VMS Software

VSI OpenVMS

VSI TCP/IP Services for OpenVMS Management Command Reference

Document Number: DO-TCPMCR-01A

Publication Date: September 2020

Revision Update Information: This is a new manual.

Operating System and Version: VSI OpenVMS Integrity Version 8.4-2
VSI OpenVMS Alpha Version 8.4-2L1

Software Version: VSI TCP/IP Services Version 5.7

VMS Software, Inc. (VSI)
Burlington, Massachusetts, USA

VSI TCP/IP Services for OpenVMS Management Command Reference

VMS Software

Legal Notice

The VSI OpenVMS documentation set is available on DVD.

Preface

The VSI TCP/IP Services for OpenVMS product is the VSI implementation of the TCP/IP networking protocol suite and internet services for VSI OpenVMS Alpha systems.

TCP/IP Services provides a comprehensive suite of functions and applications that support industry-standard protocols for heterogeneous network communications and resource sharing.

This manual describes the TCP/IP Services management commands. Use it in conjunction with the *VSI TCP/IP Services for OpenVMS Management* manual, which describes the management tasks.

Refer to the *VSI TCP/IP Services for OpenVMS Installation and Configuration* manual for information about installing, configuring, and starting this product.

1. About VSI

VMS Software, Inc. (VSI) is an independent software company licensed by Hewlett Packard Enterprise to develop and support the OpenVMS operating system.

VSI seeks to continue the legendary development prowess and customer-first priorities that are so closely associated with the OpenVMS operating system and its original author, Digital Equipment Corporation.

2. Intended Audience

This manual is for experienced OpenVMS and UNIX® system managers and assumes a working knowledge of OpenVMS system management, TCP/IP networking, and TCP/IP terminology.

If you are not familiar with the TCP/IP Services product, please review the VSI TCP/IP Services for OpenVMS Concepts and Planning manual before using this manual to configure and manage TCP/IP components.

3. Document Structure

This manual contains the following chapters:

- Chapter 1 introduces the management control program.
- Chapter 2 provides command descriptions for each management command.

4. Related Documents

The table below lists the documents available with this version of TCP/IP Services.

Table 1. TCP/IP Services Documentation

Manual	Contents
VSI TCP/IP Services for OpenVMS Concepts and Planning	This manual provides conceptual information about TCP/IP networking on OpenVMS systems, including general planning issues to consider before configuring your system to use the TCP/IP Services software.

Manual	Contents
	This manual also describes the manuals in the TCP/IP Services documentation set and provides a glossary of terms and acronyms for the TCP/IP Services software product.
VSI TCP/IP Services for OpenVMS Release Notes	The release notes provide version-specific information that supersedes the information in the documentation set. The features, restrictions, and corrections in this version of the software are described in the release notes. Always read the release notes before installing the software.
VSI TCP/IP Services for OpenVMS Installation and Configuration	This manual explains how to install and configure the TCP/IP Services product.
VSI TCP/IP Services for OpenVMS User's Guide	This manual describes how to use the applications available with TCP/IP Services such as remote file operations, email, TELNET, TN3270, and network printing.
VSI TCP/IP Services for OpenVMS Management	This manual describes how to configure and manage the TCP/IP Services product.
VSI TCP/IP Services for OpenVMS Management Command Reference	This manual describes the TCP/IP Services management commands.
VSI TCP/IP Services for OpenVMS Management Command Quick Reference Card	This reference card lists the TCP/IP management commands by component and describes the purpose of each command.
VSI TCP/IP Services for OpenVMS UNIX Command Equivalents Reference Card	This reference card contains information about commonly performed network management tasks and their corresponding TCP/IP management and UNIX command formats.
VSI TCP/IP Services for OpenVMS ONC RPC Programming	This manual presents an overview of high-level programming using open network computing remote procedure calls (ONC RPCs). This manual also describes the RPC programming interface and how to use the RPCGEN protocol compiler to create applications.
VSI TCP/IP Services for OpenVMS Sockets API and System Services Programming	This manual describes how to use the Sockets API and OpenVMS system services to develop network applications.
VSI TCP/IP Services for OpenVMS SNMP Programming and Reference	This manual describes the Simple Network Management Protocol (SNMP) and the SNMP application programming interface (eSNMP). It describes the subagents provided with TCP/IP Services, utilities provided for managing subagents, and how to build your own subagents.
VSI TCP/IP Services for OpenVMS Tuning and Troubleshooting	This manual provides information about how to isolate the causes of network problems and how to tune the TCP/IP Services software for the best performance.

Manual	Contents
VSI TCP/IP Services for OpenVMS Guide to IPv6	This manual describes the IPv6 environment, the roles of systems in this environment, the types and function of the different IPv6 addresses, and how to configure TCP/IP Services to access the IPv6 network.
VSI TCP/IP Services for OpenVMS Guide to SSH	This manual describes the way Secure Shell (SSH) is implemented on TCP/IP Services. It describes how to configure, manage, and use the optional services that are protected by secure shell security.

For a comprehensive overview of the TCP/IP protocol suite, refer to the book *Internetworking with TCP/IP: Principles, Protocols, and Architecture*, by Douglas Comer.

5. VSI Encourages Your Comments

You may send comments or suggestions regarding this manual or any VSI document by sending electronic mail to the following Internet address: <docinfo@vmssoftware.com>. Users who have OpenVMS support contracts through VSI can contact <support@vmssoftware.com> for help with this product.

6. Conventions

The following conventions may be used in this manual:

Convention	Meaning
Ctrl/ *x*	A sequence such as **Ctrl**/ *x* indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
PF1 *x*	A sequence such as PF1 *x* indicates that you must first press and release the key labeled PF1 and then press and release another key or a pointing device button.
Return	In examples, a key name enclosed in a box indicates that you press a key on the keyboard. (In text, a key name is not enclosed in a box.)
. . .	A horizontal ellipsis in examples indicates one of the following possibilities: • Additional optional arguments in a statement have been omitted. • The preceding item or items can be repeated one or more times. • Additional parameters, values, or other information can be entered.
. . .	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
( )	In command format descriptions, parentheses indicate that you must enclose the options in parentheses if you choose more than one.
[ ]	In command format descriptions, brackets indicate optional choices. You can choose one or more items or no items. Do not type the brackets on the command line. However, you must include the brackets in the syntax for OpenVMS

Convention	Meaning
	directory specifications and for a substring specification in an assignment statement.
[|]	In command format descriptions, vertical bars separate choices within brackets or braces. Within brackets, the choices are options; within braces, at least one choice is required. Do not type the vertical bars on the command line.
{ }	In command format descriptions, braces indicate required choices; you must choose at least one of the items listed. Do not type the braces on the command line.
bold text	This typeface represents the introduction of a new term. It also represents the name of an argument, an attribute, or a reason.
italic text	Italic text indicates important information, complete titles of manuals, or variables. Variables include information that varies in system output (Internal error *number*), in command lines (/PRODUCER= *name*), and in command parameters in text (where *dd* represents the predefined code for the device type).
UPPERCASE TEXT	Uppercase text indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.
`Monospace type`	Monospace type indicates code examples and interactive screen displays. In the C programming language, monospace type in text identifies the following elements: keywords, the names of independently compiled external functions and files, syntax summaries, and references to variables or identifiers introduced in an example.
-	A hyphen at the end of a command format description, command line, or code line indicates that the command or statement continues on the following line.
numbers	All numbers in text are assumed to be decimal unless otherwise noted. Nondecimal radixes—binary, octal, or hexadecimal—are explicitly indicated.

Other conventions are:

- All numbers are decimal unless otherwise noted.
- All Ethernet addresses are hexadecimal.

Chapter 1. Using TCP/IP Services Management Commands

The TCP/IP Services product provides a management command interface you use to configure and manage the software. These commands let you perform the following tasks:

- Configure and reconfigure components
- Modify parameters of components
- Configure customer-developed services
- Enable and disable running components
- Monitor the running software

1.1. Entering Commands

To start the management control program, type TCPIP at the DCL prompt. For example:

```
$ TCPIP
TCPIP>
```

At the TCPIP> prompt, you can enter commands described in this manual or display online help. Type EXIT to exit the management control program, or press Ctrl/C to abort a command.

Help is also available at the DCL prompt by typing HELP TCPIP_SERVICES.

```
$ HELP TCPIP_SERVICES
```

Note

The word **command** refers to commands for the TCP/IP Services software. DCL commands and UNIX commands are explicitly identified.

Table 1.1 provides guidelines for using management control program commands.

Table 1.1. Management Command Guidelines

Element	Guideline
Address formats	Some commands require that you specify one of the following kinds of addresses: • IP • Ethernet • FDDI • Token Ring • Hardware

Element	Guideline

	Be sure to use the appropriate format. The following examples illustrate an IP address, an Ethernet address, and a hardware address, respectively.
	
	`TCPIP> SET HOST CROW /ADDRESS=1.2.3.4`
	
	`TCPIP> SET ARP AA-BB-04-05-06-07 CONDOR`
	
	`TCPIP> SET BOOTP MACAW /HARDWARE=ADDRESS=08-dd-ff-2a-23-21`

Default	Refers to the command's behavior if optional qualifiers are omitted.
File and directory names	When you specify OpenVMS files, follow all OpenVMS file specification rules. Likewise, when you specify UNIX files, follow all UNIX file specification rules.
Service names	To specify a lowercase or mixed-case service name, enclose it in quotation marks. Service names are limited to 16 characters. Use only the following characters in a service name: • Uppercase and lowercase alphabetic characters • Numerals • Dollar sign ($) • Underscore (_) Do not define a service name equivalent to one of the TCP/IP Services for OpenVMS components (for example, do not define a service name BIND or TCPIP$BIND). In addition, the service name CUSTOMER_SERVICE is reserved by VSI.
Host names and IP addresses	To specify a host or network name on a command line, you can enter either the host's name or the host's IP address.

Keywords	You can abbreviate commands to the fewest number of characters, usually four, that identify the command. The following command lines, for example, have identical meanings:
	
	`TCPIP> SH SE NFS/FU/PER`
	
	`TCPIP> SHOW SERVICE NFS /FULL /PERMANENT`
	
	Command examples shown in this manual are expressed using full command and qualifier names for clarity.

Multiple values	To specify multiple host names, addresses, or options for parameters and qualifiers, be sure to separate elements with commas and enclose the entire list in parentheses. Wildcards are valid unless otherwise stated. A space between multiple elements is optional unless otherwise stated. For example, the following qualifiers are the same: `/ qualifier=(option_a:value1,option_b:value2,value3)`

Element	Guideline
	`/qualifier=(option_a=value1), (option_b=value2,value3)` Wildcards are valid unless otherwise stated. A space between multiple elements is optional unless otherwise stated.
Numeric values	Unless otherwise stated, all numeric values are decimal. Values are indicated by either a preceding equals sign (=) or a colon (:). For example: `TCPIP> SET NAME_SERVICE /SERVER: (SORA,JACANA,PARROT) - _TCPIP> /ACCEPT: (HOSTS:JACANA,JAY,JUNCO,999.20.40.3)`

Quotation marks	In command lines, enclose the following in quotation marks:
	
	• Lowercase and mixed-case names to be stored in a database with the exact case preserved
	
	• Directory and file specifications containing a slash (/)
	
	• Uppercase options specified with UNIX commands
	
	Consider these examples:
	
	1. To specify a path, enclose it in quotation marks:
	
	`TCPIP> MAP "/usr/songbirds/canary" CANARY $DUA2:`
	
	2. To specify host names using lowercase letters when you create a proxy entry in the database:
	
	`TCPIP> ADD PROXY COUSINS /GID=10 /UID=40 - _TCPIP> /HOST=("raven","crow","rook","daw")`
	
	Note the use of the DCL command-line continuation character " - " that allows you to continue a long command on the next line.
	
	3. To specify a lowercase host name when adding the host to the hosts database, use these commands:
	
	`TCPIP> SET HOST "eaglet" /ADDRESS = 128.33.22.1`
	`TCPIP> SHOW HOST EAGLET`
	
	Note that DCL interprets all input as uppercase unless you enclose it in quotation marks. Therefore, you must use quotation marks to enter the host name in lowercase in the hosts database. To display information about a host, you can enter either uppercase or lowercase characters.

Element	Guideline

	4. When entering a lowercase or mixed-case service name in a command, enclose it in quotations marks. For example:
	`TCPIP> SET SERVICE "hello" ...`
	5. When entering an option in uppercase in a UNIX command, enclose the option in quotation marks. For example:
	`TCPIP> sysconfig "-Q" inet`

UNIX commands	Follow UNIX syntax and case rules when entering UNIX commands at the DCL and TCPIP> prompts. For example, enter the `ifconfig` command in lowercase letters:
	`TCPIP> ifconfig` *`options`*
	When entering UNIX commands at the DCL or TCPIP> prompt, enclose uppercase options in quotation marks. For example:
	`$ TCPIP> sysconfig "-Q" inet`
	You can abbreviate commands, as shown in the following example. The abbreviation must be unique through the first four characters.
	`TCPIP> ifco` *`options`*
	If the abbreviation entered is not unique, an error message will advise you to supply more characters. In the following example, the SYSCONFIG command cannot be abbreviated because of the SYSCONFIGDB command.
	`TCPIP> sysc -q`
	`%CLI-W-ABVERB, ambiguous command verb - supply more characters`

Wildcards	If you specify a wildcard (an asterisk [*]) on a command line, you are asked for confirmation before the command executes. For example:
	`TCPIP> REMOVE PROXY *`
	`VMS User_name  Type   User_ID  Group_ID Host_name`
	`GRACKLE          N       269       48      MAPLE`
	`Remove? [N]:`
	To change this default behavior (so that you are not asked to confirm), use the /NOCONFIRM qualifier with the command.

1.1.1. Setting Configuration Parameters

Some commands allow you to enter information in the database; others modify only the run-time parameters. Table 1.2 shows the SET commands that affect one or the other.

Table 1.2. SET Commands

Modify Permanent Database Files	Modify Dynamic Memory
SET BOOTP	SET ARP
SET CONFIGURATION	SET COMMUNICATION
SET HOST	SET INTERFACE
SET MX_RECORDS	SET NAME_SERVICE
SET NETWORK	SET NFS_SERVER
SET CONTAINER	SET PROTOCOL
SET ROUTE	SET ROUTE
SET SERVICE	

Note that the SET ROUTE command affects both the permanent and dynamic routing databases.

1.1.2. Modifying the Configuration Database

Unlike the other databases, which have similar objects, the configuration database holds diverse initialization information for various TCP/IP Services components.

The following commands modify the configuration database:

- SET CONFIGURATION BIND
- SET CONFIGURATION COMMUNICATION
- SET CONFIGURATION ENABLE SERVICE
- SET CONFIGURATION INTERFACE
- SET CONFIGURATION NAME_SERVICE
- SET CONFIGURATION PROTOCOL
- SET CONFIGURATION SMTP
- SET CONFIGURATION SNMP
- SET CONFIGURATION START ROUTING
- SET CONFIGURATION TIME

1.1.3. Creating and Deleting Files

The CREATE commands create the following kinds of files:

- Database files

  VSI strongly recommends that you use the TCP/IP Services configuration procedure (TCPIP $CONFIG) instead of manually creating databases. Refer to the VSI TCP/IP Services for OpenVMS Installation and Configuration guide for instructions.

  Use the following commands to create database files:

  - CREATE BOOTP
  - CREATE CONFIGURATION
  - CREATE EXPORT
  - CREATE HOST
  - CREATE NETWORK
  - CREATE PROXY
  - CREATE ROUTE
- UNIX container directories

  These directories are used by the NFS server software. Use the following commands to create and delete container directories and files:

  - CREATE CONTAINER
  - DELETE CONTAINER
  - CREATE DIRECTORY
  - REMOVE DIRECTORY
  - REMOVE FILE

1.1.4. Adding and Deleting Records

To add and delete records from the TCP/IP Services databases, use the CONVERT, ADD, and REMOVE management commands.

Use the following commands to add records to and delete records from the proxy and export databases:

- ADD EXPORT
- REMOVE EXPORT
- ADD PROXY
- REMOVE PROXY

You cannot modify information that you enter into databases. Instead, delete the record with the REMOVE command and then issue a new ADD command.

TCP/IP Services provides the following kinds of CONVERT management commands:

- CONVERT/CONFIGURATION BIND

  Converts BIND configuration information to BIND Version 8.1 format.

- CONVERT/VMS

Populates an existing database with entries from a UNIX database file.

- CONVERT/UNIX

  Reads a TCP/IP Services database and converts the information to a UNIX-formatted (sequential) database file. For example, CONVERT/UNIX HOST reads the hosts database and converts the records into a UNIX-formatted `/etc/hosts` file.

1.1.5. Starting and Stopping Software

You can start and stop software components interactively by using the following commands:

- START MAIL
- START ROUTING
- STOP ROUTING

For example:

```
TCPIP> START ROUTING /SUPPLY
```

For the server components that are started by the auxiliary server upon an incoming client request, the ENABLE SERVICE command tells the auxiliary server to listen for requests and act upon them.

The DISABLE SERVICE command tells the auxiliary server to stop listening for incoming requests.

Use the following commands to set components to start when TCP/IP Services starts. The permanent configuration is stored in the configuration database.

- SET CONFIGURATION ENABLE SERVICE
- SET CONFIGURATION ENABLE NOSERVICE
- SET CONFIGURATION START ROUTING
- SET CONFIGURATION START NOROUTING

1.1.6. Validating Data Integrity

Use the following commands to verify the integrity of TCP/IP Services files:

- ANALYZE CONTAINER
- ANALYZE MAIL
- ANALYZE SERVICE

1.1.7. Managing NFS

The NFS server requires the following management commands:

- MAP
- SET CONFIGURATION MAP

- SHOW MAP
- SHOW CONFIGURATION MAP
- UNMAP
- SET CONFIGURATION NOMAP
- SET NFS_SERVER
- SHOW NFS_SERVER
- CREATE EXPORT
- ADD EXPORT
- REMOVE EXPORT
- SET EXPORT
- SHOW EXPORT
- CREATE PROXY
- CONVERT/VMS PROXY
- ADD PROXY
- REMOVE PROXY
- SHOW PROXY

1.1.8. Using NFS

The TCP/IP Services software includes commands for using NFS. Use the following commands to view container file systems and to copy files to and from them:

- CREATE DIRECTORY
- DIRECTORY
- REMOVE DIRECTORY
- REMOVE FILE
- CREATE CONTAINER
- DELETE CONTAINER
- EXPORT
- IMPORT

For using the NFS client (working with files that reside on remote hosts), TCP/IP Services software provides the following commands:

- MOUNT

- SHOW MOUNT
- DISMOUNT

1.1.9. Displaying Information

The SHOW and LIST commands display configuration, status, and performance information.

1.2. UNIX Management Commands

You can use UNIX management commands to manage the TCP/IP Services software. Section 1.2.1 lists the supported UNIX management commands, and Section 1.2.2 gives a few tips about using UNIX management commands and provides sources for more information.

1.2.1. Supported UNIX Management Commands

Table 1.3 describes the supported UNIX commands.

Table 1.3. UNIX Management Commands

Command	Description
`arp`	Controls and displays ARP tables for the specified host.
`ifconfig`	Configures or displays network interface parameters, redefines an address for a particular interface, or sets options such as an alias list, broadcast address, or access filter.
`netstat`	Displays network statistics of sockets, data link counters, specified protocols or aliases, network interfaces, and a host's routing table.
`nfsstat`	Displays statistical information about the network file system (NFS) and remote procedure call (RPC) interfaces in the kernel. It can also be used to reinitialize this information.
`ripquery`	Requests all routes known by a RIP gateway by sending a RIP request or a POLL command.
`route`	Allows you to manipulate the routing table manually. Normally, a system routing table management component, such as GATED or ROUTED, will tend to this task.
`sysconfig`	Manages and displays network attributes in the kernel subsystem configuration.
`sysconfigdb`	Manages and displays network attributes in the subsystem configuration table (TCPIP $ETC:SYSCONFIGTAB.DAT).
`traceroute`	Displays the route that packets take to a network host.
`whois`	Displays user, host, and organization names in the Network Information Center (NIC) database.

1.2.2. Using UNIX Management Commands

To use UNIX management commands at the DCL prompt, execute the command procedure SYS $MANAGER:TCPIP$DEFINE_COMMANDS.COM. For example:

```
$ @SYS$MANAGER:TCPIP$DEFINE_COMMANDS.COM
```

This command procedure defines process-specific commands that enable you to enter UNIX commands from the DCL prompt. Note that execution of a UNIX command in a DCL command procedure does not return an error in $STATUS, so you cannot test for the failure of a UNIX command in a DCL command procedure.

The following command shows how to obtain information about an interface configured on your host. Note the use of quotation marks to preserve the case of the command option. You must enclose uppercase options in quotation marks when entering UNIX commands.

```
$ ifconfig "WF0"
```

The following is displayed:

```
WF0: flags=c43<UP,BROADCAST,RUNNING,MULTICAST,SIMPLEX>
     rxmt 1000, reach time 30000, dad tries 1, MT 4352, hops 64, token len
 64
     inet 16.20.208.100 netmask ffff0000 broadcast 16.20.255.255 ipmtu 4470
     inet6 fe80::200:f8ff:fe66:2e35
```

For more information about UNIX command options and flags, refer to the VSI TCP/IP Services for OpenVMS Tuning and Troubleshooting manual, or enter HELP *unix_command* at the TCPIP> prompt. For example, to display information about the `netstat` command, enter:

```
TCPIP> HELP NETSTAT
```

Chapter 2. Command Descriptions

This chapter describes the TCP/IP Services management commands in alphabetical order.

For information about how to enter commands, see Chapter 1.

ADD EXPORT

ADD EXPORT — Adds an export entry, in the form of a UNIX path name, to the export database for a Network File System (NFS) file system. The path name is a name that is mapped to one of the following: an OpenVMS disk, a subdirectory on an OpenVMS disk, or a UNIX container file system. *Related commands:* MAP, REMOVE EXPORT, SET CONFIGURATION MAP, SHOW EXPORT, SHOW HOST. *Applies to:* NFS server.

Format

```
ADD EXPORT "/path/name"
           /HOST=host
           [ /OPTIONS=
           [NO]DATA_CONVERSION
           [NO]NAME_CONVERSION
           [NO]PURGE_VERSIONS
           [NO]TYPELESS_DIRECTORIES ]
```

Restrictions and Tips

This command requires read and write access to the export database. The following restrictions and instructions apply:

- Do not use wildcards within a UNIX directory specification.
- For each host, define both its host name and any alias names.
- For each entry, use uppercase and lowercase consistently.
- If you remove an export entry and replace the entry using different options; each client must dismount and remount for the new options to take effect.

Parameters

/path/name

Required.

File system to add to the export database.

Separate directory and subdirectory names with slashes.

Qualifiers

/HOST=*host*

Required.

The NFS client host or hosts that will have access to the specified NFS file system.

You can use a wildcard to allow access to all hosts.

/OPTIONS= { [NO]DATA_CONVERSION |[NO]NAME_CONVERSION
|[NO]PURGE_VERSIONS |[NO]TYPELESS_DIRECTORIES }

Optional.

Note

For clients operating in OpenVMS to OpenVMS mode, the server ignores the options in the export record and uses the settings required for OpenVMS to OpenVMS mode.

Options for the specified directory:

- DATA_CONVERSION, NODATA_CONVERSION
  - DATA_CONVERSION (default)

    Converts the following kinds of sequential files:

    - Variable
    - Variable with fixed-length control (VFC)
    - Fixed-record formats

    Converts sequential files according to the rules applied by the following record attributes:

    - Carriage return/carriage control (CR)
    - Fortran carriage control (FTN)
    - Print file-format control (PRN)

    Stream formats are returned unconverted.

    The data in files with nonstream records cannot be written back to the file.
  - NODATA_CONVERSION

    File data is considered raw and is returned without conversion. Nonstream records are returned with their record control information mixed with the record data. Files can be rewritten randomly.
- NAME_CONVERSION, NONAME_CONVERSION
  - NAME_CONVERSION

    A non-OpenVMS client can create files with mixed-case names and names containing characters that are invalid for OpenVMS file names. The server converts such names to valid OpenVMS file names, and reverses the conversion when displaying the file names to a non-OpenVMS client.

    If the EXPORT command specifies an ODS-5 volume, the NAME_CONVERSION option is ignored.

    - NONAME_CONVERSION (default)

      Clients can only create files with valid OpenVMS names. The server performs case-insensitive lookups and displays directories in lowercase.

- PURGE_VERSIONS, NOPURGE_VERSIONS

  Default: NOPURGE_VERSIONS

  Deletes multiple versions of files. (The NFS CREATE and RENAME calls can create multiple versions. The NFS READDIR call can sense multiple versions.)

- TYPELESS_DIRECTORIES, NOTYPELESS_DIRECTORIES

    - TYPELESS_DIRECTORIES

      Removes `.dir.1` from the name of directories. A naming conflict could arise if, for example, two files exist in the parent directory:

      `DOVE.;1` (regular file)

      `DOVE.DIR;1` (directory file)

      The name is returned as `dove.`, rather than `dove`, if a file and a conflicting directory exist.

    - NOTYPELESS_DIRECTORIES (default)

      Returns names as *`file.ext`* and *`file.dir`*.

Examples

1. `TCPIP> ADD EXPORT "/gold/finch" /HOST=GOLD`

   Adds the name of UNIX directory `/gold/finch` to the export database and gives NFS client users on the remote host `gold` access to this directory.

2. `TCPIP> ADD EXPORT "/gold/finch" /HOST=(PURPLE,FINCH)`

   Adds the directory `/gold/finch` to the export database and gives NFS client users on multiple remote hosts (PURPLE and FINCH) access to this directory.

ADD PROXY

ADD PROXY — Adds entries to the proxy database that give remote users an OpenVMS identity (account name). Applies to the NFS server, NFS client, PC-NFS, RSH, LPR/LPD, and customer-developed services.

Additional Information

The proxy database contains communication proxies and NFS proxies:

- Communication proxy

- Provides an identity for users of RSH, RLOGIN, REXEC, RMT/RCD, LPR/LPD, and customer-written services, if these services are marked with SET SERVICE / FLAGS=APPLICATION_PROXY.

Note

The ROOT account does not require a communication proxy in the proxy database. The /FLAGS=APPLICATION_PROXY flag, therefore, is not relevant when you are setting up a communication proxy for a UNIX root account.

- **Required qualifiers:** /HOST, /REMOTE_USER.

- NFS proxy
  - Provides an identity for users of NFS client, NFS server, PC-NFS. No two proxies can have the same UID and host combination.
  - **Required qualifiers:** /HOST, /GID, /UID.

You can combine NFS and communication proxies in one record.

Related commands: SHOW HOST, DCL command AUTHORIZE, CONVERT/VMS PROXY, REMOVE PROXY, SHOW PROXY

Format for Communication Proxies

```
ADD PROXY user { /HOST=host | /REMOTE_USER=user }
               [ /PERMANENT ]
```

Format for NFS Proxies

```
ADD PROXY user { /GID=n | /HOST=host | /UID=n }
               [ /NFS=options ]
               [ /PERMANENT ]
```

Restrictions and Tips

Requires read and write access to the proxy database and one of the following privileges:

- SYSPRV
- SYSLCK
- OPER

For each host, define both its host name and alias names.

For each entry, use uppercase and lowercase consistently.

Parameters

user

Required.

Local OpenVMS identity for the user of: NFS server, NFS client, PC-NFS, remote shell, or LPR/LPD.

Qualifiers

/GID=*n*

Required for an NFS proxy.

Group identifier (GID) for an NFS user.

Wildcards not allowed.

/HOST=*host*

Required.

Host name on which the NFS user is working.

- The host must be seen on the SHOW HOST/LOCAL display. It is important to use the SHOW HOST command to verify that the host is known to TCP/IP Services because ADD PROXY does not do this check.
- A wildcard is allowed.
- VSI recommends that you define both the host name and any alias names.

/NFS=[INCOMING | OUTGOING]

Optional. Default: /NFS=(INCOMING,OUTGOING).

Creates an NFS proxy for local clients, remote clients, or PC-NFS clients. Specify one of the following:

- /NFS=OUTGOING — Proxy to use NFS client
- /NFS=INCOMING — Proxy to use NFS server
- /NFS=(OUTGOING,INCOMING) — Proxy to use both NFS server and NFS client

/PERMANENT

Optional. Default: Both permanent and volatile databases.

Adds the entry only to the permanent proxy database.

- Communication proxy
  - If REMOTE_USER is not active, changes are made to the permanent database.
  - If REMOTE_user is active, changes to the permanent database take effect after you restart RSH, RLOGIN, or LPD.

/REMOTE_USER=*user*

Required for a communication proxy. Not valid on NFS proxy.

Specifies the remote client's user name.

A wildcard is allowed.

/UID=*n*

Required for an NFS proxy.

Specifies the user identifier (UID) for an NFS user.

Wildcards are not allowed.

Examples

1. ```
   TCPIP> ADD PROXY HAWAIIAN_GOOSE /NFS=(OUTGOING,INCOMING) -
   _TCPIP> /GID=10 /UID=444 /HOST="nene"
   ```

   Creates a proxy called HAWAIIAN_GOOSE, authorizing use of both the NFS client and the NFS server to and from host `nene`.

2. ```
   TCPIP> ADD PROXY COUSINS /GID=10 /UID=40 /NFS=OUTGOING -
   _TCPIP> /HOST=("grackle","blackbird")
   ```

   Enters an NFS proxy called COUSINS for a local NFS client user named COUSINS. This user can access remote files from NFS servers `grackle` and `blackbird`.

3. ```
   TCPIP> ADD PROXY REMOTE_UGLY_DUCKLING -
   _TCPIP> /REMOTE_USER="cygnet" /HOST="babyswan"
   ```

   Adds to the proxy database communication proxy REMOTE_UGLY_DUCKLING for remote user `cygnet` at host `babyswan`.

ANALYZE CONTAINER

ANALYZE CONTAINER — Verifies the integrity of a UNIX container file, including the following checks: Superblock validation, Inode validation, Directory validation, Internal allocation validation, and Directory hierarchy validation. Reports, and optionally corrects, problems within the structure of the container directory. Applies to the NFS server. *Related commands:* DIRECTORY, MAP, UNMAP.

Syntax

```
ANALYZE CONTAINER device:path.name
                [ /[NO]CONFIRM ]
                [ /[NO]REPAIR ]
```

Restrictions and Tips

- Before you enter the ANALYZE CONTAINER command, you must map the OpenVMS volume on which the container directory resides, using the MAP command.
- Before you enter the ANALYZE CONTAINER command, unmap the container file system to prevent users from attempting to access to it while you analyze it.
- This command requires the BYPASS privilege.
```

Parameters

device:path.name

Required.

Device and container directory to analyze.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /NOCONFIRM.

Use only with the /REPAIR qualifier.

When the software encounters a problem with the services database, it displays a description and solution and then requests confirmation before making each correction. For example:

```
%TCPIP-E-ANA_SUP_BADIICGSIZE, Problem: Bad initial inode cell group
size:bad_value
Solution: Will be replaced by default size: good_value
      CONFIRM [Y/N/G]:
```

Respond to the CONFIRM: prompt by entering one of the following:

- Y to repair the problem
- N to ignore the problem
- G to change to NO CONFIRMATION mode

/REPAIR
/NOREPAIR

Optional. Default: /NOREPAIR.

Any errors will be repaired.

Examples

```
TCPIP> UNMAP "/wren20"
TCPIP> ANALYZE CONTAINER DUA0:[WREN20]
```

Verifies the integrity of container DUA0:[WREN20].

ANALYZE MAIL

ANALYZE MAIL — Verifies the consistency of the SMTP queues with SMTP control files. *Related commands:* REMOVE MAIL, SHOW MAIL

Syntax

```
ANALYZE MAIL [ user ]
```

```
[ /[NO]CONFIRM ]
[ /DELETE[=options ]
[ /HOLD=time ]
[ /LOG=file ]
[ /[NO]REPAIR ]
```

Restrictions

Requires SYSNAM, SYSPRV, or BYPASS privilege to access mail that is not yours.

Parameters

user

Optional. Default: All users.

User whose mail you want to analyze.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /NOCONFIRM

Use only with either the /REPAIR or the /DELETE qualifier.

When the software encounters a problem, it displays a description and solution. If you specify the /CONFIRM qualifier, the software then requests confirmation before making a correction or deleting each record. Enter one of the following:

- Y to repair the problem
- N to ignore the problem
- G to change to NO CONFIRMATION mode

/DELETE [=BEFORE=*time* | =SINCE=*time*]

Optional. Default: Files not deleted.

Deletes each control file without a corresponding queue entry.

- =BEFORE=*time*
  - Deletes files created before the specified time.
  - Default: Current date and time.
- =SINCE=*time*
  - Deletes files created since the specified time.
  - Default: Deletes all files.

Use the /DELETE and /REPAIR qualifiers on the same command line only if their time frames do not conflict.

The following command requeues lost mail created since yesterday and deletes all previous mail:

```
TCPIP> ANALYZE MAIL /REPAIR /DELETE=BEFORE=YESTERDAY
```

/HOLD=*time*

Optional. Default: Immediate retransmission.

Hold, until the specified time, lost control files that you requeued.

/LOG[=*file*]

Optional. Default: [*current_default_dir*]:TCPIP$SMTP_ANALYZE.LOG.

Writes the ANALYZE MAIL log to the specified file.

/REPAIR
/NOREPAIR [=BEFORE=*time* | =SINCE=*time*]

Optional. Default: /NOREPAIR.

Corrects errors as follows:

- Resubmits for delivery each valid control file in the SMTP directory with no entry in an SMTP queue.
- Deletes each invalid control file (fails the internal consistency check) and the corresponding queue entry.
- Either requeues or deletes messages placed on hold.

Supports the following options:

- =BEFORE=*time*
  - Deletes files created before the specified time.
  - Default: Current date and time.
- =SINCE=*time*
  - Deletes files created since the specified time.
  - Default: Deletes all files.

Do not use /REPAIR with /DELETE if their time frames conflict.

Examples

1. `TCPIP> ANALYZE MAIL /REPAIR`

   Displays status for the SMTP queues, and requeues each valid control file that lacks a corresponding queue entry.

2. `TCPIP> ANALYZE MAIL /DELETE`

   Creates the summary of SMTP queues, and deletes each valid control file that lacks a corresponding SMTP queue entry.

3. `TCPIP> ANALYZE MAIL DRAKE /REPAIR /DELETE=BEFORE=24-APR-2003`

   This command does the following:

   - Creates a summary of SMTP entries and control files for user DRAKE.
   - Requeues control files that lack corresponding queue entries.
   - Deletes control files created before April 24, 2003.

ANALYZE SERVICE

ANALYZE SERVICE — Searches through the services database for corrupted definitions. Displays invalid records and, with the /REPAIR qualifier, deletes them. *Related commands:* SET SERVICE, SHOW SERVICE

Syntax

```
ANALYZE SERVICE [ /[NO]CONFIRM ]
                [ /[NO]REPAIR ]
```

Restrictions

Requires write access to the directory with the services database.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /NOCONFIRM.

Use only with the /REPAIR qualifier.

When the software encounters a problem, it displays a description and a solution. If you specify the /CONFIRM qualifier, the software then requests confirmation before making a correction.

Respond to the CONFIRM: prompt by entering one of the following:

- Y to repair the problem
- N to ignore the problem

/REPAIR
/NOREPAIR

Optional. Default: /NOREPAIR

Deletes the corrupted records.

Examples

1. `TCPIP> ANALYZE SERVICE`

```
                    Invalid IP option records
    Service             Port  Proto    Process          Address

    TOE                   25  TCP      TOED             0.0.0.0
    NESTING              560  TCP      NEW_EGGS_TCPIP   0.0.0.0

IP option records
  Total:              0
  Invalid:            0

TCP option records
  Total:              7
  Invalid:            2
```

Displays total and invalid protocol option records found in two service definitions, TOE and NESTING.

2. `TCPIP> ` **`ANALYZE SERVICE /REPAIR /CONFIRM`**

```
                    Invalid IP option records
    Service             Port  Proto    Process          Address

    TOE                   67  UDP      TOED             0.0.0.0

Remove? [N]: YES


    Service             Port  Proto    Process          Address

    NESTING               69  UDP      NEW_EGGS_TCPIP   0.0.0.0

Remove? [N]: YES
```

Displays the total protocol option records and deletes, after confirmation, the invalid records.

CONVERT/CONFIGURATION_BIND

CONVERT/CONFIGURATION_BIND — Converts the UCX BIND Version 4.*x* name server configuration to the BIND Version 8.1 format. Applies to the BIND name server. Use this command if you have a BIND configuration from an earlier release (Version 4.2 or lower) of the TCP/IP Services software. This command extracts the BIND configuration information from the file UCX $CONFIGURATION.DAT and creates the ASCII file TCPIP$BIND.CONF. *Related commands:* SET CONFIGURATION BIND, SHOW CONFIGURATION BIND

Syntax

```
CONVERT/CONFIGURATION BIND [bind_conf_file]
                           [/CLUSTER=lbroker_conf_file]
```

Parameters

bind_conf_file

Optional. Default: SYS$SPECIFIC:[TCPIP$BIND]TCPIP$BIND.CONF.

Specifies the alternate name or location of the BIND configuration file to be created.

Qualifiers

/CLUSTER=*lbroker_conf_file*

Optional. Default: SYS$SYSDEVICE:[TCPIP$LD_BKR]TCPIP$LBROKER.CONF

Creates the file used by the load broker for cluster load balancing. Here, *lbroker_conf_file* specifies the name of the load broker configuration file.

Examples

1.
```
TCPIP> CONVERT /CONFIGURATION  BIND -
_TCPIP> /CLUSTER=SYS$SYSDEVICE:[TCPIP$LD_BKR]TCPIP$LBROKER.CONF
```

   Converts the UCX BIND server configuration to BIND Version 8.1 format and creates the configuration file SYS$SPECIFIC:[TCPIP$BIND]TCPIP$BIND.CONF. The /CLUSTER qualifier creates the configuration file TCPIP$LBROKER.CONF used by the load broker.

2.
```
TCPIP> CONVERT /CONFIGURATION  BIND SITE2_BIND.CONF
```

   Converts the UCX BIND server configuration and creates the configuration file SITE2_BIND.CONF.

Refer to the VSI TCP/IP Services for OpenVMS Management manual for more information about the BIND name server, resolver, and load broker.

CONVERT/UNIX BIND

CONVERT/UNIX BIND — Creates a BIND server database and populates it with records from the local host and MX databases. This command will create either a forward translation file or a reverse translation file. If you specify a *domain.name* that ends in `IN-ADDR.arpa`, a reverse translation file is created. *Related commands:* SET HOST, SET MX_RECORD.

Syntax

```
CONVERT/UNIX BIND /DOMAIN=domain.name
                  [ /[NO]LOG ]
```

Qualifiers

/DOMAIN=*domain.name*

Required.

Domain for which to extract data. Determines whether to perform forward translation or reverse translation. The following restrictions and instructions apply:

- The domain name must be fully qualified.
- The closing dot is not required.

- Do not use wildcards.
- Specify up to four parts of the IP address for varying degrees of selectivity. For example:

  /DOMAIN=16.IN-ADDR.arpa is less selective.

  /DOMAIN=8.20.16.IN-ADDR.arpa is more selective.

The end of *domain.name* determines whether a forward or reverse translation is performed. For example:

- If the domain name ends in IN-ADDR.arpa:
  - Reverse translation is performed.
  - The domain is some part of an IP address, reversed, and added to IN-ADDR.arpa.
  - The selection includes the contents of the hosts database in the output.
- If the domain name ends in anything else (for example, `/DOMAIN=DAW.MAG.COM`):
  - Forward translation is performed.
  - All hosts in *domain.name* or in any of its subdomains are selected for the output file.
  - CONVERT/UNIX BIND does a forward translation and selects hosts in DAW.MAG.COM and in its subdomains.

/LOG
/NOLOG

Optional. Default: /NOLOG.

Shows records as they are processed.

Examples

1. `TCPIP>` **`CONVERT/UNIX BIND /DOMAIN=KESTREL.SMALL.FALCON`**

   On host KESTREL, creates a BIND server database with default file name SYS$SPECIFIC:[TCPIP$BIND]KESTREL_SMALL_FALCON.DB.

   BIND and MX records for the host's domain, `kestrel.small.falcon`, are extracted, converted, and written to KESTREL_SMALL_FALCON.DB.

2. `TCPIP>` **`CONVERT/UNIX BIND /LOG /DOMAIN=ABC.COM`**

   Creates a BIND server database. The /LOG qualifier indicates that records will be displayed as they are processed.

CONVERT/UNIX HOST

CONVERT/UNIX HOST — Reads the hosts database and converts the information to an ASCII file formatted for use as a hosts file on a UNIX system. The name and location of the hosts database is specified by the logical name TCPIP$HOST. If this name is not defined, the command looks for

TCPIP$HOST.DAT in your current directory. *Related commands:* CREATE HOST, SET HOST, SHOW HOST.

Syntax

```
CONVERT/UNIX HOST [ destination_file ]
                  [ /BYADDRESS ]
                  [ /LOG ]
```

Restrictions

Requires:

- Read access to the hosts database.
- Write access to []ETC.HOSTS.

Parameters

destination_file

Optional. Default: []ETC.HOSTS (UNIX formatted)

Allows you to specify the name for the new file.

Qualifiers

/BYADDRESS

Optional. Default: Sorts by name.

Sorts entries in the converted file by IP address.

/LOG

Optional. Default: No display.

Interactively displays the processing.

Examples

```
TCPIP> CONVERT/UNIX HOST
```

Converts the hosts database TCPIP$HOST to an ASCII file that is formatted for use as a hosts file on a UNIX system. The resulting file is named (default) ETC.HOSTS in the current directory and can be used on a UNIX system as the file `/etc/hosts`.

CONVERT/UNIX NETWORK

CONVERT/UNIX NETWORK — Converts the networks database to an ASCII file formatted for use on a UNIX system. The name and location of the networks database is specified by the logical name TCPIP$NETWORK. If this name is not defined, the command looks for TCPIP$NETWORK.DAT in your current directory. *Related commands:* CREATE NETWORK, SET NETWORK, SHOW NETWORK.

Syntax

```
CONVERT/UNIX NETWORK [ destination_file ]
                     [ /BYADDRESS ]
                     [ /LOG ]
```

Restrictions

Requires:

- Read access to the hosts database.
- Write access to the file []ETC.NETWORKS.

Parameters

destination_file

Optional. Default: []ETC.NETWORKS (UNIX formatted).

Specifies the name of the new UNIX formatted file.

Qualifiers

/BYADDRESS

Optional. Default: Sorts by name.

Sorts the converted file by network number.

/LOG

Optional. Default: No display.

Displays messages generated during processing.

Examples

1. `TCPIP> CONVERT/UNIX NETWORK /LOG`

   Converts the networks database (TCPIP$NETWORK) to the ASCII file []ETC.NETWORKS. This resulting file can be used as a networks database file `/etc/networks` on a UNIX system. The /LOG qualifier displays each record as it is converted.

2. `TCPIP> CONVERT/UNIX NETWORK NETWORKS.TXT`

   Converts the networks database (TCPIP$NETWORK) to the ASCII file []NETWORKS.TXT. The resulting file may be used as the networks database file `/etc/networks` on a UNIX system.

CONVERT/VMS BOOTP

CONVERT/VMS BOOTP — Populates the existing BOOTP database with entries from a BIND-formatted UNIX `/etc/bootptab` file. If the logical name TCPIP$BOOTP is defined, it is used to specify the directory and file name for the database. If TCPIP$BOOTP is not defined, the database

is created as [*current_directory*]TCPIP$BOOTP.DAT. *Related commands:* CREATE BOOTP, SET BOOTP, SHOW BOOTP.

Syntax

```
CONVERT/VMS BOOTP [ source_file ]
                  [ /ADD_HOST ]
                  [ /FILE=sys_image_file ]
```

Restrictions

Requires:

- Read and write access to the hosts database (if using /ADD_HOST qualifier).
- Read access to the hosts database.

Parameters

source_file

Optional. Default: []ETC.BOOTPTAB in your current directory.

File to be converted.

Qualifiers

/ADD_HOST

Optional. Default: No adding.

Adds new host names found in the UNIX `/etc/bootptab` file to TCPIP$HOST.DAT.

/FILE=*sys_image_file*

Optional. Default: None.

Specifies the name of the client's system image file to download if this file name is not in the BOOTP database.

Examples

```
TCPIP> CONVERT/VMS BOOTP BOOTP.DAT /ADD_HOST
```

Converts a UNIX ASCII boot data file to an OpenVMS indexed file. BOOTP.DAT specifies the source UNIX boot file to convert. For new hosts, the /ADD_HOST qualifier adds the host to the hosts database.

CONVERT/VMS HOST

CONVERT/VMS HOST — Populates the existing hosts database with entries from a UNIX `/etc/hosts` file. The name and location of the hosts database is specified by the logical name TCPIP $HOST. If this name is not defined, the command looks for TCPIP$HOST.DAT in your current directory. *Related commands:* CREATE HOST, SET HOST, SHOW HOST.

Syntax

```
CONVERT/VMS HOST [ source_file ]
                 [ /LOG ]
                 [ /UPCASE ]
```

Restrictions

Requires:

- Read and write access to the hosts database.
- Read access to the UNIX formatted hosts file.
- Exclusive use of the hosts database.

Parameters

source_file

Optional. Default: ETC.HOSTS in your current directory.

UNIX formatted file to be converted to the TCPIP$HOST database file.

Qualifiers

/LOG

Optional. Default: No logging.

Displays records as they are being processed.

/UPCASE

Optional. Default: Not created.

Creates an uppercase alias name for each host.

Examples

1. `TCPIP> CONVERT/VMS HOST`

   Converts the UNIX formatted hosts database file to the file TCPIP$HOST.DAT. The name of the UNIX formatted file is ETC.HOSTS in your current directory.

2. `TCPIP> CONVERT/VMS HOST HOSTS.TXT`

   Converts the UNIX formatted hosts database file HOSTS.TXT to the file TCPIP$HOST.DAT.

CONVERT/VMS NETWORK

CONVERT/VMS NETWORK — Populates the existing networks database with entries from a UNIX `/etc/networks` file. The name and location of the networks database is specified by

the logical name TCPIP$NETWORK. If this name is not defined, the command looks for TCPIP $NETWORK.DAT in your current directory. *Related commands:* CREATE NETWORK, SET NETWORK, SHOW NETWORK.

Syntax

```
CONVERT/VMS NETWORK [ source_file ]
                    [ /LOG ]
                    [ /UPCASE ]
```

Restrictions

Requires:

- Read and write access to the networks database.
- Read access to []ETC.NETWORKS.
- Exclusive use of the networks database.

Parameters

source_file

Optional. Default: ETC.NETWORKS in your current directory.

Name of the file to be converted.

Qualifiers

/LOG

Optional. Default: Log file created.

Interactively displays records as they are being processed.

/UPCASE

Optional. Default: Alias not created.

Specifies that an uppercase alias name be created for each network name.

Examples

1. `TCPIP>` **`CONVERT/VMS NETWORK`**

   Converts a UNIX formatted `/etc/networks` database file into a TCP/IP Services formatted networks database. The OpenVMS file name of `/etc/networks` is ETC.NETWORKS in your current directory.

2. `TCPIP>` **`CONVERT/VMS NETWORK /UPCASE`**

   Converts a UNIX formatted database file to a TCP/IP Services formatted networks database. An uppercase alias name is created for each network name.

CONVERT/VMS PROXY

CONVERT/VMS PROXY — Populates the existing proxy database with entries from a UNIX `/etc/passwd` file. The name of the proxy database is specified by the logical name TCPIP$PROXY. If this name is not defined, the command looks for TCPIP$PROXY.DAT in your current directory. *Related commands:* ADD PROXY, CREATE PROXY. *Applies to:* NFS server, NFS client, PC-NFS.

Syntax

```
CONVERT/VMS PROXY [ source_file ]
                  [ /LOG ]
```

Syntax

Requires:

- Read and write access to the proxy database.
- Read access to []ETC.PASSWORD.
- Exclusive use of the proxy database.

Parameters

source_file

Optional. Default: ETC.PASSWD in your current directory.

ASCII file to convert to a TCP/IP Services proxy database.

Qualifiers

/LOG

Optional. Default: No display.

Displays records as they are being processed.

Examples

1. `TCPIP> CONVERT/VMS PROXY`

   Converts a UNIX formatted `/etc/passwd` file to an OpenVMS formatted proxy database. The OpenVMS file name of `/etc/passwd` is ETC.PASSWD in your current directory.

2. `TCPIP> CONVERT/VMS PROXY UNIX_PASSWDS.TXT`

   Converts a UNIX formatted `/etc/passwd` file to an OpenVMS formatted proxy database. In this example, the file UNIX_PASSWDS.TXT contains the `/etc/passwd` data.

CREATE BOOTP

CREATE BOOTP — Creates the BOOTP database file, using the file name and location specified by the logical name TCPIP$BOOTP. If the logical name is not defined, creates the database file in

your current directory as TCPIP$BOOTP.DAT. *Related commands:* CONVERT/VMS BOOTP, SET BOOTP.

Syntax

```
CREATE BOOTP
```

Restrictions

Requires write access to the directory with the BOOTP configuration database.

Caution

Do not execute this command unless you intend to reconfigure your entire cluster.

Examples

```
TCPIP> CREATE BOOTP
```

Creates an empty BOOTP database.

CREATE CONFIGURATION

CREATE CONFIGURATION — Creates the configuration database file, using the file name and location specified by the logical name TCPIP$CONFIGURATION. If the logical name is not defined, creates the database file in your current directory as TCPIP$CONFIGURATION.DAT.

Syntax

```
CREATE CONFIGURATION
```

Restrictions

Requires write access to the directory with the configuration database.

Caution

Do not execute this command unless you intend to reconfigure your entire cluster.

Examples

```
TCPIP> CREATE CONFIGURATION
```

Creates an empty configuration database.

CREATE CONTAINER

CREATE CONTAINER — Creates a UNIX file system with: an empty OpenVMS style root directory, an empty local directory that corresponds to the UNIX root directory, and a container file in the OpenVMS style root directory. *Applies to:* NFS server

Syntax

```
CREATE CONTAINER device:directory
                 [ /HOST=host ]
                 [ /[NO]LOG ]
                 [ /OWNER=[uic] ]
                 [ /ROOT_MODE=n ]
                 [ /SIZE=option=value ]
                 [ /UID=n ]
                 [ /USER_NAME=vms_user_name ]
```

Restrictions

Requires:

- Read and write access to the specified device and directory.
- SYSPRV or BYPASS privilege.

Parameters

device:*directory* CONTAINER command)

Required.

Device and directory of the UNIX container.

Qualifiers

/HOST=*host*

Required. Default: None.

If the proxy database has multiple entries with the same user name and UID, the NFS server selects the entry specified with this qualifier.

/LOG
/NOLOG

Optional. Default: Displays host, UID, GID, and user name.

Displays a full description of the specified proxy database record for you to determine ownership (see /USER_NAME).

/OWNER=[*uic*]

Optional. Default: UIC in the selected proxy record.

OpenVMS ownership of the container file directory and container file.

The other files in this directory are owned by the OpenVMS users whose proxy database entries correspond to the UNIX owner UIDs of the individual files.

/ROOT_MODE=*n*

Optional. Default: 755 (provides the following protection for owner, group, and world rwx-rx-rx).

UNIX protection of the default container files: root directory, bit map, and superblock.

Specify octal values in the following order: for user, for group, for others. The values are:

- 0 — No access
- 1 — Execute access
- 2 — Write access
- 3 — Write and execute access
- 4 — Read access
- 5 — Read and execute access
- 6 — Read and write access
- 7 — Read, write, and execute access

For example, /ROOT_MODE=751 provides:

User	Read, write, and execute access	7	rwx
Group	Read and execute access	5	rx
Other	Execute access	1	x

/SIZE=*option=value*

Optional. Default: /SIZE=(INITIAL=8917,EXTEND=160,MAXIMUM=0).

Specifies the following file size attributes:

- INITIAL

  Specifies the initial size, in OpenVMS blocks, of the container. Maximum value: 8917.

- EXTEND

  When an extension is necessary, specifies the number of blocks by which the container is extended. Maximum value: 8192.

Note

Using large values might impact system performance and disk usage. In most cases, the default is sufficient.

- MAXIMUM

  Specifies the maximum size of the container.

  If the value is zero, the size of the container file can increase without limits.

/UID=*n*

Optional. Default: UID in the selected proxy record.

Specifies the owner of the UNIX container root directory.

/USER_NAME=*vms_user_name*

Required.

Specifies the user name of the owner of the container file system. The user name must be in the proxy database. The specified user becomes the owner of the internal root directory of the container.

Examples

```
TCPIP> CREATE CONTAINER DUCK$4:[DUCKLING] /HOST=MALLARD -
_TCPIP> /OWNER=[300,12] /ROOT_MODE=755 /UID=7015 -
_TCPIP> /USER_NAME=G_JONES
```

Creates container directory DUCK$4:[DUCKLING]. The local OpenVMS owner is [300,12]. Remote users see the root directory as owned by UID 7015. The root directory is writable only by UID 7015; it is readable and executable by all others. Before you execute this command, user G_JONES should have both an OpenVMS account with UIC [300,12] and an incoming proxy record specifying UID 7015 on host MALLARD.

CREATE DIRECTORY

CREATE DIRECTORY — Creates a directory within an existing UNIX container. *Applies to:* NFS server. **Related commands:** DIRECTORY, REMOVE DIRECTORY

Syntax

```
CREATE DIRECTORY "/path/name"
                 [ /HOST=host ]
                 [ /[NO]LOG ]
                 [ /MODE=n ]
                 [ /UID=n ]
                 [ /USER_NAME=vms_user_name ]
```

Restrictions

Requires:

- Read and write access to the parent directory.
- SYSPRV or BYPASS privilege if you specify /USER_NAME with a name other than your own.

The container file system must be mapped with the MAP command.

Parameters

"*/path/name*"

Required.

Name of the directory you want to create.

Qualifiers

/HOST=*host*

Optional. Default: Uses the first user name found.

If the proxy database has multiple entries with the same user name and UID combination, the value of /HOST determines the specified record.

/LOG
/NOLOG

Optional. Default: Displays values for host, UID, GID, and user name.

Displays a full description of the specified proxy database record for you to determine ownership (see /USER_NAME).

/MODE=*n*

Optional. Default: 755 (provides the following protection for owner, group, and world: rwx-rx-rx).

UNIX protection of the new directory.

Specify octal values in the following order: for user, for group, for others. The values are:

- 0 — No access
- 1 — Execute access
- 2 — Write access
- 3 — Write and execute access
- 4 — Read access
- 5 — Read and execute access
- 6 — Read and write access
- 7 — Read, write, and execute access

For example, /MODE=751 provides:

User	Read, write, and execute access	7	rwx
Group	Read and execute access	5	rx
Other	Execute access	1	x

/UID=*n*

Optional. Default: None.

Entry in the proxy database that determines, if necessary, the ownership of the container root directory.

In the proxy database:

- UID and GID fields determine the root directory's UNIX identity.
- User name field determines the OpenVMS ownership.

Required to access an entry in the proxy database that lacks a unique UID, user name, and host combination.

You can use /UID with the /HOST and /USER_NAME qualifiers. If you do not have SYSPRV or BYPASS privilege, the values you specify must correspond to the values for your user name in the proxy database.

/USER_NAME=*vms_user_name*

Optional. Default: UID=0 and GID=1 (if you have SYSPRV or BYPASS privilege).

Selects an entry in the proxy database and creates the UID, GID, and OpenVMS UIC for the directory files.

To select a user name that has a UIC different than the UIC of the process running the management program, you need SYSPRV or BYPASS privilege.

You can use /USER_NAME in combination with /HOST and /UID. However, if you do not have SYSPRV or BYPASS privilege, the values you specify must correspond to the values for your user name in the proxy database.

Examples

```
TCPIP> MAP "/user" dua0:[group_a]
TCPIP> CREATE DIRECTORY "/user/umbrella.bird"
```

Creates a UNIX directory for user UMBRELLA called `umbrella.bird.`

In this example, the user UMBRELLA is running the TCP/IP Services management program from the directory ([UMBRELLA.BIRD]). The UIC for [UMBRELLA] is [340,6] and the TCPIP$PROXY entry is defined as follows:

User	UID	GID	Host
UMBRELLA	300	12	*
SYSTEM	0	1	*

If UMBRELLA does not have SYSPRV or BYPASS privilege, the directory is created as follows:

UID = 300
GID = 12
UIC = [340,6]

If UMBRELLA has SYSPRV or BYPASS privilege, the directory is created as follows:

UID = 0
GID = 1
UIC = [SYSTEM]

CREATE EXPORT

CREATE EXPORT — Creates the export database file, using the file name and location specified by the logical name TCPIP$EXPORT. If the logical name is not defined, creates the database file in your current directory as TCPIP$EXPORT.DAT. *Related commands:* ADD EXPORT, SHOW EXPORT. *Applies to:* NFS server.

Syntax

```
CREATE EXPORT
```

Restrictions

Requires write access to the directory with the export database.

Caution

Do not execute this command unless you intend to reconfigure your entire cluster.

Examples

```
TCPIP> CREATE EXPORT
```

Creates an empty export database.

CREATE HOST

CREATE HOST — Creates a hosts database file with: One entry for LOCALHOST, LOCALHOST's alias, *localhost*, and LOCALHOST's address, 127.0.0.1. The hosts database file name and location are specified by the logical name TCPIP$HOST. If the logical name is not defined, the database file name will be TCPIP$HOST.DAT in your current directory. **Related commands:** SET HOST, SHOW HOST, CONVERT/VMS HOST.

Syntax

```
CREATE HOST
```

Restrictions

Requires:

- Write access to the directory with the hosts database.
- Read and write access to the hosts database.

Caution

Do not execute this command unless you intend to reconfigure your entire cluster.

Examples

```
TCPIP> CREATE HOST
```

Creates a hosts database with one entry for LOCALHOST.

CREATE NETWORK

CREATE NETWORK — Creates the networks database file, using the file name and location specified by the logical name TCPIP$NETWORK. If the logical name is not defined, creates the database file in your current directory as TCPIP$NETWORK.DAT. *Related commands:* SET NETWORK, SHOW NETWORK, CONVERT/VMS NETWORK

Syntax

```
CREATE NETWORK
```

Restrictions

Requires write access to the directory with the networks database.

Caution

Do not execute this command unless you intend to reconfigure your entire cluster.

Examples

```
TCPIP> CREATE NETWORK
```

Creates an empty networks database.

CREATE PROXY

CREATE PROXY — Creates the proxy database file, using the file name and location specified by the logical name TCPIP$PROXY. If the logical name is not defined, creates the database file in your current directory as TCPIP$PROXY.DAT. *Related commands:* ADD PROXY, SHOW PROXY, CONVERT/VMS PROXY.

Syntax

```
CREATE PROXY
```

Restrictions

Requires write access to the directory with the proxy database.

Caution

Do not execute this command unless you intend to reconfigure your entire cluster.

Examples

```
TCPIP> CREATE PROXY
```

Creates an empty proxy database.

CREATE ROUTE

CREATE ROUTE — Creates the routes database file, using the file name and location specified by the logical name TCPIP$ROUTE. If the logical is not defined, creates the database file named TCPIP $ROUTE.DAT in your current directory. **Related commands:** SET ROUTE, SHOW ROUTE.

Syntax

```
CREATE ROUTE
```

Restrictions

Requires write access to the directory with the routes database.

Caution

Do not execute this command unless you intend to reconfigure your entire cluster.

Examples

```
TCPIP> CREATE ROUTE
```

Creates an empty routes database.

DEFINE COMMUNICATION_CONTROLLER

DEFINE COMMUNICATION_CONTROLLER — Defines the mapping between a communication controller device and its corresponding Internet interface. Each mapping or controller definition is stored as a record in the configuration database.

Additional Information

To modify an existing controller definition, you must delete the old controller definition from the configuration database (using the DELETE COMMUNICATION_CONTROLLER command) and then define the new controller definition (using the DEFINE COMMUNICATION_CONTROLLER command).

Internet devices are uniquely identified using 2-character names. The first character is determined by the /INTERNET_INTERFACE qualifier. The second character is determined by the controller type you specify with the /TYPE qualifier, as follows:

Controller Type	Second Character of Interface Name
ETHERNET	E
FDDI	F
PPP	P
SERIAL	L
TOKEN_RING	T

For example, with the following command, the communication controller EW maps to the interface WE:

```
TCPIP> DEFINE COMMUNICATION_CONTROLLER EW -
_TCPIP> /INTERNET_INTERFACE=W /TYPE=ETHERNET
```

Related commands: LIST COMMUNICATION_CONTROLLER, DELETE COMMUNICATION_CONTROLLER, all INTERFACE commands

Syntax

```
DEFINE COMMUNICATION_CONTROLLER controller
                                [ /DESCRIPTION=text ]
                                /INTERNET_INTERFACE=character
                                /TYPE=(option[,...])
```

Parameters

controller

Required.

Specifies the OpenVMS device name of the communication controller (as displayed by the DCL command SHOW DEVICE) to be mapped to an Internet interface. For examples of communication controllers with their corresponding Internet interfaces, refer to the LIST COMMUNICATION_CONTROLLER command. For more information on configuring network interfaces, refer to the VSI TCP/IP Services for OpenVMS Management manual.

Qualifiers

/DESCRIPTION=*text*

Optional. Default: None.

Optional text describing the communication controller.

/INTERNET_INTERFACE=*character*

Required.

Specifies the first character of the Internet interface name. If you prefer using a standard name, call your VSI support representative.

/TYPE=(*option*[,...])

Required.

Specifies the communication controller type and cluster attribute. You must specify one of the following communication controller types: ETHERNET, FDDI, PPP, SERIAL, or TOKEN_RING.

Optionally, specify the CLUSTER attribute to indicate that the interface can join an Internet cluster. To enable a cluster alias (alias host identifier) with an interface, use the SET INTERFACE /CLUSTER command.

Examples

```
TCPIP> DEFINE COMMUNICATION_CONTROLLER ES -
_TCPIP> /INTERNET_INTERFACE=S /TYPE=(ETHERNET,CLUSTER) -
_TCPIP> /DESCRIPTION="DESVA-Class Ethernet Adapter"
```

Defines the OpenVMS device ES as the Internet interface SE, which can join an Internet cluster.

DELETE COMMUNICATION_CONTROLLER

DELETE COMMUNICATION_CONTROLLER — Deletes communication controller definitions from the configuration database. **Related commands:** DEFINE COMMUNICATION_CONTROLLER, LIST COMMUNICATION_CONTROLLER

Syntax

```
DELETE COMMUNICATION_CONTROLLER [ controller ]
                                [ /[NO]CONFIRM ]
                                [ /INTERNET_INTERFACE=character ]
```

Restrictions

Requires OPER privilege.

Parameters

controller

Required.

Specifies the OpenVMS device name of the communication controller.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM if you use wildcards; otherwise, /NOCONFIRM.

If you specify the /CONFIRM qualifier, a message displays asking you to confirm the delete request. Respond to the CONFIRM: prompt by entering one of the following:

- Y to delete the entry
- N to retain the entry

If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

/INTERNET_INTERFACE=*character*

Optional. Default: All alphabetic characters.

Specifies the first character of the Internet interface name of communication controller definitions to delete from the configuration database.

Examples

1. `TCPIP> DELETE COMMUNICATION_CONTROLLER EZ`

   Deletes from the configuration database the communication controller definition corresponding to the OpenVMS device EZ.

2. `TCPIP> DELETE COMMUNICATION_CONTROLLER * -`
   `_TCPIP> /INTERNET_INTERFACE=W`

   Deletes all the communication controller definitions having an Internet interface name that begins with the letter W.

DELETE CONTAINER

DELETE CONTAINER — Deletes a container file system and all its contents. **Applies to:** NFS server. *Related commands:* CREATE CONTAINER.

Syntax

`DELETE CONTAINER` *`container_file_system`*

Restrictions

Wildcards are not allowed.

Requires both read and delete access to the directory.

Requires BYPASS privilege.

Parameters

container_file_system

Required.

Device and directory name of the container file (no wildcards).

The container file has file type .CONTAINER.

Examples

`TCPIP> DELETE CONTAINER WORK1$:[DOVE.NEST_BUILDING]`

Deletes the container directory WORK1$:[DOVE.NEST_BUILDING] along with the container file, all subdirectories, and files.

DIRECTORY

DIRECTORY — Displays a list of files, along with typical directory information, in a UNIX container directory. **Applies to:** NFS server.

Syntax

```
DIRECTORY "/path/name"
          [ /FULL ]
          [ /VMS ]
```

Restrictions

Requires:

- Read access to the specified container directory.
- BYPASS privilege.

Parameters

"*/path/name*"

Required.

Name of the UNIX container directory for which you want a directory listing and, optionally, directory names.

Qualifiers

/FULL

Optional. Default: Brief display.

Displays a comprehensive list of information, including the OpenVMS file name, for each file.

/VMS

Optional. Default: No OpenVMS file names provided.

Provides the corresponding OpenVMS file name for each file.

Examples

1. TCPIP> **DIRECTORY/FULL "/nest_container"**

```
Directory:  /nest_container

 .
 OpenVMS file: _$1$DISK:[SYSTEM.NEST.HATCHLING]00012301$BFS.DIR;1
 Size                                File ID:   74497
  Blocks:            4               Owner
  Bytes:          1915                UID:          0
 Created:   1-NOV-2002 13:17:18.91    GID:          1
 Revised:   1-NOV-2002 13:17:19.24   Mode:        755  Type: Directory
 Accessed:  1-NOV-2002 13:16:20.52   Links:         2

     ..
 OpenVMS file: _$1$DISK:[SYSTEM.NEST.HATCHLING]00012301$BFS.DIR;1
 Size                                File ID:   74497
```

```
 Blocks:            4               Owner
 Bytes:          1915                UID:          0
Created:   1-NOV-2002 13:17:18.91    GID:          1
Revised:   1-NOV-2002 13:17:19.24   Mode:        755  Type: Directory
Accessed:  1-NOV-2002 13:16:20.52   Links:         2

    .SUPER.SYS
OpenVMS file: no corresponding file
Size                                File ID:    6145
 Blocks:            1               Owner
 Bytes:            54                UID:          0
Created:   1-NOV-2002 13:17:18.91    GID:          1
Revised:   1-NOV-2002 13:17:17.24   Mode:        644  Type: File
Accessed:  1-NOV-2002 13:16:18.52   Links:         1

    .BITMAP.SYS
OpenVMS file: no corresponding file
Size                                File ID:    6657
 Blocks:           16               Owner
 Bytes:          8187                UID:          0
Created:   1-NOV-2002 13:17:18.91    GID:          1
Revised:   1-NOV-2002 13:17:17.24   Mode:        644  Type: File
Accessed:  1-NOV-2002 13:16:18.52   Links:         1

    .HISTORY.SYS
OpenVMS file: no corresponding file
Size                                File ID:   66305
 Blocks:            1               Owner
 Bytes:           129                UID:          0
Created:   1-NOV-2002 13:17:18.91    GID:          1
Revised:   1-NOV-2002 13:17:17.24   Mode:        644  Type: File
Accessed:  1-NOV-2002 13:16:18.52   Links:         1
```

Displays a full directory listing of the container file system `/nest_container`.

2. `TCPIP> DIRECTORY "/dove/nest/plans"`

   Displays names of the files in UNIX directory `/dove/nest/plans`.

DISABLE SERVICE

DISABLE SERVICE — For most services, this command disables the specified service but does not stop the current process. This allows you to perform an orderly shutdown of the service, which prevents new connections while allowing current connections to continue. To stop and restart the current process, first wait until the process exits, or stop it using the service-specific shutdown command procedure (TCPIP$*service*_SHUTDOWN.COM), then restart the service using the service-specific startup command procedure (TCPIP$*service*_STARTUP.COM). Note that, for the NFS server, TELNET, and RLOGIN, the DISABLE SERVICE command stops the current process. *Related commands:* ENABLE SERVICE, SET SERVICE, SHOW SERVICE

Syntax

```
DISABLE SERVICE service
                [ /ADDRESS=IP_address ]
                [ /PORT=n ]
                [ /PROCESS=process ]
```

```
[ /PROTOCOL=protocol ]
```

Parameters

service

Required.

Service you want to disable. Specify any service that appears in the SHOW SERVICE display. To disable all services, use a wildcard.

Qualifiers

/ADDRESS=*IP_address*

Optional. Default: 0.0.0.0.

Disables only the services for the specified address.

/PORT=*n*

Optional. Default: All ports.

Disables the service communicating at the specified port.

/PROCESS=*process*

Optional. Default: All processes.

Disables the service running as the specified process.

/PROTOCOL=*protocol*

Optional. Default: All protocols.

Disables only the services that use the specified protocol.

Examples

1. `TCPIP> DISABLE SERVICE TELNET`

   Disables TELNET.

2. `TCPIP> DISABLE SERVICE RLOGIN /ADDRESS=130.180.4.7`

   Disables the remote login process that is bound to address 130.180.4.7.

DISCONNECT DEVICE_SOCKET

DISCONNECT DEVICE_SOCKET — Interactively terminates a TCP/IP connection.

Syntax

`DISCONNECT DEVICE_SOCKET` *`dev_sock_number`*

Parameters

dev_sock_number

Required.

Number of the device socket associated with the connection you want to terminate.

Examples

```
TCPIP> DISCONNECT DEVICE_SOCKET BG123
```

Interactively terminates the connection at DEVICE_SOCKET BG123.

DISMOUNT

DISMOUNT — Makes a physically remote file system that is currently accessible to local users inaccessible. Dismounts a remote file system or directory from local device DNFS*n*: (the mount point). *Related commands:* MOUNT, SHOW MOUNT. *Applies to:* NFS client.

Syntax

```
DISMOUNT { mount_point | logical_name }
         [ /ALL ]
         [ /HOST=host ]
         [ /[NO]WAIT ]
```

Restrictions

Dismounting a /SYSTEM mount requires SYSNAM privilege.

Dismounting a /GROUP mount requires GRPNAM privilege.

Parameters

mount_point

Required (if you omit *logical_name* and the /ALL qualifier). Default: None.

DNFS device (and optional directory tree) required to dismount. Specify this mount point as one of the following:

DNFS*n*:
DNFS*n*:[*dir.subdir*]
DNFS*n*:[*dir.subdir*]*file*

where:

n	Value from 1 to 9999.
[*dir*] or [*dir.subdir*]	Directory to mount (up to eight in addition to the [000000] directory).
file	Individual file to dismount.

If you use the /ALL qualifier, you must specify DNFS*n*: without the directory tree.

logical_name

Required (if you omit *mount_point* and the /ALL qualifier). Default: None.

Logical name that you defined with the MOUNT command of the device to dismount.

Qualifiers

/ALL

Optional.

Dismounts one of the following:

- All file systems from all servers: `DISMOUNT /ALL`
- All file systems on the specified server: `DISMOUNT /ALL /HOST=host`
- All file systems on the specified device: `DISMOUNT DNFSn: /ALL`

If you dismount using the /ALL qualifier, the dismount operation completes even if the server is not currently reachable.

/HOST=*host*

Optional. Default: None.

Dismounts all file systems from the specified NFS server.

Valid only with the /ALL qualifier.

/WAIT
/NOWAIT

Optional. Default: /NOWAIT.

- /WAIT
  - Does not dismount the mounted file system if outstanding activities exist.
  - Waits until the dismount has been completed.
  - If you try to access any files on the mount point, the dismount fails.
- /NOWAIT
  - The client completes the command immediately.
  - Dismounting does not actually occur until all file activity has been completed.

Examples

1. `TCPIP> DISMOUNT DNFS3:`

Makes the file system mounted on local device DNFS3: inaccessible to local users.

2. `TCPIP> DISMOUNT DNFS4:[USR.MNT]`

   Dismounts only the specified mount point, [USR.MNT], on local device DNFS4:.

3. `TCPIP> DISMOUNT DNFS5: /WAIT`

   Dismounts the DNFS5:[000000] mount point and waits for it to occur.

4. `TCPIP> DISMOUNT /ALL`

   Dismounts all mount points on all devices.

5. `TCPIP> DISMOUNT /ALL /HOST="robin"`

   Dismounts all mount points served by host `robin`.

ENABLE SERVICE

ENABLE SERVICE — Enables a service on the running TCP/IP Services software. *Related commands:* DISABLE SERVICE, SHOW SERVICE.

Syntax

```
ENABLE SERVICE [ service ]
               [ /ADDRESS=IP_address ]
               [ /PORT=n ]
               [ /PROCESS=process ]
               [ /PROTOCOL=protocol ]
```

Parameters

service

Optional. Default: All services.

Specifies the service to enable. The service must be defined in the services database.

Qualifiers

/ADDRESS=*IP_address*

Optional. Default: 0.0.0.0.

Binds the service only to the specified address. If your host is multihomed, use this qualifier to configure the service to be offered on a specific Internet interface.

/PORT=*n*

Optional. Starts the service on the specified port.

/PROCESS=*process*

Optional.

Runs the service as the specified process.

/PROTOCOL=*protocol*

Optional. Default: TCP.

Runs the service with the specified protocol.

Examples

1. `TCPIP> ENABLE SERVICE TELNET`

   Initializes TELNET communications.

2. `TCPIP> ENABLE SERVICE RLOGIN /ADDRESS=130.180.4.7`

   Starts the remote login service for users on the host with IP address 130.180.4.7.

3. `TCPIP> ENABLE SERVICE SMTP`

   Starts the SMTP receiver. To start the SMTP sender, see the START MAIL command. For instructions on how to start the SMTP sender when TCP/IP Services starts up, see the SET CONFIGURATION ENABLE SERVICE command.

EXIT

EXIT — Exits from the management program.

Syntax

`EXIT`

EXPORT

EXPORT — Copies a file from within a container directory to an OpenVMS file. *Related commands:* IMPORT, DIRECTORY

Syntax

`EXPORT "/path/name" vms_file_name`

Restrictions

No wildcards.

Parameters

"*/path/name*"

Required.

Specifies the container directory and name of the file you want to copy.

vms_file_name

Required.

Specifies the target OpenVMS file name for the copied file.

Example

```
TCPIP> EXPORT "/upland/sand/piper" USER1$:[BIRDY]JOBS.TXT
```

Copies the file `piper` from the container directory `/upland/sand` to the regular OpenVMS file JOBS.TXT in directory [BIRDY] on disk USER1$:.

HELP

HELP — Displays online help for using management commands.

Syntax

```
HELP [ topic ]
```

Parameters

topic

Optional.

Specifies a specific topic for which to display help. When you enter the HELP command without specifying *topic*, a list of topics is displayed.

IMPORT

IMPORT — Copies an OpenVMS file to a UNIX file located in a container directory. *Related commands:* EXPORT, DIRECTORY. *Applies to:* NFS server.

Syntax

```
IMPORT vms_file_name "/path/name"
       [ /[NO]CONVERT ]
       [ /HOST=host ]
       [ /[NO]LOG ]
       [ /MODE=n ]
       [ /UID=n ]
       [ /USER_NAME=vms_user_name ]
```

Restrictions

No wildcards.

Parameters

vms_file_name

Required.

Name of the file to copy.

"*/path/name*"

Required.

Specifies the name of the UNIX container directory into which you want to copy the file and a file name.

Qualifiers

/CONVERT
/NOCONVERT

Optional. Default: /CONVERT.

Converts OpenVMS record files to STREAM_LF files. (The NFS server stores UNIX files in STREAM_LF format.)

/HOST=*host*

Optional.

Selects specific hosts if the proxy database has multiple host entries with the same user name and UID.

/LOG
/NOLOG

Optional. Default: Displays values for host, UID, GID, and user name.

Displays a full description of the specified proxy database record for you to determine ownership.

/MODE=*n*

Optional. Default: 755 (provides the following protection for owner, group, and world: rwx-rx-rx).

Specifies a UNIX protection mask for a new directory.

Specify octal values in the following order: user, group, others. The values are:

- 0 — No access
- 1 — Execute access
- 2 — Write access
- 3 — Write and execute access
- 4 — Read access
- 5 — Read and execute access
- 6 — Read and write access

- 7 — Read, write, and execute access

For example, /MODE=751 provides:

User	Read, write, and execute access	7	rwx
Group	Read and execute access	5	rx
Other	Execute access	1	x

/UID=*n*

Optional. Default: Determined with CREATE DIRECTORY.

Selects a specific entry in the proxy database to determine the ownership of the UNIX file.

In the proxy database:

- UID and GID fields identify UNIX ownership.
- User name field identifies OpenVMS ownership.

If you want to access an entry in the proxy database without a unique UID and user name combination, you might need to specify the /HOST qualifier. For example, the same UID and user name combination could appear on multiple hosts.

You can use the /UID qualifier in any combination with the /HOST and /USER_NAME qualifiers. However, if you do not have SYSPRV or BYPASS privilege, the values you specify must correspond to the values for your user name in the proxy database.

/USER_NAME=*vms_user_name*

Optional. Default: None.

Selects a specific entry in the proxy database to determine the ownership of the UNIX file.

The UID and GID fields in this entry establish the file's UNIX identity, while the user name field provides the OpenVMS ownership.

If you want to access an entry in the proxy database without a unique UID and user name combination, you might need to specify the /HOST qualifier. For example, the same UID and user name combination could appear on multiple hosts.

If you have SYSPRV or BYPASS privilege and do not specify the /USER_NAME qualifier, the proxy record with a UID of 0 and a GID of 1 is selected.

SYSPRV or BYPASS privilege is required to select a user name that has a UIC different from the UIC of the process running the management program.

You can use the /USER_NAME qualifier in any combination with the /HOST and /UID qualifiers. However, if you do not have SYSPRV or BYPASS privilege, the values you specify must correspond to the values for your user name in the proxy database.

If you do not specify the /USER_NAME qualifier, the proxy record with a GID of 1 and a UID of 0 is selected. If there is no proxy entry for the UID of 0, IMPORT fails.

Example

```
TCPIP> IMPORT USER1$:[BIRDY]JOBS.TXT "/upland/sand/piper"
```

Copies the file JOBS.TXT to the new file `piper` in the container directory `/upland/sand`. The file's ownership depends on the directory information specified with the CREATE DIRECTORY command.

LIST COMMUNICATION_CONTROLLER

LIST COMMUNICATION_CONTROLLER — Displays the communication controller definitions defined in the configuration database. **Related commands:** DEFINE COMMUNICATION_CONTROLLER, DELETE COMMUNICATION_CONTROLLER.

Syntax

```
LIST COMMUNICATION_CONTROLLER [ controller ]
                              [ /INTERNET_INTERFACE=character ]
```

Parameters

controller

Optional. Default: All devices.

Specifies the OpenVMS device name of communication controller definitions to be displayed.

Qualifiers

/INTERNET_INTERFACE=*letter*

Optional. Default: All alphabetic characters.

Specifies the first character of the Internet interface name corresponding to the communication controller definitions to be displayed. For more information on network interfaces, refer to the VSI TCP/IP Services for OpenVMS Management manual.

Examples

```
TCPIP> LIST COMMUNICATION_CONTROLLER

                    Communication Controller Configuration

Controller:  LO   Internet Interface:  L
                         Description:
                                Type:  LOCAL

Controller:  WI   Internet Interface:  W
                         Description:
                                Type:  WIRELESS

Controller:  EC   Internet Interface:  C
                         Description:
                                Type:  CLUSTER ETHERNET
```

```
Controller:  XE   Internet Interface:  D
                         Description:
                                Type:  CLUSTER ETHERNET

Controller:  EF   Internet Interface:  F
                         Description:
                                Type:  CLUSTER ETHERNET

Controller:  CL   Internet Interface:  I
                         Description:  ATM Classical IP
                                Type:  FDDI

Controller:  EL   Internet Interface:  L
                         Description:  ATM Emulated LAN
                                Type:  FDDI

Controller:  PP   Internet Interface:  P
                         Description:  Point to Point Protocol
                                Type:  PPP

Controller:  EB   Internet Interface:  B
                         Description:  Shared Memory LAN
                                Type:  CLUSTER ETHERNET

Controller:  EI   Internet Interface:  I
                         Description:  Fast Ethernet - I82558
                                Type:  CLUSTER ETHERNET

Controller:  FA   Internet Interface:  A
                         Description:
                                Type:  CLUSTER FDDI

Controller:  FC   Internet Interface:  C
                         Description:
                                Type:  CLUSTER FDDI

Controller:  IC   Internet Interface:  C
                         Description:
                                Type:  CLUSTER TOKEN_RING

Controller:  IR   Internet Interface:  R
                         Description:
                                Type:  CLUSTER TOKEN_RING

Controller:  SL   Internet Interface:  S
                         Description:
                                Type:  SERIAL


Controller:  CL   Internet Interface:  I
                         Description:  ATM Classical IP
                                Type:  FDDI

TCPIP>
```

Displays all the information in the table used to match OpenVMS device names with Internet interface names.

LOOP

LOOP — Sends ICMP ECHO packets to hosts to determine whether they are active. Same as the PING command.

Syntax

```
LOOP [ host ]
     [ /ADDRESS=xx.xx.xx.xx ]
     [ /ALL ]
     [ /FULL ]
     [ /NUMBER_PACKETS=n ]
     [ /PACKET_SIZE=n ]
     [ /PATTERN="hexadecimal-string" ]
     [ /[NO]ROUTE ]
     [ /WAIT=n ]
```

Parameters

host

Optional. Default: None.

Specifies the host to which the test packets are sent. Omitting the *host* parameter and the /ADDRESS qualifier tests the TCP/IP Services software on the local node, as defined by the system logical TCPIP $INET_HOST.

Qualifiers

/ADDRESS=*xx.xx.xx.xx*

Optional.

Specifies the IP address of the host to which the test packets are sent.

/ALL

Optional. Default: Not all requests.

Displays all ICMP ECHO_REQUESTs, even if not in direct response to this operation.

/FULL

Optional.

Numeric output only. No attempt is made to look up symbolic names for host addresses. This occurs only when displaying ICMP packets other than ECHO_RESPONSE.

/NUMBER_PACKETS=*n*

Optional. Default: 4 packets

Specifies the number of packets to send. If you specify 0, packets are sent continuously until you terminate the LOOP command with Ctrl/C.

/PACKET_SIZE=*n*

Optional. Default: 64 bytes.

Specifies the size of the ICMP ECHO_REQUEST.

/PATTERN=*hexadecimal-string*

Optional.

Fills out the packet you send with up to 16 bytes, which is useful for diagnosing data-dependent problems. The *hexadecimal-string* is a string of hexadecimal digits of up to 32 characters (16 bytes).

For example, /PATTERN="ff" causes the sent packet to be filled with ones (1).

/ROUTE
/NOROUTE

Optional. Default: /ROUTE.

/ROUTE	Request is routed through the normal routing tables.
/NOROUTE	Normal routing tables are bypassed. If the host is not on the LAN, you get an error.

/WAIT=*n*

Optional.

Specifies the number of seconds to wait between sending packets.

Syntax

1. `TCPIP> LOOP thrush`

   Tests the connectivity path to UNIX host `thrush`.

2. `TCPIP> LOOP`

   Tests the local TCP/IP Services software.

3. `TCPIP> LOOP /NOROUTE thrush`

   Tests the path to the UNIX host `thrush` without using normal routing tables.

MAP

MAP — Maps (logically links) one of the following to the NFS server: an openVMS disk (requires one execution of MAP to map the disk to a UNIX path name) or a container file system (requires two executions of MAP. The first maps the disk, and the second maps the file system). Mapping creates a logical file system, also called an NFS file system. A logical file system (with an entry in the export database) is accessible to NFS client users for mounting. To make a file system available on all nodes of a cluster, map it on each node. Mapping is one step in the tasks necessary to give remote users access to a file system that physically resides on an OpenVMS host running an NFS

server: map the file system, export the file system (add an entry in the export database), give potential users entries in the proxy database. MAP settings are not permanent. To map file systems in the permanent configuration database, issue SET CONFIGURATION [NO]MAP. *Related commands:* ADD EXPORT, SHOW EXPORT, REMOVE EXPORT, SET CONFIGURATION MAP, SET CONFIGURATION NOMAP, SHOW MAP, SHOW CONFIGURATION MAP, UNMAP.

Syntax

`MAP "file system name" logical_file_system`

Restrictions

Requires SYSPRV and BYPASS privileges.

Parameters

"*file system name*"

Required.

Specifies the name for the file system or disk. In the case of mapping a disk, the "*/path*" can be only one level from the root. This parameter specifies the name by which users will access the file system.

logical_file_system

Required.

Specifies the file system to make known to the NFS server.

To map an OpenVMS file system, specify its disk:

`MAP "/disk" disk:`

To map a container file system, specify the disk and the directory name:

`TCPIP> MAP "/container_name" disk:[vms.directory.name]`

Examples

1. `TCPIP>  (TCPIP>)MAP "/usr" CANARY$DUA2:`

   Maps local disk CANARY$DUA2: to `/usr`. This disk can be exported as `/usr` to users on remote NFS clients.

2. ```
   TCPIP> MAP "/remote" VERDIN$DUA3:
   TCPIP> MAP "/flyers" VERDIN$DUA3:[UNIX_BIRD_FILES]
   ```

   Maps [UNIX_BIRD_FILES], a container file system on disk VERDIN$DUA3:, to `/flyers`. This file system can be exported as `/flyers` to NFS server users. (The first MAP command maps the underlying OpenVMS file system.)

MOUNT

MOUNT — Makes a physically remote file system accessible to local users. *Applies to:* NFS client. Mounts a remote directory to local device DNFS*n*:. Similar in function to the UNIX `/etc/mount`
   ```

command, MOUNT gives a file system a UNIX path name. (In format and style, MOUNT resembles the DCL command MOUNT.) You can mount either OpenVMS or UNIX file systems. *Related commands:* DISMOUNT, SHOW MOUNT.

Syntax

```
MOUNT mount_point [ volume_label ] [ logical_name ]
      [ /HOST=host ]
      [ /PATH="/path/name" ]
      [ /ACP_PARAMS=options ]
      [ /[NO]ADF[=option] ]
      [ /AUTOMOUNT[=INACTIVITY:time] ]
      [ /BACKGROUND[=options] ]
      [ /CACHE_TIMEOUT[=options] ]
      [ /[NO]CONVERT ]
      [ /DATA=[options] ]
      [ /FILEIDS[=options] ]
      [ /[NO]FORCE ]
      [ /GID=gid ]
      [ /GROUP ]
      [ /OWNER_UIC=n ]
      [ /PROCESSOR=acp_option ]
      [ /PROTECTION=protections ]
      [ /RETRIES=n ]
      [ /SERVER_TYPE=type ]
      [ /SHARE ]
      [ /STRUCTURE ]
      [ /[NO]SUPERUSER=uid ]
      [ /SYSTEM ]
      [ /TIMEOUT=OpenVMS_delta_time ]
      [ /UID=uid ]
      [ /USER=user ]
      [ /[NO]WRITE ]
```

Restrictions

If you mount remote OpenVMS directories where the NFS server is running TCP/IP Services software, use the /NOADF qualifier on the MOUNT command line unless you are using the OpenVMS-to-OpenVMS integration feature.

The /NOADF requirement applies only if the remote NFS server is running versions of TCP/IP Services earlier than Version 3.3 and cannot participate in OpenVMS-to-OpenVMS mode operation. Other tips include:

- For the qualifiers that require a time value, specify OpenVMS delta time.
- Whenever you specify multiple options and values, use the following syntax:

  /qualifier=(option_a:value1,option_b:value2,value3)

Parameters

mount_point

Required.

Local device (and optional directory tree) on which to mount the remote NFS file system. Specify this mount point as one of the following:

```
DNFSn:
DNFSn:[dir.subdir]
DNFSn:[dir.subdir]file
```

where:

n	Specifies the unit number. Specify a value from 0 to 9999. Specifying 0 causes the client to choose the next available unit number. (It does not mount a device named DNFS0:.)
[*dir*] or [*dir.subdir*]	Specifies the directory to mount (up to eight subdirectories in addition to the [000000] directory).
file	Specifies the individual file to mount.

volume_label

Optional. Default: First 12 characters of the combined values of the /HOST and /PATH qualifiers. The default label is a combination of /HOST and /PATH with a dollar sign ($) separating the two.

Specifies the Files-11 (ODS-2 or ODS-5) volume label to be associated with the remote path name. You can use this parameter to provide a unique volume label on a system where there is a label conflict. The client does the following:

- Accepts only the first 12 characters for all other entries.
- Applies *volume_label* only on the first mount of a particular disk.
- Ignores *volume_label* with subsequent mounts on that disk.

VSI recommends that if you use the SET FILE /STATISTICS command on a file mounted with DNFS, do not include any colons (:) in the *volume_label*.

logical_name

Optional. Default: None.

Specifies the logical name associated with the volume.

The client creates the following logical definitions, depending on what you specify:

- If you mount DNFS*n*:[000000], the client defines the logical name as DNFS*n*:
- If you mount DNFS*n*:[*dir.dir*], the client defines the logical name as DNFS*n*:[*dir.dir.*] The extra dot allows for relative directory specifications. If you issue the following command:

  ```
  $ SET DEFAULT logical:[subdir]
  ```

  The full default definition becomes:

  ```
  DNFSn:[dir.dir.subdir]
  ```

The client places the logical name in the SYSTEM logical name table, unless you specify the /GROUP or /SHARE qualifier. The client deletes the logical name from the SYSTEM table when you dismount the volume. The process must have SYSNAM privilege to mount a system mount point. Without SYSNAM or GRPNAM privilege, the user must specify /SHARE for a JOB mount. (See the /SHARE qualifier for more information.)

Qualifiers

/ACP_PARAMS= { BUFFER_LIMIT=*n* | DUMP | IO_DIRECT=*n* | IO_BUFFERED=*n* | MAX_WORKSET=*pages* | PAGE_FILE=*file* | PRIORITY=*base-priority* | WORKSET=*pages* }

Optional.

Specifies modifiable process parameters for the ancillary control process (ACP).

These parameters are dynamic. The NFS client applies your settings at each first start of an ACP.

For descriptions of these options, see the section on RUN (PROCESS) in the *OpenVMS DCL Dictionary*.

/ADF=CREATE
/NOADF

Optional. Default: /ADF=CREATE.

If attributes data files (ADFs) exist on the NFS server, the /ADF qualifier lets you use them.

The server uses ADFs to store OpenVMS file attributes. These files appear on the server as `.$ADF$`*`file`* files, but you cannot view them directly on the local client system.

The option is:

- CREATE

  The client uses and updates the ADFs, and creates ADFs for new files.

/NOADF — No ADFs are created or used.

/AUTOMOUNT[=INACTIVITY:*time*]

Optional. Defaults:

- If you omit this qualifier, automounting is not enabled for this file system.
- If you include the /AUTOMOUNT qualifier but omit the INACTIVITY keyword, file systems are automatically dismounted after five minutes of inactivity.

This qualifier enables automounting for the file system. The file system is automatically mounted when you access its path name.

You can include the optional INACTIVITY keyword to specify the number of minutes of inactivity before automatically dismounting the file system. Be sure to specify the *time* as *hh:mm:ss*. When this inactive period expires, the NFS client dismounts the path name.

/BACKGROUND [= {DELAY:*OpenVMS_delta_time* | RETRY:*n*}]

Optional. Defaults:

- If you omit this qualifier, background mode mounting is not attempted.
- If you omit the DELAY keyword, background mode mounting is set up with /BACKGROUND=(DELAY:00:00:30,RETRY:10).

This qualifier enables background mode for mounting the file system.

The optional DELAY time specifies amount of time to wait if the mount attempt fails before trying again. Specify the time as *hh:mm:ss*. The maximum delay period you can specify is approximately 49 days. The default delay time is 30 seconds.

The optional RETRY keyword specifies the number of times to repeat the attempt to mount the file system. RETRY:0 means that the client uses the first try only. The default number of times to retry is 10.

If you use the /BACKGROUND qualifier, you must also use the /RETRIES qualifier and specify a nonzero value. For example:

```
$ TCPIP MOUNT DNFS0: /BACKGROUND=RETRY:9 /RETRIES=4 /HOST="robin" -
_$  /PATH="/USR/USERS/GEORGE"
```

In this example, you are asking for four data retries on each mount attempt and nine mount attempts, for a total of 36 tries. If you use the default value for /RETRIES, the first mount attempt can never complete except by succeeding, and the process doing the mount will hang until the server becomes available.

/CACHE_TIMEOUT= [DIRECTORY:*OpenVMS_delta_time*] [ATTRIBUTE:*OpenVMS_delta_time*] [READ_DIRECTORY]

Optional. Defaults:

If you omit this qualifier, the file system is set up with caching timeouts as follows:

```
/CACHE_TIMEOUT=(DIRECTORY:00:00:30,ATTRIBUTE:00:00:15)
```

Specifies the following caching timeout information for the mount point:

- DIRECTORY:*OpenVMS_delta_time*
  - Amount of time that the client waits between rereading a directory's status or contents.
  - Specify *OpenVMS_delta_time* as *hh:mm:ss*.
- ATTRIBUTE:*OpenVMS_delta_time*
  - Amount of time that the client waits between rereading a file's attributes from the NFS server.
  - Specify *OpenVMS_delta_time* as *hh:mm:ss*.
- READ_DIRECTORY
  - Forces the client to read the contents of the directory requested when the cache timeout occurs rather than rely on the directory's modified time.
  - By reading the directory contents, the client can be aware of any changes to the number of files within the directory, even if the directory's modify time was not updated.

/CONVERT
/NOCONVERT

Optional. Default: /CONVERT.

Converts files with the following attributes to STREAM_LF files:

- Sequential
- Variable length
- Carriage return/carriage control (VAR-CR)

The convert feature works with some utilities and DCL commands but not with others. For example, it works with the CREATE command and with EDIT/TPU, but it does not work with COPY, BACKUP, or EDIT/EDT. There is no simple way to identify what works. However, for the feature to take effect, the following conditions must be satisfied:

- The file attributes must be sequential, variable length, and carriage return/carriage control.
- The file must be opened for exclusive write access (generally true for newly created files).
- The file must be opened with the FAB$M_SQO bit set in the FAB$L_FOP field.
- The file creation and the open for write must be done in one step. That is, if the program first creates the file and afterward opens it for write, the convert feature does not work.

You can convert only those files that were opened using RMS sequential access. For additional information, refer to the VSI TCP/IP Services for OpenVMS Management manual.

/DATA [= {*read_bytes* | *write_bytes*}]

Optional. Default: /DATA=(8192,8192).

Largest amount of NFS data received or transmitted in one network operation. The options mean:

- *read_bytes* — Data received. Minimum value = 512.
- *write_bytes* — Data transmitted. Minimum value = 512.

If you specify only one value, it applies to both READ and WRITE.

You do not need to use /DATA unless a remote NFS server imposes a restriction on data size. If the server requests a smaller transfer size than you specified, the server's requested value overrides the one you set.

/FILEIDS [= {UNIQUE | NONUNIQUE}]

Optional. Default: /FILEIDS=UNIQUE.

With UNIQUE, the client uses the file name and 32-bit NFS file ID when processing the directory information returned by the server to determine whether cached information is valid.

With NONUNIQUE, the client uses the file handle instead of the file ID. This can refresh directory entries in the client's cache more quickly. However, this can degrade performance because the client must issue additional RPC requests to get the file handle.

/FORCE
/NOFORCE

Optional. Default: /NOFORCE.

Performs an overmount or a mount that can cause file system occlusion.

Required privileges:

- OPER
- SYSPRV (for overmounting a /SYSTEM mount)
- GRPNAM (for overmounting a /GROUP mount)

/GID=*n*

Optional. Default: –2.

Default GID if no GID mapping exists for file access.

Restriction: Requires OPER privilege.

/GROUP

Optional. Default: User mounted.

Adds the logical name to the group logical name table. If the mount is the first one on the volume, /GROUP marks the volume as being group-mounted.

Restrictions:

- Requires GRPNAM privilege.
- /GROUP and /SYSTEM are mutually exclusive.

/HOST=*host*

Required.

Remote NFS server on which the physical files reside. Type either domain-name or IP-address format.

/OWNER_UIC=*n*

Optional. Default: Ownership recorded on the volume.

UIC-assigned ownership of the volume while you mount it.

Applied only on the first mount of an NFS disk.

/PATH="*/path/name*"

Required.

Path name on the NFS server (specified by /HOST). Must match an exported directory, subdirectory, or file of an exported file system on the server.

The */path/name* is mounted as the master file directory (MFD) of the specified device.

/PROCESSOR= {UNIQUE | SAME:DNFS*n*: | FILE:*file*}

Optional. Default: New ACP for each mount.

Associates an ancillary control process (ACP) to process the volume, overriding the default manner in which the client associates ACPs with NFS devices (starting a new ACP for each mount request). The options are:

- UNIQUE
  - Creates a new ACP for the new NFS device.
  - Requires OPER privilege.
- SAME:DNFS*n*:
  - Uses the same ACP as the specified device.
  - Requires OPER privilege.
- FILE:*file*
  - Creates a new ACP running the image specified by *file*. Do not use wildcards, host names, or directory names.
  - Requires CMKRNL or OPER privilege.

/PROTECTION=*protections*

Optional. Default: /PROTECTION=(S:RWED,O:RWED,G:RWED,W:RWED).

Protection code for the volume. If you omit a category, the client denies access to that category of user.

Applied only on the first mount of an NFS device and ignored with subsequent mounts on that device.

Restrictions: Requires OPER privilege.

/RETRIES=*n*

Optional. Default: /RETRIES=0 (the client retries the request forever or until the server responds).

Maximum number of read or write retries if the NFS server fails to respond.

/RETRIES=0 is a close equivalent to a UNIX hard mount. If you attempt to abort a command or program while it is still retrying the I/O operation on a client device, the process enters RWAST state and remains in that state until the NFS server responds. A process in RWAST state cannot be terminated. If the server does not become available, the only way remove the process without rebooting the client host is to dismount the client device with the /ALL qualifier.

/SERVER_TYPE=*type*

Optional. Default: UNIX.

Operating system of the host running NFS server. The values for *type* are:

- UNIX
- IBM_VM

If the server is TCP/IP Services Version 3.3 or later, this qualifier is ignored because the client and server always operate in OpenVMS-to-OpenVMS mode.

/SHARE

Places the logical name in the job logical name table and increments the volume mount count regardless of the number of job mounts. When the job logs out, all job mounts are dismounted, causing the volume mount count to be decremented. Refer to the VSI TCP/IP Services for OpenVMS Management manual for more information.

/STRUCTURE

Optional. Default: /STRUCTURE=2

Specifies whether the volume should be formatted in Files-11 On-Disk Structure Level 2 (ODS-2), which is the default, or Files-11 On-Disk Structure Level 5 (ODS-5).

For more information about ODS-5 disks, refer to the VSI OpenVMS System Manager's Manual: Essentials.

/SUPERUSER=*uid*
/NOSUPERUSER

Optional. Default: /NOSUPERUSER.

Maps users with SYSPRV, BYPASS, or READALL privileges to the superuser UID. The NFS server must allow superuser access.

The normal superuser UID is 0.

/NOSUPERUSER: No mapping.

/SYSTEM

Optional. Default: System mounted.

Places the logical name in the system logical name table unless you specify the /GROUP or /SHARE qualifier. The client deletes the logical name from the system table when you dismount the volume.

Restrictions:

- Requires SYSNAM privilege.
- The /GROUP, /SYSTEM, and /SHARE qualifiers are mutually exclusive.
- Without SYSNAM or GRPNAM privilege, you must use /SHARE for a job mount.

/TIMEOUT=*OpenVMS_delta_time*

Optional. Default: ::01 (1 second).

Minimum timeout period for initial remote procedure call (RPC) request retransmissions.

Specify the timeout period as your estimate of the typical round-trip time for RPC requests. For slower-speed links — for example, NFS traffic over SLIP — specify a value that is larger than the default.

Example: For a maximum read/write size of 8192 (see the /DATA qualifier) over a 19,200-baud SLIP line, set the absolute minimum timeout value as follows:

```
10240 bytes*8 bits per byte
--------------------------- = 4.27 seconds
   19,200 bits per second
```

Here, the 10240 bytes is 8192 data bytes plus the worst-case RPC overhead. Because 4.27 seconds is the absolute minimum, a more realistic value for this link is 15 to 30 seconds to allow for other traffic.

/UID=*n*

Optional. Default: –2.

Default UID if no UID mapping exists for file access.

Restriction: Requires OPER privilege.

Both the NFS server and NFS client use the proxy database for access control. VSI strongly recommends that you provide a proxy with a unique UID for every NFS client user.

If you need to provide universal access to world-readable files, you can use the default UID to avoid creating a proxy for every NFS client user. You should avoid using the default UID if clients require additional file access; otherwise, client users may see unpredictable and confusing results when they try to create files. Refer to the VSI TCP/IP Services for OpenVMS Management manual for a detailed discussion about using proxies, the default user, and security considerations.

/USER=*user*

Optional. Default: USER account.

Existing OpenVMS account to which the NFS client maps unknown UIDs.

If the client does not find the USER account, the DECnet account becomes the default. If the client does not find the DECnet account, [200,200] becomes the default.

/WRITE
/NOWRITE

Optional. Default: /WRITE.

Mounts files with WRITE privilege.

/NOWRITE mounts files as read only.

Examples

1. `TCPIP> MOUNT DNFS2: /HOST="loon" /PATH="/usr/users/curlew"`

   Mounts, on local device DNFS2:, the remote directory `/usr/users/curlew`, which resides on NFS server `loon`.

2. ```
   TCPIP> MOUNT DNFS3: /HOST="sigma" -
   _TCPIP> /PATH="/usr" /AUTO=(INACT:00:10:00)
   ```

   Using automounting, this command mounts the `/usr` file system from `sigma` onto the OpenVMS mount point when it references the path name. The client keeps the path mounted for an inactive period of 10 minutes, after which it dismounts the path name.

3. ```
   TCPIP> MOUNT DNFS4: /HOST="sigma" /PATH="/usr" -
   _TCPIP> /BACKGROUND=(DEL:00:01:00,RET:20) /RETRIES=4
   ```

   Attempts to mount the `/usr` file system. If it cannot, it waits 1 minute and retries the connection up to 20 times.

4. ```
   TCPIP> MOUNT DNFS5:[USERS.MNT] /HOST="sigma" /PATH="/usr"
   %DNFSMOUNT-S-MOUNTED, /usr mounted on _DNFS5:[USERS.MNT]

   TCPIP> MOUNT DNFS5:[USERS.MNT] /HOST="sigma" /PATH="/usr/users" /FORCE
   %DNFSMOUNT-S-REMOUNTED, _DNFS5:[USERS.MNT] remounted as /usr/users on
    SIGMA
   ```

   Specifies a lower level in the NFS server path with the second mount. This constitutes another path name and qualifies for an overmount.

5. ```
   TCPIP> MOUNT DNFS22:[USERS.SMITH.MNT] /HOST="sigma" /PATH="/usr"
   %DNFSMOUNT-S-MOUNTED, /usr mounted on _DNFS22:[USERS.SMITH.MNT]

   TCPIP> MOUNT DNFS22:[USERS.SMITH] /HOST="sigma" /PATH="/usr" /FORCE
   %DNFSMOUNT-S-MOUNTED, /usr mounted on _DFS22:[USERS.SMITH]
   %TCPIP-I-OCCLUDED, previous contents of _DNFS22:[USERS.SMITH] occluded
   ```

   The /FORCE qualifier performs an occluded mount. If you issue the DIRECTORY command, the NFS client occludes (hides from view) the subdirectory dropped from the first MOUNT command.

   To make the directory visible again, either issue the SHOW MOUNT command (both mounts will be visible) or dismount DNFS22:[USERS.SMITH].

6. ```
   TCPIP> MOUNT DNFS0: BOOK1 BEATRICE -
   _TCPIP> /PATH="/INFERNO" -
   _TCPIP> /HOST="FOO.BAR.EREWHON" -
   _TCPIP> /STRUCTURE=5 -
   _TCPIP> /SYSTEM
   ```

   Mounts path INFERNO with label BOOK1 and logical name BEATRICE. Specifies the volume structure as ODS-5.

PING

PING — Sends ICMP ECHO packets to hosts to determine whether they are active. Same as the LOOP command.

Syntax

```
PING [ host ]
     [ /ADDRESS=xx.xx.xx.xx ]
     [ /ALL ]
```
```

```
[ /FULL ]
[ /NUMBER_PACKETS=n ]
[ /PACKET_SIZE=n ]
[ /PATTERN="hexadecimal-string" ]
[ /[NO]ROUTE ]
[ /WAIT=n ]
```

Parameters

host

Specifies the host to which the test packets are sent. Omitting *host* tests the TCP/IP Services software on the local node, as defined by the system logical TCPIP$INET_HOST.

Qualifiers

/ADDRESS=*xx.xx.xx.xx*

Optional.

Specifies the IP address of the host to which the test packets are sent.

/ALL

Optional. Default: Not all requests.

Displays all ICMP ECHO_REQUESTs, even if not in direct response to this operation.

/FULL

Optional.

Numeric output only. No attempt is made to look up symbolic names for host addresses. This occurs only when displaying ICMP packets other than ECHO_RESPONSE.

/NUMBER_PACKETS=*n*

Optional. Default: 4 packets

Specifies the number of packets to send. If you specify 0, packets are sent continuously until you terminate it with Ctrl/C.

/PACKET_SIZE=*n*

Optional. Default: 64 bytes.

Specifies the size of the ICMP ECHO_REQUEST.

/PATTERN=*"hexadecimal-string"*

Optional.

Fills out the packet you send with up to 16 bytes, which is useful for diagnosing data-dependent problems. The string is a hexadecimal string of up to 32 characters (16 bytes).

For example, /PATTERN="ff" causes the sent packet to be filled with ones (1).

/ROUTE
/NOROUTE

Optional. Default: /ROUTE.

/ROUTE	Request is routed through the normal routing tables.
/NOROUTE	Normal routing tables are bypassed. If the host is not on the LAN, you get an error.

/WAIT=*n*

Optional.

Specifies the number of seconds to wait between sending packets.

Example

```
TCPIP> PING dented
```

Specifies that the local host test the connectivity path to host `dented`.

REMOVE DIRECTORY

REMOVE DIRECTORY — Removes a link to a directory within a UNIX container directory. If there are no other links to it, the directory is deleted. **Related commands:** CREATE DIRECTORY, DIRECTORY. *Applies to:* NFS server.

Syntax

```
REMOVE DIRECTORY "/path/name"
```

Restrictions

Requires:

- Read and write access to the parent directory
- BYPASS privilege

Parameters

"*/path/name*"

Required.

Directory with the link you want to remove.

Example

```
TCPIP> REMOVE DIRECTORY "/eagles/eaglet"
```

Removes a link to the directory `/eagles/eaglet`.

REMOVE EXPORT

REMOVE EXPORT — Deletes directory names from the export database so that they are not available for mounting by an NFS client. *Related commands:* ADD EXPORT, SHOW EXPORT, MAP, SET CONFIGURATION MAP, SET CONFIGURATION NOMAP, SHOW MAP, SHOW CONFIGURATION MAP. *Applies to:* NFS server.

Syntax

```
REMOVE EXPORT "/path/name"
              [ /[NO]CONFIRM ]
              [ /HOST=host ]
```

Restrictions

Requires read and write access to the export database.

Parameters

"*/path/name*"

Required.

Directory name to delete from the export database.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM if you use a wildcard.

When the software encounters a match, it displays a description and solution. If /CONFIRM is enabled, the software then requests confirmation before deleting each directory name. Enter one of the following:

- Y to delete the name
- N to retain the name

If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

/HOST=*host*

Optional. Default: /HOST=* (all hosts).

Host, running NFS client, that will become unable to access the specified container directory.

Format for multiple hosts:

/HOST=("*host0*","*host1*", "*host2*")

Examples

1. `TCPIP> REMOVE EXPORT "/house/finch"`

   Removes the name of container directory `/house/finch` from the export database. This directory is now inaccessible to NFS client users.

2. `TCPIP> REMOVE EXPORT "/oceans/swamps" /HOST=("tern","crane")`

   Modifies the accessibility of local UNIX directory `/oceans/swamps`. This directory is now unavailable to users working on hosts `tern` and `crane`, which run NFS client software.

REMOVE FILE

REMOVE FILE — Removes a link to a file within a container directory. If there are no other links to it, the file is deleted. *Related commands:* DIRECTORY, REMOVE DIRECTORY. *Applies to:* NFS server.

Syntax

`REMOVE FILE "/path/name"`

Restrictions

Requires:

- Read and write access to the parent directory
- BYPASS privilege

Parameters

"*/path/name*"

Required.

File with the link you want to remove.

Example

`TCPIP> REMOVE FILE "/peacock/feather.care/preening"`

Removes the NFS link to the file `preening`.

REMOVE MAIL

REMOVE MAIL — Deletes mail messages from SMTP queues. Without the *user* parameter, all messages from the user name that correspond to your process's user name are deleted. *Related commands:* SEND MAIL, SHOW MAIL. *Applies to:* SMTP.

Syntax

```
REMOVE MAIL [ user ]
            [ /[NO]COPY=[directory] ]
            [ /[NO]CONFIRM ]
            [ /ENTRY=n ]
```

Restrictions

Requires SYSPRV or BYPASS privilege for mail messages that are not yours.

Parameters

user

Optional. Default: All mail messages with your process's user name.

Removes messages sent from the specified user name.

Qualifiers

/COPY=[*directory*]
/NOCOPY=[*directory*]

Optional. Default: Messages copied to the user's default directory.

Copies messages to be deleted to the specified directory.

/NOCOPY: Messages are not copied.

/CONFIRM
/NOCONFIRM

Optional. Defaults:

- With an entry number specified — /NOCONFIRM
- Without an entry number specified — /CONFIRM

If you specify /CONFIRM, or if you omit an entry number, requests confirmation before deleting each message. Respond to the CONFIRM: prompt by entering one of the following:

- Y to delete the mail message
- N to retain the mail message
- G to change to NO CONFIRMATION mode

If you specify the /NOCONFIRM qualifier or an entry, the operation is performed without asking you to confirm the request.

/ENTRY=*n*

Optional. Default: All.

Queue entry numbers to remove from the SMTP queue.

Examples

1. `TCPIP> REMOVE MAIL`

   Removes all messages for your process's user name, or deletes everything in the SMTP queue if you have either SYSPRV or BYPASS privilege.

2. `TCPIP> REMOVE MAIL /ENTRY=781`

   Removes message 781, if it corresponds to your process's user name, or if you have either SYSPRV or BYPASS privilege.

3. `TCPIP> REMOVE MAIL BROOD`

   Removes all messages for BROOD, if your process's user name is BROOD, or if you have either SYSPRV or BYPASS privilege.

4. `TCPIP> REMOVE MAIL /USER_NAME=COCKATOO /COPY=[COCKATOO.OLD_MAIL]`

   Removes all messages for COCKATOO, if this is your process's user name, or if you have either SYSPRV or BYPASS privilege. Before deletion, copies this queued mail to the specified directory.

REMOVE PROXY

REMOVE PROXY — Deletes entries from the volatile and permanent proxy database. *Related commands:* ADD PROXY, SHOW PROXY. *Applies to:* NFS server, NFS client, PC-NFS, remote shell, LPR/LPD, and customer-developed services.

Syntax

```
REMOVE PROXY [ user_name ]
             [ /COMMUNICATION ]
             [ /[NO]CONFIRM ]
             [ /GID=n ]
             [ /HOST=host ]
             [ /NFS=options ]
             [ /PERMANENT ]
             [ /REMOTE_USER=user ]
             [ /UID=n ]
```

Restrictions

Requires:

- Read and write access to the proxy database
- One of the following privileges:
  - SYSPRV
  - SYSLCK
  - OPER

Parameters

user_name

Optional. Default: All entries (REMOVE PROXY *).

Deletes the specified entries from the proxy database.

Qualifiers

/COMMUNICATION

Optional. Default: Both communication and NFS entries.

Deletes communication (non-NFS) proxies.

/CONFIRM
NOCONFIRM

Optional. Default: /CONFIRM if you use a wildcard.

With /CONFIRM enabled, the software requests confirmation before deleting records. At the CONFIRM: prompt, enter one of the following:

- Y to delete the entry
- N to retain the entry
- G to change to NO CONFIRMATION mode

If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

/GID=*n*

Optional. Default: All GIDs.

Deletes only proxies for the specified group identifier (GID).

/HOST=*host*

Optional. Default: All hosts.

Deletes only proxies for the specified host.

/NFS=INCOMING
/NFS=OUTGOING

Optional. Default: /NFS=(INCOMING,OUTGOING).

Deletes an NFS proxy. Specify one of the following:

/NFS=OUTGOING	Proxy to use NFS client

/NFS=INCOMING	Proxy to use NFS server
/NFS=(OUTGOING,INCOMING)	Proxy to use NFS client and NFS server

/PERMANENT

Optional. Default: None.

Deletes entries only from the permanent proxy database.

/REMOTE_USER=*user*

Optional. Default: None.

Deletes entries for the specified remote user name.

/UID=*n*

Optional. Default: All UIDs.

Limits the search of entries to delete to proxies for the specified UID.

Examples

1. `TCPIP> REMOVE PROXY "peacock" /HOST=GOLDEN /UID=83`

   Removes authorization for UID 83 on host GOLDEN from OpenVMS account `peacock`.

2. `TCPIP> REMOVE PROXY /HOST=GOLDEN /UID=83`

   Removes authorization for UID 83 from host GOLDEN.

3. `TCPIP> REMOVE PROXY /HOST=("goose","grouse")`

   Removes authorization for all users on hosts `goose` and `grouse`.

4. `TCPIP> REMOVE PROXY /UID=83`

   Totally removes authorization for UID 83.

5. `TCPIP> REMOVE PROXY VMS_USER /REMOTE=PARTRIDGE /HOST=*`

   Removes authorization for remote user PARTRIDGE on all hosts.

SEND MAIL

SEND MAIL — Requeues a mail message for delivery. Releases jobs that are in a hold state. *Related commands:* REMOVE MAIL, SHOW MAIL. *Applies to:* SMTP.

Syntax

```
SEND MAIL [ user ]
          [ /AFTER=time ]
          [ /[NO]CONFIRM ]
          [ /ENTRY=n ]
```

Restrictions

SYSPRV or BYPASS privilege required to requeue mail messages that do not correspond to your process's user name.

Parameters

user

Optional. Default: All.

Requeues messages sent from the specified user name.

Qualifiers

/AFTER=*time*

Optional. Default: Immediate delivery attempt.

Time after which delivery is to be attempted.

/CONFIRM
/NOCONFIRM

Optional. Defaults:

- With an entry number specified — /NOCONFIRM
- Without an entry number specified — /CONFIRM

With /CONFIRM enabled, the software requests confirmation before deleting each message when you omit an entry number. At the CONFIRM: prompt, enter one of the following:

- Y to delete the message
- N to retain the message
- G to change to NO CONFIRMATION mode

With /NOCONFIRM enabled, the operation is performed without asking you to confirm the request.

/ENTRY=*n*

Optional.

Queue number of the mail message to be re-queued for delivery.

SET ARP

SET ARP — Provides the dynamic mapping from an IP address to the corresponding physical network address (hardware address) on an FDDI, Ethernet, or Token Ring LAN segment. SET

NOARP removes an address-mapping pair (IP address to physical network address). *Related command:* SHOW ARP.

Syntax

```
SET ARP mac_address host
        [ /[NO]PERMANENT ]
        [ /[NO]PUBLIC ]

SET NOARP [host]
```

Restrictions

Requires OPER privilege.

Parameters

mac_address

Required.

Specifies the physical network address (the hardware address) on an FDDI, Ethernet, or Token Ring LAN segment to be mapped to an IP address.

For *mac_address*, specify *hh-hh-hh-hh-hh-hh*, where *hh* are pairs of hexadecimal digits.

host

Required.

Specifies the host on the targeted LAN segment. If you do not supply a host name, you must supply its corresponding IP address.

Qualifiers

/PERMANENT
/NOPERMANENT

Optional. Default: /PERMANENT.

Specifies whether the mapping information is cached.

/NOPERMANENT removes ARP mapping after the caching interval.

Not valid with SET NOARP.

/PUBLIC
/NOPUBLIC

Optional. Default: /PUBLIC.

Specifies whether the local ARP responds to ARP requests from other hosts to the specified host.

/NOPUBLIC maps only for the local host.

Not valid with SET NOARP.

Example

```
TCPIP> SET ARP AA-BB-04-05-06-07 CONDOR
```

Permanently maps CONDOR's host name to FDDI address AA-BB-04-05-06-07.

SET BOOTP

SET BOOTP — Creates client entries in the BOOTP database. SET NOBOOTP does not require any qualifiers. *Related commands:* CONVERT/VMS BOOTP, SHOW BOOTP.

Syntax

```
SET [NO]BOOTP host
              [ /FILE=file]
              /HARDWARE=ADDRESS=hex_address
              [ /GATEWAYS=hosts ]
              [ /NETWORK_MASK=IP_address ]
              [ /SERVERS=type=host ]
              [ /TIME_OFFSET=seconds ]
```

Restrictions

Requires read, write, and delete access to the BOOTP database.

Parameters

host

Required.

Specifies the client to which your system will download files upon request. Enter a host name or IP address.

Qualifiers

/FILE=*file*

Optional.

Specifies the name of the client's system image or other file to download upon request.

- By default, upon receiving a request, BOOTP looks for this file in TCPIP$TFTP_ROOT:[*host*], where *host* is the client's host name, excluding the domain.
- If this directory does not exist, BOOTP uses:

  ```
  TCPIP$TFTP_ROOT:[000000].
  ```

- When the TCP/IP Services software receives a boot request, BOOTP verifies the existence and size of this file.

/GATEWAYS=*hosts*
/NOGATEWAYS=*hosts*

Optional. Default: None.

Specifies the gateways used for routing.

/HARDWARE=ADDRESS=*hex_addr*

Required.

Specifies the client's hardware address. For *hex_addr*, specify: *hh-hh-hh-hh-hh-hh*.

/NETWORK_MASK=*IP_address*

Required if you use subnets; otherwise optional.

Specifies the part of the host field of an IP address identified as the subnet.

The software calculates the default by setting the following:

- The bits representing the network field to 1
- The bits representing the host field to 0

You can divide the host field into a site-specific subnetwork and a host field. If you use subnets, you must specify a subnet field.

/SERVERS=*type=host*

Optional.

Specifies other servers whose names BOOTP can supply to clients. Here, *host* specifies a host name or IP address and *type* can be one or more of the following:

[NO]COOKIE	Cookie server
[NO]IEN_NAME	IEN-116 name server
[NO]IMPRESS	Impress network image server (IMAGEN)
[NO]LPR	Berkeley 4BSD print server
[NO]LOG	MIT-LCS UDP logging server
[NO]NAME	BIND name server
[NO]RESOURCE	Resource Location Protocol (RLP) server (RFC-887)
[NO]TIME	Internet time server (RFC-868)

/TIME_OFFSET=*seconds*

Optional. Default: 0 seconds.

Specifies the time difference, in seconds, between the client's time zone and Universal Coordinated Time (UTC) expressed in seconds. This value is zero (0) in the British Isles and parts of Europe, a positive number for locations east of the zero meridian, and a negative number for locations west of the zero meridian.

Examples

1. TCPIP> **SET BOOTP PLOVER /HARDWARE=ADDRESS=08-00-2D-20-23-21 -**
   _TCPIP> **/FILE=PLOVER.SYS**

   Adds client host PLOVER, with hardware address 08-00-2D-20-23-21 to the BOOTP database. BOOTP can respond to a remote boot request from client PLOVER with a reply packet containing the name of the file to down load and its IP address.

2. TCPIP> **SET BOOTP ERN /HARDWARE=ADDRESS=98-00-2D-20-23-21 -**
   _TCPIP> **/SERVERS=COOKIE=(PLOVER,GULL)**

   Adds client host ERN to the BOOTP database and specifies that ERN will use PLOVER AND GULL as cookie servers.

3. TCPIP> **SET BOOTP PLOVER /HARDWARE=ADDRESS=08-00-2D-20-23-21 -**
   _TCPIP> **/SERVERS=(COOKIE=GULL,NAME=BIRDS)**

   Adds client host PLOVER to the BOOTP database and specifies that PLOVER will use GULL as a COOKIE server and BIRDS as its name server.

SET COMMUNICATION

SET COMMUNICATION — Modifies the IP, TCP, UDP, and INET_ACP software on the running system. **Related commands:** SET CONFIGURATION COMMUNICATION, SHOW COMMUNICATION.

Syntax

```
SET COMMUNICATION [ /ACCEPT=options ]
                  [ /DOMAIN=domain ]
                  [ /LOCAL_HOST=host ]
                  [ /PROXIES=n ]
                  [ /REJECT=options ]
```

Restictions

Requires OPER privilege.

Qualifiers

/ACCEPT { =[NO]HOSTS=(*hosts*) | =[NO]NETWORKS=(*networks*) }

Optional. Default: All hosts and all networks.

Accepts communication from the hosts and networks specified.

Do not specify the same hosts or networks for both /ACCEPT and /REJECT.

To delete an /ACCEPT entry, specify it again using the NOHOSTS or NONETWORKS option.

Specify one of the following:

- [NO]HOSTS=*hosts*

Hosts that can access TCP/IP Services. Maximum is 32. For example:

/ACCEPT=HOSTS=(*host1_name*,*host2_name*, *host3_address*)

- [NO]NETWORKS=*networks*

  Networks that can access TCP/IP Services. Maximum is 16.

  Use the following syntax:

  NETWORKS=(*net1[:net1mask],net2[:net2mask],...*)

  For each network, specify: *network*:[*network_mask*]. The network mask is optional. (Default: class number of your network. For example, the default for 11.200.0.0. is 255.0.0.0.). For example:

  /ACCEPT=NETWORKS=(*net1_name*,*net2_addr*,*net3_addr*:*net3_mask*)

/DOMAIN=*domain*

Optional.

Specifies your system's local domain. This qualifier requires either SYSPRV or BYPASS privilege.

/LOCAL_HOST=*host*

Optional.

Defines the following logical names for the local host:

- TCPIP$INET_HOST=*host-name*

  This logical is always set with the primary host name even if the alias name was specified as host.

- TCPIP$INET_HOSTADDR=*host-IP-address*

  If the local host has multiple IP addresses, this logical name is set with a name for each address, called TCPIP$INET_HOSTADDR*n*, where *n* is a number starting at 2.

This qualifier requires either SYSPRV or BYPASS privilege.

/PROXIES=*n*

Optional. Default: Number of communication proxies plus 10, with a minimum of 20.

Specifies the maximum size of the proxy cache. If you plan to add entries to the proxy database after you start the TCP/IP Services software, set /PROXIES to a value higher than the default.

You cannot change this value if the TCP/IP Services software is running.

/REJECT {=[NO]HOSTS=(*hosts*) | =[NO]NETWORKS=(*networks*) | =[NO]MESSAGE=(*message*) }

Optional. Default: No rejections.

Specifies the hosts or networks that cannot access the TCP/IP Services software, including the rejection message that TCP/IP might return.

(For remote login, remote shell, and remote executive, the rejection message is preceded by a byte with a value of 1 and terminated by a byte with a value of zero.)

Do not specify the same hosts or networks for both /ACCEPT and /REJECT.

To delete a /REJECT entry, specify it again using the NOHOSTS or NONETWORKS option.

Specify one of the following:

- [NO]HOSTS=*hosts* to list hosts that cannot access TCP/IP Services. Maximum is 32. The syntax is:

  /REJECT=HOSTS=(*host1_name*,*host2_name*,*host3_address*)

- [NO]NETWORKS=*networks* to list networks that cannot access TCP/IP Services Maximum is 16. The syntax is:

  NETWORKS=(*net1[:net1mask],net2[:net2mask],...*)

  For each network, specify *network*:*network_mask*. The network mask is optional. Default: Class number of your network. For example, the default for 11.200.0.0. is 255.0.0.0. The syntax is:

  /REJECT=NETWORKS=(*net1_name*,*net2_address*,*net3_addr*:*net3_mask*)

Example

```
TCPIP> SET COMMUNICATION -
_TCPIP> /REJECT=NETWORK=(16.30.0.0:255.255.0.0,16.40.0.0:255.255.0.0)
```

Sets all the services to be inaccessible to the two specified networks.

SET CONFIGURATION BIND

SET CONFIGURATION BIND — Configures the BIND name server. Creates the BIND server configuration file, which holds the following information: cluster alias or aliases, server type (primary, secondary, or forwarding), domains to be served, and location from which the BIND server gets initial information for lookups. You can configure the BIND server as follows: for one or more Internet domains, as one kind of BIND server (primary, secondary, or forwarding), as multiple kinds of BIND servers, or on TCP/IP clusters for cluster load balancing. This command does not create a BIND 8.1 configuration. If you want to take full advantage of the new features available with the BIND 8.1 implementation, you should set up your BIND environment by editing the TCPIP$BIND.CONF configuration file. Refer to the VSI TCP/IP Services for OpenVMS Management manual for detailed instructions. If you choose to configure your BIND environment with the SET CONFIGURATION BIND command, you must enter the command CONVERT/CONFIGURATION BIND before running BIND. *Related commands:* SHOW CONFIGURATION BIND, CONVERT /CONFIGURATION BIND

Syntax

```
SET CONFIGURATION [NO]BIND [ /CACHE=options ]
```

```
[ /[NO]CLUSTER=names ]
[ /FORWARDERS=options ]
[ /PRIMARY=options ]
[ /SECONDARY=options ]
```

Restrictions

Requires SYSPRV or BYPASS privilege.

Qualifiers

/CACHE=([NO]DOMAIN:*do*,[NO]FILE:*file*)

Optional. Default: None.

Specifies the cache server for the specified domain. Do not use with /FORWARDERS. Use with /PRIMARY and /SECONDARY.

The cache tells the primary or secondary server how to use hints to find the file. These hints let a server find a root name server. With this ability, the server can answer requests even if it does not have the information. You can use the following options:

- DOMAIN keys to a particular record within a type.

  NODOMAIN deletes the entry.

- FILE specifies the name of the hints file.

If you use /CACHE with no options:

- DOMAIN defaults to "." ("`root`").
- FILE defaults to NAMED.CA.

/CLUSTER=*name*
/NOCLUSTER=*name*

Required to configure cluster load balancing.

Identifies the name of a TCP/IP cluster as the first step to setting up cluster load balancing.

For information about the remaining procedure, refer to the VSI TCP/IP Services for OpenVMS Management manual.

/NOCLUSTER=*name* deletes the specified name as a cluster load-balancing host.

/FORWARDERS=([NO]HOST:*host*)

Optional.

Specifies the forwarding server.

NOHOST deletes hosts.

/PRIMARY=([NO]DOMAIN:*do*,[NO]FILE:*file*)

Optional. Default: None.

Specifies the primary server for the specified zone. Multiple primary servers are allowed if each is associated with a different domain.

- DOMAIN keys to a particular domain.

  NODOMAIN deletes the entry.

- FILE specifies the domain to be served.

  If you do not specify a file, the default file name is created from the value that you supply with the DOMAIN option.

  NOFILE specifies that no file is created.

/SECONDARY=([NO]DOMAIN:*do*,[NO]FILE:*file*,[NO]HOST:*host*)

Optional. Default: None.

Specifies the secondary server for the specified zone. Multiple secondary servers are allowed if each is associated with a different domain.

- DOMAIN keys to a particular record within a type.

  NODOMAIN deletes the entry.

- FILE specifies the name of the boot file.

  If you do not specify a file, the default file name is created from the value that you supply with the DOMAIN option.

  NOFILE specifies that no file is created.

- HOST is a list of hosts from which the secondary server copies the database file.

  NOHOST deletes hosts from the host list.

Examples

1. ```
   TCPIP> SET CONFIGURATION BIND -
   _TCPIP> /PRIMARY=(DOMAIN:RHEA.LAB.UBIRD.EDU)
   ```

   Configures the host as the primary server for domain RHEA.LAB.UBIRD.EDU.

2. ```
   TCPIP> SET CONFIGURATION BIND -
   _TCPIP> /SECONDARY=(DOMAIN:JACANA.LAB.UBIRD.EDU) -
   _TCPIP> /SECONDARY=(FILE:JACANA.DB,HOST=MARSHY)
   ```

   Configures the host as a secondary server for domain JACANA.LAB.UBIRD.EDU and names the boot file JACANA.DB.

   Omitting the file name would default to file JACANA_LAB_UBIRD_EDU.DB.

3. ```
   TCPIP> SET CONFIGURATION BIND -
   _TCPIP> /SECONDARY=(DOMAIN:0.192.IN-ADDR.ARPA,HOST:WEBBED)
   ```

   Configures the host as a secondary server for the reverse lookup domain for addresses that have the form 192.0.*.*.
   ```

The boot file name defaults to 0_192_IN-ADDR_ARPA.DB and the host copies this file from the host WEBBED.

4. `TCPIP> SET CONFIGURATION BIND /CACHE`

   Points the server to the cache file (NAMED.CA), which contains hints about the root name servers.

SET CONFIGURATION COMMUNICATION

SET CONFIGURATION COMMUNICATION — Enters information into the configuration database to start the IP, TCP, UDP, and INET_ACP software when the system starts up. When TCP/IP Services starts up, this configuration overrides the default settings. **Related commands:** SHOW CONFIGURATION COMMUNICATION, SET COMMUNICATION.

Syntax

```
SET CONFIGURATION COMMUNICATION [ /ACCEPT=options ]
                                [ /DOMAIN=domain ]
                                [ /LOCAL_HOST=host ]
                                [ /PROXIES=n ]
                                [ /REJECT=options ]
```

Restrictions

Requires OPER privilege.

Qualifiers

/ACCEPT { =[NO]HOSTS=(*hosts*) | =[NO]NETWORKS=(*networks*) }

Optional. Default: All hosts and all networks.

Accepts communication from the hosts and networks specified.

Do not specify the same hosts or networks for both /ACCEPT and /REJECT.

To delete an /ACCEPT entry, specify it again using the NOHOSTS or NONETWORKS option.

Specify one of the following:

- [NO]HOSTS=*hosts*

  Hosts that can access TCP/IP Services. Maximum is 32. For example:

  /ACCEPT=HOSTS=(*host1_name*,*host2_name*,*host3_address*)

- [NO]NETWORKS=*networks*

  Networks that can access TCP/IP Services. Maximum is 16.

  The syntax is:

  NETWORKS=(*net1[:net1mask],net2[:net2mask],...*)

For each network, specify: *network*:[*network_mask*]. The network mask is optional. (Default: class number of your network. For example, the default for 11.200.0.0. is 255.0.0.0.). For example:

/ACCEPT=NETWORKS=(*net1_name*,*net2_addr*,*net3_addr*:*net3_mask*)

/DOMAIN=*domain*

Optional.

Specifies your system's local domain. This qualifier requires either SYSPRV or BYPASS privilege.

/LOCAL_HOST=*host*

Optional.

Defines the following logical names for the local host:

- TCPIP$INET_HOST=*host-name*

  This logical is always set with the primary host name, even if the alias name was specified as host.

- TCPIP$INET_HOSTADDR=*host-IP-address*

  If the local host has multiple IP addresses, this logical name is set with a name for each address, called TCPIP$INET_HOSTADDR*n*, where *n* is a number starting at 2.

This qualifier requires either SYSPRV or BYPASS privilege.

/PROXIES=*n*

Optional. Default: Number of communication proxies plus 10, with a minimum of 20.

Specifies the maximum size of the proxy cache. If you plan to add entries to the proxy database after you start the TCP/IP Services software, set /PROXIES to a value higher than the default.

You cannot change this value if the TCP/IP Services software is running.

/REJECT { =[NO]HOSTS=(*hosts*) | =[NO]NETWORKS=(*networks*) | =[NO]MESSAGE=(*message*) }

Optional. Default: No rejections.

Specifies the hosts or networks that cannot access the TCP/IP Services software, including the rejection message that TCP/IP might return.

(For remote login, remote shell, and remote executive, the rejection message is preceded by a byte with a value of 1 and terminated by a byte with a value of 0.)

Do not specify the same hosts or networks for both /ACCEPT and /REJECT.

To delete a /REJECT entry, specify it again using the NOHOSTS or NONETWORKS option.

Specify one of the following:

- [NO]HOSTS=*hosts* to list hosts that cannot access TCP/IP Services. Maximum is 32. The syntax is:

  /REJECT=HOSTS=(*host1_name*, *host2_name*, *host3_address*)

- [NO]NETWORKS=*networks* to list networks that cannot access TCP/IP Services Maximum is 16. The syntax is:

  NETWORKS=(*net1[:net1mask],net2[:net2mask],...*)

  For each network, specify *network*:*network_mask*. The network mask is optional. Default: Class number of your network. For example, the default for 11.200.0.0. is 255.0.0.0. The syntax is:

  /REJECT=NETWORKS=(*net1_name*,*net2_address*,*net3_addr*:*net3_mask*)

Examples

```
TCPIP> SET CONFIGURATION COMMUNICATION -
_TCPIP> /REJECT=NETWORK=(16.30.0.0:255.255.0.0,16.40.0.0:255.255.0.0)
```

In the configuration database, sets all the services to be inaccessible to the two specified networks.

SET CONFIGURATION ENABLE SERVICE

SET CONFIGURATION ENABLE SERVICE — Modifies service-related information in the permanent configuration database that enables (or disables) services for startup. Allows you to specify that the service be enabled or disabled for startup on the current node only or on all nodes in the cluster. To specify clusterwide enabling or disabling of services, use the /COMMON qualifier. SET CONFIGURATION ENABLE SERVICE adds an entry for a service to the list of enabled services in the configuration database. SET CONFIGURATION ENABLE NOSERVICE removes an entry for a service from the list of enabled services in the configuration database. **Related commands:** SHOW CONFIGURATION ENABLE SERVICE, ENABLE SERVICE.

Syntax

```
SET CONFIGURATION ENABLE [NO]SERVICE service
                                    [ /COMMON ]
                                    [ /[NO]CONFIRM ]
```

Parameters

service

Required.

Specifies the service to add or delete from the configuration database.

Qualifiers

/COMMON

Optional. Default (when /COMMON is not specified): node-specific enabling or disabling of services.

Modifies service-related information in the configuration database for the clusterwide enabling or disabling of services.

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM if you use wildcards; otherwise, /NOCONFIRM.

Use only with SET CONFIGURATION ENABLE NOSERVICE. Controls whether the software requests you to confirm before it deletes an entry. With /CONFIRM enabled, the software requests confirmation. At the CONFIRM: prompt, enter one of the following:

- Y to delete the entry
- N to retain the entry

The /NOCONFIRM qualifier eliminates all user confirmation when deleting service entries.

Examples

1. `TCPIP> SET CONFIGURATION ENABLE SERVICE TELNET`

   In the configuration database, enables the TELNET service for startup on this node.

2. `TCPIP> SET CONFIGURATION ENABLE SERVICE FTP /COMMON`

   In the configuration database, enables the FTP service for startup on every node in the cluster.

3. ```
   TCPIP> SET CONFIGURATION ENABLE NOSERVICE *
   Enable service TELNET
   Remove? [N]: Y
   ```

   In the configuration database, disables any service enabled for startup on this node, if confirmed by the user.

SET CONFIGURATION INTERFACE

SET CONFIGURATION INTERFACE — Enters information into the configuration database, which defines one of the following when TCP/IP Services starts up: an Internet interface (hardware connection to the network), a serial line Internet interface (a form of hardware connection to the network), or a pseudointerface (a data structure that extends subnet routing so that, on the same physical network, an interface acts as a gateway between multiple subnets). **Related commands:** SHOW INTERFACE, SET INTERFACE. **Applies to:** Routing.

Syntax

```
SET CONFIGURATION [NO]INTERFACE interface
                                [/[NO]ARP ]
                                [/[NO]AUTO_START ]
                                [/BROADCAST_MASK=IP_address ]
                                [/C_BROADCAST_MASK=IP_address]
                                [/C_NETWORK=IP_address ]
                                [/[NO]CLUSTER=host ]
                                [/COMPRESS=options ]
                                [/DESTINATION=IP_address ]
```
```

```
[/[NO]DHCP ]
[/FLOWCONTROL ]
[/HOST=host ]
[/[NO]LOOPBACK ]
[/NETWORK_MASK=IP_address ]
[/[NO]PRIMARY ]
[/SERIAL_DEVICE=device ]
```

Restrictions

This command requires:

- OPER privilege
- Read access to the hosts database
- Read access to the networks database
- Read, write, and delete access to the routes database

Every host on the same network must have the same network mask.

Parameters

interface

Required.

Specifies an interface name for the communication controller, such as RF1, RT1, ZE0, XE0, SL0, SL1, SL2, PP0, PP1, PP2. Refer to the chapter on configuring network interfaces in the VSI TCP/IP Services for OpenVMS Management manual for more information.

Qualifiers

/ARP
/NOARP

Optional. Default: /ARP.

Enables IP address-to-hardware address (Ethernet or FDDI) mapping.

/ARP is valid when you create an interface but not when you modify an existing interface.

/AUTO_START
/NOAUTO_START

Optional. Default: /AUTO_START.

Valid for a SLIP or PPP interface. Automatically creates the interface when TCP/IP Services starts.

/BROADCAST_MASK=*IP_address*

Optional.

Sets the Internet interface to receive all broadcast messages.

TCP/IP Services calculates the default by the following methods:

- Using the network number
- Setting all bits in the host number field to 1

/C_BROADCAST_MASK=*IP_address*

Optional.

Sets the cluster broadcast mask to receive all broadcast messages.

The software calculates the default by the following methods:

- Using the network number
- Setting all bits in the host number field to 1

/C_NETWORK=*IP_address*

Optional.

Sets the network mask of the cluster network. This mask is specific to the cluster host network.

The software calculates the default by using the following methods:

- Setting the bits representing the network fields to 1
- Setting the bits representing the host field to 0

/CLUSTER=*host*
/NOCLUSTER

Optional. Default: None.

Specifies the cluster host name (alias host identifier).

Before using this qualifier, first define the same name in the hosts database.

/CLUSTER=*host* associates the alias host identifier with each interface in a cluster.

/NOCLUSTER disables Internet cluster processing on the specified interface.

Caution

When you specify /NOCLUSTER, active communication is aborted for applications bound to the cluster alias name.

/COMPRESS= {ON | OFF | AUTOMATIC}

Optional. Defaults: For PPP interface, /COMPRESS=ON; for SLIP interface, /COMPRESS=OFF.

Valid for SLIP and PPP interfaces.

Enables or disables TCP header compression.

/COMPRESS=AUTOMATIC turns off compression unless the remote end begins to use it.

/DESTINATION=*IP_address*

Optional.

Valid for a PPP interface.

Used on the local host to provide dialup access to remote systems. The value specified is the IP address to be given to remote clients for use while the PPP connection is active. If using /DESTINATION, you must provide the address of the local host by using the /HOST qualifier.

/DHCP
/NODHCP

Optional.

Designates the interface as a DHCP-controlled interface in the permanent database.

/FLOWCONTROL

Optional. Default: No flow control.

Valid for a SLIP interface. Enables the handling of XON and XOFF characters to interoperate properly with modems that are configured to interpret these characters locally.

Specify /FLOWCONTROL only if the host at the other end of the line is running TCP/IP Services.

/HOST=*host*

Required when first setting the interface; optional if the interface is already defined. Always required for a SLIP interface. Optional for a PPP interface unless you are setting up the local host as a dialup provider by using the /DESTINATION qualifier.

Local host name or IP address using the interface. If not specified for a PPP interface, PPP obtains the correct address from the remote host.

If your host is multihomed, specify an address.

/LOOPBACK
/NOLOOPBACK

Optional. Default: /NOLOOPBACK.

Sets loopback mode.

/NETWORK_MASK=*IP_address*

Required if you use subnets.

The part of the host field of the IP address identified as the subnet.

The software calculates the default by the following methods:

- Setting the bits representing the network fields to 1
- Setting the bits representing the host field to 0

An IP address consists of a network number and a host number. You can also divide the host field into a site-specific subnetwork and host field.

/PRIMARY
/NOPRIMARY

Optional.

For DHCP-controlled interfaces, designates the interface from which system-wide configuration options (such as the IP address of the BIND server) are used.

/SERIAL_DEVICE=*device*

Required for SLIP and PPP interfaces; otherwise, not used.

Identifies the OpenVMS terminal device used as a serial device. Specify an arbitrary terminal device name. (Unlike Ethernet, FDDI, and Token Ring interface names, a serial interface name is not related to the OpenVMS device name.)

Examples

1. ```
   TCPIP> SET CONFIGURATION INTERFACE SL5 /HOST=LARK -
   _TCPIP> /NETWORK_MASK=255.255.255.0 /SERIAL_DEVICE=TTA3: -
   _TCPIP> /COMPRESS=ON /FLOWCONTROL
   ```

   Configures SLIP interface SL5, using the local IP address assigned to host LARK, with a subnet mask of 255.255.255.0.

   The interface uses the terminal device TTA3:.

   The /COMPRESS qualifier enables TCP header compression (CSLIP).

   The /FLOWCONTROL qualifier enables special handling of XON and XOFF characters for proper interoperation with modems that are configured to interpret these characters locally.

2. ```
   TCPIP> SET CONFIGURATION INTERFACE FF0 /HOST=KESTREL -
   _TCPIP> /NETWORK_MASK=255.255.0.0 -
   _TCPIP> /BROADCAST_MASK=128.30.0.0 /ARP
   ```

   For new interface `FF0` on host KESTREL, sets the network mask to 255.255.0.0, sets the broadcast mask to 128.30.0.0, enables ARP, and activates the interface.

3. ```
   TCPIP> SET CONFIGURATION INTERFACE PP0 /SERIAL_DEVICE=TTA0: -
   _TCPIP> /HOST=10.10.1.2 /DESTINATION=10.10.1.3
   ```

   Configures the interface as a PPP serial device. This command specifies that the local host is a dialup provider. The address specified with the /DESTINATION qualifier (10.10.1.3) is the address assigned to the client system requesting an address.

   Refer to the VSI TCP/IP Services for OpenVMS Management manual for more information about setting up interfaces for SLIP and PPP communication.

SET CONFIGURATION MAP

SET CONFIGURATION MAP — Adds information to the configuration database that maps (logically links) one of the following to the NFS server: an OpenVMS disk (requires one execution
   ```

of SET CONFIGURATION MAP to map the disk to a UNIX path name (logical file system)) or a container file system (requires two executions of SET CONFIGURATION MAP. The first maps the disk, and the second maps the file system). Mapping creates a logical file system, also called an NFS file system. When the NFS server starts up, it issues a GENERATE MAP command, which creates the mappings for disks and container file systems; these mappings are viewable with the SHOW MAP command. *Related commands:* ADD EXPORT, SHOW EXPORT, REMOVE EXPORT, MAP, UNMAP, SET CONFIGURATION NOMAP, SHOW MAP, SHOW CONFIGURATION MAP

Syntax

```
SET CONFIGURATION MAP "file system name" logical_file_system
```

Restrictions

Requires SYSPRV and BYPASS privileges.

Parameters

"*file system name*"

Required.

Specifies the name for the file system or disk. When mapping a disk, the "*/path*" can be only one level from the root. This parameter specifies the name by which users access the file system.

logical_file_system

Required.

Specifies the file system to make known to the NFS server.

To map an OpenVMS file system, specify its disk as follows:

```
$ SET CONFIGURATION MAP "/disk" disk:
```

To map a container file system, specify the disk and the directory name as follows:

```
TCPIP> SET CONFIGURATION MAP "/container_name" -
_TCPIP> disk:[vms.directory.name]
```

Examples

1. ```
   TCPIP> SET CONFIGURATION MAP "/usr" CANARY$DUA2:
   ```

   Maps local disk CANARY$DUA2: to `/usr`. This disk can be exported to users on remote NFS clients as `/usr`.

2. ```
   TCPIP> SET CONFIGURATION MAP "/remote" VERDIN$DUA3: -
   _TCPIP> SET CONFIGURATION MAP "/flyers" VERDIN$DUA3:[UNIX_BIRD_FILES]
   ```

   Maps [UNIX_BIRD_FILES], a container file system on disk VERDIN$DUA3:, to `/flyers`. This file system can be exported to NFS server users as `/flyers`. (The first MAP command maps the underlying OpenVMS file system.)

SET CONFIGURATION NAME_SERVICE

SET CONFIGURATION NAME_SERVICE — When TCP/IP Services starts up, configures the BIND resolver and designates a BIND server. All settings are systemwide. **Related commands:** SET NAME_SERVICE, SHOW CONFIGURATION NAME_SERVICE

Syntax

```
SET CONFIG [NO]NAME_SERVICE [ /[NO]SERVER=host]
                            [ /[NO]DOMAIN=domain ]
                            [ /[NO]PATH=domain ]
                            [ /RETRY=number of retries ]
                            [ /TIMEOUT=seconds ]
                            [ /TRANSPORT=protocol ]
```

Qualifiers

/CLUSTER=*dev:[directory]*

Optional.

Specifies the common BIND directory. By default, the clusterwide common database *common-disk:*[TCPIP$BIND_*common*] is used. This qualifier reloads the BIND database on every master BIND server running the OpenVMS cluster.

/DOMAIN=*domain*
/NODOMAIN

Optional. Default: The local domain.

Defines the default domain.

/NODOMAIN deletes the definition of the domain.

/PATH=*domain*
/NOPATH=*domain*

Optional. SYSNAM privilege is required for this command.

Defines the BIND resolver domain search list. The /NOPATH qualifier removes domains from the list.

To specify multiple domains, list them by search preference. The resolver starts with the first domain on the list, and continues to search each domain until the name is found (or until all domains have been exhausted and the lookup fails).

If you define a domain list and then issue another SET CONFIGURATION NAME_SERVICE / PATH command, TCP/IP Services appends the new domains to the end of the list.

If a search list is not defined, the default behavior of the BIND resolver is to do a lookup on the name as you typed it. If that lookup fails, then the default domain is appended and the lookup is attempted again.

/RETRY=*number of retries*

Optional. Default: Four retries.

Number of times that the BIND resolver attempts to contact a BIND server if previous tries failed.

/SERVER=*host*
/NOSERVER=*host*

Optional.

Host name or address of the BIND server or servers that the BIND resolver will query.

To specify multiple hosts, list them by request preference. The resolver sends the first lookup request to the first host on the list.

/NOSERVER removes hosts from the list.

If you define a server list and then issue another SET CONFIGURATION NAME_SERVICE / SERVER command, TCP/IP Services appends the new servers to the end of the list.

/TIMEOUT=*seconds*

Optional. Default: 4 seconds.

Timeout interval for the BIND resolver's requests to a BIND server. Represents the length of time to wait for a reply after each retry attempt.

The total timeout period will be:

*timeout_value * retry_value * number_servers*

/TRANSPORT=*protocol*

Optional. Default: UDP.

Protocol used for communicating with a BIND server. Specify one:

- UDP
- TCP

Examples

1. `TCPIP> SET CONFIGURATION NAME_SERVICE /SERVER=(PARROT,SORA,JACANA)`

   When TCP/IP Services starts, defines hosts PARROT, SORA, and JACANA as BIND servers.

2. `TCPIP> SET CONFIGURATION NAME_SERVICE /SERVER=OSPREY -`
   `_TCPIP> /PATH=(abc.dec.com,xyz.dec.com)`

   When TCP/IP Services starts, defines host OSPREY as the BIND server. The BIND resolver searches the `abc.dec.com` domain first, and then searches the `xyz.dec.com` domain.

SET CONFIGURATION NOMAP

SET CONFIGURATION NOMAP — Removes map records from the configuration database that were previously added with SET CONFIGURATION MAP. When the NFS server starts up, it issues

a GENERATE MAP command that creates the mappings for disks and container file systems. *Related commands:* SET CONFIGURATION MAP, SHOW CONFIGURATION MAP, ADD EXPORT, SHOW EXPORT, REMOVE EXPORT, MAP, UNMAP, SHOW MAP.

Syntax

```
SET CONFIGURATION NOMAP "/path/name" [ /[NO]CONFIRM ]
```

Restrictions

Requires SYSPRV and BYPASS privilege.

Parameters

"/path/name"

Required.

UNIX name of the file system to unmap.

You can use wildcards.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM if you use a wildcard.

With /CONFIRM enabled, requests confirmation before unmapping each file system. If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

Example

```
TCPIP> SET CONFIGURATION NOMAP "/disk_host"
```

Unmaps the NFS file system `/remote`, making it unavailable to client users when TCP/IP Services starts.

SET CONFIGURATION PROTOCOL

SET CONFIGURATION PROTOCOL — Enters information into the configuration database that sets the parameters for ICMP, IP, TCP, and UDP when TCP/IP Services starts up. **Related commands:** SET PROTOCOL, SHOW CONFIGURATION PROTOCOL.

Syntax

```
SET CONFIGURATION PROTOCOL ICMP [ /[NO]REDIRECT ]

SET CONFIGURATION PROTOCOL IP [ /[NO]FORWARD ]
                              [ /REASSEMBLY_TIMER=seconds ]
```

```
SET CONFIGURATION PROTOCOL TCP [ /[NO]MTU_SEGMENT_SIZE ]
                               [ /[NO]DELAY_ACK ]
                               [ /DROP_COUNT=n ]
                               [ /PROBE_TIMER=seconds ]
                               [ /QUOTA=[ SEND=bytes,RECEIVE=bytes ] ]
                               [ /[NO]WINDOW_SCALE ]

SET CONFIGURATION PROTOCOL UDP [ /[NO]BROADCAST ]
                               [ /[NO]FORWARD ]
                               [ /QUOTA=options ]
```

Restrictions

Requires OPER privilege.

Parameters

{ICMP | IP | TCP | UDP}

Required.

Specifies the protocol software to configure.

ICMP Qualifiers

/REDIRECT
/NOREDIRECT

Optional. Default: /NOREDIRECT.

Sends ICMP_REDIRECT messages.

IP Qualifiers

/FORWARD
/NOFORWARD

Optional. Default: /NOFORWARD.

Forwards IP messages to other hosts.

/REASSEMBLY_TIMER=*n*

Optional. Default: 7 seconds. Valid range: 1 to 126.

Sets the maximum time for trying to reassemble a received datagram.

TCP Qualifiers

/MTU_SEGMENT_SIZE
/NOMTU_SEGMENT_SIZE

Optional. Default: /NOMTU_SEGMENT_SIZE.

If a connection is more than one hop away, sets the segment size. Specify one of the following:

/MTU_SEGMENT_SIZE	Sets the segment size as close as possible to the maximum transfer unit (MTU) size.
/NOMTU_SEGMENT_SIZE	Sets the segment size as close as possible to the standard 512 bytes.

/DELAY_ACK
/NODELAY_ACK

Optional. Default: /DELAY_ACK.

Enables or disables a delay before sending acknowledgments:

/DELAY_ACK	ACKs are generated with a delay.
/NODELAY_ACK	ACKs are generated without any delay.

/DROP_COUNT=*n*

Optional.

Number of idle probes that can go unsatisfied before the software declares a TCP connection dead and closes it.

/PROBE_TIMER=*n*

Optional. Default: 75 seconds.

Number of seconds between probes for idle TCP connections (when the SO_KEEPALIVE option is set). If the remote system fails to respond, the connection is removed. Also, when initiating a TCP connection request, indicates the maximum number of seconds that the software waits for a response from the remote system before the request times out.

/QUOTA=[SEND=*bytes*,RECEIVE=*bytes*]

Optional.

Specifies the queue size (in bytes) for messages.

The options for setting TCP message queue size are:

- RECEIVE:*n* — Receive queue size. Default: 4096 bytes.
- SEND:*n* — Send queue size. Default: 4096 bytes.

/WINDOW_SCALE
/NOWINDOW_SCALE

Optional.

Turns TCP window scaling on and off. Default is on.

Scaling allows windows larger than 64 KB to be represented in the normal 16-bit TCP window field. Large windows allow improved throughput. Turning this option off may help troubleshoot communication problems with another TCP/IP implementation.

UDP Qualifiers

/BROADCAST
/NOBROADCAST

Optional. Default: /NOBROADCAST.

Enables privilege checking for broadcast messages.

- /BROADCAST — Nonprivileged users can send broadcast messages.
- /NOBROADCAST — To send broadcast messages, users need a privileged UIC or SYSPRV, BYPASS, or OPER privilege.

Sun RPC applications use broadcast messages and need privilege checking disabled.

/FORWARD
/NOFORWARD

Optional. Default: /NOFORWARD.

Forwards IP messages.

/QUOTA=*options*

Optional.

Specifies the queue size (in bytes) for messages.

The options for setting UDP message queue size are:

- RECEIVE:*n* — Receive queue size. Default: 9000 bytes.
- SEND:*n* — Send queue size. Default: 9000 bytes.

Examples

1. `TCPIP> SET CONFIGURATION PROTOCOL IP /FORWARD`

   Sets IP to forward messages to other hosts, including other Internet cluster nodes.

2. `TCPIP> SET CONFIGURATION PROTOCOL TCP /PROBE_TIMER=50`

   Sets the TCP protocol probe timer parameter to 50 seconds.

SET CONFIGURATION SMTP

SET CONFIGURATION SMTP — Modifies the SMTP configuration in the configuration database. SET CONFIGURATION NOSMTP with no qualifiers deletes all SMTP records. *Related commands:* SHOW CONFIGURATION SMTP

Syntax

```
SET CONFIGURATION [NO]SMTP [ /ADDRESS_RETRIES=n ]
```

```
                        [ /GATEWAY=option=host ]
                        [ /HOP_COUNT_MAXIMUM=n ]
                        [ /INTERVAL=options ]
                        [ /[NO]LOG=[file] ]
                        [ /OPTIONS=options ]
                        [ /QUEUES=n ]
                        [ /RECEIVE_TIMEOUT=minutes ]
                        [ /SEND_TIMEOUT=minutes ]
      [ /SUBSTITUTE_DOMAIN=[NO]NAME=fully-qualified-name ]
                        [ /[NO]ZONE[=domain] ]
```

Restrictions

For clusters, issue this command only on the nodes where the SMTP queues reside — that is, on nodes that are not using clusterwide queues and are not managing clusterwide queues for other nodes.

Requires SYSPRV or BYPASS privilege.

Qualifiers

/ADDRESS_RETRIES=*n*

Optional. Default: 16.

Maximum number of different addresses to which SMTP will send as it tries to deliver mail. Beyond this number of attempts, the message is undeliverable.

A message is also undeliverable if SMTP fails to deliver after it attempts all the possible addresses from an MX lookup.

/GATEWAY=*option=host*

Optional. Default: None.

An alternate route through which SMTP sends mail if delivery fails.

- [NO]ALTERNATE=*host*
  - Alternate host or domain to which delivery is attempted.
  - Used by ZONE, if a zone is defined, as the last chance for delivery (see the /ZONE qualifier).
  - NOALTERNATE deletes an existing alternate destination.
- [NO]GENERAL_PURPOSE=*host*
  - Gateway to handle non-SMTP mail, for example, UUCP addresses.
  - NOGENERAL_PURPOSE deletes the specified destination for protocols other than SMTP.

/HOP_COUNT_MAXIMUM=*n*

Optional. Default: 16.

Maximum number of relays (hops) between routers until SMTP considers the mail undeliverable.

/INTERVAL={ INITIAL="*OpenVMS_delta_time*" | RETRY="*OpenVMS_delta_time*" | MAXIMUM="*OpenVMS_delta_time*" }

Optional. Defaults: INITIAL=30 minutes, RETRY=60 minutes, MAXIMUM=3 days.

Time intervals related to repeated attempts before delivery fails. Specify the value within quotation marks as follows: "*dddd hh:mm:ss:cc.*" For example:

dddd	= days (0–9999)
h	= hours (0–24)
m	= minutes
s	= seconds
cc	= milliseconds

You can modify the following options:

- INITIAL="*OpenVMS_delta_time*" is the amount of time that SMTP waits before making a second attempt to deliver.
- RETRY="*OpenVMS_delta_time*" is the time SMTP waits between retries, starting with the second attempt. (Recommended time: twice the initial interval.)
- MAXIMUM="*OpenVMS_delta_time*" is the maximum elapsed time that SMTP retries delivery.

/LOG=[*file*]
/NOLOG=[*file*]

Optional. Default: SYS$SPECIFIC:[TCPIP_SMTP]TCPIP$SMTP_LOGFILE.LOG.

File to which SMTP queue activity is logged.

/OPTIONS=*options*

Optional. Defaults: NOEIGHT_BIT, HEADERS, NORELAY.

The following SMTP options are available:

- [NO]EIGHT_BIT

  All characters must have the eighth bit clear. Allows the transmission of 8-bit characters.

- Header control. Specify one of the following:

HEADERS	Headers are printed at bottom of messages.
NOHEADERS	Headers are omitted.
TOP_HEADERS	Headers are printed at top of messages.
NOTOP_HEADERS	Resets TOP_HEADERS to the default.

- [NO]RELAY

Relays mail to other hosts by functioning as an end node.

/QUEUES=*n*

Optional. Default: 1.

Number of execution queues for the specified nodes.

Use this qualifier only on nodes that own the SMTP queues — that is, nodes not using clusterwide SMTP queues or managing SMTP clusterwide queues for other nodes.

/RECEIVE_TIMEOUT=*minutes*

Optional. Default: 5 minutes.

Maximum time between socket receipts of a message for a particular dialog.

If a message is not received within this interval, the connection is broken and the mail control file is deleted.

/SEND_TIMEOUT=*minutes*

Optional. Defaults:

DATA — 3 minutes
INITIAL — 5 minutes
MAIL — 5 minutes
RECEIPT — 5 minutes
TERMINATION — 10 minutes

Maximum time between remote host acknowledgments of a particular SMTP command.

If an acknowledgment is not received within the specified time, it is assumed that there are communication problems with the remote host. If the next delivery attempt takes place before the mail's delivery date, the mail is rescheduled for later delivery.

/SUBSTITUTE_DOMAIN=[NO]NAME=*fully-qualified-domain*

Optional.

By default, the `From:` and `Return-Path` fields display the sender's name and fully qualified domain. NONAME causes the sender's domain name to be omitted from the `Return-Path` field. If you specify a fully qualified domain name (/SUBSTITUTE_DOMAIN=NAME=*fully-qualified-domain*), that specified domain name is displayed as the sender's domain name.

For example, suppose you specify the fully qualified domain name `eagle` for the sender's return path (/SUBSTITUTE_DOMAIN=NAME=`eagle`). When user `magpie` on host `condor.hawk.eagle.org` sends mail to `daw` on another host, user `daw` sees the return path as `magpie@eagle` rather than `magpie@condor.hawk.eagle.org`.

This is what `daw` sees:

```
    #707        18-NOV-2002 14:02:02.71                  MAIL
From:   SMTP%"magpie@eagle"
To:     SMTP%"daw@crow.ravin.rook.org"
CC:
Subj:   Big sale today!
```

Note

For changes made with the /SUBSTITUTE_DOMAIN qualifier to take effect, you must stop and restart SMTP. For more information about stopping and starting SMTP, refer to the VSI TCP/IP Services for OpenVMS Management manual.

/ZONE[=*domain*]
/NOZONE[=*domain*]

Optional. Default: /NOZONE (no gateway searching).

Domain for your environment (probably a superset of your local domain).

Mail sent to another network must be sent to this gateway.

With no value, /ZONE defaults to one level higher than your local domain.

For example, if your local domain is `a.b.com`, the default value of /ZONE is `b.com` because TCP/IP Services has been started; this assumes that the domain is known.

Mail for delivery outside of your zone is sent to its destination by the alternate gateway (see the /GATEWAY qualifier).

Examples

1. `TCPIP> SET CONFIGURATION SMTP /INTERVAL=(INIT="0 00:10:00.00")`

   The system waits 10 minutes before making its first attempt to deliver the message.

2. `TCPIP> SET CONFIGURATION SMTP /INTERVAL=(RETRY="0 00:20:00.00")`

   Specifies the wait time between retries.

3. `TCPIP> SET CONFIGURATION SMTP /INTERVAL=(MAX="3 00:20:00.00")`

   Specifies the maximum amount of time to retry before an error message is issued.

4. `TCPIP> SET CONFIGURATION SMTP /GATEWAY=(ALTERNATE:route_gateway)`

   Specifies the alternate host or domain to which delivery is attempted if mail cannot be delivered to the primary destination.

5. `TCPIP> SET CONFIGURATION SMTP /GATEWAY=(GENERAL:uucp_gateway)`

   Specifies a general-purpose gateway to handle non-SMTP mail.

6. `TCPIP> SET CONFIGURATION SMTP /ZONE=rsch.opt.com`

   Specifies that `rsch` is a domain that can be used to divert messages to nodes outside the local domain.

SET CONFIGURATION SNMP

SET CONFIGURATION SNMP — Configures SNMP on an individual host. SET CONFIGURATION NOSNMP does not require any qualifiers. After making changes to the SNMP

configuration, shut down and restart the master agent and any subagents. Issue the following commands: `$ @SYS$STARTUP:TCPIP$SNMP_SHUTDOWN`, and `$ @SYS$STARTUP:TCPIP $SNMP_STARTUP`. *Related command:* SHOW CONFIGURATION SNMP.

Syntax

```
SET CONFIGURATION [NO]SNMP [ /[NO]ADDRESS=host ]
                           [ /[NO]COMMUNITY="name" ]
                           [ /[NO]CONFIRM ]
                           [ /CONTACT=name ]
                           [ /FLAGS=options]
                           [ /LOCATION=options ]
                           [ /TYPE=options ]
```

Restrictions

Requires SYSPRV or BYPASS privilege.

If you add a new community and do not specify the /TYPE qualifier, the value of /TYPE defaults to read only.

If you add a new community and do not specify the /ADDRESS qualifier, the default address is 0.0.0.0.

Qualifiers

/ADDRESS=(*IP_address*)
/NOADDRESS=(*IP_address*)

Optional. Default: 0.0.0.0

Specifies hosts that belong to a particular community. You can specify multiple addresses.

This qualifier is meaningful only if you include the /COMMUNITY qualifier. A remote host cannot access information from this host unless its address appears in one or more communities of type READ or WRITE. For communities of type TRAP, the addresses specify the hosts that receive trap messages. For more information, see the /TYPE qualifier.

If you add a new community and do not specify this qualifier, the new entry's address is 0.0.0.0.

If you use the /ADDRESS qualifier with a community that already exists, these addresses are added to the existing address list.

/NOADDRESS deletes addresses from an existing list. If the deleted address is the only address listed for the community name, this qualifier also deletes the community.

/COMMUNITY=*"name"*
/NOCOMMUNITY=*"name"*

Optional. Default: To enable the standard "public" community, you can run the TCPIP$CONFIG procedure.

Used with the /ADDRESS qualifier. Name of the community that the SNMP agent recognizes. Optionally, specify a type of access and a list of host addresses. Enclose the name in quotation marks to preserve lowercase characters. See the /TYPE and /ADDRESS qualifiers for more information.

[NO]COMMUNITY=*"name"* removes a community name.

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM with if you use a wildcard; otherwise, /NOCONFIRM

When you delete communities (with the /NOCOMMUNITY qualifier), first asks for your confirmation.

If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

/CONTACT=*name*

Optional. Default: None.

Name of the system administrator (or other contact person) of the host on which the SNMP agent runs. The *name* field has a maximum length of 235.

/FLAGS=*options*

Optional.

The options include:

- SETS

  Lets the master agent process SET commands from SNMP clients.

- AUTHEN_TRAPS

  Lets the master agent send trap messages in response to unauthorized community strings from SNMP clients.

/LOCATION=*options*

Optional. Default: None.

Location of the system on which the SNMP agent runs. Maximum total length is 215 characters.

The options include:

- [NO]FIRST=*text*

  Specifies the first part of the location. Maximum length of *text* is 200 characters.

- [NO]SECOND=*text*

  Specifies the last part of the location. Maximum length of *text* is 200 characters.

If you specify two options, they are appended when sent to a client in response to an SNMP request for `syslocation`. For example, if FIRST is **abc** and SECOND is **def**, the value of the location is **abcdef** with no spaces. The total number of characters must not exceed 215.

/TYPE= {[NO]READ | [NO]TRAP | [NO]WRITE}

Optional. Default: READ.

Sets the type of access (to your local MIB data) to allow for a specified community.

- Type READ allows the master agent to accept GET, GETNEXT, and GETBULK commands from clients (management stations).
- Type TRAP allows the local master agent to issue traps to members of a specified community. Members of a trap community receive SNMP Trap-PDUs for significant events, including `coldStart` traps when the agent is initialized, and `authenticationFailure` traps when the agent receives an SNMP request that specifies an unauthorized community string.
- Type WRITE allows the master agent to accept SET commands from clients (management stations).

READ access is present by default when specifying TRAP or WRITE. Also, you can remove the read access without affecting the way the agent responds to a read request. For example:

```
$ SET CONFIGURATION SNMP /COMMUNITY="name" /TYPE=NOREAD
```

Examples

1. ```
   TCPIP> SET CONFIGURATION SNMP /COMMUNITY="public" -
   _TCPIP> /CONTACT="Sam Spade" -
   _TCPIP> /LOCATION=(FIRST="Falcon Building",SECOND="Los Angeles,
   California")
   ```

   Configures SNMP with the standard public community, taking the default type (READ) and address (0.0.0.0) for that community. Both contact and location are specified.

   The first and second parts of the location text are concatenated when displayed by an SNMP client. For example:

   ```
   Falcon BuildingLos Angeles, California
   ```

   If no update to the location text is done by an SNMP client, the display produced by SHOW CONFIGURATION SNMP is as follows:

   ```
   Location
   First:  Falcon Building
   Second: Los Angeles, California
   ```

   If the text is updated by an SNMP client (for example, to change "Falcon" to "Falconi"), the original formatting is not preserved and the display produced by SHOW CONFIGURATION SNMP is as follows:

   ```
   Location
   First:  Falconi BuildingLos Angeles, California
   ```

2. ```
   TCPIP> SET CONFIGURATION SNMP /COMMUNITY="rw" /TYPE=WRITE -
   _TCPIP> /ADDRESS=136.20.100.10 /FLAGS=SETS
   ```

   Configures a community with only read/write access to the host with the address specified. Other hosts still have read access through the public community. Also sets the SETS flag to enable the SNMP agents to process write requests from SNMP clients on host 136.20.100.10.

3. ```
   TCPIP> SET CONFIGURATION SNMP /NOCOMMUNITY="rw"
   ```
   ```

Removes the `rw` (read/write) community (set in example 2.)

4. `TCPIP> SET CONFIGURATION SNMP /COMMUNITY="trapit" /TYPE=TRAP -`
   `_TCPIP> /ADDRESS=136.20.0.10`

   Configures SNMP so that agents can send trap messages to the well-known UDP port 162 on the host identified with the address 136.20.0.10.

5. `TCPIP> SET CONFIGURATION SNMP /FLAGS=AUTHEN_TRAPS -`
   `_TCPIP> /COMMUNITY="trapit2" /TYPE=TRAP -`
   `_TCPIP> /ADDRESS=(136.20.0.12,136.20.0.15)`

   Configures SNMP with the AUTHEN_TRAPS flag so that the master agent sends trap messages when it detects a client request containing an invalid community name. Also configures an additional trap community. Trap messages, including authentication traps, go to all three addresses specified in the trap communities configured in this example and in example 4.

6. `TCPIP> SET CONFIGURATION SNMP  /COMMUNITY="rw2" /TYPE=WRITE -`
   `_TCPIP> /ADDRESS=(136.20.0.15,136.20.0.100)`

   Configures community `rw2`, which gives read/write access to two hosts. Note that one address can appear for more than one community, although a given address cannot be specified more than once for a single community.

SET CONFIGURATION START ROUTING

SET CONFIGURATION START ROUTING — Enters information into the configuration database to start dynamic routing when TCP/IP Services starts. **Related commands:** SHOW CONFIGURATION START ROUTING, START ROUTING

Syntax

```
SET CONFIGURATION START [NO]ROUTING [ /GATED ]
                                    [ /LOG ]
                                    [ /SUPPLY[=DEFAULT] ]
```

Qualifiers

/GATED

Optional.

Enables the gateway routing daemon (GATED).

If you enable dynamic GATED routing, you will be able to configure this host to use any combination of the following routing protocols to exchange dynamic routing information with other hosts on the network:

- RIP (Routing Information Protocol), Versions 1 and 2
- RDISC (Router Discovery Protocol)
- OSPF (Open Shortest Path First)

- EGP (Exterior Gateway Protocol)
- BGP (Border Gateway Protocol), BGP-4
- Static routes

/LOG

Optional. Default: No logging.

Applies to ROUTED. Do not use with /GATED.

Logs routing activity to SYS$SYSDEVICE:[TCPIP$ROUTED]TCPIP$ROUTED.LOG.

/SUPPLY[=DEFAULT]

Optional. Applies only to ROUTED. Do not use with /GATED.

Broadcasts routing information to other hosts in 30-second intervals.

If you specify /SUPPLY=DEFAULT, the local host supplies the default network route.

Examples

```
TCPIP> SET CONFIGURATION START ROUTING /SUPPLY
```

Starts ROUTED dynamic routing when TCP/IP Services is started. The local host both broadcasts and receives network routing information.

SET GATED

SET GATED — Configures the Gateway Routing Daemon (GATED). GATED obtains information from several routing protocols and selects the best routes based on that information. These protocols are configured in the file TCPIP$GATED.CONF. *Related commands:* START ROUTING /GATED, STOP ROUTING /GATED

Syntax

```
SET GATED [ /CHECK_INTERFACES ]
          [ /FILE=file ]
          [ /SAVE_STATE ]
          [ /TOGGLE_TRACE ]
```

Qualifiers

/CHECK_INTERFACES

Optional.

Instructs GATED to scan the kernel interface list for changes.

/FILE=*file*

Optional.

Specifies the name of the GATED configuration file. Use with the /SAVE_STATE qualifier.

/SAVE_STATE

Optional.

Causes GATED to save the current state of all tasks, timers, protocols, and tables to the file SYS $SYSDEVICE:[TCPIP$GATED]TCPIP$GATED.DMP (default).

Use the /FILE qualifier to specify a file name other than the default.

/TOGGLE_TRACE

Optional.

Use to close the trace file. A subsequent set GATED /TOGGLE_TRACE command reopens the trace file. This allows the file to be copied regularly. Valid only when a trace file is specified in the GATED configuration file.

Examples

1. `TCPIP> SET GATED /SAVE_STATE`

   This example causes GATED to save its current state to the file SYS$SYSDEVICE:[TCPIP $GATED]TCPIP$GATED.DMP.

2. `TCPIP> SET GATED /SAVE_STATE /FILE=STATE.DMP`

   This example causes GATED to save its current state to the file named STATE.DMP.

SET HOST

SET HOST — Defines or deletes an entry in the hosts database. Equivalent to maintaining the `/etc/hosts` file on UNIX hosts. *Related command:* SHOW HOST, CONVERT/VMS HOST.

Syntax

```
SET [NO]HOST host
            /ADDRESS=IP_address
            [ /[NO]ALIAS=alias ]
            [ /[NO]CONFIRM ]
```

Syntax

Requires read, write, and delete access to the hosts database.

Parameters

host

Required.

Name of a host that is a source or destination of Internet communications.

Note

If you define a mixed-case name, also define an alias in either all uppercase or all lowercase characters.

You cannot delete a host by specifying its alias.

Qualifiers

/ADDRESS=*IP_address*

Required SET HOST.

Host's IP address.

/ALIAS=*alias*
/NOALIAS=*alias*

Optional.

Add or remove an alternate name for a host.

Do not use with SET NOHOST.

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM if you use a wildcard.

Used with the SET NOHOST command, prompts you to confirm the delete request. For example:

```
TCPIP>  SET NOHOST MOA /ADDRESS=11.33.33.8 /CONFIRM

     LOCAL database

Host address    Host name

11.33.33.8 MOA
Remove? [N]:
```

If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

Examples

1. ```
   TCPIP> SET HOST MOA /ADDRESS=11.33.33.8 -
   _TCPIP> /ALIAS=("moa","bigbrd","nofly")
   ```

   Sets the IP address of host MOA to 11.33.33.8 and establishes `moa, bigbrd,` and `nofly` as aliases for host MOA.

2. ```
   TCPIP> SET HOST MOA /ALIAS="MOA_2"
   ```

   Establishes `MOA_2` as an alias for host MOA.

3. ```
   TCPIP> SET HOST MOA /ADDRESS = 128.33.33.9
   ```
   ```

Establishes a second IP address for host MOA.

4. `TCPIP> SET HOST MOA /ADDRESS = 128.33.33.9 /ALIAS="MOA_3"`

   Establishes `MOA_3` as an alias for host MOA's second IP address 128.33.33.9.

5. `TCPIP> SET HOST MOA /NOALIAS="MOA_2"`

   Deletes `MOA_2` as an alias for host MOA.

6. `TCPIP> SET NOHOST MOA /NOCONFIRM`

   Deletes MOA and all of its associated aliases.

SET INTERFACE

SET INTERFACE — Defines one of the following: an Internet interface, a serial line IP (SLIP) or point-to-point (PPP) connection, or a pseudointerface (a data structure that extends subnet routing). Before you issue SET INTERFACE, do the following to identify the name of an interface: first, issue the LIST COMMUNICATION_CONTROLLER command to find your system's controller. Then use the first character of the associated interface name. Refer to the VSI TCP/IP Services for OpenVMS Management manual for more information about specifying an interface name. SET NOINTERFACE deletes a record. No qualifiers are required. **Related commands:** SHOW INTERFACE, SET CONFIGURATION INTERFACE.

Syntax

```
SET [NO]INTERFACE interface
                  [ /[NO]ARP ]
                  [ /[NO]AUTO_START ]
                  [ /BROADCAST_MASK=IP_address ]
                  [ /C_BROADCAST_MASK=IP_address ]
                  [ /C_NETWORK=IP_address ]
                  [ /[NO]CLUSTER=host ]
                  [ /COMPRESS=options ]
                  [ /DESTINATION=IP_address ]
                  [ /DHCP ]
                  [ /FLOWCONTROL ]
                  [ /HOST=host ]
                  [ /[NO]LOOPBACK ]
                  [ /NETWORK_MASK=IP_address ]
                  [ /PRIMARY ]
                  [ /SERIAL_DEVICE=device ]
```

Restrictions

Before you issue the SET INTERFACE command, disable the interface by using the SET NOINTERFACE command.

This command requires:

- OPER privilege
- Read access to the hosts database

- Read access to the networks database
- Read, write, and delete access to the routes database

Every host on the same network must have the same network mask.

Parameters

interface

Required.

Specifies an interface name for the communication controller, such as RF1, RT1, ZE0, XE0, SL0, SL1, SL2, PP0, PP1, PP2.

Qualifiers

/ARP
/NOARP

Optional. Default: /ARP.

Enables IP address-to-hardware address (Ethernet or FDDI) mapping.

/ARP is valid when you create an interface but not when you modify an existing interface.

/AUTO_START
/NOAUTO_START

Optional. Default: /AUTO_START.

Valid for a SLIP or PPP interface. Automatically creates the interface when TCP/IP Services starts.

/BROADCAST_MASK=*IP_address*

Optional.

Sets the Internet interface to receive all broadcast messages.

TCP/IP Services calculates the default by:

- Using the network number from the network mask
- Setting all bits in the host number field to 1

/C_BROADCAST_MASK=*IP_address*

Optional.

Sets the cluster broadcast mask to receive all broadcast messages.

The software calculates the default by:

- Using the network number from the network mask

- Setting all bits in the host number field to 1

/C_NETWORK=*IP_address*

Optional.

Sets the network mask of the cluster network. This mask is specific to the cluster host network.

The software calculates the default by:

- Setting the bits representing the network fields to 1
- Setting the bits representing the host field to 0

/CLUSTER=*host*
/NOCLUSTER

Optional. Default: None.

Specifies the cluster host name (alias host identifier).

Before using this qualifier, first define the same name in the hosts database.

/CLUSTER=*host* associates the alias host identifier with each interface in a cluster.

/NOCLUSTER disables Internet cluster processing on the specified interface.

Caution

When you specify /NOCLUSTER, active communication is aborted for applications bound to the cluster alias name.

/COMPRESS= {ON | OFF | AUTOMATIC}

Optional. Default: For PPP interface: /COMPRESS=ON; for SLIP interface: /COMPRESS=OFF

Valid for SLIP and PPP interfaces.

Enables or disables TCP header compression.

/COMPRESS=AUTOMATIC turns off compression unless the remote end begins to use it.

/DESTINATION=*IP_address*

Optional.

Valid for a PPP interface.

Used on the local host to provide dialup access to remote systems. The value specified is the IP address to be given to remote clients for use while PPP connection is active. If you use /DESTINATION, you must provide the address of the local host with the /HOST qualifier.

/DHCP

Optional.

Designates the interface as a DHCP-controlled interface in the volatile database. This qualifier affects only the currently running interface.

Before you enter the SET INTERFACE command, be sure to enter the SET NOINTERFACE command first and specify the interface you are changing.

/FLOWCONTROL

Optional. Default: No flow control.

Valid for a SLIP interface. Enables the handling of XON and XOFF characters to properly interoperate with modems that are configured to interpret these characters locally.

Specify /FLOWCONTROL only if the host at the other end of the line is another host running TCP/IP Services.

/HOST=*host*

Required when first setting the interface; optional if the interface is already defined. Always required for a SLIP interface. Optional for a PPP interface unless you are setting up the local host as a dialup provider by using the /DESTINATION qualifier.

Local host name or IP address using the interface. If this information is not specified for a PPP interface, PPP obtains the correct address from the remote host.

If your host is multihomed, specify an address.

/LOOPBACK
/NOLOOPBACK

Optional. Default: /NOLOOPBACK.

Sets loopback mode.

/NETWORK_MASK=*IP_address*

Required if you use subnets.

The part of the host field of the IP address identified as the subnet.

The software calculates the default by:

- Setting the bits representing the network fields to 1
- Setting the bits representing the host field to 0

An IP address consists of a network number and a host number. You can also divide the host field into a site-specific subnetwork and host field.

/PRIMARY

Optional.

For DHCP-controlled interfaces, designates the interface from which system-wide configuration options (such as the IP address of the BIND server) are used.

/SERIAL_DEVICE=*device*

Required for SLIP and PPP interfaces; otherwise not used.

Identifies the OpenVMS terminal device used as a serial device. Specify an arbitrary terminal device name. (Unlike Ethernet, FDDI, and Token Ring interface names, a serial interface name is not related to the OpenVMS device name.)

Examples

1. ```
   TCPIP> SET INTERFACE SL5 /HOST=LARK /NETWORK_MASK=255.255.255.0 -
   _TCPIP> /SERIAL_DEVICE=TTA3: /COMPRESS=ON /FLOWCONTROL
   ```

   Configures SLIP interface SL5, using the local IP address assigned to host LARK, with a subnet mask of 255.255.255.0.

   The interface uses the terminal device TTA3:.

   The /COMPRESS qualifier enables TCP header compression (CSLIP).

   The /FLOWCONTROL qualifier enables special handling of XON and XOFF characters, to ensure proper interoperation with modems that are configured to interpret these characters locally.

2. ```
   TCPIP> SET INTERFACE FF0 /HOST=KESTREL /NETWORK_MASK=255.255.0.0 -
   _TCPIP> /BROADCAST_MASK=128.30.255.255 /ARP
   ```

   For new interface `FF0` on host KESTREL, sets the network mask to 255.255.0.0, sets the broadcast mask to 128.30.0.0, enables ARP, and activates the interface.

3. ```
   TCPIP> SET INTERFACE PP0 /SERIAL_DEVICE=TTA0: -
   _TCPIP> /HOST=10.10.1.2 /DESTINATION=10.10.1.3
   ```

   Configures the interface as a PPP serial device. This command specifies that the local host is a dialup provider. The address specified with the /DESTINATION qualifier (10.10.1.3) is the address assigned to the client system requesting an address.

   Refer to VSI TCP/IP Services for OpenVMS Management manual for more information on setting up interfaces for SLIP and PPP communication.

4. ```
   TCPIP> SET NOINTERFACE DE2
   TCPIP> SET INTERFACE DE2 /LOOPBACK
   .
   .
   .
   TCPIP> SET INTERFACE DE2
   ```

   Deletes interface DE2, sets loopback mode for testing this interface, and, after testing, reactivates it.

SET MX_RECORD

SET MX_RECORD — For routing mail, adds routing information to the local Mail Exchanger (MX) database. Each entry contains a list of hosts that can accept mail for the specified destination. The list is in order of routing preference. The local MX information is stored in the routes database. The MX entry is one of the record types in the BIND database. In addition, a BIND server might provide an

MX record. SMTP is designed to determine where the sending system should try to relay mail. It is also designed to identify where the sending system actually tries to relay mail. To find a destination address, the MX routing lookup process follows this sequence: local MX database, remote MX database, BIND database, Local hosts database. *Related command:* SHOW MX_RECORD

Formats

```
SET MX_RECORD destination
              /GATEWAY=host
              /PREFERENCE=n

SET NOMX_RECORD destination
                [ /GATEWAY=host ]
```

Restrictions

Requires read and write access to the routes database.

Parameters

destination

Required.

Host name or domain name to which mail will be sent.

Qualifiers

/GATEWAY=*host*

Required with SET MX_RECORD. Optional with SET NOMX_RECORD.

Gateway through which mail will be relayed. Must have an address in either the local hosts database or the BIND database.

A destination can have multiple gateways, each with an associated preference value.

/PREFERENCE=*n*

Required.

Arbitrary number for ranking multiple gateways for a destination. The smaller the number, the higher the preference in sending mail by way of that gateway.

Do not use with SET NOMX_RECORD.

Examples

1. `TCPIP> SET MX_RECORD JUNO /GATEWAY=MARS /PREFERENCE=100`

   Assigns MARS as the gateway for host JUNO with a preference of 100.

2. `TCPIP> SET MX_RECORD JUNO /GATEWAY=VENUS /PREFERENCE=200`

   Assigns VENUS as the gateway for host JUNO with a preference of 200.

SET NAME_SERVICE

SET NAME_SERVICE — Configures the BIND resolver and designates a BIND server. By default, all settings are process specific. To make modifications that are systemwide, use the /SYSTEM qualifier. The local host's domain is used as the default domain unless you also specify /DOMAIN. To reload the BIND server databases, use the /INITIALIZE qualifier. /NOSERVER does not require any options. **Related commands:** SHOW NAME_SERVICE, SET CONFIGURATION NAME_SERVICE

Syntax

```
SET NAME_SERVICE [ /CLUSTER=dev:[directory] ]
                                 [ /DISABLE ]
                                 [ /[NO]DOMAIN=domain ]
                                 [ /ENABLE ]
                                 [ /INITIALIZE ]
                                 [ /[NO]PATH=domain ]
                                 [ /RETRY=number of retries ]
                                 [ /[NO]SERVER=host ]
                                 [ /SYSTEM ]
                                 [ /TIMEOUT=seconds ]
                                 [ /TRANSPORT=protocol ] )
```

Restrictions

The /SYSTEM qualifier requires the SYSPRV or BYPASS and SYSNAM privileges. The /INITIALIZE qualifier requires the BYPASS, READALL, or SYSPRV privileges.

Qualifiers

/CLUSTER=*dev:*[*directory*]

Optional.

Specifies the common BIND directory. By default, the clusterwide common directory is *common-disk:*[TCPIP$BIND_COMMON]. This qualifier reloads the BIND database on every master BIND server running in the OpenVMS Cluster.

This qualifier must be used with the /INITIALIZE qualifier.

/DISABLE

Optional.

Disables the BIND resolver. All name and address lookups are now directed to the local hosts database. Use with /SYSTEM.

/DOMAIN=*domain*
/NODOMAIN=*domain*

Optional.

Defines the default domain. The default domain is appended to host name references made from the local process.

/NODOMAIN deletes the process-specific definition of the domain. Do not use with /SYSTEM.

/ENABLE

Optional. Default: Not enabled.

Enables the BIND resolver. Must be used with /SYSTEM.

/INITIALIZE

Optional. Default: No reloading.

Reloads all BIND server databases and the BIND configuration file. Either TCPIP $ETC:RNDC.CONF or TCPIP$ETC:RNDC.KEY must be set up to allow for secure communication between the user and the BIND server. For more information, refer to the VSI TCP/IP Services for OpenVMS Management manual.

/PATH=*domain*
/NOPATH=*domain*

Optional. Requires the SYSNAM privilege.

Defines the BIND resolver's domain search list.

To specify multiple domains, list them by search preference. The resolver starts with the first domain on the list, and continues to search each domain until the name is found or until all domains have been exhausted and the lookup fails.

/NOPATH removes domains from the list.

If you define a domain list and then issue another SET NAME_SERVICE /PATH command, TCP/IP Services appends the new domains to the end of the list.

If no search list is defined, the default behavior of the BIND resolver is to do a lookup on the name as you typed it. If that lookup fails, then the default domain is appended and the lookup is attempted again.

/RETRY=*number of retries*

Optional. Default: four retries.

Specifies the number of times that the BIND resolver attempts to contact a BIND server if previous tries fail.

/SERVER=*host*
/NOSERVER=*host*

Optional.

Specifies the host name or address of the BIND server or servers that the resolver will query.

To specify multiple hosts, list them by request preference. The resolver sends the first lookup request to the first host on the list.

/NOSERVER removes hosts from the list.

If you define a server list and then issue another SET NAME_SERVICE /SERVER command, TCP/IP Services appends the new servers to the end of the list.

Do not use /NOSERVER with /SYSTEM.

/SYSTEM

Optional. Default: Changes are process specific.

Makes your settings systemwide.

/TIMEOUT=*seconds*

Optional. Default: 4 seconds.

Specifies the timeout interval for the BIND resolver's requests to a server.

When the BIND resolver is used by the auxiliary server, the following conditions are true:

- The number of retries is one.
- The timeout interval is 1 second.
- The timeout interval increases by the power of two for each retry, as shown in the following table:

Current settings:	**If Retries=4 and Timeout=4**
Retry	**Timeout interval is:**
First	4 seconds
Second	8 seconds
Third	16 seconds
Last	32 seconds
	Total = 1 minute for one server
	If a second BIND server exists and both servers time out, total = 2 minutes

/TRANSPORT=*protocol*

Optional. Default: UDP.

Protocol used for communicating with a BIND server. Specify one of the following:

- UDP
- TCP

Examples

1. `TCPIP> SET NAME_SERVICE /SERVER=(PARROT,SORA,JACANA) /SYSTEM /ENABLE`

   Defines hosts PARROT, SORA, and JACANA as systemwide BIND servers. Also enables the BIND resolver.

2. `TCPIP> SET NAME_SERVICE /SERVER=OSPREY`

   For your process, defines host OSPREY as the BIND server. The servers that are defined systemwide will not be queried.

SET NETWORK

SET NETWORK — Defines or deletes an entry in the networks database. Equivalent to maintaining the `/etc/networks` file on UNIX hosts. *Related commands:* SHOW NETWORK, CONVERT/VMS NETWORK.

Examples

```
SET [NO]NETWORK network
                [ /ADDRESS=IP_address ]
                [ /[NO]ALIAS=alias ]
                [ /[NO]CONFIRM ]
```

Restrictions

Requires read, write, and delete access to the networks database.

Parameters

network

Required.

Name of the network.

You cannot delete a network by specifying an alias name.

Qualifiers

/ADDRESS=*IP_address*

Required for a new entry.

IP address of the network.

/ALIAS=*alias*
/NOALIAS=*alias*

Optional.

Alternate name for the network.

- Do not use with SET NONETWORK.
- /NOALIAS=`alias` deletes an alias.
- /NOALIAS=* deletes all aliases.

/CONFIRM
NOCONFIRM

Optional. Default: /CONFIRM if you use a wildcard.

When used with SET NONETWORK, prompts you to confirm the delete request.

If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

Examples

1. `TCPIP> SET NETWORK MYNA /ADDRESS=128.30.30.10 /ALIAS=MYNA_1`

   Creates an entry for network MYNA at IP address 128.30.30.10, and its alias MYNA_1, to the networks database.

2. `TCPIP> SET NETWORK MYNA /ALIAS=MYNA_2`

   Adds a second alias for network MYNA.

3. `TCPIP> SET NETWORK MYNA /NOALIAS=MYNA_2`

   Deletes the alias MYNA_2 from the network MYNA entry in the networks database.

4. `TCPIP> SET NETWORK "jungle" /ALIAS=("parrot","canary","motmot")`

   For network `jungle`, creates the aliases `parrot`, `canary`, and `motmot`.

SET NFS_SERVER

SET NFS_SERVER — Modifies dynamic configuration parameters that control NFS server operation. To modify a dynamic configuration parameter, use one or more of the command qualifiers. Static configuration parameters can take effect only by restarting the NFS server. For more information, refer to the VSI TCP/IP Services for OpenVMS Management manual. Edit the TCPIP $ETC:SYSCONFIGTAB.DAT file to modify static configuration parameters, as explained in the VSI TCP/IP Services for OpenVMS Tuning and Troubleshooting manual. *Related commands:* SHOW NFS_SERVER, ZERO NFS_SERVER

Syntax

```
SET NFS_SERVER [ /DISABLE=NOPROXY ]
               [ /ENABLE=NOPROXY ]
               [ /GID_DEFAULT=n ]
               [ /INACTIVITY_TIMER=n ]
               [ /UID_DEFAULT=n ]
```

Restrictions

Requires the following privileges:

- SYSNAM
- WORLD

- SYSPRV or BYPASS

Qualifiers

/DISABLE=NOPROXY

Optional. Default: If the SYSCONFIG attribute `noproxy_enabled` is set to 0, then proxies are required for server access. Otherwise, the values of `noproxy_uid` (for the user ID) and `noproxy_gid` (for the group ID) become the default for users who have no proxies defined.

Disables the use of default UIDs and GIDs.

/ENABLE=NOPROXY

Optional.

Enables the use of default UIDs and GIDs.

/GID_DEFAULT=*n*

Optional. Default: –2.

Default GID associated with files owned by a UIC that has no corresponding proxy mapping.

/INACTIVITY_TIMER=*n*

Optional. Default: 120 seconds.

Maximum time period (in seconds) that unaccessed NFS files remain open.

/UID_DEFAULT=*n*

Optional. Default: –2.

Default UID associated with files owned by a UIC that has no corresponding proxy mapping.

Example

```
TCPIP> SET NFS_SERVER /INACTIVITY_TIMER=180
```

Sets the time period that unaccessed NFS files remain open to 180 seconds.

SET PROTOCOL

SET PROTOCOL — Sets parameters for ICMP, IP, TCP, and UDP. **Related commands:** SET CONFIGURATION PROTOCOL, SHOW PROTOCOL.

Syntax

```
SET PROTOCOL ICMP [ /[NO]REDIRECT ]

SET PROTOCOL IP [ /[NO]FORWARD ]
                [ /REASSEMBLY_TIMER=seconds ]
```

```
SET PROTOCOL TCP [ /[NO]MTU_SEGMENT_SIZE ]
                 [ /[NO]DELAY_ACK ]
                 [ /DROP_COUNT=n ]
                 [ /PROBE_TIMER=seconds ]
                 [ /QUOTA=[ SEND=bytes,RECEIVE=bytes ]
                 [ /[NO]WINDOW_SCALE ]

SET PROTOCOL UDP [ /[NO]BROADCAST ]
                 [ /QUOTA=options ]
```

Restrictions

Requires OPER privilege.

Parameters

{ICMP | IP | TCP | UDP}

Required.

Specifies the protocol software to configure.

Qualifiers for ICMP

/REDIRECT
/NOREDIRECT

Optional. Default: /NOREDIRECT.

Sends ICMP_REDIRECT messages.

Qualifiers for IP

/FORWARD
/NOFORWARD

Optional. Default: /NOFORWARD.

Forwards IP messages to other hosts.

/REASSEMBLY_TIMER=*n*

Optional. Default: 7 seconds. Valid range: 1 to 126.

Maximum time for trying to reassemble a received datagram.

Qualifiers for TCP

/MTU_SEGMENT_SIZE
/NOMTU_SEGMENT_SIZE

Optional. Default: /NOMTU_SEGMENT_SIZE.

If a connection is more than one hop away, sets the segment size. Specify one of the following:

/MTU_SEGMENT_SIZE	Sets the segment size as close as possible to the maximum transmission unit (MTU) size.
/NOMTU_SEGMENT_SIZE	Sets the segment size as close as possible to the standard 512 bytes.

/DELAY_ACK
/NODELAY_ACK

Optional. Default: /DELAY_ACK.

Enables or disables a delay before sending the following acknowledgments:

/DELAY_ACK	ACKs are generated with a delay.
/NODELAY_ACK	ACKs are generated without any delay.

/DROP_COUNT=*n*

Optional.

Number of idle probes that can go unsatisfied before the software declares a TCP connection dead and closes it.

/PROBE_TIMER=*n*

Optional. Default: 75 seconds.

Number of seconds between probes for idle TCP connections (when the SO_KEEPALIVE option is set). If the remote system fails to respond, the connection is removed. Also, when initiating a TCP connection request, indicates the maximum number of seconds that the software waits for a response from the remote system before the request times out.

/QUOTA=[SEND=*bytes*,RECEIVE=*bytes*]

Optional.

Queue size (in bytes) for messages.

The options for setting TCP message queue size are:

- RECEIVE:*n* — Receive queue size. Default: 4096 bytes.
- SEND:*n* — Send queue size. Default: 4096 bytes.

/WINDOW_SCALE
/NOWINDOW_SCALE

Optional.

Turns TCP window scaling on and off. Default is on.

Scaling allows windows larger than 64 KB to be represented in the normal 16-bit TCP window field. Large windows allow improved throughput. Turning this option off may help you to troubleshoot communication problems with another TCP/IP implementation.

Qualifiers for UDP

/BROADCAST
/[NO]BROADCAST

Optional. Default: /NOBROADCAST.

Enables privilege checking for broadcast messages.

- /BROADCAST — Nonprivileged users can send broadcast messages.
- /NOBROADCAST — To send broadcast messages, users need a privileged UIC or the SYSPRV, BYPASS, or OPER privilege.

ONC RPC applications use broadcast messages and need privilege checking disabled.

/QUOTA=*options*

Optional.

Specifies the queue size (in bytes) for messages.

The options for setting UDP message queue size are:

- RECEIVE:*n* — Receive queue size. Default: 9000 bytes.
- SEND:*n* — Send queue size. Default: 9000 bytes.

Examples

1. `TCPIP> SET PROTOCOL IP /FORWARD`

   Sets IP to forward messages to other hosts, including other Internet cluster nodes.

2. `TCPIP> SET PROTOCOL TCP /PROBE_TIMER=50`

   Sets the TCP probe timer parameter to 50 seconds.

SET ROUTE

SET ROUTE — Defines a routing path in either the permanent or volatile routes database. Routes in the permanent, on-disk routes database are static. Static routes can be supplemented by routes that the dynamic routing server receives. Defaults are as follows: If the network is not active, the command affects the permanent database. If the network is active, the command affects the volatile database. (To modify the permanent database, use the /PERMANENT qualifier.) Note the following restrictions: you can add routes, you cannot use SET NOROUTE to remove a route that is maintained by the routing daemon, to have full manual control over your routing table, first issue STOP ROUTING and then use SET NOROUTE, SET NOROUTE does not require any qualifiers. **Related commands:** SHOW ROUTE, STOP ROUTING. VSI strongly recommends that you do not specify alias names with the *destination* parameter or with the /GATEWAY=*host* qualifier.

Syntax

```
SET [NO]ROUTE destination
```

```
[ /[NO]CONFIRM ]
[ /DEFAULT_ROUTE ]
[ /GATEWAY=host ]
[ /MASK=mask_length ]
[ /NETWORK ]
[ /PERMANENT ]
```

Restrictions

Requires OPER privilege if:

- The TCP/IP Services product is running.
- The routes database requires read and write access.

Parameters

destination

Required unless you specify the /DEFAULT_ROUTE qualifier.

Host or network through which to route packets. Specify one of the following:

- A host, as it is defined in the hosts database
- A network, as it is defined in the networks database

Not valid with /DEFAULT_ROUTE.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM if you use a wildcard.

Prompts you to confirm the change.

If you specify the /NOCONFIRM qualifier, the operation is performed without asking you to confirm the request.

/DEFAULT_ROUTE

Optional. Default: 0.0.0.0.

Defines a second route to use if the first try to route a packet fails.

You must also specify a value for /GATEWAY.

Not valid with the *destination* parameter.

/GATEWAY=*host*

Optional. Default: None.

Gateway for the route. Necessary to send packets to a host on another network.

/MASK=*mask_length*

Optional. Default: None.

Defines the Classless Inter-Domain Routing (CIDR) mask length. (The mask length is sometimes referred to as the prefix length.)

CIDR is a method of associating blocks of Internet addresses through the use of a mask. With CIDR, a route is a combination of the IP address and a value describing the length of the leftmost contiguous set of bits.

/NETWORK

Optional. Defaults:

- Destination is classified based on its Internet network class (A, B, or C).
- If the address is clearly a network number, SET ROUTE interprets the number correctly.

Defines the route as a network route.

Use this qualifier if the network number could be misinterpreted as an IP host address; for example, if a network mask is nonstandard, or if the IP address is abbreviated.

This qualifier is required if you are creating a network route that specifies a CIDR mask (for example, /MASK=*mask_length*).

/PERMANENT

Optional. Defaults:

If the network is not active, the permanent routes database is changed. If the network is active, the volatile routes database is changed.

Changes only the permanent routes database.

Examples

1. `TCPIP> SET ROUTE DODO /GATEWAY=RHEA`

   Defines a route for local host DODO to send packets.

2. `TCPIP> SET ROUTE 101.81 /GATEWAY=100.42`

   Defines a gateway for routing packets for the host with IP address 101.81.

3. `TCPIP> SET ROUTE 100.45.0 /GATEWAY=REMOTE /NETWORK`

   Sets a route through the network whose IP address is 100.45.0.

4. `TCPIP> SET ROUTE /DEFAULT /GATEWAY=DEFGATE /PERMANENT`

   Sets a default route with host DEFGATE as the default gateway. Adds the definition to the permanent routes database.

SET SERVICE

SET SERVICE — Defines a new entry or modifies an existing entry in the services database. The /FILE, /PORT, /PROCESS_NAME, and /USER_NAME qualifiers are required when defining a new entry and optional when modifying an existing one. For changes to service parameters to take effect, you must disable and reenable the service. **Related command**: SHOW SERVICE

Syntax

```
SET [NO]SERVICE service
                { /FILE=startup_file
                /PORT=n
                /PROCESS_NAME=process
                /USER_NAME=vms_user_account }
                [ /ACCEPT=options ]
                [ /ADDRESS=IP_address ]
                [ /FLAGS=options ]
                [ /LIMIT=n ]
                [ /LOG_OPTIONS=options ]
                [ /PROTOCOL=protocol=options ]
                [ /REJECT=options ]
                [ /RPC=values ]
                [ /SEPARATOR=option ]
                [ /SOCKET_OPTIONS=options ]
```

Restrictions

You cannot modify the following fields in an existing entry:

- *service*
- /ADDRESS
- /PORT
- /PROCESS_NAME
- /PROTOCOL (except for the optional settings)

To make changes to these fields, use SET NOSERVICE to delete the entry and then re-create the entry.

Note

There is no RCP service. RCP uses the RSH server process.

VSI strongly suggests that, for the services provided by TCP/IP Services, you do not use this command to reset the following:

- The required qualifiers
- The /FLAGS qualifier, except for the APPLICATION_PROXY and CASE_INSENSITIVE options

Using SET NOSERVICE without either a specified service or specified qualifiers deletes all entries for all services.

Requires write access to the directory with the services database.

Parameters

service

Required for SET SERVICE; optional for SET NOSERVICE.

Service you want to modify or enter into the services database.

Qualifiers

/ACCEPT {=[NO]HOSTS=(*hosts*) | =[NO]NETWORKS=(*networks*)}

Optional. Default: Offers the service to all hosts on all networks.

- /ACCEPT=HOST=(*host*)
  - Grants *host* or *hosts* access to the service.
  - Denies access to all other hosts.
- /ACCEPT=NOHOST=*host* removes access to the service for a host that previously gained access with /ACCEPT=HOST.

The following options are available:

Option	Meaning
HOSTS=*hosts*	Makes the service available to the specified hosts. Denies all other hosts access to the service. Maximum is 32. **Example:** `/ACCEPT=HOSTS=(host1_name,host2_name,host3_address)`
NOHOSTS=*hosts*	Removes the specified hosts from the accept list so they cannot gain access to the service. You can specify a wildcard character (*) in place of the *hosts* list to remove all hosts from the accept list. Maximum is 32. **Example:** `/ACCEPT=NOHOSTS=(host1_name,host2_name, host3_address)`
NETWORKS=*networks*	Makes the service available to the specified networks. Denies access to the service to all other networks. Maximum is 16.

Option	Meaning
	For each network, you can optionally specify the network mask. The default network mask equals network's class number. For example, for the network 11.200.0.0., the default mask is 255.0.0.0. **Example:** `/ACCEPT=NETWORKS=(net1_name,net2_address,net3:net3mask)`
NONETWORKS[=*networks*]	Removes the specified networks from the accept list so they cannot gain access to the service. You can specify a wildcard character (*) in place of the *networks* list to remove all networks from the accept list. Maximum is 16. For each network, you can optionally specify the network mask. The default net mask equals network's class number. For example, for network 11.200.0.0., the default mask is 255.0.0.0. `/ACCEPT=NONETWORKS=(net1_name,net2_address,net3:net3mask)`

/ADDRESS=*IP_address*

Optional. Default: 0.0.0.0 (all local interfaces receive incoming requests for the service).

If you have multiple Internet interfaces and, therefore, more than one IP address, /ADDRESS specifies the particular address on which incoming requests are received.

To define a service name more than once, use /ADDRESS with different values for each instance. A reason to duplicate a service name, for example, is that your local host has three interfaces and you want to make a service available on two of them. Each service/interface pair must be unique.

/FILE=*startup_file*

Required if defining a new service entry; optional if modifying an existing one.

Name of the service's startup command file.

/FLAGS= {[NO]APPLICATION_PROXY | [NO]MULTITHREAD | [NO]PROXY | [NO]CASE_INSENSITIVE}

Optional.

The flag options are:

- [NO]APPLICATION_PROXY. Default: NOAPPLICATION_PROXY.

  The service does its own proxy checking. This allows connections based on defined proxies.

  Applies to: remote shell (RSH) and line printer daemon (LPD).

Note

The ROOT account does not require a communication proxy in the proxy database. The setting of /FLAGS=APPLICATION_PROXY flag is not relevant.

- [NO]MULTITHREAD. Default: NOMULTITHREAD.

  While connecting a socket to a remote host and passing the socket to the requested server, the auxiliary server continues to listen for incoming requests.

- [NO]PROXY. Default: NOPROXY.

  User account information is from the proxy database.

- [NO]CASE_INSENSITIVE. Default: CASE_INSENSITIVE.

  Case sensitivity of the remote user name in the proxy database.

  Use with /PROXY.

/LIMIT=*n*

Optional.

Maximum number of copies of the requested service allowed to run on the system. If the maximum number is reached, any additional requests for the service are rejected.

/LOG_OPTIONS= [[NO]ACCEPT] [[NO]ACTIVATE] [[NO]ADDRESS] [[NO]ALL] [[NO]CONNECT] [[NO]DEACTIVATE] [[NO]ERROR] [[NO]EXIT_CLEANUP] [[NO]LOGIN] [[NO]LOGOUT] [[NO]MODIFY] [[NO]REJECT]

Sets the specified logging options for the service you are configuring.

The logging options have the following meanings:

Option	Meaning
[NO]ACCEPT	Message is logged when a request is accepted.
[NO]ACTIVATE	Message is logged when the service is activated.
[NO]ADDRESS	For auxiliary server messages and OpenVMS security events, the message displays the IP address as a host name. If host names are not relevant, VSI recommends that you specify [NO]ADDRESS.
[NO]ALL	Messages are logged for all events.
[NO]CONNECT	Message is logged when the auxiliary server issues a connect request back to the client. The services that usually make this request (on a second socket) are remote shell and remote execute.
[NO]DEACTIVATE	Message is logged when the service is being deactivated.

Option	Meaning
[NO]ERROR	Message is logged when an error is detected while processing a request to the service.
[NO]EXIT_CLEANUP	Message is logged when the service fails to complete startup (that is, the server did not assign the BG device, with logical name SYS$NET, or did not issue a C socket before exiting).
[NO]LOGIN	Message is logged when a connected terminal server accepts a remote login request.
[NO]LOGOUT	Message is logged when a connected terminal server terminates a connection.
[NO]MODIFY	Message is logged when the active service is being modified.
[NO]REJECT	Message is logged when a request is rejected.

/PORT=*n*

Required if defining a new service entry. Cannot be modified; use SET NOSERVICE to delete the entry and then re-create the entry with the modification you want to make.

Port number that the service will use. Specify a number from 1 to 65535.

/PROCESS_NAME=*process*

Required if defining a new service entry. Cannot be modified (use SET NOSERVICE to delete the entry and then re-create the entry with the modification you want to make).

Name of the service's process.

Specify a character string up to 15 characters long. The name is truncated to 15 characters if it exceeds that limit.

/PROTOCOL=*protocol* [=*options*]

Optional. Default: TCP.

Protocol, and its parameters, that the service will use. To set these parameters, use the following options:

Protocol	Option	Meaning
IP	TYPE_OF_SERVICE=*n*	Type of service, expressed as a value between 0 and 255.
	TIME_TO_LIVE=*n*	Maximum number of hops that packets can traverse before being dropped.
TCP (stream socket type)	[NO]DELAY	Delays sending packets, allowing multiple packets to be combined into a single larger packet before transmission. Default: DELAY.
	DROP_COUNT=*n*	TCP connection-request timeout interval for the service.

Protocol	Option	Meaning
		Maximum number of seconds to probe for idle TCP connections before such a connection times out and closes.
	PROBE_TIMER=*seconds*	Number of seconds between probes for idle connections.
UDP	None	Datagram socket type.

/REJECT {=[NO]HOSTS=(*hosts*) |=[NO]NETWORKS=(*networks*) |=[NO]MESSAGE="*text*"] }

Optional. Default: No rejections if /ACCEPT is set to its default (service all hosts).

- /REJECT=HOST=*host* denies *host* access to the service.
- /REJECT=NOHOST=*host* regrants *host* access to the service.

The following options are available.

Option	Meaning
HOSTS=*hosts*	Makes the service unavailable to the specified hosts. Maximum is 32. **Examples:** /REJECT=HOSTS=(*host1_name*,*host2_name*, *host3_address*) /REJECT=HOSTS=*
NOHOSTS=*hosts*	Removes the specified hosts from the reject list. You can use the wildcard character (*) in place of the *hosts* list to remove all hosts from the reject list. Maximum is 32. **Examples:** /REJECT=NOHOSTS=(*host1_name*,*host2_name*,*host3_address*) /REJECT=NOHOSTS=*
NETWORKS=*networks*	Makes the service unavailable to the hosts on the specified networks. Maximum is 16. For each network, you can optionally specify the network mask. The default net mask equals network's class number. For example, for network 11.200.0.0., the default mask is 255.0.0.0. **Example:** /REJECT=NETWORKS=(*net1_name*,*net2_address*, *net3*:*net3mask*)

Option	Meaning
NONETWORKS[=*networks*]	Removes the specified networks from the reject list. You can use the wildcard character (*) in place of the *networks* list to remove all networks from the reject list. Maximum is 16. For each network, you can optionally specify the network mask. The default net mask equals network's class number. For example, for network 11.200.0.0., the default mask is 255.0.0.0. **Example:** /REJECT=NONETWORKS=(*net1_name*,*net2_address*, *net3*:*net3mask*)
[NO]MESSAGE=*text* NOMESSAGE	Message sent to "reject-list" clients when TCP/IP Services rejects their request for the service. Optional. Specify a character string up to 63 characters. Enclose the string in quotation marks. Use this option only for a service whose clients require and support reject messages. Messages are sent with a carriage return/line feed at the end. For RLOGIN, RSH, and REXEC, this message is preceded by a byte with a value of 1 and is terminated with a byte with a value of 0. /REJECT=NOMESSAGE deletes the stored message text.

/RPC=(PROGRAM_NUMBER=*n*, VERSION_NUMBER=(LOW=*n*, HIGH=*n*))

Required for services that use the Portmapper; otherwise, not valid. Defaults:

Service	Program Number	Highest Version	Lowest Version
MOUNT	100005	1	1
NFS server	100003	2	2
PCNFS	150001	1	2
PORTMAPPER	100000	1	1

Information that identifies the service to the Portmapper. Use this qualifier for all applications that use RPCs.

/SEPARATOR=*option=character*

Optional. Default: 0 (null).

Character that separates the following fields in received packets:

- PORT=*character*
- USER_NAME=*character*
- PASSWORD=*character*
- COMMAND=*character*

/SOCKET_OPTIONS=(*options*)

Optional.

The following socket options are available.

Option	Description	Default
BROADCAST NOBROADCAST	Sockets are UDP broadcast. Sockets are not UDP broadcast.	Null character (hexadecimal 00)
KEEPALIVE NOKEEPALIVE	Sockets are TCP keepalive. Sockets are not TCP keepalive.	Null character (hexadecimal 00)
RECEIVE	Receive socket quota.	Null character (hexadecimal 00)
SEND	Send socket quota.	Null character (hexadecimal 00)

/USER_NAME=*vms_user_account*

Required if defining a new service entry; optional if modifying an existing service.

OpenVMS account information for users working on client systems. Required for a user to access the service.

The user must also be defined in the system user authorization file (SYSUAF.DAT).

Examples

1.
```
TCPIP> SET SERVICE TOE /USER_NAME=LITTLE_PIGGY -
_TCPIP> /PROCESS_NAME=TOEd /PORT=1050 /PROTOCOL=UDP -
_TCPIP> /FILE=SYS$COMMON:[SYSMGR]TOE_STARTUP.COM
```

   Modifies the service TOE to implement UDP on port 1050. This service has the OpenVMS process context of user LITTLE_PIGGY.

   After you issue a SET SERVICE TOE command, the auxiliary server executes TOE_STARTUP.COM when a request arrives for service TOE.

2.
```
TCPIP> SET SERVICE LPD -
_TCPIP> /REJECT=NETWORK=(11.30.0.0:255.255.0.0,11.40.0.0)
```

   Sets the LPD service to be inaccessible to the two specified networks.

3.
```
TCPIP> SET SERVICE RSH /FLAGS=(PROXY,CASE_INSENSITIVE)
```

   Sets the proxy and case-sensitivity flags for the RSH service.

SHOW ARP

SHOW ARP — Displays ARP information. *Related command:* SET ARP

Syntax

```
SHOW ARP [host]
         [/[NO]LOCAL]
```

Parameters

host

Optional. Default: All hosts; same as specifying /NOLOCAL.

Specifies the host about which you want information.

Qualifiers

/LOCAL
/[NO]LOCAL

Optional. Default: /NOLOCAL.

If you either omit this qualifier or specify /NOLOCAL, the software checks the hosts database. If a lookup fails, it also checks the BIND resolver.

Limits name-to-address lookups to the local hosts database.

Examples

1. `TCPIP> SHOW ARP`

```
Cnt  Flags   Timer  Host                    Phys Addr
  1: UC        425  *                       00-00-f8-40-ad-91
  2: UC         60  150.110.4.191           08-00-2b-39-4b-40
  3: UC        438  150.110.5.117           00-00-f8-8d-24-d4
  4: UCS       878  150.110.5.109           00-00-f8-4f-f1-63
  5: UCS       426  150.110.5.31            08-00-2b-a1-b1-93
  7: UC        104  150.110.4.2             aa-00-04-00-6d-10
```

Displays the contents of the ARP table.

The flags have the following meanings:

Flag	Meaning
U	Up, or in use
C	Complete
S	Stale
P	Permanent
D	Dead

2.
```
TCPIP> SHOW ARP PARROT
parrot (11.170.6.45) at 08-CC-2B-12-C2-BB
```

   Displays IP address-to-hardware address mapping for host PARROT.

SHOW BOOTP

SHOW BOOTP — Displays client entries in the BOOTP database. *Related command:* SET BOOTP

Syntax

```
SHOW BOOTP [ host ]
           [ /HARDWARE=ADDRESS=hex_address ]
           [ /LOCAL ]
           [ /OUTPUT=file ]
```

Parameters

host

Optional. Default: Displays all entries.

Host entry you want to display.

Qualifiers

/HARDWARE=ADDRESS=*hex_address*

Optional. Default: None.

Identifies the entry by hardware address.

Specify the address as *hh-hh-hh-hh-hh-hh*.

/LOCAL

Optional. Default: The command checks the hosts database; if a lookup fails, it also checks the BIND resolver.

To display hardware-address/IP-identification matches, limits host-name-to-IP-address lookup for each entry in the BOOTP database to the local hosts database.

/OUTPUT=*file*

Optional. Default: Screen display.

Output is directed to the specified file.

Example

```
TCPIP> SHOW BOOTP MYNA /FULL

Host:         17.22.222.255   myna
```

```
                                        Hardware Address: 07-06-2B-05-04-03

Network mask: 255.0.0.0                          Type:    Ethernet
File:         MYNA.SYS

Time offset:           0               Vendor: Birds, Inc.

Gateways:     not defined

Servers:
 Cookie:      not defined
 IEN:         not defined
 Impress:     not defined
 Log:         not defined
 LPR:         not defined
 Name:        owl.wise.mouser.edu
 Resource:    not defined
 Time:        not defined
TCPIP>
```

Displays the entry for client `myna` in the BOOTP database. When `myna` requests a download, TCP/IP Services sends system image MYNA.SYS.

SHOW COMMUNICATION

SHOW COMMUNICATION — Displays the settings and status for the IP, TCP, UDP, and INET_ACP software, including routing, buffers, sockets, and memory on the running system. *Related commands:* SET COMMUNICATION.

Syntax

```
SHOW COMMUNICATION [ /MEMORY ]
                   [ /ROUTE ]
                   [ /SECURITY ]
```

Qualifiers

/MEMORY

Optional.

Displays information about memory allocated to data structures associated with network operations.

/ROUTE

Optional.

Displays routing statistics.

/SECURITY

Optional.

Displays the security settings specified with the following SET COMMUNICATION qualifiers:

- /ACCEPT
- /REJECT

Examples

1. TCPIP> **SHOW COMMUNICATION/ROUTE**

```
Route statistics

       Created by redirects            0
       Modified by redirects           0
       Wildcarded lookups              6

       Invalid redirect calls          0
       Failed lookups                  0
```

   Displays routing statistics.

2. TCPIP> **SHOW COMMUNICATION /MEMORY**

```
1585 mbufs in use:
        490 mbufs allocated to data
        221 mbufs allocated to ancillary data
        2 mbufs allocated to packet headers
        131 mbufs allocated to socket structures
        198 mbufs allocated to protocol control blocks
        180 mbufs allocated to routing table entries
        246 mbufs allocated to socket names and addresses
        21 mbufs allocated to socket options
        76 mbufs allocated to interface addresses
        1 mbufs allocated to kernel table
        2 mbufs allocated to ip multicast options
        7 mbufs allocated to ip multicast address
        10 mbufs allocated to interface multicast addess
        1 mbufs allocated to <mbuf type 100>
        1 mbufs allocated to <mbuf type 108>
        3 mbufs allocated to <mbuf type 109>
        7 mbufs allocated to <mbuf type 112>
        28 mbufs allocated to <mbuf type 113>
        28 mbufs allocated to <mbuf type 114>
        3 mbufs allocated to <mbuf type 115>
        3 mbufs allocated to <mbuf type 117>
        5 mbufs allocated to <mbuf type 118>
```

   Displays information about memory usage.

3. TCPIP> **SHOW COMMUNICATION /SECURITY**

```
Communication Security Parameters

Allow Log:  Acpt Actv Dactv Conn Error Exit Logi Logo Mdfy Rjct TimO
 Addr
Force Log:  None

Security device:  disabled

Access lists
 Accept host: 0.0.0.0
```

```
Accept netw: 0.0.0.0
```

Displays information about security parameters that were specified with the SET COMMUNICATION command.

SHOW CONFIGURATION

SHOW CONFIGURATION — Displays the information in the configuration database.

Additional Information

This database, read by the TCP/IP Services startup procedure, holds information to:

- Configure the lower-layer protocols, including dynamic routing.
- Configure and enable the Internet interfaces.
- Configure the services you want enabled when TCP/IP Services starts up.
- Configure the BIND resolver.
- Configure the BIND server, including the default domain.
- Configure SMTP.
- Configure SNMP.
- Configure TIME.
- Map file systems.

Because the SET CONFIGURATION commands do not take effect until the next time TCP/IP Services starts up, the SHOW CONFIGURATION command might not display the current settings for the running software. For example, the information you see from SHOW CONFIGURATION PROTOCOL might differ from the SHOW PROTOCOL /PARAMETERS output.

Related commands: All SET CONFIGURATION commands

Syntax

```
SHOW CONFIGURATION { component | ENABLE SERVICE [service] | START ROUTING }
                   [ /COMMON ]
                   [ /FULL ]
                   [ /OUTPUT=file ]
```

Parameters

component

Required if you do not specify ENABLE SERVICE or START ROUTING.

Component, defined in the configuration database, whose configuration you want to display.

ENABLE SERVICE [*service*]

Required if you do not specify the *component* parameter or START ROUTING.

Displays either the specified component or all the components that are enabled by the TCP/IP Services startup procedure.

START ROUTING

Required if you do not specify *component* or ENABLE SERVICE.

Shows whether or not routing is configured to start running by the TCP/IP Services startup procedure.

Qualifiers

/COMMON

Optional. Default: Node specific.

Valid only with SHOW CONFIGURATION ENABLE SERVICE.

Displays the information in the configuration database for the clusterwide enabling or disabling of services.

/FULL

Optional. The default is to give a brief listing of the information.

Displays complete information.

Use with the *component* parameter.

/OUTPUT=*file*

Optional. Default: Screen display.

Writes output to the specified file.

Examples

1.
```
TCPIP> SHOW CONFIGURATION ENABLE SERVICE

Enable service

 FTP, FTP_CLIENT, LPD, MOUNT, NFS, NFS_CLIENT, PCNFS,
  PORTMAPPER, REXEC, RSH, SMTP, SNMP
```

   Displays the services configured in the services database that will be enabled by the TCP/IP Services startup procedure.

2.
```
TCPIP> SHOW CONFIGURATION SNMP

SNMP Configuration

Flags:    AuthenTraps Sets

Contact:  Sam Spade
```

```
Location
  First:  Falcon Building
  Second: Los Angeles, California

Community           Type       Address_list

public              Read       0.0.0.0

trapit              Read Trap  136.20.0.10

trapit2             Read Trap  136.20.0.12, 136.20.0.15

rw2                 Read Write 136.20.0.15, 136.20.0.100
```

Displays the SNMP configuration.

3.
```
TCPIP> SHOW CONFIGURATION NAME_SERVICE
BIND Resolver Configuration

  Transport:  UDP
  Domain:     fred.parrot.brd.com
  Retry:         4
  Timeout:       4
  Servers:    11.20.208.10, 11.20.208.53
  Path:       No values defined
TCPIP>
```

Displays, from the configuration database, the BIND resolver configuration.

4.
```
TCPIP> SHOW CONFIGURATION INTERFACE


 Interface: DE1
   IP_Addr: 11.20.208.100     NETWRK: 255.255.0.0  BRDCST: 11.20.255.255


 Interface: LO0
   IP_Addr: 137.0.0.1         NETWRK: 255.0.0.0    BRDCST:
TCPIP>
```

Displays the permanent definition of Internet interfaces DE1 and LO0.

5.
```
TCPIP> SHOW CONFIGURATION SMTP


SMTP Configuration

 Options
Initial interval:   0 00:30:00.00       Address_max:    16
 NOEIGHT_BIT
Retry interval:     0 01:00:00.00       Hop_count_max:  16       NORELAY

Maximum interval:   3 00:00:00.00
 TOP_HEADERS
```

```
   Timeout             Initial       Mail    Receipt       Data  Terminate

     Send:                   5          5          5          3         10

     Receive:                5

   Alternate gateway:  not defined
   General gateway:    not defined

   Substitute domain:  not defined
   Zone:               not defined

   Postmaster:         TCPIP_SMTP
   Log file:           SYS$SPECIFIC:[TCPIP$SMTP]TCPIP$SMTP_LOGFILE.LOG

   Generic queue       Queues   Participating nodes

   TCPIP$SMTP_CRANE_00     1     CRANE
```

Displays the SMTP configuration.

SHOW CONFIGURATION PROTOCOL

SHOW CONFIGURATION PROTOCOL — Displays information in the configuration database which sets the parameters for ICMP, IP, TCP, and UDP when TCP/IP Services starts up. **Related commands:** SET CONFIGURATION PROTOCOL, SET PROTOCOL

Syntax

```
SHOW CONFIGURATION PROTOCOL [ protocol ]
                           [ /PARAMETERS ]
```

Parameters

protocol

Optional. Default: All protocols.

Specify one of the following: ICMP, IP, TCP, UDP.

Qualifiers

/PARAMETERS

Optional.

Shows parameter settings. You can specify any protocol.

Example

```
TCPIP> SHOW CONFIGURATION PROTOCOL
```

```
ICMP
  Redirect:               enabled
  Unreachable:           disabled

IP
  Forward:               disabled

  Reassembly timer:             0

TCP
  Delay ACK:              enabled
  Window scale:           enabled
  Drop count:                   0
  Probe timer:                  0

                          Receive                Send

  Push:                  disabled            disabled
  Quota:                        0                   0

UDP
  Unpriv. broadcast:     disabled

                          Receive                Send

  Checksum:               enabled             enabled
  Quota:                        0                   0
```

Displays information in the configuration database.

SHOW DEVICE_SOCKET

SHOW DEVICE_SOCKET — Displays: device socket counters or current settings from the running communications software. A device socket is composed of two parts: the Internet device (interface) and the socket.

Syntax

```
SHOW DEVICE_SOCKET [ device_socket ]
                   [ /CONTINUOUS[=n] ]
                   [ /FULL ]
                   [ /HOST=host ]
                   [ /[NO]LOCAL ]
                   [ /PORT=n ]
                   [ /SERVICE=service ]
                   [ /TYPE=socket_type ]
```

Parameters

device_socket

Optional. Default: All device sockets.

Device socket for which you want information.

Specify the characters BG and the device's unit number, for example, BG3, BG17.

Qualifiers

/CONTINUOUS[=*n*]

Optional. Default: Static display; /CONTINUOUS=4.

Automatically updates the display.

Screen update interval in seconds. Valid only for a specific device socket.

To terminate the display, press Ctrl/C.

/FULL

Optional. The default is to give a brief listing of the information.

Displays complete information, including:

- Application option settings, for example, ACCEPT or FULL_DUPLEX_CLOSE
- State of the service
- Counters for receiving and sending buffers

/HOST=*host*

Optional. Default: All hosts.

Displays information for device sockets bound to the specified host.

This does not include device sockets bound to the "ANY host" (address 0.0.0.0).

/LOCAL
/NOLOCAL

Optional. Default: /LOCAL.

Displays information for device sockets bound to hosts in the local hosts database.

/NOLOCAL: Displays information for device sockets bound to hosts in the hosts database and hosts known by the BIND resolver.

/PORT=*n*

Optional. Default: All local ports.

Displays information for device sockets bound to the specified local port.

/SERVICE=*service*

Optional. Default: All services.

Displays information for device sockets used for the specified service.

/TYPE=*socket_type*

Optional. Default: /TYPE=ALL.

Displays information for the specified type of device socket. Specify one of the following:

- STREAM
- DGRAM

Examples

1. TCPIP> **SHOW DEVICE_SOCKET BG21 /FULL**

```
Device_socket:  BG21        Type: STREAM
                      LOCAL                             REMOTE
         Port:           21                                0
         Host:  *                                *
      Service:  FTP

                                                     RECEIVE    SEND
                          Queued I/O                       0          0
        Q0LEN        0    Socket buffer bytes              0          0
        QLEN         0    Socket buffer quota          61440      61440
        QLIMIT    1024    Total buffer alloc               0          0
        TIMEO        0    Total buffer limit          491520     491520
        ERROR        0    Buffer or I/O waits              1          0
        OOBMARK      0    Buffer or I/O drops              0          0
                          I/O completed                    0          0
                          Bytes transferred                0          0

  Options:  ACCEPT REUSEADR
  State:    PRIV
  RCV Buff: WAIT
  SND Buff: None
```

Displays complete information about device socket BG21, including the options that are set, for example, the FULL_DUPLEX_CLOSE option.

(With the FULL_DUPLEX_CLOSE option set, the first data transmission on a TCP connection that has been closed by the remote application returns an EPIPE error. Subsequent `send` operations on the half-closed socket proceed normally.)

The following table describes the counters in the first column of the display.

Counter	**Meaning**
Q0LEN	Number of sockets that are about to be connected to the specified socket
QLEN	Number of sockets that have established a connection but have not yet been accepted by the specified socket
QLIMIT	Number of sockets for the Q0LEN and QLEN sockets
TIMEO	Not used

Counter	Meaning
ERROR	Error code temporarily stored on the socket
OOBMARK	Out-of-band mark

2. `TCPIP> SHOW DEVICE_SOCKET BG75 /CONTINUOUS=10`

   Displays information about device socket BG75 every 10 seconds.

3. `TCPIP> SHOW DEVICE_SOCKET /HOST="lark"`

   Displays information about all device sockets for remote host `lark`.

4. `TCPIP> SHOW DEVICE_SOCKET BG1898`

```
                              Port                          Remote
Device_socket  Type    Local  Remote  Service               Host

  bg1898      STREAM      23    2568  TELNET              16.20.176.227
```

   Displays socket type, service, and host information for device BG1898.

SHOW EXPORT

SHOW EXPORT — Displays disks/directories available for mounting by NFS clients, in the form of UNIX path names. It also displays the clients allowed to mount these path names. *Related commands:* ADD EXPORT, REMOVE EXPORT, MAP, SET CONFIGURATION MAP, SET CONFIGURATION NOMAP, SHOW MAP, SHOW CONFIGURATION MAP

Syntax

```
SHOW EXPORT [ "/path/name" ]
            [ /HOST=host ]
            [ /OUTPUT=file] ]
```

Restrictions

Requires read access to the export database.

Parameters

"*/path/name*"

Optional. Default: All exported file systems.

Exported files for which to show access rights.

To specify multiple directory names, separate them with slashes.

Qualifiers

/HOST="*host_name*"

Optional. Default: All hosts.

NFS client hosts for which you want to display access rights.

/OUTPUT=*file*

Optional. Default: Screen display. Sends output to the specified file.

Examples

1.
```
TCPIP> SHOW EXPORT

File system                Host name

/TOUCAN                    TOUCAN, toucan
/talkers                   parrot
/aviary_dua0               *
/condor_root/root          condor
/condor_root/work4         condor
/nene_d                    *
/nfstest_unix/lark1        *
/nfstest_unix/lark2        *
/spoonbill                 dove, nuthatch, dove.tree.branch.com
                           toucan, oriole.tree.branch.com
/spoonbill/birdy           dove, DOVE, nuthatch, thrush,
                           thrush.tree.branch.com
/spoonbill/bigbirdy        dove
```

Displays exported NFS file systems with the clients that have access rights.

2.
```
TCPIP> SHOW EXPORT/HOST="condor"

File System                              Host name

/condor_root/root            condor
/condor_root/work4                       condor
```

Displays exported NFS file systems for clients on host `condor`.

SHOW HOST

SHOW HOST — Displays information from the hosts database. If the BIND resolver is enabled, information from the BIND database is also displayed. *Related commands:* SET HOST

Syntax

```
SHOW HOST [ host ]
          [ /ADDRESS=IP_address ]
          [ /DOMAIN=domain ]
          [ /LOCAL ]
          [ /OUTPUT=file ]
          [ /SERVER=server ]
```

Restrictions

Requires read access to the hosts database.

Parameters

host

Optional. Default: All hosts.

All alias names for the specified host are displayed.

- If a host has more than one IP address and you specify the name, all its addresses and aliases are displayed.
- If a host has multiple IP addresses and you specify an alias that is defined on multiple IP addresses, only the first IP address and aliases are displayed.
- If you do not specify the *host* parameter or if you use a wildcard, all hosts from the local and BIND databases are displayed.
- If you use a wildcard to complete a host name, no BIND information is displayed.
- If you specify a host, entries are displayed first from the local hosts database, if they exist; otherwise, entries from the BIND database are displayed, if they exist.

Qualifiers

/ADDRESS=*IP_address*

Optional. Default: None.

Allows you to select a host by IP address.

- If a host has more than one IP address and you specify the name, all IP addresses and aliases for the host are displayed.
- If a host has multiple IP addresses and you specify an alias that is defined on multiple IP addresses, only the first IP address and aliases are displayed.
- Recommended: Use the *host* parameter instead of this qualifier.

/DOMAIN=*domain*

Optional. Default: Name service domain.

Domain to be used by the local host. However, the definition of the domain name is valid only during the execution of the current SHOW HOST command. The BIND request is sent to the specified domain.

/LOCAL

Optional.

Limits name-to-address lookups to the local hosts database.

/OUTPUT=*file*

Optional. Default: Screen display.

Specifies a file for the output of the SHOW HOST command.

/SERVER=*server*

Optional. Default: Name server list.

BIND servers to be used. The definition of the server name list is valid only during the execution of the current SHOW HOST command. The request is sent to the specified server.

The list is ordered by request preference. For example, the initial request is sent to the first host in the list. If that host is unavailable, the request is sent to the second host in the list, and so on.

You can specify a maximum of three servers.

Examples

1. TCPIP> **SHOW HOST /LOCAL**

```
     LOCAL database

Host address    Host name

11.180.6.60    aa80z, AA80Z
11.180.4.1     abbss.zz3.ddd.com, abbss, ABBSS, ab, a
11.180.6.8     alibam, ALIBAM, alb
11.180.5.5     alLpin, ALLPIN, allpine.zz3.ddd.com
11.180.6.30    amfer, AMFER
11.180.6.2     ankles, ANKLES
11.180.6.73    auntie, AUNTIE, maitai
  .
  .
  .
11.180.4.200   zlepin, ZLEPIN
11.180.20.1    zooley, ZOOLEY, zoo
11.180.6.37    zxtra, ZXTRA
```

The /LOCAL qualifier displays only the hosts in the local database.

2. TCPIP> **SHOW HOST ABCXYZ**

```
    BIND database

Server:    128.182.4.164    ZSERVE

Host address     Host name

128.180.5.164    ABCXYZ.one.nam.com
```

Displays information about a host found in the BIND database. Note that the display includes the name and address of the BIND server that supplied the information.

3. TCPIP> **SHOW HOST ***

Displays the entire hosts database and BIND database (if the resolver is enabled).
<EXAMPLES_INTRO> In the following examples, host `heron` has the following IP addresses and aliases:

IP Address	Host	Aliases
100.1	heron	HOST_1A HOST_1B HOST_ALIAS
100.2	heron	HOST_2A HOST_2B HOST_ALIAS

4. `TCPIP> SHOW HOST HERON`

   Shows all the IP addresses and aliases for the host HERON.

5. `TCPIP> SHOW HOST HOST_1A`

   Shows IP address 100.1 and the aliases HOST_1A, HOST_1B, and HOST_ALIAS.

6. `TCPIP> SHOW HOST HOST_ALIAS`

   Shows the host and all aliases for addresses 100.1 and 100.2. Shows all the IP addresses and aliases for host `heron`.

SHOW INTERFACE

SHOW INTERFACE — Displays information from the running system for Internet interfaces and pseudointerfaces. **Related commands:** SHOW CONFIGURATION INTERFACE, SET INTERFACE.

Additional Information

The flags that can appear in the display include:

- AMCST – The interface will receive multicast packets.
- BRDCAST – Indicates the interface supports broadcast messages.
- LOOP – The interface is a loopback mode. Packets transmitted on this interface will be looped back in the driver and not be transmitted out on the network.
- MCAST – The interface supports multicast packets. However, this does not mean that a multicast address is configured for the interface.
- NOARP – The interface is not using address resolution protocol (ARP). It will neither transmit nor respond to ARP requests.
- PFCPY – All packets transmitted on this interface are copied and passed to the packet filter program.
- PTP – The interface is point-to-point link. This is a read-only flag that is set by the driver.

- RUN – Indicates the interface is operational. The driver has allocated resources for the interface and is ready to transmit and receive packets. This option is not applicable to loopback devices, for example, LO0.
- SMPX – The interface cannot hear its own transmissions.
- UP – Indicates the interface is enabled for use.
- VMTU – The interface supports variable maximum transmission unit (MTU) sizes.

Syntax

```
SHOW INTERFACE [ interface ]
               [ /CLUSTER ]
               [ /FULL ]
```

Parameters

interface

Optional. Default: All interfaces.

Specifies the name of an Internet interface or pseudointerface. Examples include ZE0, LO0, QE2, QE3.

Qualifiers

/CLUSTER

Optional. Default: None.

Displays information about the cluster of which the interface is a member.

/FULL

Optional. Default: Brief description is displayed.

Displays full information.

Example

```
TCPIP> SHOW INTERFACE WE0 /FULL

Interface: WE0
   IP_Addr: 126.65.100.102   NETWRK: 255.255.255.0   BRDCST: 126.65.100.255

               Ethernet_Addr:AA-00-05-CC-2D-2B       MTU: 65535
     Flags: UP BRDCST RUN
                                  RECEIVE        SEND
   Packets                        3817269      595744
      Errors                            0
   Collisions:                          0
```

Displays information about interface WE0.

SHOW MAIL

SHOW MAIL — Displays SMTP queue information. *Related commands:* REMOVE MAIL, SEND MAIL.

Syntax

```
SHOW MAIL [ user ]
          [ /FULL ]
          [ /RECIPIENT[=options] ]
          [ /ENTRY=n ]
```

Restrictions

Requires SYSPRV or BYPASS privilege to display information for other users.

Parameters

user

Optional. Default: All users.

Displays SMTP process information of the specified user.

Qualifiers

/FULL

Optional. Default: Brief description is displayed.

Displays detailed information.

/RECIPIENT[=*options*]

Optional. Default: ALL.

Used with /FULL, displays selected recipient classes. Available options include the following:

ALL	Shows failed, sent, and unsent mail messages.
FAILED	Shows messages that could not be read for a particular recipient.
SENT	Shows successful deliveries to a particular recipient.
UNSENT	Shows messages that are as yet unsent.

/ENTRY=*n*

Optional. Default: Your queue entries.

Displays information about the specified queue entry number.

Examples

1. `TCPIP> SHOW MAIL`

   Displays information about mail messages queued to your process's user name.

2. `TCPIP> SHOW MAIL /ENTRY=1234`

   Displays information about the mail message 1234 in the queue.

3. `TCPIP> SHOW MAIL /FULL /RECIPIENT=ALL`

   Displays detailed information about all mail messages sent by the user of your process's user name.

SHOW MAP

SHOW MAP — Displays the names of mapped (logically linked) file systems, also called NFS file systems. *Applies to:* NFS server. *Related commands:* MAP, UNMAP, ADD EXPORT, SHOW EXPORT, REMOVE EXPORT, SET CONFIGURATION MAP, SET CONFIGURATION NOMAP, SHOW CONFIGURATION MAP.

Syntax

`SHOW MAP [ /path/name ]`

Parameters

/path/name

Optional.

Name of the file system (the first element of the UNIX file specification).

Examples

1. `TCPIP> SHOW MAP`

```
             Dynamic Filesystem Map
Pathname                                 Logical File System

/water     USER$DKC100:
/water                                   USER$DKC100:[WATER]
/duck/pond                               USER$DKC100:[DUCK.POND.TEAL]

TCPIP>
```

   Displays all mapped file systems.

2. `TCPIP> SHOW MAP "/bird"`

```
             Dynamic Filesystem Map
Pathname                                 Logical File System

/bird                                    $1$DUA7:
```

```
TCPIP>
```

Lists mapped file system `/bird`.

SHOW MOUNT

SHOW MOUNT — Displays a list of mounted directories at all mount points or at a particular mount point. *Related commands:* MOUNT, DISMOUNT.

Syntax

```
SHOW MOUNT [ device ]
           [ /ALL ]
           [ /FULL ]
           [ /HOST=host ]
```

Parameters

device

Optional. Default: All mounted file systems.

Local device for which to display mount information. Specify one of the following:

- DNFS*n*: — the full NFS device name and directory tree, for example, DNFS3:[USER.NOTES]
- Volume label
- Logical name for the device

You can use abbreviations and wildcards.

Qualifiers

/ALL

Optional.

If you also specify *device*, displays information for all NFS server hosts with mounted file systems on this device.

If you do not specify *device*, displays information for all NFS server hosts with mounted file systems on any device.

Not valid with /HOST.

/FULL

Optional. Default: Brief description is displayed.

Displays the full, current operating parameters related to each mount.

/HOST=*host*

Optional. Default: All NFS servers with file systems currently mounted.

NFS server on which the physical files reside.

Not valid with /ALL.

Examples

1.
```
TCPIP> SHOW MOUNT

_DNFS1:[000000]  automount (inactivity timer  0 00:23:00.00), mounted
        SIGMA.PROCESS.COM:/usr
_DNFS2:[000000]  mounted
        IRIS.PROCESS.COM:/usr/users
```

   Shows the characteristics of all mounted file systems on all local NFS devices.

2.
```
TCPIP> SHOW MOUNT DNFS3: /ALL

_DNFS3:[A.B]  mounted
        SIGMA.PROCESS.COM:/usr
_DNFS3:[A.C]  mounted
        SIGMA.PROCESS.COM:/work
```

   Shows the characteristics of all mounted file systems on local device DNFS3:.

SHOW MX_RECORD

SHOW MX_RECORD — Displays SMTP routing information. If you omit *destination*, you see the entries in the local Mail Exchange (MX) database. If you specify *destination*, you see all the entries in all the databases that TCP/IP Services would look at, if necessary, to resolve the address. To send mail, SMTP looks up addresses in one or more databases (if necessary) in the following order: local MX database, remote MX database, BIND server database, local hosts database. *Related command:* SET MX_RECORD.

Syntax

```
SHOW MX_RECORD [ destination ]
               [ /GATEWAY=host ]
               [ /OUTPUT=file ]
```

Parameter

destination

Optional. Default: All entries in the local MX database.

Final destination host name.

Qualifiers

/GATEWAY=*host*

Optional. Default: All destinations.

Displays the destinations that are accessed through the specified gateway.

/OUTPUT=*file*

Optional. Default: Screen display.

Sends the output to the specified file.

Examples

1. TCPIP> **SHOW MX_RECORD SWAN**

```
                                BIND MX database

Server:           18.18.218.10     GREAT.HORNED.OWL.COM

Gate address      Preference       Gate name

18.18.218.10      50               WATER.PIPIT.WEBBED.FEET.COM
18.1.218.16       100              bd-gw.purple.martin.com
188.88.206.2      200              great.horned.owl.com
199.9.214.1       300              bird.food.seeds.worms.com


                                BIND database

Server:           18.18.218.10     WATER.PIPIT.WEBBED.FEET.COM

Host address                       Host name

18.18.100.10                       SWAN.WEBBED.FEET.COM
```

Displays, in order of preference, the routing hops to reach host SWAN if an attempt fails. The local host tries to route through:

a. `WATER.PIPIT.WEBBED.FEET.COM`

b. `bd-gw.purple.martin.com`

c. `great.horned.owl.com`

d. `bird.food.seeds.worms.com`

Both the alternate gateway and the zone affect how SMTP determines where to relay nonlocal mail.

MX records tell mailers where to relay mail that is destined for a given host. In the display:

- The `Gate name` field tells where to relay the mail.
- The `Gate address` field gives the gateway's IP address.
- The `Preference` field gives each MX record a precedence. A lower preference number means a higher precedence.

2. TCPIP> **SHOW MX_RECORD CROW.COM**

```
                                BIND MX database
```

```
Server:          18.18.218.10     WATER.PIPIT.WEBBED.FEET.COM

Gate address     Preference       Gate name

159.228.12.253   1                cawcaw.crow.com
159.228.12.254   2                scare.crow.com
TCPIP>
```

Displays the MX record for destination host `crow.com`. In the display:

- The `Gate name` field tells where to relay the mail.
- The `Gate address` field gives the gateway's IP address.
- The `Preference` field gives each MX record a precedence. A lower preference number means higher precedence.

In this example, the local host name is WATER, the alternate gateway is `scare.crow.com`, and the zone is `crow.com`. The first preference for delivering mail to `crow.com` is to send to `cawcaw.crow.com`.

If you have not defined an alternate gateway, SMTP tries to relay the mail to `scare.crow` at IP address 158.228.12.253. It uses the MX records to determine the host to which to relay mail. SMTP tries to relay the mail to each gateway host, in order of preference, until it either successfully transfers the mail or runs out of MX records to try. If there is no alternate gateway, the zone is not used.

If you have defined an alternate gateway, SMTP goes through the list of MX records, but it does not automatically try to relay the mail directly to the gateway. SMTP checks whether the gateway host name is outside or inside the SMTP zone (as defined with SET SMTP CONFIGURATION). If the gateway is inside the SMTP zone, SMTP tries to relay the mail directly to the gateway host. If the gateway is outside the zone, SMTP sends the mail to the alternate gateway.

SHOW NAME_SERVICE

SHOW NAME_SERVICE — Logs information about the BIND resolver. **Related commands:** SET NAME_SERVICE, SHOW CONFIGURATION NAME_SERVICE.

Syntax

```
SHOW NAME_SERVICE [ /STATISTICS ]
```

Restrictions

The /STATISTICS qualifier requires BYPASS, READALL, or SYSPRV privilege.

Qualifiers

/STATISTICS

Optional.

Dumps statistics to SYS$SPECIFIC:[TCPIP$BIND]TCPIP$BIND.STATS.

Either TCPIP$ETC:RNDC.CONF or TCPIP$ETC:RNDC.KEY must be set up to allow for secure communication between the user and the BIND server. For more information, refer to the VSI TCP/IP Services for OpenVMS Management manual.

Examples

1.
```
TCPIP> SHOW NAME_SERVICE
BIND Resolver Parameters

 Local domain: TCPIP.OWL.ROC.COM

 System

  State:     Started, Enabled

  Transport: UDP
  Domain:    tcpip.owl.roc.com
  Retry:     4
  Timeout:   4
  Servers:   LOCALHOST, tcpip.owl.roc.com

 Process

  State:     Started, Enabled

  Transport: UDP
  Domain:    11.180.34.3
  Retry:     4
  Timeout:   4
  Servers:   LOCALHOST, lark, crow.moa.awk.com
TCPIP>
```

   Shows systemwide and process-specific parameter settings for the BIND resolver.

2.
```
TCPIP> SHOW NAME_SERVICE /STATISTICS
```

   Logs current BIND server statistics to the file TCPIP$BIND_SERVER_STATISTICS.LOG. The following sample shows such a log file.

```
+++ Statistics Dump +++ (922292822) Wed Mar 24 11:27:02
34250   time since boot (secs)
15670   time since reset (secs)
12      Unknown query types
20000   A queries
540     SOA queries
2399    MX queries
867     ANY queries
3       AXFR queries

++ Name Server Statistics ++
(Legend)
        RR      RNXD    RFwdR   RDupR   RFail
        RFErr   RErr    RAXFR   RLame   ROpts
        SSysQ   SAns    SFwdQ   SDupQ   SErr
        RQ      RIQ     RFwdQ   RDupQ   RTCP
```

```
        SFwdR   SFail   SFErr   SNaAns  SNXD
(Global)
        2 0 0 0 0  0 0 0 0 0  2 0 0 0 0  0 0 0 0 5  0 0 0 0 0
– Name Server Statistics --
++ Memory Statistics ++
      3:           9 gets,           2 rem
      4:           7 gets,           0 rem (1 bl, 1022


      5:          16 gets,           1 rem
      6:           7 gets,           5 rem
      7:          10 gets,           5 rem
      8:          97 gets,          16 rem (1 bl, 485 ff)
     13:           6 gets,           4 rem
.
.
.
    664:           5 gets,           1 rem (1 bl, 5 ff)
    732:           2 gets,           0 rem (1 bl, 5 ff)
   1040:           1 gets,           1 rem (1 bl, 2 ff)
>= 1100:          23 gets,           9 rem
– Memory Statistics --
– Statistics Dump – (907337687) Fri Jan 2  10:14:47 2003
```

SHOW NETWORK

SHOW NETWORK — Displays information about the networks database. *Related command:* SET NETWORK

Syntax

```
SHOW NETWORK [ network ]
             [ /ADDRESS=address ]
             [ /OUTPUT=file ]
```

Restrictions

Requires read access to the networks database.

Parameters

network

Optional. Default: All known networks.

Network about which to display information.

- Displays all alias names of the specified network.
- If you specify an alias, the network name and all its alias names are displayed.

Qualifiers

/ADDRESS=*address*

Optional. Default: None.

Selects networks by address.

Not valid with the *network* parameter.

/OUTPUT=*file*

Optional. Default: Screen display.

Output is written to the specified file.

Examples

1.
```
TCPIP> SHOW NETWORK COBNET

Network address    Network name

4.0.0.0            COBNET
```

Displays the entry for COBNET in the networks database.

2.
```
TCPIP> SHOW NETWORK Z*
Network address    Network name

138.180.4.0        zznet, ZZNET
120.45.30.0        zzo-net, ZZO-NET, zz01-net
```

From the networks database, displays the entries for all the networks with names or aliases beginning with the letter Z.

SHOW NFS_SERVER

SHOW NFS_SERVER — Displays NFS server performance counters and statistics. *Related commands:* SET NFS_SERVER, ZERO NFS_SERVER.

Syntax

```
SHOW NFS_SERVER [ /CONTINUOUS[=seconds] ]
                [ /RPC]
                [ /SERVER ]
                [ /VERSION=versions ]
```

Restrictions

Requires SYSNAM and WORLD privilege.

Qualifiers

/CONTINUOUS[=*seconds*]

Optional. Defaults: Static display; if you specify /CONTINUOUS without a value, the default is 4 seconds.

Provides a dynamic display with optional screen-update interval.

To terminate the display, press Ctrl/Y.

/RPC

Optional.

Displays only RPC-related performance counters and statistics.

/SERVER

Optional.

Displays NFS server-related performance counters and statistics.

/VERSION=*versions*

Optional. Default: Displays both Version 2 and Version 3.

Displays version-specific NFS server performance counters and statistics. You can specify *versions* as follows:

Qualifier	**Displays**
/VERSION=V2	Only Version 2
/VERSION=V3	Only Version 3
/VERSION=(V2,V3)	Both Version 2 and Version 3

SHOW PORTMAPPER

SHOW PORTMAPPER — Displays a list of all registered remote procedure call (RPC) programs. The Portmapper running on the specified host gets this list.

Syntax

```
SHOW PORTMAPPER [ host ]
```

Parameter

host

Optional. Default: Local host.

Host with the Portmapper you want to query.

Examples

1. `TCPIP> SHOW PORTMAPPER`

```
Program Number    Version Protocol  Port-number Process   Service-name
----------------- ------- --------  ----------- --------  ------------
000186A0 (100000)       2  TCP         111       56E0021D  PORTMAPPER
000186A0 (100000)       2  UDP         111       56E0021D  PORTMAPPER
000186A3 (100003)       3  UDP        2049       56E0021F  NFS
000186A5 (100005)       1  UDP          10       56E00220  MOUNT
000249F1 (150001)       1  UDP        5151       56E00222  PCNFS
```

Lists information about all of the currently registered applications.

2. TCPIP> **SHOW PORTMAPPER PARROT**

```
Program Number    Version Protocol  Port-number Process   Service-name
----------------- ------- --------  ----------- --------  ------------
000186A0 (100000)       2  TCP          111       24800126  PORTMAPPER
000186A0 (100000)       2  UDP          111       24800126  PORTMAPPER
```

Queries host PARROT for a list of registered applications.

SHOW PROTOCOL

SHOW PROTOCOL — Displays statistics and configuration information for the specified protocol.
Related commands: SET PROTOCOL SET CONFIGURATION PROTOCOL

Syntax

```
SHOW PROTOCOL [ protocol ]
              [ /PARAMETERS ]
```

Parameter

protocol

Optional. Default: All protocols.

Specify one of the following: ICMP, IP, TCP, UDP.

Qualifier

/PARAMETERS

Optional.

Shows parameter settings. You can specify any protocol.

Examples

1. TCPIP> **SHOW PROTOCOL TCP**

```
tcp:
        64213 packets sent
                56262 data packets (44164814 bytes)
                49 data packets (39372 bytes) retransmitted
                7792 ack-only packets (7923 delayed)
                0 URG only packets
                0 window probe packets
                10 window update packets
                100 control packets
        50000 packets received
                37102 acks (for 44165036 bytes)
                381 duplicate acks
```

```
                0 acks for unsent data
                23176 packets (194520 bytes) received in-sequence
                233 completely duplicate packets (290 bytes)
                50 packets with some dup. data (65 bytes duped)
                57 out-of-order packets (43 bytes)
                4 packets (4294967292 bytes) of data after window
                0 window probes
                916 window update packets
                0 packets received after close
                0 discarded for bad checksums
                0 discarded for bad header offset fields
                0 discarded because packet too short
        54 connection requests
        35 connection accepts
        89 connections established (including accepts)
        91 connections closed (including 3 drops)
        1 embryonic connection dropped
        30253 segments updated rtt (of 30286 attempts)
        14 retransmit timeouts
                0 connections dropped by rexmit timeout
        1 persist timeout
        2 keepalive timeouts
                2 keepalive probes sent
                0 connections dropped by keepalive
```

Displays the TCP statistics.

The following abbreviations are used for the TCP counters display:

ack — acknowledge
URG — urgent
dup. — duplicate
embryonic connections – connections not yet established
rtt — retries
rexmt — retransmit

2. `TCPIP> SHOW PROTOCOL TCP /PARAMETERS`

```
TCP
  Delay ACK:              enabled
  Window scale:           enabled
  Drop count:                   8
  Probe timer:                150

                          Receive                Send

  Push:                  disabled            disabled
  Quota:                    32768               32768
```

Displays the TCP parameters.

SHOW PROXY

SHOW PROXY — Displays entries in the proxy database. *Related commands:* ADD PROXY, REMOVE PROXY. *Applies to:* NFS server, NFS client, PC-NFS, remote shell, LPR/LPD, and customer-developed services.

Syntax

```
SHOW PROXY [ user_name ]
           [ /COMMUNICATION ]
           [ /GID=n ]
           [ /HOST=host ]
           [ /NFS =[options] ]
           [ /UID=n ]
```

Restrictions

Requires read access to the proxy database.

Parameters

user_name

Optional. Default: SHOW PROXY * (all entries).

Specifies the local OpenVMS identity for the user of the NFS server, NFS client, PC-NFS, remote shell, or LPR/LPD.

Qualfiers

/COMMUNICATION

Optional. Default: Displays both communication and NFS proxies.

Displays communication proxies.

/GID=*n*

Optional. Default: Displays all NFS proxies.

Displays the database entries for all clients with the specified GID.

/HOST=*host*

Optional. Default: Displays information for all hosts (same as /HOST=*).

Specifies the remote host from which information is to be displayed.

/NFS=*option*

Optional. Default:

- If you omit this qualifier, displays both communication and NFS proxies.
- If you omit *option*, displays both incoming and outgoing proxies.

Displays NFS proxies.

These entries might be for local clients, remote clients, or PC-NFS clients. You can include the following options:

/NFS=OUTGOING	Proxy to use NFS client

/NFS=INCOMING	Proxy to use NFS server
/NFS=(OUTGOING,INCOMING)	Proxy to use both the NFS client and the NFS server

/UID=*n*

Optional. Default: All NFS proxies.

Displays the database entry for the client with the specified UID.

Examples

1.
```
TCPIP> SHOW PROXY /NFS

VMS User_name     Type      User_ID    Group_ID   Host_name

WEBSTER           OD            311          10   *
SHERMAN           ND            115          10   *
COHEN             OND           115          10   *
SILK              ON            115          10   *
```

Shows the NFS entries in the proxy database:

- WEBSTER has authorization to use the local NFS client (outgoing rights).
- SHERMAN can use the local NFS server (incoming rights).
- COHEN can use both the NFS server and client.
- SILK can use both the NFS server and client. This information is not currently known to NFS because SILK is not loaded in the dynamic database.

In the display, the values in the `Type` field mean:

N	NFS server
O	NFS client
ON	NFS server and client
C	Communication
OD, ND, OND	Loaded in the NFS cache
CD	Loaded in the communication cache

To set up N, O, or ON proxies, see ADD PROXY /NFS=INCOMING=OUTGOING.

2.
```
TCPIP> SHOW PROXY /COMMUNICATION

VMS User_name     Type   Remote User_name         Host_name

BLUEJAY           CD     JAY                      *
QUETZAL           CD     quetzal                  central.america.com
FALCON            CD     FALCON                   HAWK
MYNA              C      MYNA                     PARROT,parrot
```

```
CANVASBACK          CD      CBACK                       DUCK,duck
```

Shows all the communication proxies.

SHOW ROUTE

SHOW ROUTE — Displays the permanent or volatile routes database. To display the permanent database, use the /PERMANENT qualifier. Looks up the destination you specify first in the hosts database and then, if this lookup fails, in the networks database. Displays the following routes and their types: A — Active route (created manually or associated with an interface), D — Dynamic route (created by ROUTED or GATED routing daemon), H — Host route (a route to a host), N — Network route (a route to a network), P — Permanent (from the routes database). **Related command:** SET ROUTE.

Syntax

```
SHOW ROUTE [ destination ]
           [ /FULL ]
           [ /GATEWAY=host ]
           [ /LOCAL ]
           [ /OUTPUT=file ]
           [ /PERMANENT ]
```

Restrictions

Requires read access to the routes database.

Parameters

destination

Optional. Default: Displays all routes.

Destination host.

Qualifiers

/FULL

Optional. Default: Displays routes as specified in the routes database.

Displays mapping between destination addresses and names and gateway addresses and names.

/GATEWAY=*host*

Optional. Default: All gateways.

Displays information for the specified host that performs as a gateway.

/LOCAL

Optional. Default: The command checks the hosts database; if a lookup fails, it checks the BIND resolver.

Limits name-to-address lookups to the local hosts database.

/OUTPUT=*file*

Optional. Default: Screen display.

Sends output to the specified file.

/PERMANENT

Optional.

Displays only the permanent routes database.

- If TCP/IP Services is running and you omit /PERMANENT, the volatile database is displayed.
- If TCP/IP Services is not running, the permanent database is displayed.

Examples

1. `TCPIP> SHOW ROUTE`

```
                                DYNAMIC

Type           Destination                          Gateway

AN    0.0.0.0                                16.20.0.173
AN    16.20.0.0/16                           16.20.208.100
AH    16.20.208.100                          16.20.208.100
AH    127.0.0.1                              127.0.0.1
```

   Displays all defined routes.

2. `TCPIP> SHOW ROUTE "robin"`

   Displays the network route to host `robin`.

SHOW SERVICE

SHOW SERVICE — Displays the following information about configured services: Service name, Port for listening, Protocol, Process name, IP address, State, RPC information. *Related commands:* SET SERVICE, DISABLE SERVICE, ENABLE SERVICE.

Syntax

```
SHOW SERVICE [ service ]
             [ /ADDRESS=address ]
             [ /FULL ]
             [ /PERMANENT ]
             [ /PORT=n ]
             [ /PROCESS=process ]
             [ /PROTOCOL=protocol ]
             [ /RPC ]
```

Parameters

service

Optional. Default: All services.

Service for which you want information.

Qualifiers

/ADDRESS=*address*

Optional. Default: All services.

Displays information for only the services that use the specified address.

/FULL

Optional. Default: Brief description is displayed.

Provides a full display.

/PERMANENT

Optional.

Defaults:

- If TCP/IP Services is running and you omit the /PERMANENT qualifier, the volatile database is displayed.
- If TCP/IP Services is not running, the permanent database is displayed.

You must include the /PERMANENT qualifier when you specify the /RPC qualifier.

/PORT=*n*

Optional. Default: All services.

Displays information only for services that use the specified port.

/PROCESS=*process*

Optional. Default: All services.

Displays information for only the services that use the specified process.

/PROTOCOL=*protocol*

Optional. Default: All services.

Displays information only for services that use the specified protocol.

/RPC

Optional. Default: No RPC information is displayed.

Displays a brief summary of the services that are configured with RPC information. You must include the /PERMANENT qualifier when you specify the /RPC qualifier.

Examples

1. TCPIP> **SHOW SERVICE /RPC /PERMANENT**

```
                          RPC              Protocol Versions
Service             Program Number          Lowest / Highest

MOUNT                     100005                 1        1
NFS                       100003                 2        2
PCNFS                     150001                 1        2
PORTMAPPER                100000                 2        2
TCPIP>
```

   Displays all previously set RPC information.

2. TCPIP> **SHOW SERVICE NFS /FULL /PERMANENT**

```
Service: NFS

Port:            2049     Protocol:  UDP            Address:  0.0.0.0
Inactivity:         0     User_name: TCPIP$NFS      Process:  TCPIP$NFS

Limit:              1

File:        TCPIP$SYSTEM:TCPIP$NFS_RUN.COM
Flags:       TCPIP

Socket Opts: Rcheck Scheck
 Receive:        64000     Send:           64000

Log Opts:    Acpt Actv Dactv Conn Error Exit Logi Mdfy Rjct TimO Addr
 File:       SYS$SYSDEVICE:[TCPIP$NFS]TCPIP$NFS_RUN.LOG

RPC Opts
 Program number:   100003  Lowest version:   2   Highest version:   2

Security
 Reject msg:  not defined
 Accept host: 0.0.0.0
 Accept netw: 0.0.0.0
TCPIP>
```

   The /FULL and /PERMANENT qualifiers display RPC information for the NFS server, whose program number is 100003, lowest version is 2, and highest version is 2. This information is required for the NFS server to run.

3. TCPIP> **SHOW SERVICE PCNFS /FULL /PERMANENT**

```
Service: PCNFS

Port:             5151  Protocol:  TCP,UDP         Address:  0.0.0.0
Inactivity:          0  User_name: TCPIP$PCNFS     Process:  TCPIP
$PCNFSD
Limit:               1
```

```
File:         TCPIP$SYSTEM:TCPIP$PCNFSD_RUN.COM
Flags:        TCPIP Prot

Socket Opts:  Rcheck Scheck
 Receive:            0     Send:               0

Log Opts:     Acpt Actv Dactv Conn Error Exit Logi Mdfy Rjct TimO Addr
 File:        SYS$SYSDEVICE:[TCPIP$PCNFS]TCPIP$PCNFSD_STARTUP.LOG

RPC Opts
 Program number: 150001  Lowest version:     1    Highest version:     2


Security
 Reject msg:  not defined
 Accept host: 0.0.0.0
 Accept netw: 0.0.0.0
```

Shows the full configuration in the permanent database for PC-NFS. The RPC information shows that PC-NFS runs as program 150001; its lowest version number is 1 and its highest version number is 2.

4.
```
TCPIP> SHOW SERVICE PORTMAPPER
Service     Port      Protocol        Process     Address     State

PORTMAPPER   111       TCP,UDP      TCPIP$PORTM     0.0.0.0     Enabled
```

Monitors the Portmapper service process, showing that the service is enabled.

5.
```
TCPIP> SHOW SERVICE LBROKER /FULL /PERMANENT
Service: LBROKER

Port:             6570     Protocol:  UDP             Address:  0.0.0.0
Inactivity:          0     User_name: TCPIP$LD_BKR    Process:  TCPIP
$LBROKER
Limit:               1

File:         TCPIP$SYSTEM:TCPIP$LBROKER_RUN.COM
Flags:        None

Socket Opts:  Rcheck Scheck
 Receive:            0     Send:               0

Log Opts:     Acpt Actv Dactv Conn Error Exit Logi Logo Mdfy Rjct TimO
 Addr
 File:        SYS$SYSDEVICE:[TCPIP$LD_BKR]TCPIP$LBROKER_RUN.LOG

Security
 Reject msg:  not defined
 Accept host: 0.0.0.0
 Accept netw: 0.0.0.0
```

Displays the settings for cluster load balancing.

6.
```
TCPIP> SHOW SERVICE REXEC /FULL /PERMANENT

Service: REXEC
```

```
Port:           512  Protocol:  TCP              Address: 0.0.0.0
Inactivity:       5  User_name: not defined      Process: TCPIP$REXECD
Limit:            3

File:          TCPIP$SYSTEM:TCPIP$REXEC_RUN.COM
Flags:         Case Listen Rexe TCPIP

Socket Opts:   Rcheck Scheck
 Receive:          0     Send:               0

Log Opts:      Acpt Actv Dactv Error Exit Mdfy Rjct TimO Addr
 File:         TCPIP$REXEC.LOG

Separators:
 Port:   0    User_name: 0    Password:  0    Command:  0

Security
 Reject msg:  not defined
 Accept host: 0.0.0.0
 Accept netw: 0.0.0.0
```

Shows the full configuration in the permanent database for REXEC.

SHOW VERSION

SHOW VERSION — Displays the version of the TCP/IP Services software that is currently running, including individual components.

Syntax

```
SHOW VERSION [ /ALL ]
```

Qualifiers

/ALL

Optional. Default: TCP/IP Services version.

Displays the version of all running TCP/IP Services components.

Example

```
TCPIP> SHOW VERSION
HP TCP/IP Services for OpenVMS Alpha Version 5.4
    on an AlphaServer 1000 4/200 running OpenVMS V7.3-1
```

Displays the following information:

- Version of TCP/IP Services that is running.
- Model of hardware platform.
- Version of OpenVMS that is running.

START MAIL

START MAIL — Manually starts the SMTP sender queues (not the receiver [server]). *Related commands:* SHOW MAIL, ENABLE SERVICE SMTP, SHOW CONFIGURATION SMTP.

Syntax

```
START MAIL
```

Restrictions

Requires SYSPRV or BYPASS privilege.

SMTP consists of the sender and the receiver. Start the sender before you enable the receiver.

Do not issue this command unless SMTP has been configured (with the SET CONFIGURATION SMTP command).

Examples

```
TCPIP> START MAIL
```

Starts the SMTP sender.

(To start the SMTP sender when TCP/IP Services starts up, see the SET CONFIGURATION ENABLE SERVICE command. To start the SMTP receiver, see the ENABLE SERVICE command.)

START ROUTING

START ROUTING — Starts dynamic routing with ROUTED or GATED. If you want to change from one to the other, you must stop the current dynamic routing daemon then start the desired daemon. You cannot run both GATED and ROUTED at the same time. **Related commands:** STOP ROUTING, SET GATED.

Syntax

```
START ROUTING [/GATED ]
              [ /LOG ]
              [ /SUPPLY[=DEFAULT] ]
```

Qualifiers

/GATED

Optional.

Enables the gateway routing daemon (GATED).

If you enable dynamic GATED routing, you will be able to configure this host to use any combination of the following routing protocols to exchange dynamic routing information with other hosts on the network:

- RIP (Routing Information Protocol), Versions 1 and 2

- RDISC (Router Discovery Protocol)
- OSPF (Open Shortest Path First)
- EGP (Exterior Gateway Protocol)
- BGP (Border Gateway Protocol), BGP-4
- Static routes

/SUPPLY[=DEFAULT]

Optional. Applies only to ROUTED. Do not use with /GATED.

Broadcasts routing information to other hosts in 30-second intervals.

If you specify /SUPPLY=DEFAULT, the local host supplies the default network route.

/LOG

Optional. Applies to ROUTED. Do not use with /GATED.

Logs routing activity to SYS$SYSDEVICE:[TCPIP$ROUTED]TCPIP$ROUTED.LOG.

Default: No logging.

Examples

1. `TCPIP> START ROUTING /GATED`

   Starts GATED dynamic routing on the running system.

2. `TCPIP> START ROUTING`

   Interactively starts ROUTED dynamic routing on the running system.

3. `TCPIP> START ROUTING /SUPPLY`

   Immediately starts ROUTED dynamic routing. The local host both broadcasts and receives network routing information.

STOP ROUTING

STOP ROUTING — Stops dynamic routing. If GATED routing is used, stops dynamic routing but preserves GATED routes in the routing table. Use with SET NOROUTE when you require full manual control over the routing table. **Related command:** START ROUTING.

Syntax

```
STOP ROUTING [ /GATED ]
```

Qualifiers

/GATED

Optional.

Use to stop GATED dynamic routing and to remove all GATED routes from the routing table.

UNMAP

UNMAP — Makes unknown to the NFS server either a mapped (logically linked) OpenVMS disk or a container file system. Unmapping removes a logical file system, also called Network File System (NFS). Unmapped file systems are not accessible to remote users working on NFS clients. *Related commands:* MAP, SHOW MAP, SET CONFIGURATION NOMAP, ADD EXPORT, SHOW EXPORT, REMOVE EXPORT, MAP, UNMAP, SET CONFIGURATION MAP, SET CONFIGURATION NOMAP, SHOW MAP, SHOW CONFIGURATION MAP. *Applies to:* NFS server.

Syntax

```
UNMAP "/path/name" [ /[NO]CONFIRM ]
```

Restrictions

Requires SYSPRV and BYPASS privilege.

Parameters

"/path/name"

Required.

UNIX name of the file system to unmap.

You can use wildcards.

Qualifiers

/CONFIRM
/NOCONFIRM

Optional. Default: /CONFIRM if you use a wildcard.

Requests confirmation before unmapping each file system.

Examples

```
TCPIP> UNMAP "/disk_host"
```

Unmaps the NFS file system `/remote`, making it unavailable to client users.

ZERO NFS_SERVER

ZERO NFS_SERVER — Resets the NFS server performance counters. *Related commands:* SET NFS_SERVER, SHOW NFS_SERVER.

Syntax

```
ZERO NFS_SERVER [ /HOST=host ]
```

```
[ /SERVICES ]
[ /USER_NAME=vms_user_name ]
```

Restrictions

Requires SYSNAM and WORLD privileges.

Qualifiers

/HOST=*host*

Optional. Default: All users, all hosts.

With /USER_NAME, clears the counters relating to the specified users sharing the specified OpenVMS account.

/SERVICES

Optional. Default: NFS server services.

Resets the counters for the NFS server and the Mount and Portmapper services.

/USER_NAME=*vms_user_name*

Optional. Default: All users, all hosts.

With /HOST, clears the counters relating to the specified users sharing the specified OpenVMS account.

Do not specify a list of names; specify only a single name.

Examples

```
TCPIP> ZERO NFS_SERVER /USER_NAME=NESTING /HOST="pigeon"
```

Clears the NFS server counters for the remote NFS clients from host `pigeon` who use the OpenVMS account NESTING.