.

host_names
Contains a comma-delimited list of host names to delete from the specified NFS group. If no client is specified, or if an asterisk (*) is specified, the group itself will be deleted.

> **Note:** Some clients utilize the UNIX automount facility, and may require you to use the non-domain form of the host name. Before requesting access to a file system, these clients check the mount restriction list and refuse to attempt the mount request if it is not listed in the recognized format (as opposed to trying the request and letting it fail or succeed).

EXAMPLES

This example shows how to delete the system FANG.EXAMPLE.COM from the NFS group ENGINEERING.

```
NFS-CONFIG>DELETE NFS-GROUP ENGINEERING FANG.EXAMPLE.COM
NFS-CONFIG>
```

This example shows how to delete the NFS group MARKETING.

```
NFS-CONFIG>DELETE NFS-GROUP MARKETING *
NFS-CONFIG>
```

DELETE NFS-PASSWD-FILE

Deletes an NFS password file entry.

FORMAT

`DELETE NFS-PASSWD-FILE` *`file_name [group_name]`*

PARAMETERS

`file_name`
Specifies the OpenVMS name of the copied `/etc/password` file.

`group_name`
Specifies the NFS group name with which the translations specified by the password file are associated (by default, the default NFS group-all translations that are not explicitly in a group).

EXAMPLE

This example shows how to delete the file `MULTINET:NFS.PASSWD` from the NFS password file list for the default NFS group.

```
NFS-CONFIG>DELETE NFS-PASSWD-FILE MULTINET:NFS.PASSWD
NFS-CONFIG>
```

DELETE UID-TRANSLATION

Deletes an OpenVMS user name-to-UID/GID translation.

FORMAT

`DELETE UID-TRANSLATION [group_name/] username`

PARAMETER

[group_name/]username

Specifies the name for the user's OpenVMS account. To delete a UID translation in a group other than the default, specify the group name, a slash, then the OpenVMS user name.

EXAMPLES

This example shows how to delete the user name JOHN from the configuration.

```
NFS-CONFIG>DELETE UID-TRANSLATION JOHN
NFS-CONFIG>
```

This example shows how to delete the user name JETSON in the NFS group ENGINEERING from the configuration.

```
NFS-CONFIG>DELETE UID-TRANSLATION ENGINEERING/JETSON
NFS-CONFIG>
```

EXIT

Saves the current configuration, if it has been modified, then exits the configuration program. Use the `STATUS` command to display whether the configuration was modified.

FORMAT

```
EXIT
```

EXAMPLES

When the configuration has not changed, a message displays indicating that the configuration file is not updated.

```
$ MULTINET CONFIGURE  /NFS
NFSS-CONFIG>EXIT
$
```

When the configuration has changed, a message displays indicating that the configuration file has been updated.

```
$ MULTINET CONFIGURE  /NFS
MultiNet NFS Configuration Utility 5.6(76)
[Reading in NFS file server configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>GET MULTINET:NFS.CONFIGURATION
NFS-CONFIG>EXIT
$
```

GET

Reads in a configuration file.

FORMAT

GET *filename*

PARAMETER

filename
Specifies the file name of the configuration to be read (by default, NETWORK_DEVICES.CONFIGURATION in the current working directory). GET resets any unspecified parameters. To augment the existing configuration, use the APPEND command.

EXAMPLE

This example retrieves the configuration file MULTINET:TEST.CONFIGURATION into the NFS-CONFIG workspace.

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>GET MULTINET:TEST.CONFIGURATION
```

HELP

Invokes command help.

FORMAT

`HELP [topics]`

PARAMETER

topics

Contains a space-delimited list of topics that begins with a topic followed by subtopics. The default topic is `HELP`.

NETCONTROL

Transfers control to an NFS configuration manager subsystem that contacts the NETCONTROL server at local or remote sites.

After invoking NETCONTROL, you can issue commands to the NETCONTROL server to affect MULTINET_SERVER operations at that site.

FORMAT

NETCONTROL [*host*]

RESTRICTION

The NETCONTROL server is normally protected from unauthorized access by a restriction list.

PARAMETER

host
Specifies the name of the host to which to connect (by default, the local host).

EXAMPLE

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>NETCONTROL
Connected to NETCONTROL server on "127.0.0.1"
<EXAMPLE.COM Network Control 5.6 (nnn) at Mon 15-Mar-2019 7:42am-EST
NFS>
```

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command.

If the MULTINET_DISABLE_SPAWN logical is set, PUSH does not work.

FORMAT

PUSH

QUIT

Prompts you to save the current configuration if it was modified, and then exits.

FORMAT

```
QUIT
```

EXAMPLE

```
$ MULTINET CONFIGURE  /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>GET MULTINET:NFS.CONFIGURATION
NFS-CONFIG>QUIT
Configuration modified, do you want to save it ? [NO]RETURN
$
```

RELOAD

Reloads the NFS and RPCMOUNT server databases, the NFS client UID/GID translation table, and the NFS group configuration.

If the configuration has been modified since the last save, `RELOAD` prompts you to save it before reloading. `RELOAD` allows you to update the NFS UID-translations and exported mount points without flushing the file cache and causing a temporary performance degradation.

FORMAT

```
RELOAD
```

RESTART

Restarts the MultiNet NFS server process and reloads the RPCMOUNT server's databases. If the configuration has been modified since the last save, RESTART prompts you to save it before restarting. In general, you should use RELOAD instead of RESTART.

FORMAT

RESTART

EXAMPLE

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6(nnn)
[Reading in NFS server configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>RESTART
Connected to NETCONTROL server on "127.0.0.1"
< simple.example.com Network Control 5.6(nnn) at Tue 27-Apr-2019 2:27PM
< NFS/RPCLockMgr Server Started
< RPCMOUNT database reloaded
NFS-CONFIG>
```

SAVE

Saves the configuration parameters.

FORMAT

`SAVE [filename]`

PARAMETER

filename

Specifies the file name for the configuration file being saved (by default, the file from which the configuration was read).

SELECT

Selects the mount point to be acted on by the ADD and DELETE commands if the mount point is not explicitly specified. (Functionally equivalent to CURRENT.)

FORMAT

SELECT *mount_point_name* | NONE

PARAMETER

mount_point_name
Specifies the name of the mount point. A value of NONE unselects the current mountpoint.

EXAMPLE

This example shows how to select the mount point SYS$SYSDEVICE:.

```
NFS-CONFIG>SELECT SYS$SYSDEVICE:
[Current Exported File System set to "SYS$SYDEVICE:]
NFS-CONFIG>
```

SET APPROXIMATE-TEXT-SIZE-THRESHOLD

When set to a positive value, the MultiNet NFS server permits UNIX `ls` commands to execute faster by approximating file sizes when the OpenVMS file length exceeds the specified threshold.

> **Note:** The NFS specification requires that NFS servers return exact file sizes.

FORMAT

`SET APPROXIMATE-TEXT-SIZE-THRESHOLD` *`threshold`*

PARAMETER

`threshold`
Specifies the minimum OpenVMS file length (in bytes) required before the NFS server approximates the file size.

The client must use the `/APPROXIMATE_TEXT_SIZE` qualifier as a mount point option for the threshold to take effect.

SET DIRECTORY-INFO-FLUSH-AGE

Controls how long cached headers and data buffers for a directory can remain in the cache.

Unless the cache-interrupt parameters are on, cached headers and buffers are not automatically discarded when an OpenVMS user attempts to access their directories on disk. The DIRECTORY-INFO-FLUSH-AGE parameter specifies a period after which the server discards cached information (requiring rereads from disk if the information is needed again).

This parameter is a trade-off between response time and concurrency between information stored in the cache and on the disk.

You can raise or lower the default setting; however, if you set this parameter below 15 seconds, the server cannot complete any directory operations.

FORMAT

SET DIRECTORY-INFO-FLUSH-AGE *seconds*

PARAMETER

seconds
Specifies the length of time, in seconds (by default, 300).

EXAMPLE

This example shows how to set the DIRECTORY-INFO-FLUSH-AGE interval to 600 seconds.

```
NFS-CONFIG>SET DIRECTORY-INFO-FLUSH-AGE 600
NFS-CONFIG>
```

SET DIRECTORY-INFO-IDLE-FLUSH-AGE

This parameter controls how long cached headers and data buffers can remain in the cache for a directory that is not being accessed by any client.

Unless the cache-interrupt parameters are on, cached headers and buffers are not discarded automatically when an OpenVMS user attempts to access their directories on disk. The DIRECTORY-INFO-IDLE-FLUSH-AGE parameter specifies a period after which the server discards cached information unless a client is actively accessing it (requiring rereads from disk if the information is needed again).

This parameter is a trade-off between response time and concurrency between information stored in the cache and on the disk.

You can raise or lower the default setting; however, if you set this parameter below 15 seconds, the server cannot complete any directory operations.

FORMAT

SET DIRECTORY-INFO-IDLE-FLUSH-AGE *seconds*

PARAMETER

seconds
Specifies the length of time, in seconds (by default, 150).

EXAMPLE

This example shows how to set the DIRECTORY-INFO-IDLE-FLUSH-AGE interval to 300 seconds.

```
NFS-CONFIG>SET DIRECTORY-INFO-IDLE-FLUSH-AGE 300
NFS-CONFIG>
```

SET FILE-CACHE-TIMER-INTERVAL

Determines how often the MultiNet NFS server scans the cache, checking the other parameters to see if their timers have expired, and processes those that have.

FORMAT

`SET FILE-CACHE-TIMER-INTERVAL` *seconds*

PARAMETER

seconds
Specifies the length of time, in seconds (by default, 30).

> **Note:** The default setting for the `FILE-CACHE-TIMER-INTERVAL` parameter, 30 seconds, is not changed during configuration.

EXAMPLE

This example shows how to set the `FILE-CACHE-TIMER-INTERVAL` interval to 15 seconds.

```
NFS-CONFIG>SET FILE-CACHE-TIMER-INTERVAL 15
NFS-CONFIG>
```

SET FILE-INFO-FLUSH-AGE

Controls how long cached headers and data buffers for a file can remain in the cache.

Unless the cache-interrupt parameters are on, cached headers and buffers are not automatically discarded when an OpenVMS user attempts to access their files on disk. The FILE-INFO-FLUSH-AGE parameter specifies a period after which the server discards cached information (requiring rereads from disk if the information is needed again).

This parameter is a trade-off between response time and concurrency of information stored in the cache and on the disk.

You can raise or lower the default setting; however, if you set this parameter below 15 seconds, the server cannot complete any file operations.

FORMAT

SET FILE-INFO-FLUSH-AGE *seconds*

PARAMETER

seconds
Specifies the length of time, in seconds (by default, 1200).

EXAMPLE

This example shows how to set the FILE-INFO-FLUSH-AGE interval to 2400 seconds.

```
NFS-CONFIG>SET FILE-INFO-FLUSH-AGE 2400
NFS-CONFIG>
```

SET FILE-INFO-IDLE-FLUSH-AGE

Controls how long cached headers and data buffers for a file that is not being accessed by a client can remain in the cache.

Unless the cache-interrupt parameters are on, cached headers and buffers are not automatically discarded when an OpenVMS user attempts to access their files on disk. The FILE-INFO-IDLE-FLUSH-AGE parameter specifies a period after which the server discards cached information unless a client is actively accessing it (requiring rereads from disk if the information is needed again).

This parameter is a trade-off between response time and the concurrency between information stored in the cache and on the disk.

You can raise or lower the default setting; however, if you set this parameter below 15 seconds, the server cannot complete any file operations.

FORMAT

SET FILE-INFO-IDLE-FLUSH-AGE *seconds*

PARAMETER

seconds
Specifies the length of time, in seconds (by default, 600).

EXAMPLE

This example shows how to set the FILE-INFO-IDLE-FLUSH-AGE interval to 1200 seconds.

```
NFS-CONFIG>SET FILE-INFO-IDLE-FLUSH-AGE 1200
NFS-CONFIG>
```

SET MAXIMUM-CACHE-BUFFERS

Determines the maximum number of cached data buffers allowed simultaneously for the cache as a whole.

FORMAT

`SET MAXIMUM-CACHE-BUFFERS` *buffers*

PARAMETER

buffers

Specifies the number of data buffers. Each data buffer holds 16 disk blocks; the default is 3000.

> **Note:** Unless the setting for `MAXIMUM-CACHE-BUFFERS` is large enough to allow the cache to hold the largest files the client will access, performance will be severely degraded for those files. Each cached data buffer holds 16 disk blocks.

EXAMPLE

This example shows how to set the `MAXIMUM-CACHE-BUFFERS` parameter to 250 buffers.

```
NFS-CONFIG>SET MAXIMUM-CACHE-BUFFERS 250
NFS-CONFIG>
```

SET MAXIMUM-CACHE-FILES

Determines the maximum number of cached file headers allowed simultaneously for the cache as a whole.

FORMAT

`SET MAXIMUM-CACHE-FILES` *files*

PARAMETER

files
Specifies the maximum number of files.

EXAMPLE

This example shows how to set the `MAXIMUM-CACHE-FILES` parameter to 1500 files.

```
NFS-CONFIG>SET MAXIMUM-CACHE-FILES 1500
NFS-CONFIG>
```

SET MAXIMUM-DIRTY-BUFFERS

Controls the functions of the optional writeback feature of the directory and file cache. If the writeback cache is enabled by setting the SECONDS-BEFORE-WRITEBACK parameter to a non-zero value, and the number of modified buffers in the cache exceeds this limit, a write operation is started immediately. A value of zero means that there is no limit to the number of buffers the NFS server can create.

FORMAT

SET MAXIMUM-DIRTY-BUFFERS *buffers*

PARAMETER

buffers
Specifies the number of buffers (by default, 0).

EXAMPLE

This example shows how to set the MAXIMUM-DIRTY-BUFFERS parameter to 10 buffers.

```
NFS-CONFIG>SET MAXIMUM-DIRTY-BUFFERS 10
NFS-CONFIG>
```

SET MAXIMUM-FILESYSTEM-BUFFERS

Determines the maximum number of cached data buffers allowed simultaneously for a single file system on a per-mount-point basis.

FORMAT

`SET MAXIMUM-FILESYSTEM-BUFFERS` *buffers*

PARAMETER

buffers

Specifies the number of data buffers. Each data buffer holds 16 disk blocks; the default is 500.

> **Note:** Unless the setting for `MAXIMUM-FILESYSTEM-BUFFERS` is large enough to allow the cache to hold the largest files the client will access, performance will be severely degraded for those files. Each cached data buffer holds 16 disk blocks.

EXAMPLE

This example shows how to set the `MAXIMUM-FILESYSTEM-BUFFERS` parameter to 250 buffers.

```
NFS-CONFIG>SET MAXIMUM-FILESYSTEM-BUFFERS 250
NFS-CONFIG>
```

SET MAXIMUM-FILESYSTEM-CHANNELS

Determines the maximum number of open channels allowed simultaneously for a single file system on a per-mount-point basis.

FORMAT

```
SET MAXIMUM-FILESYSTEM-CHANNELS channels
```

PARAMETER

channels
Specifies the maximum number of open channels (by default, 50).

EXAMPLE

This example shows how to set the `MAXIMUM-FILESYSTEM-CHANNELS` to 10 channels.

```
NFS-CONFIG>SET MAXIMUM-FILESYSTEM-CHANNELS 10
NFS-CONFIG>
```

SET MAXIMUM-FILESYSTEM-FILES

Determines the maximum number of cached file headers allowed simultaneously for single file systems on a per-mount-point basis.

FORMAT

SET MAXIMUM-FILESYSTEM-FILES *files*

PARAMETER

files
Specifies the maximum number of cached files (by default, 3000).

EXAMPLE

This example shows how to set the MAXIMUM-FILESYSTEM-FILES to 1500 files.

```
NFS-CONFIG>SET MAXIMUM-FILESYSTEM-FILES 1500
NFS-CONFIG>
```

SET MAXIMUM-OPEN-CHANNELS

Determines the maximum number of open channels allowed simultaneously for the cache as a whole.

FORMAT

`SET MAXIMUM-OPEN-CHANNELS` *channels*

PARAMETER

channels
Specifies the maximum number of open channels (by default, 50).

EXAMPLE

This example shows how to set the `MAXIMUM-OPEN-CHANNELS` to 100 channels.

```
NFS-CONFIG>SET MAXIMUM-OPEN-CHANNELS 100
NFS-CONFIG>
```

SET MAXIMUM-QUEUED-REMOVES

Sets a limit on the number of remove operations that can be queued in the delete-behind cache.

This parameter affects how client users perceive the speed at which directories and files are deleted. The OpenVMS file deletion operation is very slow. The MultiNet NFS server uses its delete-behind queue to hide some of the deletion delay from the client user. When a request to delete a directory or file arrives, the request is answered immediately, but usually the delete request is only enqueued to the OpenVMS file system.

The `MAXIMUM-QUEUED-REMOVES` parameter limits the number of requests that can be enqueued. When that number is reached, the next delete request must wait until the next enqueued request has completed.

> **Note:** This delay can be significant if the next request is to delete a large directory; directory deletions always occur synchronously, and each file in a directory must be deleted before the directory itself is deleted. Therefore, the parameter setting defines when, in a series of deletions, the client user will perceive the OpenVMS deletion delay.

FORMAT

`SET MAXIMUM-QUEUED-REMOVES` *`value`*

PARAMETER

`value`
Specifies the maximum number of queued operations. A value of 0 disables the delete-behind cache, making all delete operations synchronous. The default is 25.

EXAMPLE

This example shows how to set the `MAXIMUM-QUEUED-REMOVES` parameter to 10 files.

```
NFS-CONFIG>SET MAXIMUM-QUEUED-REMOVES 10
NFS-CONFIG>
```

SET MAXIMUM-WRITE-JOBS

Limits the number of simultaneous write operations that can occur when the writeback cache is enabled.

FORMAT

`SET MAXIMUM-WRITE-JOBS` *limit*

PARAMETER

limit
Specifies the maximum number of simultaneous write operations. A value of zero (the default) means there is no limit.

EXAMPLE

This example shows how to set the `MAXIMUM-WRITE-JOBS` parameter to 5 simultaneous write operations.

```
NFS-CONFIG>SET MAXIMUM-WRITE-JOBS 5
NFS-CONFIG>
```

SET NUMBER-OF-DUPLICATE-REQUESTS-CACHED

Limits the number of requests that can be retained in the NFS server's duplicate-request detection cache. The NFS server uses this cache to store the most recent responses it has sent to clients that request directory and file access.

The duplicate-request detection cache operates with the cache that the RPC protocol module keeps of the transaction IDs (XIDs) of the last 400 requests it has seen. The RPC layer uses its cache to detect duplicate requests.

For example, if the network layer dropped a UDP packet containing a response to a client, the client would repeat the request after an interval, and the RPC protocol would notify the MultiNet NFS server that the request was a duplicate. The server would look in its duplicate-request detection cache for the response to resend without repeating the original operation.

> **Note:** Too low a value causes the following error message to display frequently on the OpenVMS console: "Duplicate Detected but not in cache." Too low a value can also cause an incorrect answer to be sent. A value above 400 has the same effect as 400 (400 is the maximum number of XIDs stored by the RPC protocol).

FORMAT

`SET NUMBER-OF-DUPLICATE-REQUESTS-CACHED` *value*

PARAMETER

value
Specifies the size of the duplicate request cache. By default, the cache stores the last 250 responses sent.

EXAMPLE

This example shows how to set the `NUMBER-OF-DUPLICATE-REQUESTS-CACHED` parameter to 300.

```
NFS-CONFIG>SET NUMBER-OF-DUPLICATE-REQUESTS-CACHED 300
NFS-CONFIG>
```

SET NUMBER-OF-RPC-TRANSPORTS

Limits the number of client requests that the MultiNet NFS server can process simultaneously. When the set limit is reached, no new requests are processed until one of the requests in progress completes. Processing multiple requests simultaneously prevents a single client from locking out other clients while it is performing a slow operation.

FORMAT

`SET NUMBER-OF-RPC-TRANSPORTS` *value*

PARAMETER

value

Specifies the number of simultaneous operations. You can change this value to adjust the trade-off between concurrency and memory requirements. The default setting (10) allows the server to process 10 requests simultaneously.

EXAMPLE

This example shows how to set the `NUMBER-OF-RPC-TRANSPORTS` to 100 transports.

```
NFS-CONFIG>SET NUMBER-OF-RPC-TRANSPORTS 100
NFS-CONFIG>
```

SET READ-ONLY-FLUSH-AGE

Limits how long idle channels can remain assigned to a file.

Applies to files that have been opened for read operations only; the READ-WRITE-FLUSH-AGE parameter applies to files that have been opened for both read and write operations. Closing a channel does not discard the data in the file headers and data buffers, and clients can continue to access the cached data without requiring that the file be reopened.

You can shorten or lengthen the timer interval to adjust trade-offs between improved response time and the overhead of keeping channels assigned.

FORMAT

SET READ-ONLY-FLUSH-AGE *seconds*

PARAMETER

seconds
Specifies the length of time, in seconds (by default,180).

EXAMPLE

This example shows how to set the READ-ONLY-FLUSH-AGE interval to 60 seconds.

```
NFS-CONFIG>SET READ-ONLY-FLUSH-AGE 60
NFS-CONFIG>
```

SET READ-WRITE-FLUSH-AGE

Limits how long idle channels can remain assigned to a file.

Applies to files that have been opened for read operations only; the READ-WRITE-FLUSH-AGE parameter applies to files that have been opened for both read and write operations. Closing a channel does not discard the data in the file headers and data buffers, and clients can continue to access the cached data without requiring that the file be reopened. However, the file is locked from access via OpenVMS until the NFS server releases the channel.

You can shorten or lengthen the timer interval to adjust trade-offs between improved response time and the overhead of keeping channels assigned.

FORMAT

SET READ-WRITE-FLUSH-AGE *seconds*

PARAMETER

seconds
Specifies the length of time, in seconds (by default, 60).

EXAMPLE

This example shows how to set the READ-WRITE-FLUSH-AGE interval to 15 seconds.

```
NFS-CONFIG>SET READ-WRITE-FLUSH-AGE 15
NFS-CONFIG>
```

SET SECONDS-BEFORE-WRITEBACK

Sets the length of time write operations are deferred before the data is written to disk.

FORMAT

`SET SECONDS-BEFORE-WRITEBACK` *seconds*

PARAMETER

seconds
Specifies the length of time in seconds. The default (`0`) disables the writeback cache.

DESCRIPTION

The directory and file cache normally function as a write-through cache. In this case, whenever a client is notified that a write request has completed, the data has been stored on the disk, and data integrity is guaranteed.

The optional writeback feature greatly increases the speed of write operations, as perceived by the user, by notifying the client that write operations are complete when the data is stored in cache memory on the server, but before it is written to disk. This increase in perceived write performance is achieved at the risk of data loss if the OpenVMS server crashes while a write operation is in progress or if, during a write operation, the server encounters an error such as insufficient disk space, insufficient disk quota, or a hardware write error.

When the server is unable to complete a writeback write operation, it discards the write operation, flags the file's cached header to indicate the error, and sends an error message in response to the next request for the file. However, if there is no new request before the affected header is discarded, or if the next request is from another user, data can be lost.

The `SECONDS-BEFORE-WRITEBACK` parameter determines whether the writeback feature is enabled and specifies how long the server will delay initiating a write operation after receiving data for a write request. The longer the delay, the greater the chance that the server can merge multiple small write operations into fewer, larger, and more efficient operations.

The default setting (`0`) disables the writeback feature. Any other value enables the feature. The recommended value for writeback delay is 5 seconds; little performance is gained from longer delays.

EXAMPLE

This example shows how to set the SECONDS-BEFORE-WRITEBACK interval to 5 seconds.

```
NFS-CONFIG>SET SECONDS-BEFORE-WRITEBACK 5
NFS-CONFIG>
```

SET USE-DIRECTORY-BLOCKING-ASTS

Determines whether the server flushes the cache when an OpenVMS user attempts to access a directory from which cached information came.

Enabling blocking ASTs causes the server to discard the cached file header and all data buffers for a directory when an OpenVMS user attempts to access it on disk.

You must enable this parameter to allow PC clients to use the PC-NFSD remote printing function. Enabling this parameter also ensures that client users almost always receive the directory as it exists on disk. This concurrency is at the expense of the overhead of the additional interrupts and disk reads.

FORMAT

`SET USE-DIRECTORY-BLOCKING-ASTS` *value*

PARAMETER

value
Specify this parameter as `1` to enable blocking ASTs on directories (the default), or `0` to disable blocking ASTs.

EXAMPLE

This example shows how to turn off the `USE-DIRECTORY-BLOCKING-ASTS` parameter.

```
NFS-CONFIG>SET USE-DIRECTORY-BLOCKING-ASTS 0
NFS-CONFIG>
```

SET USE-FILE-BLOCKING-ASTS

Determines whether the server flushes the cache when an OpenVMS user attempts to access a file from which cached information came.

Enabling blocking ASTs causes the server to discard the cached file header and all data buffers for a file when an OpenVMS user attempts to access it on disk.

You must enable this parameter to allow PC clients to use the PC-NFSD remote printing function. Enabling this parameter also ensures that client users almost always receive the file as it exists on disk. This concurrency is at the expense of the overhead of the additional interrupts and disk reads.

FORMAT

`SET USE-FILE-BLOCKING-ASTS` *value*

PARAMETER

value
Specify this parameter as 1 to enable blocking ASTs on files (the default), or 0 to disable blocking ASTs.

EXAMPLE

This example shows how to turn off the `USE-FILE-BLOCKING-ASTS` parameter.

```
NFS-CONFIG>SET USE-FILE-BLOCKING-ASTS 0
NFS-CONFIG>
```

SHOW

Displays information about the configuration of the NFS server and NFS client.

FORMAT

`SHOW [mount_point_name] | [option]`

PARAMETERS

mount_point_name

Specifies the name of a specific file system's mount point. If not specified, `SHOW` displays:

- The file system export list - a list of the file systems available to the network. A mount restrictions list appears next to the entry for each file system, showing the clients that can access the file system (unless all clients can access it).
- The UID/GID-to-OpenVMS user name translation list.
- The global parameter list - the names and settings of the server's global parameters.

option

Specifies the set of configuration parameters to be viewed. Accepted values are `exported-file-systems`, `nfs-groups`, `nfs-passwd-files`, `parameters`, and `uid-translations`.

QUALIFIER

/FULL

Displays information in greater detail. Without `/FULL`, `SHOW` truncates the mount restriction list at 80 columns and displays ellipses (...) to indicate there are more entries. With `/FULL`, `SHOW` displays the full mount restriction list and the settings of each global parameter.

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. If the MULTINET_DISABLE_SPAWN logical is set, SPAWN does not work.

FORMAT

SPAWN [*command*]

PARAMETER

command
Specifies a command to execute. If you omit command, a DCL command line subprocess is created.

QUALIFIERS

/INPUT=*file-spec*
Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES
/NOLOGICAL_NAMES
Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS
/NOSYMBOLS
Specifies that global and local names are not passed to the subprocesses.

/WAIT
/NOWAIT
Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=*file-spec*

Specifies a file that retains the output of the command invoked with SPAWN. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after SPAWN or other qualifiers.

EXAMPLES

This example displays terminal information, captures the output in a file, then displays the information with the TYPE command.

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
NFS-CONFIG>SPAWN/OUTPUT=FOO. SHOW TERM
NFS-CONFIG>SPAWN TYPE FOO.
 . .
```

This example invokes a command procedure.

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
NFS-CONFIG>SPAWN @COMPROC
 . .
```

This example displays help information about NFS-CONFIG. Use the LOGOUT command to return control to NFS-CONFIG.

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
NFS-CONFIG>SPAWN
$ HELP MULTINET CONFIGURE /NFS
 . .
$ LOGOUT
NFS-CONFIG>
```

STATUS

Displays the status of the current configuration.

FORMAT

```
STATUS
```

EXAMPLE

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in NFS file server configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>STATUS
This is the MultiNet NFS configuration program Version 5.6(76)
There are 19/1024 entries in the exported NFS filesystem list.
There are 1/100 entries in the NFS passwd file list.
There are 10/5000 entries in the UID translation list.
There is NO selected FILESYSTEM entry.
The configuration MULTINET:NFS.CONFIGURATION has not been modified.
NFS-CONFIG>QUIT
$
```

USE

Reads in a configuration file. (Functionally equivalent to `GET`.)

FORMAT

`USE` *`config_file`*

PARAMETER

`config_file`
Specifies the name of the configuration file to read.

VERSION

Displays the NFS-CONFIG version and release information.

FORMAT

VERSION

EXAMPLE

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6
[Reading in NFS file server configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>VERSION
This is the MultiNet NFS configuration program Version 5.6
NFS-CONFIG>QUIT
```

WRITE

Writes the current configuration to a file. (Functionally equivalent to SAV.)

FORMAT

WRITE [*config_file*]

PARAMETER

config_file
Specifies the name of the configuration file to write, by default, the file from which the configuration was read.

EXAMPLE

```
$ MULTINET CONFIGURE /NFS
MultiNet NFS Configuration Utility 5.6 (nnn)
[Reading in NFS file server configuration from MULTINET:NFS.CONFIGURATION]
NFS-CONFIG>WRITE
[Writing configuration to MULTINET_ROOT:[MULTINET]NFS.CONFIGURATION.7]
NFS-CONFIG>EXIT
[Writing configuration to MULTINET:NFS.CONFIGURATION]
[Writing Startup file MULTINET:START_MULTINET.COM]
[Changes take effect after the next VMS reboot]
$
```

6. NTYCP Command Reference

This chapter describes the Network Terminal Device Control Program (NTYCP) commands you can use to create terminal devices.

To invoke NTYCP as an OpenVMS "foreign" command:

```
$ NTYCP := $MULTINET:NTYCP
```

To invoke NTYCP interactively:

```
$ RUN MULTINET:NTYCP
NTYCP> CREATE PORT NTYnnnn /NODE=host-name /PORT=port-number
NTYCP> EXIT
```

To set up the terminal characteristics:

```
$ SET TERMINAL NTYnnn:/PERMANENT/NOBROADCAST/NOTYPEAHEAD/NOWRAP
```

To set up spooling:

```
$ SET DEVICE/SPOOLED=(queue-name,SYS$SYSDEVICE:) NTYnnnn:
```

To initialize and start the queue:

```
$ INITIALIZE/QUEUE/ON=NTYnnnn: queue-name
/PROCESSOR=MULTINET_NTYSMB/START
```

This example shows how to set up a print queue connected to an HP LaserJet printer with a JetDirect card:

```
$ NTYCP := $MULTINET:NTYCP
$ NTYCP CREATE PORT NTY1001/NODE=hp-laserjet/PORT=9100
%NTYCP-S-CREPORT, device _NTY1001: created to host 192.1.1.5, port 9100
$ SET TERMINAL/PERMANENT NTY1001:/NOBROADCAST/NOTYPEAHEAD/NOWRAP
$ INITIALIZE/QUEUE/ON=NTY1001:
HP_LASERJET/PROCESSOR=MULTINET_NTYSMB/START
```

Command Summary

The below table lists the NTYCP commands.

NTYCP Command	Description
Error! Reference source not found.	Creates a new network terminal port device.
Error! Reference source not found.	Deletes an NTY device created by the CREATE_PORT command.
Error! Reference source not found.	Exits from NTYCP to DCL command mode.
Error! Reference source not found.	Displays help text about NTYCP commands.
MODIFY PORT	Modifies (changes) an existing network terminal port device.

CREATE PORT

Creates a new network terminal port device. The device links the VMS terminal driver to a TCP/IP network connection directed to the destination address specified by the /HOST qualifier and the /PORT or /SERVICE qualifier.

You can use network terminal (NTY) devices with the MULTINET_NTYSMB print symbiont to provide VMS print queue support for network-connect printers. You can also use them with user-written applications that need a simple terminal-style I/O interface to a remote terminal, plotter, etc.

FORMAT

CREATE PORT *device-name*

Command Qualifiers	Defaults
/[NO]LOG /LOGICAL=(*logical-name-options...*) /NODE=*node-name-or-address* /PORT=*port-number* /SERVICE=*service-name*	/LOG

PARAMETER

device-name

Name of the NTY port device to be created. If specified, the device name must be of the form NTY*n*, where *n* is a device unit number, which must be in the range 1-9999. The specified device must not already exist. If omitted, the next available unit number will be used.

QUALIFIERS

/LOG
/NOLOG

Controls whether a log message is generated on successful completion of the command. The default is /LOG.

/LOGICAL=(*logical-name-options*...)

Causes NTYCP to create a logical name for the created NTY device. This qualifier takes one or more of the keyword options specified in the below table. If you specify multiple options, separate them by commas. You must have access to the specified logical name table and sufficient privilege to create the logical name.

Keyword	Description
NAME=*logical-name*	The logical name to be created. You must specify this option if you use /LOGICAL.
TABLE=*table-name*	Specifies the logical name table in which the logical name should be created. This can be the actual name of a table, or one of the key-words PROCESS, GROUP, SYSTEM. The default is TABLE=PROCESS.
MODE=*mode-name*	Specifies the access mode in which the logical name should be created. The keywords are EXECUTIVE, SUPERVISOR, USER. The default is MODE=SUPERVISOR.

/NODE=node-name-or-address
Specifies the name or numeric IP address of the remote node. You must specify this qualifier.

If you specify a node name, it is translated into an IP address. Only one IP address may be configured per NTY device. If the destination system is a host with multiple IP addresses, and not all addresses are directly reachable from your local system, you should specify the IP address numerically to ensure that a reachable address is configured.

/PORT=port-number
Specifies a TCP port number on the remote node to which the connection will be made. You must specify either the /PORT qualifier or the /SERVICE qualifier with the command.

/SERVICE=service-name
Specifies the name of a TCP service that translates to a port number to which the connection will be made. You may specify any TCP service name present in the local hosts/services table. You must specify either the /PORT qualifier or the /SERVICE qualifier with the command.

DELETE PORT

Deletes an NTY device created by the `NTYCP CREATE PORT` command.

FORMAT

`DELETE PORT` *`device-name`*

Command Qualifiers	Defaults
`/[NO]LOG`	`/LOG`

PARAMETER

`device-name`
Name of the NTY port device to be deleted.

QUALIFIER

`/LOG`
`/NOLOG`
Controls whether a log message is generated on successful completion of the command. The default is `/LOG`.

EXIT

Causes NTYCP to exit back to DCL command mode.

FORMAT

```
EXIT
```

HELP

Displays help text about NTYCP commands.

FORMAT

`HELP [topic]`

PARAMETER

topic

A command name or other topic in the NTYCP help library. If omitted, a list of topics displays.

MODIFY PORT

Modifies (changes) an existing network terminal port device. The device links the VMS terminal driver to a TCP/IP network connection directed to the destination address specified by the /HOST qualifier and the /PORT or /SERVICE qualifier.

FORMAT

MODIFY_PORT *device-name*

Command Qualifiers	Defaults
/[NO]LOG /LOGICAL=(*logical-name-options...*) /NODE=*node-name-or-address* /PORT=*port-number* /SERVICE=*service-name*	/LOG

PARAMETERS

modify-object
Name of the NTY object to be modified (changed).

port-name
Name of the NTY port device to be modified (changed).

QUALIFIERS

/LOG
/NOLOG
Controls whether a log message is generated on successful completion of the command. The default is /LOG.

/LOGICAL=(*logical-name-options...*)
This is the logical name for the NTY device you want to modify. This qualifier takes one or more of the keyword options specified in the below table. If you specify multiple options, separate them by commas.

You must have access to the specified logical name table and sufficient privilege to modify the logical name.

Keyword	Description
NAME=*logical-name*	The logical name to be created. You must specify this option if you use /LOGICAL.
TABLE=*table-name*	Specifies the logical name table in which the logical name should be created. This can be the actual name of a table, or one of the keywords PROCESS, GROUP, SYSTEM. The default is TABLE=PROCESS.
MODE=*mode-name*	Specifies the access mode in which the logical name should be created. The keywords are EXECUTIVE, SUPERVISOR, USER. The default is MODE=SUPERVISOR.

/NODE=*node-name-or-address*
Specifies the name or numeric IP address of the remote node. You must specify this qualifier.

If you specify a node name, it is translated into an IP address. Only one IP address may be configured per NTY device. If the destination system is a host with multiple IP addresses, and not all addresses are directly reachable from your local system, you should specify the IP address numerically to ensure that a reachable address is configured.

/PORT=*port-number*
Specifies a TCP port number on the remote node to which the connection will be made. You must specify either the /PORT qualifier or the /SERVICE qualifier with the command.

/SERVICE=*service-name*
Specifies the name of a TCP service that translates to a port number to which the connection will be made. You may specify any TCP service name present in the local hosts/services table. You must specify either the /PORT qualifier or the /SERVICE qualifier with the command.

7. PRINTER-CONFIG Command Reference

This chapter describes the commands you can run from the PRINTER-CONFIG command line. With PRINTER-CONFIG you can examine, modify, and save configuration files for MultiNet remote print queues.

To invoke PRINTER-CONFIG:

```
$ MULTINET CONFIGURE /PRINTERS
```

At the PRINTER-CONFIG prompt, type **?** to list the available commands. For online help use the PRINTER-CONFIG HELP command.

Changes do not take effect until you do one of the following:

- Restart the MultiNet remote printer queues with the @MULTINET:REMOTE-PRINTER-QUEUES.COM command.
- Restart your system.

For details on configuring MultiNet remote printer queues, refer to the *MultiNet Installation and Administrator's Guide*.

Command Summary

The below table lists the commands you can run from the PRINTER-CONFIG prompt.

PRINTER-CONFIG Command	Description
ADD	Adds a new VMS print queue to the current configuration.
ATTACH	Switches the terminal to another process.

CLEAR	Removes all printer queues from the current configuration.
DELETE	Removes a single printer queue from the current configuration.
ERASE	Removes all printer queues from the current configuration (same as CLEAR).
EXIT	Saves the current printer configuration and leaves PRINTER-CONFIG mode.
GET	Reads in a printer configuration file.
HELP	Displays information about one or all commands.
MODIFY	Changes a printer configuration file.
PUSH	Accesses the DCL command line and pauses PRINTER-CONFIG.
QUIT	Exits PRINTER-CONFIG and prompts to save changes.
SAVE	Writes out the current printer configuration file (same as WRITE).
SELECT	Picks the printer that will be modified by subsequent SET commands.
SET ALLOW-USER-SPECIFIED-PRINTER	Controls whether the print queue allows the use of PRINT/PARAMETER=(...) for specifying the destination address or printer for an LPD job (to override the original queue configuration).
SET BASE-PRIORITY	Specifies the base process priority at which jobs are initiated from a batch execution queue.

SET BLOCK-LIMIT-LOWER	Limits the size of print jobs that can be processed on an output execution queue.
SET BLOCK-LIMIT-UPPER	Limits the size of print jobs that can be processed on an output execution queue.
SET BURST	Controls whether two file flag pages with a burst bar between them are printed preceding output.
SET CHARACTERISTICS	Specifies one or more characteristics for processing jobs on an execution queue.
SET DEFAULT-FORM	Sets the default form used when submitting a print job to this printer.
SET DESCRIPTION	Specifies a string of up to 255 characters used to provide operator-supplied information about the queue.
SET FLAG	Forces a VMS banner page to print at the beginning of each file, by default, on the print queue.
SET LIBRARY	Sets the device control library for the print queue.
SET NOFEED	Prevents the VMS print symbiont formatting code from inserting a form feed between pages.
SET OWNER	Sets the owner of the print queue.
SET PROTECTION	Sets the protection of the print queue.
SET RETAIN-ON-ERROR	Retains jobs that terminate in an error in the queue.
SET SCHEDULE-NOSIZE	Prints jobs in the order they were submitted, regardless of size.
SET SEPARATE-BURST	Specifies whether two job flag pages with a burst bar between them are printed at the beginning of each job.

SET SEPARATE-FLAG	Specifies whether a job flag page is printed at the beginning of each job.
SET SEPARATE-RESET	Specifies one or more device control library modules that contain the job reset sequence for the queue.
SET SEPARATE-TRAILER	Specifies whether a job flag page is printed at the end of each job.
SET SUPPRESS-EOJ-FF	When set on a STREAM queue, the VMS print symbiont formatting code does not add a form feed to the end of the job.
SET SUPPRESS-REMOTE-BANNER	When set on an LPD queue, the remote LPD is informed not to print a banner page; many LPD servers do not support this option.
SET SUPPRESS-TELNET	When set on a STREAM queue, MultiNet does not try to use the TELNET protocol to communicate with the printer.
SET TAB-EXPAND	Forces the VMS print symbiont formatting code to expand TAB characters into the correct number of SPACE characters.
SET TRAILER	Controls whether a file trailer page is printed following output.
SET WS-DEFAULT	Defines for a batch job a working set default, the default number of physical pages that the job can use.
SET WS-EXTENT	Defines for the batch job a working set extent, the maximum amount of physical memory that the job can use.
SET WS-QUOTA	Defines for a batch job a working set quota, the amount of physical memory that is guaranteed to the job.

SHOW	Displays the current printer configuration.
SPAWN	Invokes a DCL command in PRINTER-CONFIG, or starts a subprocess.
STATUS	Displays the status of the printer configuration.
USE	Reads in a configuration file (same as GET).
VERSION	Displays the PRINTER-CONFIG version and release information.
WRITE	Writes out the current printer configuration file.

ADD

Adds a new VMS print queue to the current MultiNet configuration, and prompts for queue configuration parameters.

FORMAT

ADD *queue_name*

PARAMETER

queue_name
Specifies the name of the queue to add to the configuration.

EXAMPLES

This example adds a remote printer queue that prints via LPD on remote system 192.0.0.15.

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility V5.6(nn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>ADD SYS$LASER
[Adding new configuration entry for queue "SYS$LASER"]
Remote Host Name or IP address: 192.0.0.15
Protocol Type: [LPD] LPD
Remote Queue Name: [lp]: lp
[SYS$LASER => 192.0.0.15, lp]
PRINTER-CONFIG>
```

This example adds a remote printer queue that prints data by connecting to TCP port 1395 at address 192.0.0.98.

```
PRINTER-CONFIG>ADD SYS$LPTERM
[Adding new configuration entry for queue “SYS$LPTERM”]
Remote Host Name or IP address: 192.0.0.98
Protocol Type: [LPD] STREAM
TCP Port  Number: [23] 1395
[SYS$LPTERM => 192.0.0.98, TCP port 1395 (no telnet option negotiation)]
PRINTER-CONFIG>
```

Note: The Remote Queue Name specified may be case-sensitive. In particular, if the server is a UNIX system, you must specify it in the same case as it occurs in the UNIX `/etc/printcap` file, usually lowercase.

If the server is an Ethernet card in a printer, the name is not arbitrary. Check the Ethernet card documentation for the correct remote queue name.

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the `SPAWN SHOW PROCESS /SUBPROCESSES` command to list the names of subprocesses. Use the DCL `LOGOUT` command to return to the original process. `ATTACH` does not work if the `MULTINET_DISABLE_SPAWN` logical is enabled.

FORMAT

`ATTACH` *process-name*

PARAMETER

process-name
Specifies the name of a process to which you want your terminal attached. (Not all subprocesses can be attached; some testing may be required.)

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>SPAWN
$ MM
MM>SPAWN SHOW PROCESS /SUB
...
There are 3 processes in this job:
_TWA42:
 PROC_1
  PROC_2  (*)
MM>ATTACH  _TWA42:
PRINTER-CONFIG>ATTACH PROC_1
MM>QUIT
$ LOGOUT
PRINTER-CONFIG>
```

This example shows the use and exit of attached subprocesses.

1. The first command uses `SPAWN` to create a subprocess. `MM` is invoked from the DCL command line. Next, the `SPAWN SHOW PROCESS/SUB` command is used to list all the subprocess names. The display shows that three subprocesses are active. (`Process`

_TWA42: is PRINTER-CONFIG, PROC_1 is MM, and PROC_2 is the SPAWN SHOW PROCESS/SUB command.)

2. The MM>ATTACH command returns control to the PRINTER-CONFIG process. From this utility, ATTACH returns control to MM. To exit, QUIT is invoked from MM, and LOGOUT is invoked at the original spawned DCL command line; control returns back to PRINTER-CONFIG. (If SPAWN SHOW PROCESS /SUB had been entered, only this command and the configuration processes would be active.)

CLEAR

Removes all remote printer queues from the current MultiNet configuration.

FORMAT

```
CLEAR
```

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>CLEAR
```

DELETE

Removes the specified remote print queue from the current MultiNet configuration.

FORMAT

DELETE *queue_name*

PARAMETER

queue_name
Specifies the name of the remote print queue to remove.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>DELETE SYS$LASER
```

ERASE

Removes all printer queues from the current MultiNet configuration. (Functionally equivalent to `CLEAR`.)

EXIT

Saves the current configuration, if it has been modified, then quits.

FORMAT

```
EXIT
```

EXAMPLES

When the configuration has not changed, a message displays indicating that the configuration file is not updated.

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>EXIT
[Configuration not modified, so no update needed]
$
```

When the configuration has been changed, a message displays indicating that the configuration file is updated.

```
PRINTER-CONFIG>EXIT
[Writing configuration to MULTINET:REMOTE-PRINTER-QUEUES.COM]
$
```

GET

Reads in a MultiNet remote printer configuration file that defaults to MULTINET:REMOTE-PRINTER-QUEUES.COM.

After a GET, you can use the various configuration commands to modify the printer configuration.

FORMAT

GET *config-file*

PARAMETER

config-file
Specifies the name of the configuration file to read in.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>GET
[Reading in configuration from MULTINET_ROOT:[MULTINET]REMOTE-PRINTER-
QUEUES.COM.68]
```

HELP

Invokes the command help.

FORMAT

`HELP [topics]`

PARAMETER

topics

Contains a space-delimited list of topics that begins with a topic followed by subtopics. The default topic is `HELP`.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>HELP ?
ADD       ATTACH    CLEAR    DELETE    ERASE     EXIT      GET       HELP
MODIFY    PUSH      QUIT     SAVE      SHOW      SPAWN     STATUS    USE
VERSION   WRITE
PRINTER-CONFIG>
```

MODIFY

Changes the parameters of the specified queue in the MultiNet remote printer configuration.

FORMAT

MODIFY *queue_name*

PARAMETER

queue_name
Specifies the name of the queue whose parameters you want to change.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>MODIFY REMOTE_LASER
[Modifying configuration entry for queue "REMOTE_LASER"]
Remote Host Name: [192.0.0.1] 192.0.0.2
Protocol Type: [LPD] RETURN
Remote Queue Name: [LASER] RETURN
[REMOTE_LASER => 192.0.0.2, LASER]
PRINTER-CONFIG>
```

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command.

PUSH does not work if the MULTINET_DISABLE_SPAWN logical is set.

FORMAT

PUSH

QUIT

If the configuration file has been edited, `QUIT` prompts you to save the file before leaving.

FORMAT

```
QUIT
```

SAVE

Functionally equivalent to `WRITE`.

SELECT

Picks the printer that will be modified by any subsequent `SET` commands.

FORMAT

`SELECT` *printer*

PARAMETER

printer
Specifies the name of the printer to pick for modification.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>
```

SET ALLOW-USER-SPECIFIED-PRINTER

Controls whether the print queue allows the use of `PRINT/PARAMETER=(...)` for specifying the destination address and/or printer for an LPD job (to override the original queue configuration).

FORMAT

```
SET ALLOW-USER-SPECIFIED-PRINTER {enable | disable}
```

PARAMETER

{enable | disable}
Specifies whether this function is enabled or disabled.

EXAMPLE

This example shows how to enable the use of `PRINT/PARAMETER=(...)` to override the original queue configuration.

```
$ MULTINET CONFIGURE /PRINTER
MultiNet Remote Printer Configuration Utility V5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SEL HP-PRINT
[The Selected Printer is now HP-PRINT]
PRINTER-CONFIG>SET ALLOW-USER-SPECIFIED-PRINTER ENABLE
```

SET BASE-PRIORITY

Establishes the base priority of the symbiont process when the symbiont process is created.

By default, if you omit this, the symbiont process is initiated at the same priority as the base priority established by `DEFPRI` at system generation (usually 4).

FORMAT

`SET BASE-PRIORITY` *priority*

PARAMETER

priority
Specifies the base priority in decimal format, 0 to 15.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET BASE-PRIORITY 4
PRINTER-CONFIG>
```

SET BLOCK-LIMIT-LOWER

Limits the size of print jobs that can be processed on the queue. Allows you to reserve certain printers for certain size jobs. You can set the lower block limit only if the upper block limit is also set (see SET BLOCK-LIMIT-UPPER).

FORMAT

SET BLOCK-LIMIT-LOWER *lowlim*

PARAMETER

lowlim

The *lowlim* parameter is a decimal number referring to the minimum number of blocks accepted by the queue for a print job. If a print job is submitted that contains fewer blocks than the *lowlim* value, the job remains pending until the block limit for the queue is changed. After the block limit for the queue is decreased sufficiently, the job is processed.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET BLOCK-LIMIT-LOWER 25
PRINTER-CONFIG>
```

SET BLOCK-LIMIT-UPPER

Limits the size of print jobs that can be processed on the queue. Allows you to reserve certain printers for certain size jobs.

FORMAT

SET BLOCK-LIMIT-UPPER *uplim*

PARAMETER

uplim

The *uplim* parameter is a decimal number referring to the maximum number of blocks that the queue accepts for a print job. If a print job is submitted that exceeds this value, the job remains pending until the block limit for the queue is changed. After the block limit for the queue is increased sufficiently, the job is processed.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET BLOCK-LIMIT-UPPER 300
PRINTER-CONFIG>
```

SET BURST

Controls whether two file flag pages with a burst bar between them are printed preceding output.

FORMAT

`SET BURST` *keyword*

PARAMETER

keyword

If you specify the keyword...	Description
`ALL` (default)	These flag pages are printed before each fill in the job.
`ONE`	These flag pages are printed once before the first file in the job.
`NONE`	No flag pages are printed. It is equivalent to `NOBURST`.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET BURST ONE
PRINTER-CONFIG>
```

SET CHARACTERISTICS

Specifies one or more characteristics for processing print jobs. If a queue does not have all the characteristics that have been specified for a job, the job remains pending. Only the characteristics specified are established for the queue.

FORMAT

`SET CHARACTERISTICS` *`characteristic,...`*

PARAMETER

`characteristics,...`

Queue characteristics are installation specific. The characteristic parameter can be either a value from 0 to 127 or a characteristic name that has been defined by the `DEFINE /CHARACTERISTIC` command.

Parenthesis are not required; they are added automatically.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET CHARACTERISTICS 56
PRINTER-CONFIG>
```

SET DEFAULT-FORM

Specifies the default form used when submitting a print job to the printer.

FORMAT

`SET DEFAULT-FORM` *formname*

PARAMETER

formname

Specifies the name of a form previously defined on the system with the `DEFINE /FORM` command.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET DEFAULT-FORM POSTSCRIPT
[Default Form POSTSCRIPT]
PRINTER-CONFIG>
```

SET DESCRIPTION

Specifies a string of up to 255 characters used to provide operator-supplied information about the queue.

FORMAT

SET DESCRIPTION *string*

PARAMETER

string

Sequence of any printable characters, including spaces. Case of input is preserved. The string may optionally be enclosed in quotation marks (" ").

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET DESCRIPTION THIS IS A TEST QUEUE.
PRINTER-CONFIG>
```

SET FLAG

By default, forces a VMS banner page to print at the beginning of each file on the print queue.

FORMAT

`SET FLAG` *`mode`*

PARAMETER

mode

If mode is `ENABLE`, banner pages are printed; if `DISABLE`, banner pages are not printed.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET FLAG ENABLE
PRINTER-CONFIG>
```

SET LIBRARY

Sets the print queue's device control library.

FORMAT

`SET LIBRARY` *`libraryfile`*

PARAMETER

`libraryfile`

Specifies the name of a text library located in `SYS$LIBRARY` to be used as the device control library for the print queue.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET LIBRARY SYSDEVCTL
[LIBRARY SYSDEVCTL]
PRINTER-CONFIG>
```

SET NOFEED

Prevents the VMS print symbiont formatting code from inserting a form feed between pages.

FORMAT

`SET NOFEED` *mode*

PARAMETER

mode

If mode is `ENABLE`, form feeds are not inserted; if `DISABLE`, they are inserted.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET NOFEED ENABLE
PRINTER-CONFIG>
```

SET OWNER

Sets the owner of the print queue.

FORMAT

SET OWNER *owner*

PARAMETER

owner
Specifies the identifier or UIC of a user on the system.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET OWNER OPERATOR
[OWNER OPERATOR]
PRINTER-CONFIG>
```

SET PROTECTION

Sets the protection of the print queue.

FORMAT

SET PROTECTION *protection_string*

PARAMETER

protection_string
Specifies a VMS queue protection mask.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET PROTECTION (S:RWED,O:REW,G:RE,W:RE)
[PROTECTION (S:RWED,O:REW,G:RE,W:RE)]
PRINTER-CONFIG>
```

SET RETAIN-ON-ERROR

Retains jobs in the queue that terminate in an error.

FORMAT

SET RETAIN-ON-ERROR *mode*

PARAMETER

mode

If *mode* is ENABLE, jobs are retained; if DISABLE, jobs are not retained.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET RETAIN-ON-ERROR ENABLE
PRINTER-CONFIG>
```

SET SCHEDULE-NOSIZE

Prints jobs in the order they were submitted, regardless of size.

FORMAT

`SET SCHEDULE-NOSIZE` *mode*

PARAMETER

mode

If *mode* is ENABLE, jobs will print in the order they are submitted; if *mode* is DISABLE, jobs will print in order by size (shorter prints before longer).

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SCHEDULE-NOSIZE ENABLE
PRINTER-CONFIG>
```

SET SEPARATE-BURST

Specifies whether two job flag pages with a burst bar between them are printed at the beginning of each job.

FORMAT

`SET SEPARATE-BURST` *mode*

PARAMETER

mode

If *mode* is `ENABLE`, prints the flag pages; if `DISABLE`, will not print the flag pages.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SEPARATE-BURST ENABLE
PRINTER-CONFIG>
```

SET SEPARATE-FLAG

Specifies whether a job flag page is printed at the beginning of each job.

FORMAT

`SET SEPARATE-FLAG` *`mode`*

PARAMETER

mode

If *`mode`* is `ENABLE`, job flag page will print; if it is `DISABLE`, job flag page will not print.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SEPARATE-FLAG ENABLE
PRINTER-CONFIG>
```

SET SEPARATE-RESET

Specifies one or more device control library modules that contain the job reset sequence for the queue. The specified modules from the queue's device control library (by default SYS$LIBRARY:SYSDEVCTL) are used to reset the device at the end of each job. The RESET sequence occurs after any file trailer and before any job trailer. Thus, all job separation pages are printed when the device is in its RESET state.

FORMAT

SET SEPARATE-RESET *module,...*

PARAMETER

module
This is the name of the device control library module.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SEPARATE-RESET cosmos
PRINTER-CONFIG>
```

SET SEPARATE-TRAILER

Specifies whether a job flag page is printed at the end of each job.

FORMAT

`SET SEPARATE-TRAILER` *mode*

PARAMETER

mode

If *mode* is `ENABLE`, job flag page will print; if it is `DISABLE`, job flag page will not print.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SEPARATE-TRAILER ENABLE
PRINTER-CONFIG>
```

SET SUPPRESS-EOJ-FF

When set on a STREAM queue, the VMS print symbiont formatting code does not add a form feed to the end of the job.

FORMAT

`SET SUPPRESS-EOJ-FF` *mode*

PARAMETER

mode

If *mode* is `ENABLE`, a form feed is not inserted at the end of each job; if `DISABLE`, a form feed is inserted at the end of each job.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6(nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SUPPRESS-EOJ-FF ENABLE
PRINTER-CONFIG>
```

SET SUPPRESS-REMOTE-BANNER

When set on an LPD queue, the remote LPD does not print a banner page. (Many LPD servers do not support this option.)

FORMAT

```
SET SUPPRESS-REMOTE-BANNER mode
```

PARAMETER

mode

If *mode* is ENABLE, banner pages may or may not be generated on the remote system; if DISABLE, banner pages are generated on the remote system.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SUPPRESS-REMOTE-BANNER ENABLE
PRINTER-CONFIG>
```

SET SUPPRESS-TELNET

When set on a STREAM queue, MultiNet does not try to use the TELNET protocol to negotiate options with the remote printer. Most terminal servers expect MultiNet to negotiate TELNET options, and most printers that connect directly to an IP network expect MultiNet not to do so.

FORMAT

`SET SUPPRESS-TELNET` *`mode`*

PARAMETER

mode

If *`mode`* is `ENABLE`, TELNET options are not negotiated; if `DISABLE`, TELNET options are negotiated.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET SUPPRESS-TELNET ENABLE
PRINTER-CONFIG>
```

SET TAB-EXPAND

Forces the VMS print symbiont formatting code to expand TAB characters into the correct number of SPACE characters.

FORMAT

`SET TAB-EXPAND` *mode*

PARAMETER

mode

If *mode* is `ENABLE`, tabs are converted to SPACES; if `DISABLE`, tabs are not changed.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET EXPAND-TAB ENABLE
PRINTER-CONFIG>
```

SET TRAILER

Controls whether a file trailer page is printed following output.

FORMAT

`SET TRAILER` *keyword*

PARAMETER

keyword

If you specify the keyword...	Description
`ALL` (default)	These flag pages are printed before each fill in the job.
`ONE`	These flag pages are printed once before the first file in the job.
`NONE`	No flag pages are printed. It is equivalent to `NOBURST`.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET TRAILER ALL
PRINTER-CONFIG>
```

SET WS-DEFAULT

Establishes the working set default of the symbiont process for the queue when the symbiont process is created.

The value set by this command overrides the value defined in the user authorization file (UAF) of any user submitting a job to the queue.

FORMAT

`SET WS-DEFAULT` *`quota`*

PARAMETER

quota
Specify the value as a number of 512-byte pagelets on Alpha systems or 512-byte pages on VAX.

> **Note:** OpenVMS rounds this value up to the nearest CPU-specific page so that actual amount of physical memory allowed may be larger than the specified amount on Alpha. For further information, see the *OpenVMS System Manager's Manual*.

If you specify `0` or `NONE`, the working set default value defaults to the value specified in the UAF or by the `SUBMIT` command (if it includes a WSDEFAULT value).

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET WS-DEFAULT 27
PRINTER-CONFIG>
```

SET WS-EXTENT

Establishes the working set extent of the symbiont process for the queue when the symbiont process is created.

The value set by this command overrides the value defined in the user authorization file (UAF) of any user submitting a job to the queue.

FORMAT

`SET WS-EXTENT` *quota*

PARAMETER

quota
Specify the value as a number of 512-byte pagelets on Alpha or and 512-byte pages on VAX.

> **Note:** OpenVMS rounds this value up to the nearest CPU-specific page so that actual amount of physical memory allowed may be larger than the specified amount on Alpha.

If you specify `0` or `NONE`, the working set extent value defaults to the value specified in the UAF or by the `SUBMIT` command (if it includes a WSEXTENT value).

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET WS-EXTENT 0
PRINTER-CONFIG>
```

SET WS-QUOTA

Establishes the working set quota of the symbiont process for the queue when the symbiont process is created. The value set by this command overrides the value defined in the user authorization file (UAF) of any user submitting a job to the queue.

FORMAT

`SET WS-QUOTA` *quota*

PARAMETER

quota

Specify the value as a number of 512-byte pagelets on OpenVMS Alpha or 512-byte pages on OpenVMS VAX. OpenVMS rounds this value up to the nearest CPU-specific page so that actual amount of physical memory allowed may be larger than the specified amount on OpenVMS Alpha. For further information, see the *OpenVMS System Manager's Manual*.

If you specify `0` or `NONE`, the working set quota value defaults to the value specified in the UAF or by the `SUBMIT` command (if it includes a WSQUOTA value).

Working set default, working set quota, and working set extent values are included in each user record in the system UAF. You can specify working set values for individual jobs or for all jobs in a given queue. The decision table shows the action taken for different combinations of specifications that involve working set values.

Is the SUBMIT command value specified?	Is the queue value specified?	Action taken
No	No	Use the UAF value.
No	Yes	Use value for the queue.
Yes	Yes	Use smaller of the two values.
Yes	No	Compare specified value with UAF value; use the smaller.

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:REMOTE-PRINTER-QUEUES.COM]
PRINTER-CONFIG>SELECT TEST
[The Selected Printer is now TEST]
PRINTER-CONFIG>SET WS-QUOTA 12
PRINTER-CONFIG>
```

SHOW

Displays the current MultiNet printer configuration.

FORMAT

```
SHOW
```

EXAMPLES

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>SHOW
Queue Name          IP Destination        Remote Queue Name
----------          --------------        -----------------
SYS$LASER           192.0.0.15            laser
SYS$LPTERM          192.0.0.98            TCP port 1395
PRINTER-CONFIG>
```

This example shows detailed queue characteristics for a specific printer called HP5.

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>SHOW HP5
Queue Name          IP Destination        Remote Queue Name
----------          --------------        -----------------
HP5                 192.0.0.9             TCP PORT 9100
Device Control Library = HPLF3SI
Queue Owner = [SUPPORT,*]
Default Form = WHITEPAPER
End of Job Form Feed will be suppressed
Telnet Options Processing will be suppressed
PRINTER-CONFIG>
```

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. SPAWN does not work if the MULTINET_DISABLE_SPAWN logical is set.

FORMAT

SPAWN [*command*]

PARAMETER

command
Specifies a command to execute. If you omit *command*, a DCL command line subprocess is created.

QUALIFIERS

/INPUT=*file-spec*
Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES
/NOLOGICAL_NAMES
Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS
/NOSYMBOLS
Specifies that global and local names are not passed to the subprocess.

/WAIT
/NOWAIT
Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=*file-spec*
Specifies a file that retains the output of the command invoked with SPAWN. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after SPAWN or other qualifiers.

EXAMPLES

This example displays terminal information, captures the output in a file, then displays the information with the TYPE command.

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>SPAWN/OUTPUT=FOO. SHOW TERM
PRINTER-CONFIG>SPAWN TYPE FOO.
...
```

This example invokes a command procedure.

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>SPAWN @COMPROC
...
```

This example displays help information about the PRINTER-CONFIG utility. Use the LOGOUT command to return control to PRINTER-CONFIG.

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>SPAWN
$ HELP MULTINET CONFIGURE /PRINTERS
...
$ LOGOUT
PRINTER-CONFIG>
```

STATUS

Shows the status of the MultiNet remote printer configuration program.

FORMAT

```
STATUS
```

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6 (nnn)
PRINTER-CONFIG>STATUS
This is the MultiNet Remote Printer configuration program Version Example
There are 1/1000 queues in the current configuration.
The configuration MULTINET:REMOTE-PRINTER-QUEUES.COM is not modified.
PRINTER-CONFIG>
```

USE

Functionally equivalent to `GET`.

VERSION

Shows the version and release information of the MultiNet remote printer configuration program.

FORMAT

```
VERSION
```

EXAMPLE

```
$ MULTINET CONFIGURE /PRINTERS
MultiNet Remote Printer Configuration Utility 5.6(nnn)
PRINTER-CONFIG>VERSION
This is the MultiNet Remote Printer configuration program Version 5.6(nnn)
PRINTER-CONFIG>
```

WRITE

Writes out the current MultiNet remote printer configuration to a MultiNet remote printer configuration file.

FORMAT

`WRITE` *`config_file`*

PARAMETER

`config_file`
Specifies the name of the file to which to write the current MultiNet printer configuration (by default, the same file from which the configuration was read).

8. SERVER-CONFIG Command Reference

This chapter describes the commands you can run from the SERVER-CONFIG command line. SERVER-CONFIG lets you examine, modify, and save configuration files for MultiNet services.

To invoke SERVER-CONFIG:

```
$ MULTINET CONFIGURE /SERVERS
```

SERVER-CONFIG commands affect the configuration of the currently selected service. You can select services with the `SELECT` command. By default, no service is selected.

At any SERVER-CONFIG prompt, type **?** to list the available commands. Use the SERVER-CONFIG `HELP` command to view online help for each SERVER-CONFIG command.

Changes do not take effect until you do one of the following:

- Reload and restart the MultiNet server process with the `MULTINET NETCONTROL` command.
- Restart your system.

For details on configuring MultiNet services, refer to the *MultiNet Installation and Administrator's Guide*.

Command Summary

The below table lists the commands you can run from the SERVER-CONFIG prompt.

SERVER-CONFIG Command	Description
`ADD`	Adds a service to the current configuration.
`ATTACH`	Switches terminal control to another process.
`COPY`	Copies a service entry to the current configuration.

DELETE	Deletes a service from the current configuration.
DISABLE	Disables a service in the current configuration.
ENABLE	Enables a service in the current configuration.
EXIT	Exits from the SERVER-CONFIG session.
Error! Reference source not found.	Reads a server configuration file; same as GET.
HELP	Displays command information.
NETCONTROL	Contacts the NETCONTROL server at another site.
PUSH	Accesses the DCL command line while pausing SERVER-CONFIG.
QUIT	Exits SERVER-CONFIG and prompts to save changes.
RESTART	Restarts the master server process.
SAVE	Writes out the current server configuration file.
SELECT	Selects a server for SET commands.
SET ACCEPT-HOSTS	Specifies which hosts can access the server.
SET ACCEPT-NETS	Specifies which networks can access the server.
SET BACKLOG	Specifies the server connection queue limits.
SET CONNECTED	Specifies the connection-request-received routine.
SET DISABLED-NODES	Specifies which VMScluster nodes cannot execute the service.
SET ENABLED-NODES	Specifies which VMScluster nodes can execute the service.

SET FLAGS	Specifies the flag bit mask for service operation control.
SET INIT	Specifies the initialize-service routine.
SET KEEPALIVE-TIMERS	Sets keep alive timers for a service.
SET LISTEN	Specifies the listen-for-connections routine.
SET LOG-ACCEPTS	Enables/disables successful connections logging.
SET LOG-FILE	Specifies the log message destination.
SET LOG-REJECTS	Enables/disables failed connections logging.
SET MAX-SERVERS	Specifies the service process limit.
SET PARAMETERS	Specifies service-dependent parameters.
SET PRIORITY	Specifies a VMS priority for the created processes.
SET PROCESS	Specifies that the service is to run in an auxiliary master server process rather than in the main master server process.
SET PROGRAM	Specifies a VMS file name for run or merged images.
SET REJECT-BY-DEFAULT	Enables/disables conditional connection rejection.
SET RECEIVE-BUFFER-SPACE**Error! Reference source not found.**	Specifies the size of the receive socket buffers.
SET REJECT-HOSTS	Specifies which hosts are not allowed service access.
SET REJECT-MESSAGE	Specifies a rejected connection message.
SET REJECT-NETS	Specifies which networks are not allowed service access.
SET SEND-BUFFER-SPACE	Specifies the size of the send socket buffers.

SET SERVICE	Specifies the perform-service routine.
SET SERVICE-NAME	Changes the service name.
SET SERVICE-TYPE	Sets the service type advertised for a particular service.
SET SOCKET-FAMILY	Specifies the service family address.
SET SOCKET-OPTIONS	Specifies the `setsockopt()` options.
SET SOCKET-PORT	Specifies the port for connection listening.
SET SOCKET-TYPE	Specifies the socket type.
SET USERNAME	Specifies the user name under which the selected service is started.
SET WORKING-SET-EXTENT	Specifies how much memory the process will be allowed to use if there are free pages available.
SET WORKING-SET-QUOTA	Specifies the maximum amount of memory the process can lock into its working set.
SHOW	Shows the current server configuration.
SHUTDOWN	Stops the master server process.
SPAWN	Invokes the DCL command or creates a subprocess.
STATUS	Shows the SERVER-CONFIG service status.
USE	Reads a server configuration file; same as `GET`.
VERSION	Shows the SERVER-CONFIG version.
WRITE	Writes the current server configuration; same as `SAVE`.

ADD

Adds a new service to the current server configuration and prompts you for an initial set of parameters for the service.

FORMAT

ADD *service*

PARAMETER

service
Specifies the name of the service to add to the configuration.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>ADD NNTP
[Adding new configuration entry for service "NNTP"]
Protocol: [TCP]  TCP
TCP Port number: 119
Program to run: USER$DISK:[NNTP]NNTP_SERVER.EXE
[Added service NNTP to configuration]
[Selected service is now NNTP]
SERVER-CONFIG>
```

ATTACH

Detaches the terminal from the calling process and reattaches it to another process. Use the SPAWN SHOW PROCESS /SUBPROCESSES command to list the name of subprocesses. Use the DCL LOGOUT command to return to the original process. If the MULTINET_DISABLE_SPAWN logical is enabled, ATTACH does not work.

FORMAT

ATTACH *process-name*

PARAMETER

process-name
Specifies the name of a process to which you want your terminal attached. (Not all subprocesses can be attached; some testing may be required.)

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SPAWN
$ MM
MM>SPAWN SHOW PROCESS /SUB
...
There are 3 processes in this job:
_TWA42:
 PROC_1
  PROC_2 (*)
MM>ATTACH _TWA42:
SERVER-CONFIG>ATTACH PROC_1
MM>QUIT
$ LOGOUT
SERVER-CONFIG>
```

This example shows the use and exit of attached subprocesses.

1. The first command uses SPAWN to create a subprocess. MM is invoked from the DCL command line. Next, the SPAWN SHOW PROCESS/SUB command is used to list all the subprocess names. The display shows that three subprocesses are active. (Process

_TWA42: is SERVER-CONFIG, PROC_1 is MM, and PROC_2 is the SPAWN SHOW PROCESS/SUB command.)

2. In the next command, the MM ATTACH command returns control to the SERVER-CONFIG process. From this utility, ATTACH returns control to MM. To exit, QUIT is invoked from MM, and LOGOUT is entered at the original spawned DCL command line; finally control returns to SERVER-CONFIG. (If SPAWN SHOW PROCESS/SUB had been entered, only this command and the configuration processes would be active.)

COPY

Copies a given service entry. When a service is copied, the copy is disabled automatically. Enable the copy after changing any conflicting parameters, such as the port number.

FORMAT

COPY *input-service output-service*

PARAMETERS

input-service
Specifies the name of the service to duplicate.

output-service
Specifies the name of the service to create.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>COPY FINGER LOCALFINGER
[Adding service LOCALFINGER to configuration]
[Disabling service LOCALFINGER]
SERVER-CONFIG>
```

DELETE

Deletes a given service from the current configuration. Once a service is deleted, all information about that service is removed. See the `DISABLE` command for disabling a service.

FORMAT

`DELETE` *`service`*

PARAMETER

`service`
Specifies the name of the service to delete from the configuration.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>DELETE NNTP
SERVER-CONFIG>
```

DISABLE

Disables (removes) a given service from the current configuration. Unlike the DELETE command, all information about the service is retained and the service can be re-enabled at any time after a DISABLE.

FORMAT

DISABLE *service*

PARAMETER

service
Specifies the name of the service to disable from the configuration.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>DISABLE NNTP
SERVER-CONFIG>
```

ENABLE

Enables a previously disabled service. (See `DISABLE` for information on disabling a service.)

FORMAT

`ENABLE` *service*

PARAMETER

service
Specifies the name of the service to enable in the configuration.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>ENABLE NNTP
SERVER-CONFIG>
```

EXIT

Saves the current configuration, if it has been modified, then quits.

FORMAT

```
EXIT
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>ENABLE NNTP
SERVER-CONFIG>EXIT
[Writing configuration to MULTINET_COMMON_ROOT:[MULTINET]
SERVICES.MASTER_SERVER]
$
```

GET

Reads in a MultiNet server configuration file. After a `GET`, you can use the various configuration commands to modify this server configuration. (Functionally equivalent to `USE`.)

FORMAT

`GET` *`config_file`*

PARAMETER

`config_file`
Specifies the name of the server configuration file to read in.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>GET ST_TMP:FOO.CONFIGURATION
[Reading in configuration from ST_ROO:[TMP]FOO.CONFIGURATION.1]
SERVER-CONFIG>
```

HELP

Invokes command help.

FORMAT

`HELP [topics]`

PARAMETER

topics

Contains a space-delimited list of topics that begins with a topic followed by subtopics. The default topic is `HELP`.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>HELP  ?
ADD         ATTACH    COPY        DELETE      DISABLE     ENABLE
EXIT        GET       HELP        NETCONTROL  PUSH        QUIT
RESTART     SAVE      SELECT      SET         SHOW        SHUTDOWN
SPAWN       STATUS    USE         VERSION     WRITE
SERVER-CONFIG>
```

NETCONTROL

Transfers control to a configuration manager subsystem that contacts the NETCONTROL server at local or remote sites.

After invoking `NETCONTROL`, you can issue commands to the NETCONTROL server to affect MULTINET_SERVER operations at that site.

FORMAT

`NETCONTROL [`*`host`*`]`

RESTRICTION

The NETCONTROL server is usually protected from unauthorized access by a restriction list.

PARAMETER

`host`
Specifies the name of the host to which to connect. If not specified, the default is the local host.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>NETCONTROL
Connected to NETCONROL server on "127.0.0.1"
<EXAMPLE.COM Network Control 5.6 (nnn) at Mon 15-Mar-2004 7:42am-EST
NETCONTROL>
```

PUSH

Starts and attaches a DCL subprocess. If a parent process exists, attach to it. To return from DCL, use the ATTACH or the LOGOUT command. To switch back from a DCL subprocess, use the ATTACH command.

If the MULTINET_DISABLE_SPAWN logical is set, PUSH does not work.

FORMAT

```
PUSH
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>PUSH
$ LOGOUT
Process foobar_1 logged out at 16-Jun-2019 16:36:22.13
SERVER-CONFIG>
```

QUIT

If the configuration file has been edited, QUIT prompts you to save the file before quitting.

FORMAT

QUIT

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>QUIT
$
```

RESTART

Kills the old master server (MULTINET_SERVER) process and starts a new one. Any connections in progress are not interrupted. If the configuration has been modified since the last save, RESTART prompts you to save the configuration before restarting.

FORMAT

RESTART

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>RESTART
%RUN-S-PROC_ID, identification of created process is 2060005c
SERVER-CONFIG>
```

SAVE

Writes the current MultiNet server configuration to a server configuration file. (Functionally equivalent to WRITE.)

FORMAT

SAVE *config_file*

PARAMETER

config_file
Specifies the name of the file to which to write the current MultiNet server configuration (by default, the same file from which the configuration was read).

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SAVE
[Writing configuration to
MULTINET_COMMON_ROOT:[MULTINET]SERVICES.MASTER_SERVER.1103]
SERVER-CONFIG>
```

SELECT

Selects which service will be modified by any subsequent `SET` commands.

FORMAT

`SELECT` *service*

PARAMETER

service
Specifies the name of the service to select for modification.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT NNTP
[The Selected SERVER entry is not NNTP]
SERVER-CONFIG>
```

SET ACCEPT-HOSTS

Specifies and maintains a list of hosts allowed access to the service. For IPv6 services IPv6 addresses and IPv4 addresses can be used. IPv4 addresses that are specified for an IPv6 service are treated as IPv4 mapped addresses.

FORMAT

```
SET ACCEPT-HOSTS
```

EXAMPLE

This example shows how to delete host 192.0.0.1 from the accept-hosts list, and add host 192.0.0.4.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET ACCEPT-HOSTS
Delete address "IP-192.0.0.1" ? [NO] Y
[Address "IP-192.0.0.1" deleted from TELNET]
Delete address "IP-192.0.0.2" ? [NO] RETURN
Delete address "IP-192.0.0.3" ? [NO] RETURN
You can now add new addresses for TELNET. An empty line terminates.
Add Address: 192.0.0.4
Add Address: RETURN
SERVER-CONFIG>
```

SET ACCEPT-NETS

Invokes an interactive utility that prompts you for the addresses of networks that are allowed access to the selected service.

Specify each network as follows:

IP_address [*subnetmask*]

When done, press **RETURN** at the `Add:` prompt.

For more information about restricting access to services, see the *MultiNet Installation and Administrator's Guide*.

FORMAT

```
SET ACCEPT-NETS
```

EXAMPLE

This example shows how to delete network address 192.0.0.0 from the accept-nets list, and add network address 128.1.0.0.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET ACCEPT-NETS
Delete address "IP-192.0.0.0" ? [NO] Y
[Address "IP-192.0.0.0" deleted from TELNET]
Delete address "IP-192.12.19.0" ? [NO] RETURN
You can now add new addresses for TELNET. An empty line terminates.
Add Address: 128.1.0.0
Add Address:
SERVER-CONFIG>
```

SET BACKLOG

Specifies the number of server connections to queue up before refusing to accept additional connections when MAX-SERVERS is reached.

FORMAT

SET BACKLOG *backlog*

PARAMETER

backlog
Specifies the number of connections to queue - but not process - while waiting for connections that are already running to exit.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET BACKLOG 5
[Backlog of TELNET set to 5]
SERVER-CONFIG>
```

SET CONNECTED

Specifies the name of the internal MULTINET_SERVER routine to call when a connection request is received.

FORMAT

```
                [none]
                [tcp_connected]
                [udp_chargen]
                [udp_connected]
SET CONNECTED   [udp_connected_single]
                [udp_daytime]
                [udp_discard]
                [udp_echo]
                [udp_time]
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET CONNECTED UDP_CONNECTED_SINGLE
[Connected action of TELNET set to UDP_Connected_Single]
SERVER-CONFIG>
```

SET DISABLED-NODES

Specifies and maintains a list of VMScluster nodes that cannot execute the service.

The master server (`MULTINET_SERVER`) can be tailored to enable or disable services on a per-node basis in a VMScluster. Use the `SET ENABLED-NODES` or `SET DISABLED-NODES` commands to specify a list of VMScluster nodes on which the service runs or does not run.

> **Note:** The service must also be enabled via the `ENABLE` command.

> **Note:** When entering the nodes to be disabled, use only the VMScluster node name or the DECnet node name; do not use the IP address.

FORMAT

```
SET DISABLED-NODES
```

EXAMPLE

This example shows how to delete the node FLEET from the disabled-nodes list, and add the node DRAGO.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET DISABLED-NODES
Delete VMScluster node "FLEET" ? [NO] Y
[Node "FLEET" deleted from TELNET]
Delete VMScluster node "NINET9" ? [NO] RETURN
You can now add new VMScluster nodes for TELNET. An empty line terminates.
Add VMScluster node: DRAGO
Add Address: RETURN
SERVER-CONFIG>
```

SET ENABLED-NODES

Specifies and maintains a list of VMScluster nodes that can execute the service.

The master server (`MULTINET_SERVER`) can be tailored to enable or disable services on a per-node basis in a VMScluster. Use the `SET ENABLED-NODES` or `SET DISABLED-NODES` commands to specify a list of VMScluster nodes on which the service runs or does not run.

Note: The service must also be enabled via the `ENABLE` command.

Note: When entering the nodes to be enabled, use only the VMScluster node name or the DECnet node name; do not use the IP address.

Note: The service must also be enabled via the `ENABLE` command.

Note: When entering the nodes to be disabled, use only the VMScluster node name or the DECnet node name; do not use the IP address.

FORMAT

```
SET ENABLED-NODES
```

EXAMPLE

This example shows how to delete the node DRAGO from the enabled-nodes list, and add the node FLEET.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET ENABLED-NODES
Delete VMScluster node "DRAGO" ? [NO] Y
[Node "DRAGO" deleted from TELNET]
```

```
Delete VMScluster node "NINET9" ? [NO] RETURN
You can now add new VMScluster nodes for TELNET. An empty line terminates.
Add VMScluster node: FLEET
Add Address: RETURN
SERVER-CONFIG>
```

SET FLAGS

Specifies a bit mask of flags that control the operation of the service. If you do not specify a flag for this command, all existing flags are cleared.

FORMAT

```
SET FLAGS [flag1 | flag2, ...]
```

FLAGS

EUNICE_SERVER

Indicates the connection is not closed until the server process exits, instead of when the server process closes the connection. This flag is required for servers compiled under EUNICE, as the EUNICE runtime opens and closes the channel to the connection many times.

EXOS_SERVER

Indicates that the MULTINET_SERVER process passes the connection to a new process using the EXOS conventions.

MANUAL

Indicates that the MULTINET_SERVER process does not accept a CHAOSnet connection; instead, the created process performs the accept in a nonstandard way.

NO_TELNET_NEGOTIATIONS

Indicates that TELNET option negotiations do not take place. This flag is used with services that use the `Internal_Telnet INIT()` routine.

SNMP_MONITORED

Tells the service handling routines to connect to SNMP Agent X and provide information about the state of the service. You need to define `MULTINET_SNMP_AGENTX 1` and add `AGENTX_PEER 127.0.0.1` in the `SNMPD.CONF` file. The values returned when `SNMP_MONITORED` flag is set are the Network Services Monitoring MIB (RFC2788). See the SNMP chapter in the *MultiNet Installation and Administrator's Guide* for an example of the Network Services Monitoring MIB. The following values are displayed within `enterprises.105.4.service_port`:

1. Service name
2. Service port number
3. Maximum servers allowed
4. Number of active servers
5. Number of times the service processing code has been called
6. Last I/O status value for the service (generally 0 or 1)

START_AUX_SERVER
Causes the main master server to start the auxiliary master server process automatically.

UCX_SERVER
Configures services written for HP TCP/IP Services for OpenVMS to work with MultiNet.

EXAMPLES

This example sets the EUNICE_SERVER flag.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET FLAGS EUNICE_SERVER
[TELNET flags set to <EUNICE_SERVER>]
SERVER-CONFIG>
```

This example clears all flags that are set on a service.

```
SERVER-CONFIG>SET FLAGS
[TELNET flags set to <NONE>]
SERVER-CONFIG>
```

SET INIT

Specifies the name of the internal MULTINET_SERVER routine to call to initialize a service.

FORMAT

```
         [cluster_alias_init]
         [merge_image]
         [netcontrol_init]
         [none]
         [rpc_init]
         [rpc_portmap_init]
SET INIT [tcp_init]
         [ucxqio_init]
         [udp_init]
         [viadecnet_init]
         [viapsi_init]
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET INIT MERGE_IMAGE
[Init action of TELNET set to Merge_Image]
SERVER-CONFIG>
```

SET KEEPALIVE-TIMERS

Sets keepalive timers for a service.

FORMAT

`SET KEEPALIVE-TIMERS` *`idle-time prove-interval probe-count`*

PARAMETERS

`idle-time`
Specifies the amount of time, in seconds, that a connection should be idle before the first keep alive probe is sent.

`probe-interval`
Specifies the number of seconds between keep alive probes.

`probe-count`
Specifies the number of probes that can be sent, with no reply from the other side of the connection, before the connection should be destroyed.

DESCRIPTION

Keep alives are useful in situations when other systems that connect to services provided by your system are subject to frequent crashing, resets, or power-offs (as with personal computers).

TCP/IP connections must pass through a three-way handshake sequence to be closed and removed from the connection table. However, if a connection is open but idle, and the remote system is shut off, reset, or crashes, the connection cannot be closed down until an attempt to communicate with the remote system is made. If an application or service does not do this, a keep alive probe can be used to ensure that these dormant connections are cleaned up.

If you set the `SO_KEEPALIVE` socket option for a service, but you do not explicitly set the `KEEPALIVE-TIMERS`, the default values are:

- *`idle-time`* 2 hours
- *`probe-interval`* 75 seconds

- *probe-count* 8

If you do not set the SO_KEEPALIVE socket option for a service, no keep alive probes will be sent for connections to that service.

SET LISTEN

Specifies the name of the internal MULTINET_SERVER routine to call to listen for connections to the service.

FORMAT

```
           [none]
SET LISTEN [tcp_listen]
           [udp_listen]
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET LISTEN TCP_LISTEN
[Listen action of TELNET set to TCP_Listen]
SERVER-CONFIG>
```

SET LISTEN-ADDRESS

Specifies the IP address that a service listens on. This parameter can be used to restrict a service to only accept incoming connections to a particular address configured on the system. Either an IPv4 or IPv6 address can be specified depending upon the socket-family that is set for the service.

FORMAT

```
SET LISTEN-ADDRESS 192.168.1.1
SET LISTEN-ADDRESS 2002:c0a8:101:1::1
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET LISTEN-ADDRESS 192.168.1.1
[Server will listen on 192.168.1.1]
SERVER-CONFIG>
```

SET LOG-ACCEPTS

Specifies whether to log successful connections to the service.

FORMAT

`SET LOG-ACCEPTS` *mode*

PARAMETER

mode

If mode is `TRUE`, accepted connections are logged to OPCOM or to the log file; if it is `FALSE`, accepted connections are not logged.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET LOG-ACCEPTS TRUE
SERVER-CONFIG>
```

SET LOG-FILE

Specifies the destination of log messages: a VMS file name, or OPCOM to direct messages to the VMS OPCOM process.

FORMAT

```
SET LOG-FILE [opcom]
             [file_spec]
```

PARAMETER

file_spec

Specifies the file specification to which to write the audit records; specify `OPCOM` to write the audit records to OPCOM.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET LOG-FILE OPCOM
SERVER-CONFIG>
```

SET LOG-REJECTS

Specifies whether to log rejected connections to the service. A connection can be rejected because of the values of the REJECT-HOSTS, REJECT-NETS, and REJECT-BY-DEFAULT parameters.

FORMAT

SET LOG-REJECTS *mode*

PARAMETER

mode

If mode is TRUE, rejected connections are logged to OPCOM or to the log file; if it is FALSE, rejected connections are not logged.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET LOG-REJECTS TRUE
SERVER-CONFIG>
```

SET MAX-SERVERS

Specifies the maximum number of service processes to allow at any one time. If this limit is reached, additional connections up to BACKLOG are accepted but are not processed until one of the previous connections completes.

FORMAT

SET MAX-SERVERS *number*

PARAMETER

number
Specifies the maximum number of server processes to create for this service.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET MAX-SERVERS 4
[Max_Servers of TELNET set to 4]
SERVER-CONFIG>
```

SET PARAMETERS

Specifies service-dependent parameters. These parameters are passed to the initialization routine of built-in services. (This is normally not used for user-written services. The parameters and their arguments differ on a per-service basis.)

FORMAT

```
SET PARAMETERS
```

EXAMPLE

This example enables debugging for the Domain Name Service (DNS).

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT DOMAINNAME
SERVER-CONFIG>SET PARAMETERS
Delete parameter "bootfile MULTINET:DOMAIN-NAME-SERVICE.CONFIGURATION"? [NO]
You can now add new parameters for DOMAINNAME. An empty line terminates.
Add Parameter: debug 3
Add Parameter:
[Service specific parameters for DOMAINNAME changed]
SERVER-CONFIG>
```

SET PRIORITY

Specifies the VMS process priority to assign to created processes.

FORMAT

`SET PRIORITY` *priority*

PARAMETER

priority
Specifies the VMS process priority to assign to created processes.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT SMTP
SERVER-CONFIG>SET PRIOITY 5
[Priority of SMTP set to 5]
SERVER-CONFIG>
```

SET PROCESS

Specifies that the service is to run in an auxiliary master server process rather than in the main master server process.

The `SET PROCESS` command should be used in conjunction with the `SET FLAGS start_aux_server` command, unless the administrator wants to start the auxiliary server manually.

FORMAT

`SET PROCESS` *`process_name`*

PARAMETER

`process_name`
Specifies the name of the process to run in auxiliary master process mode.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET FLAG start_aux_server
SERVER-CONFIG>SET PROCESS MNSERVER_1
SERVER-CONFIG>SELECT RLOGIN
SERVER-CONFIG>SET FLAG start_aux_server
SERVER-CONFIG>SET PROCESS MNSERVER_1
SERVER-CONFIG>SELECT RSHELL
SERVER-CONFIG>SET FLAG start_aux_server
SERVER-CONFIG>SET PROCESS MNSERVER_1
SERVER-CONFIG>
```

SET PROGRAM

Specifies the VMS file name of the image to run or merge.

FORMAT

`SET PROGRAM file_spec`

PARAMETER

file_spec
Specifies the name of the file containing the server image.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT SMTP
SERVER-CONFIG>SET PROGRAM MULTINET:SMTP_SERVER.EXE
[Program to run for SMTP set to MULTINET:SMTP_SERVER.EXE]
SERVER-CONFIG>
```

SET REJECT-BY-DEFAULT

Specifies whether to reject a connection from a host that does not match any of the ACCEPT-HOSTS, ACCEPT-NETS, REJECT-HOSTS, and REJECT-NETS lists.

FORMAT

SET REJECT-BY-DEFAULT *mode*

PARAMETER

mode

If mode is TRUE, the default is to reject connections; if it is FALSE, the default is to accept them.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT SMTP
SERVER-CONFIG>SET REJECT-BY-DEFAULT TRUE
SERVER-CONFIG>
```

SET RECEIVE-BUFFER-SPACE

Specifies the size of the receive socket buffers.

FORMAT

`SET RECEIVE-BUFFER-SPACE` *`size`*

PARAMETER

`size`
Specifies the size of the receive socket buffers for the selected service.

SET REJECT-HOSTS

Specifies and maintains a list of hosts that are not allowed to access the service. For IPv6 services IPv6 addresses and IPv4 addresses can be used. IPv4 addresses that are specified for an IPv6 service are treated as IPv4 mapped addresses.

FORMAT

```
SET REJECT-HOSTS
```

EXAMPLE

This example shows how to delete host 192.0.0.1 from the reject-hosts list, and add host 192.0.0.4.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET REJECT-HOSTS
Delete address "IP-192.0.0.1" ? [NO] Y
[Address "IP-192.0.0.1" deleted from TELNET]
Delete address "IP-192.0.0.2" ? [NO] RETURN
Delete address "IP-192.0.0.3" ? [NO] RETURN
You can now add new addresses for TELNET. An empty line terminates.
Add Address: 192.0.0.4
Add Address:
SERVER-CONFIG>
```

SET REJECT-MESSAGE

Specifies a text string to send down the network connection when a service is rejected.

FORMAT

`SET REJECT-MESSAGE` *`string`*

PARAMETER

`string`
This parameter is written down the network connection before closing the connection when a request is rejected.

RESTRICTION

This parameter is ignored on UDP services.

EXAMPLE

This example sets the rejection message to "Service refused."

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET REJECT-MESSAGE Service refused
SERVER-CONFIG>
```

SET REJECT-NETS

Specifies and maintains a list of networks or subnetworks that are not allowed to access the service.

FORMAT

```
SET REJECT-NETS
```

DESCRIPTION

`SET REJECT-NETS` invokes an interactive utility that prompts you for the addresses of networks that are not allowed to access the selected service.

Specify each network as follows:

IP_address [*subnetmask*]

When done, press **RETURN** at the `Add:` prompt.

For more information about restricting access to services, see the *MultiNet Installation and Administrator's Guide*.

EXAMPLE

This example deletes network address 192.0.0.0 from the reject-nets list, and adds network address 128.1.0.0.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET REJECT-NETS
Delete address "IP-192.0.0.0" ? [NO] Y
[Address "IP-192.0.0.0" deleted from TELNET]
Delete address "IP-192.12.19.0" ? [NO] RETURN
You can now add new addresses for TELNET. An empty line terminates.
Add Address: 128.1.0.0
Add Address:
SERVER-CONFIG>
```

SET SEND-BUFFER-SPACE

Specifies the size of the send socket buffers.

FORMAT

`SET RECEIVE-BUFFER-SPACE` *size*

PARAMETERS

size

Specifies the size of the send socket buffers for the selected service.

SET SERVICE

Specifies the name of the internal MULTINET_SERVER routine to call to perform the service.

FORMAT

```
             [internal_ftp]
             [internal_pop2]
             [internal_pop3]
             [internal_telnet]
SET SERVICE [no_service]
             [none]
             [run_program]
             [tcp_daytime]
             [tcp_time]
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET SERVICE INTERNAL_TELNET
[Service action of TELNET set to Internal_Telnet]
SERVER-CONFIG>
```

SET SERVICE-NAME

Changes the name of the service.

FORMAT

`SET SERVICE-NAME` *name*

PARAMETER

name
Specifies the new service name.

EXAMPLE

This example shows how to rename a service to "FOO."

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET SERVICE-NAME FOO
[Name of selected service changed to FOO]
SERVER-CONFIG>
```

SET SERVICE-TYPE

For SPX- or IPX-based servers, sets the service type advertised for the service. This setting is generally not changed by users.

FORMAT

SET SERVICE-TYPE *service-type*

PARAMETER

service-type
Specifies the service type for SAP advertisements. This value is a decimal number in the range of 1 through 6535.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT servicename
SERVER-CONFIG>SET SERVICE-TYPE n
[servicename service type is n]
SERVER-CONFIG>
```

SET SOCKET-FAMILY

Specifies the address family of the service.

FORMAT

`SET SOCKET-FAMILY` *family*

PARAMETER

family
Specifies the new protocol family for this service.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET SOCKET-FAMILY AF_INET
[TELNET socket family is AF_INET]
SERVER-CONFIG>
```

SET SOCKET-OPTIONS

Specifies socket options to be set via `setsockopt()`. See the *MultiNet Programmer's Reference* for more information on socket options.

FORMAT

`SET SOCKET-OPTIONS` *options*

PARAMETER

options
Specifies a list of socket options separated by a vertical bar (`|`).

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET SOCKET-OPTIONS SO_DEBUG | SO_KEEPALIVE
[TELNET socket options set to <SO_DEBUG | SO_KEEPALIVE]
SERVER-CONFIG>
```

SET SOCKET-PORT

Specifies the port number on which to listen for connections.

FORMAT

`SET SOCKET-PORT` *`port`*

PARAMETER

`port`
Specifies the name or number of the port on which to listen.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TELNET
SERVER-CONFIG>SET PORT 10
[TELNET socket port is now 10]
SERVER-CONFIG>
```

SET SOCKET-TYPE

Specifies the type of socket; for example, SOCK_STREAM (TCP) or SOCK_DGRAM (UDP).

FORMAT

SET SOCKET-TYPE *type*

PARAMETER

type
Specifies the socket type to listen on, usually SOCK_ DGRAM (UDP) or SOCK_STREAM (TCP).

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT TFTP
SERVER-CONFIG>SET SOCKET-TYPE SOCK_DGRAM
[Socket type of TFTP set to SOCK_DGRAM]
SERVER-CONFIG>
```

SET USERNAME

Specifies the user name under which the selected service is started.

FORMAT

`SET USERNAME` *`username`*

PARAMETER

`username`
Specifies the name of the user under which the selected service is started.

SET WORKING-SET-EXTENT

Specifies how much memory the process will be allowed to use if there are free pages available.

FORMAT

```
SET WORKING-SET-EXTENT
```

EXAMPLES

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT FTP
SERVER-CONFIG>SET WORKING-SET 2000
[Working_Set of FTP set to 2000]
SERVER-CONFIG>
```

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6(nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT FTP
[The Selected SERVER entry is now FTP]
SERVER-CONFIG>SET WORKING-SET-QUOTA 4096
[Working Set Quota of FTP set to 4096]
SERVER-CONFIG>SET WORKING-SET-EXTENT 8192
[Working Set Extent of FTP set to 8192]
```

SET WORKING-SET-QUOTA

Specifies the maximum amount of memory the process can lock into its working set.

FORMAT

```
SET WORKING-SET-QUOTA
```

EXAMPLES

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT FTP
SERVER-CONFIG>SET WORKING-SET 2000
[Working_Set of FTP set to 2000]
SERVER-CONFIG>
```

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6(nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SELECT FTP
[The Selected SERVER entry is now FTP]
SERVER-CONFIG>SET WORKING-SET-QUOTA 4096
[Working Set Quota of FTP set to 4096]
SERVER-CONFIG>SET WORKING-SET-EXTENT 8192
[Working Set Extent of FTP set to 8192]
```

SHOW

Shows the current server configuration.

FORMAT

```
SHOW
```

QUALIFIER

/FULL
Provides a more detailed listing.

EXAMPLE

This example displays detailed information about SMTP.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SHOW/FULL SMTP
Service "SMTP"
TCP socket (AF_INET,SOCK_STREAM), Port 25
Socket Options = SO_KEEPALIVE
INIT() = TCP_Init
LISTEN() = TCP_Listen
CONNECTED() = TCP_Connected
SERVICE() = Run_Program
Program = "MULTINET:SERVER_SMTP.EXE"
SERVER-CONFIG>
```

SHUTDOWN

Stops the master server (MULTINET_SERVER) process. After a SHUTDOWN, any subsequent network service requests are rejected by MultiNet until the RESTART command is executed.

FORMAT

SHUTDOWN

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SHUTDOWN
SERVER-CONFIG>
```

SPAWN

Executes a single DCL command, or if entered without options, starts a subprocess with the same effect as PUSH. To return from DCL, use the LOGOUT command. If the MULTINET_DISABLE_SPAWN logical is set, SPAWN does not work.

FORMAT

SPAWN [*command*]

PARAMETER

command
Specifies a command to execute. If you omit *command*, a DCL command line subprocess is created.

QUALIFIERS

/INPUT=*file-spec*
Specifies an input file to the command you enter with SPAWN.

/LOGICAL_NAMES
/NOLOGICAL_NAMES
Specifies that logical names and logical name tables are not copied to the subprocess.

/SYMBOLS
/NOSYMBOLS
Specifies that global and local names are not passed to the subprocess.

/WAIT
/NOWAIT
Returns control without waiting for the command to complete. Do not use this qualifier with commands that have prompts or screen displays.

/OUTPUT=*file-spec*

Specifies a file that retains the output of the command invoked with SPAWN. This qualifier only works when a single command is entered without creating a DCL subprocess. In addition, this qualifier is positional; you must enter it immediately after SPAWN or other qualifiers.

EXAMPLES

This example displays terminal information, captures the output in a file, then displays the information with the TYPE command.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SPAWN/OUTPUT=FOO. SHOW TERM
SERVER-CONFIG>SPAWN TYPE FOO.
...
```

This example invokes a command procedure.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SPAWN @COMPROC
...
```

This example displays help information about the SERVER-CONFIG utility. Use the LOGOUT command to return control to SERVER-CONFIG.

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>SPAWN
$ HELP MULTINET CONFIGURE /SERVER
...
$ LOGOUT
SERVER-CONFIG>
```

STATUS

Shows the status of the MultiNet server configuration program.

FORMAT

```
STATUS
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6(nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>STATUS
This is the MultiNet server configuration program Version 5.5(nnn)
There are 65/8192 entries in the current server configuration.
There is NO selected SERVER entry.
The configuration MULTINET:SERVICES.MASTER_SERVER is not modified.
SERVER-CONFIG>
```

USE

Reads in a MultiNet server configuration file. After a `USE`, you can use the various configuration commands to modify the server configuration. (Functionally equivalent to `GET`.)

FORMAT

`USE` *config-file*

PARAMETER

config-file
Specifies the name of the server configuration file to read in.

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>USE ST_TMP:FOO.CONFIGURATION
[Reading in configuration from ST_ROOT:[TMP]FOO.CONFIGURATION.1]
SERVER-CONFIG>
```

VERSION

Displays the MultiNet server configuration program version and release information.

FORMAT

```
VERSION
```

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>VERSION
This is the MultiNet Server configuration program Version 5.6 (nnn)
SERVER-CONFIG>
```

WRITE

Writes the current server configuration file. (Functionally equivalent to SAVE.)

FORMAT

WRITE *config-file*

PARAMETER

config-file
Specifies the name of the file to which to write the current MultiNet server configuration (by default, the same file from which the configuration was read).

EXAMPLE

```
$ MULTINET CONFIGURE /SERVERS
MultiNet Server Configuration Utility 5.6 (nnn)
[Reading in configuration from MULTINET:SERVICES.MASTER_SERVER]
SERVER-CONFIG>WRITE
[Writing configuration to
MULTINET_COMMON_ROOT:[MULTINET]SERVICES.MASTER_SERVER.1103]
SERVER-CONFIG>
```