HP TCP/IP Services for OpenVMS

Release Notes

January 2005

This document describes the new features and changes introduced with Version 5.5 of the HP TCP/IP Services for OpenVMS software product.

Revision/Update Information:	This is a new document.
Software Version:	HP TCP/IP Services for OpenVMS Version 5.5
Operating Systems:	OpenVMS I64 Version 8.2 HP OpenVMS Alpha Version 8.2

Hewlett-Packard Company
Palo Alto, California

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Printed in the US

The HP TCP/IP Services for OpenVMS documentation is available on CD-ROM.

This document was prepared using DECdocument, Version 3.3-1b.

Contents

3 Restrictions and Limitations

4 Corrections

5 Documentation Update

Tables

Preface

The HP TCP/IP Services for OpenVMS product is the HP implementation of the TCP/IP protocol suite and internet services for OpenVMS Alpha and OpenVMS Industry Standard 64 for Integrity Servers (I64) systems. This document describes the latest release of the HP TCP/IP Services for OpenVMS product.

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

For installation instructions, see the *HP TCP/IP Services for OpenVMS Installation and Configuration* manual.

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.

Intended Audience

These release notes are intended for experienced OpenVMS and UNIX® system managers and assumes a working knowledge of OpenVMS system management, TCP/IP networking, TCP/IP terminology, and some familiarity with the TCP/IP Services product.

Document Structure

These release notes are organized into the following chapters:

- Chapter 1 describes new features and special changes to the software that enhances its observed behavior.
- Chapter 2 describes changes to the installation, configuration, and startup procedures, and includes other related information that is not included in the *HP TCP/IP Services for OpenVMS Installation and Configuration* manual.
- Chapter 3 describes information about problems and restrictions, and includes notes describing changes to particular commands or services.
- Chapter 4 describes problems identified in previous versions of TCP/IP Services that have been fixed.
- Chapter 5 describes updates to information in the TCP/IP Services product documentation.

Related Documents

Table 1 lists the documents available with this version of TCP/IP Services.

Table 1 TCP/IP Services Documentation

Manual	Contents
Compaq 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. This manual also describes the other manuals in the TCP/IP Services documentation set and provides a glossary of terms and acronyms for the TCP/IP Services software product.
HP 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.
HP TCP/IP Services for OpenVMS Installation and Configuration	This manual explains how to install and configure the TCP/IP Services product.
HP 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, e-mail, TELNET, TN3270, and network printing.
HP TCP/IP Services for OpenVMS Management	This manual describes how to configure and manage the TCP/IP Services product.
HP TCP/IP Services for OpenVMS Management Command Reference	This manual describes the TCP/IP Services management commands.
HP 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.
HP 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.
HP 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 RPC). This manual also describes the RPC programming interface and how to use the RPCGEN protocol compiler to create applications.
HP TCP/IP Services for OpenVMS Guide to SSH	This manual describes how to configure, set up, use, and manage the SSH for OpenVMS software.
HP TCP/IP Services for OpenVMS Sockets API and System Services Programming	This manual describes how to use the Berkeley Sockets API and OpenVMS system services to develop network applications.
HP 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.

(continued on next page)

Table 1 (Cont.) TCP/IP Services Documentation

Manual	Contents
HP 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. It also provides information about using UNIX network management utilities on OpenVMS.
HP 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.

For additional information about HP OpenVMS products and services, visit the following World Wide Web address:

`http://www.hp.com/go/openvms`

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.

Reader's Comments

HP welcomes your comments on this manual. Please send comments to either of the following addresses:

Internet	**openvmsdoc@hp.com**
Postal Mail	Hewlett-Packard Company OSSG Documentation Group, ZKO3-4/U08 110 Spit Brook Rd. Nashua, NH 03062-2698

How to Order Additional Documentation

For information about how to order additional documentation, visit the following World Wide Web address:

`http://www.hp.com/go/openvms/doc/order`

Conventions

In the product documentation, the name TCP/IP Services means any of the following:

- HP TCP/IP Services for OpenVMS Alpha
- HP TCP/IP Services for OpenVMS I64
- HP TCP/IP Services for OpenVMS VAX

In addition, please note that all IP addresses are fictitious.

The following conventions are used in the documentation.

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.) In the HTML version of this document, this convention appears as brackets, rather than 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 choices in parentheses if you specify 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 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 optional; 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 type	Bold type represents the introduction of a new term. It also represents the name of an argument, an attribute, or a reason.
italic type	Italic type 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 TYPE	Uppercase type indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.
`Example`	This typeface indicates code examples, command examples, and interactive screen displays. In text, this type also identifies URLs, UNIX commands and pathnames, PC-based commands and folders, and certain elements of the C programming language.
-	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.

1

New Features and Behavioral Enhancements

This chapter describes new features of TCP/IP Services Version 5.5 as well as behavioral enhancements.

Note

TCP/IP Services Version 5.5 is supported on OpenVMS Alpha and OpenVMS Industry Standard 64 for Integrity Servers (I64) systems only. On VAX systems, use TCP/IP Services Version 5.3.

To use TCP/IP Services Version 5.5, you must upgrade to OpenVMS Version 8.2 or higher.

For information about installing and configuring TCP/IP Services, see the *HP TCP/IP Services for OpenVMS Installation and Configuration* guide.

Table 1–1 lists the new features of TCP/IP Services Version 5.5 and the sections that describe them.

Table 1–1 TCP/IP Services for OpenVMS New Features

Feature	Section	Description
Support for HP Industry Standard 64 Server Platforms	1.1	TCP/IP Services runs on I64 platforms as well as Alpha platforms.
failSAFE IP Support for IPv6	1.2	failSAFE IP supports IPv6. This version of TCP/IP Services includes new `ifconfig` commands for managing failSAFE IP.
Secure IMAP	1.3	Secure IMAP uses the Secure Sockets Layer (SSL).
IPv6 Updates and Enhancements	1.4	Neighbor discovery and IPv6 APIs have been enhanced.
libpcap API Support	1.5	The `libpcap` application programming interface (API) is supported in this release of TCP/IP Services.
Support for Network Time Protocol (NTP) V4.2	1.6	NTP has been upgraded to Version 4.2 and supports IPv6.
SSH Features	1.7	SSH has been upgraded to Version 3.2 and supports IPv6.
TCPDUMP Version 3.8.3	1.8	TCPDUMP has been upgraded to Version 3.8.3.

(continued on next page)

Table 1–1 (Cont.) TCP/IP Services for OpenVMS New Features

Feature	Section	Description
Updated Header Files in TCPIP$EXAMPLES	1.9	Header files residing in TCPIP$EXAMPLES have been updated.

1.1 Support for HP Industry Standard 64 Server Platforms

TCP/IP Services now runs on HP Itanium®-based (I64) platforms, providing essentially the same functionality as on Alpha platforms. Further information about I64 support is provided in the following sections:

- Section 2.5, Adding a System to an OpenVMS Cluster
- Section 3.1, Restrictions on OpenVMS I64 Platforms

1.2 failSAFE IP Support for IPv6

The failSAFE IP service has been upgraded in this release. The IPv6 environment is now supported.

The `ifconfig` utility has been updated as well. For more information about this utility, enter the following command:

```
$ TCIPIP HELP IFCONFIG
```

For more information about using the `ifconfig` utility to monitor interface failover, refer to the *HP TCP/IP Services for OpenVMS Management* guide.

1.3 Secure IMAP

This release of TCP/IP Services includes secure IMAP, which supports the Secure Sockets Layer (SSL). Secure IMAP provides secure retrieval and management of messages. Secure IMAP accepts connections on port 993 and encrypts passwords, data, and IMAP commands. It is compatible with clients that use SSL, such as Outlook Express, Netscape, and Mozilla. To use this feature, you must download the HP SSL kit for OpenVMS Alpha from the HP OpenVMS Security web site:

```
http://h71000.www7.hp.com/openvms/security.html
```

If the HP SSL software is not installed, the IMAP server will communicate in non-SSL mode.

The SSL logical names are defined by the SSL startup procedure. Therefore, if you have IMAP configured to use SSL logical names for locating the certificate and key files, you must ensure that the SSL startup procedure is run before the TCP/IP Services startup procedure.

The secure IMAP configuration is controlled by the configuration file SYS$SYSDEVICE:[TCPIP$IMAP]TCPIP$IMAP.CONF.

Use the following new configuration options and logical names to manage secure IMAP:

- `SSL-Server-Port`

This option defines the port at which the IMAP server is available for SSL connections. By default, the port is 993. For example, the following setting specifies port 1004:

```
SSL-Server-Port:1004
```

- `Disable-Clear-Text`

  Enabling this option prevents the IMAP server from serving cleartext connections. Thus, client connection requests at port 143 will receive the following error message:

  ```
  The IMAP server serves ONLY SSL client requests. Please reconfigure your
  client for SSL.
  ```

  For example, the following setting enables this option:

  ```
  Disable-Clear-Text:YES
  ```

- `Disable-SSL`

  Enabling this option prevents IMAP from serving SSL client connections. For example, the following setting enables this option:

  ```
  Disable-SSL:YES
  ```

- `TCPIP$IMAP_CERT_FILE`

  This logical name specifies the name of the certificate file that IMAP uses for SSL. If it is not defined, the default is `SSL$CERTS:SERVER.CRT`.

  You can specify the full or partial file specification as the value assigned to this logical name. That is, you can specify the directory, the file name, or both. The part of the file specification that you do not specify is supplied from the default.

  For example, the following command changes the file name to TCPIP$IMAP.CRT:

  ```
  $ DEFINE/SYSTEM/EXECUTIVE_MODE TCPIP$IMAP_CERT_FILE SSL$CERTS:TCPIP$IMAP.CRT
  ```

- `TCPIP$IMAP_KEY_FILE`

  This logical name specifies the name of the key file that IMAP uses for SSL. If it is not defined, the default is `SSL$CERTS:SERVER.KEY`.

  You can specify the full or partial file specification as the value. That is, you can specify the directory, the file name, or both. The part of the file specification that you do not specify is supplied from the default.

  For example, the following command changes the file name to TCPIP$IMAP.KEY:

  ```
  $ DEFINE/SYSTEM/EXECUTIVE_MODE TCPIP$IMAP_KEY_FILE SSL$KEY:TCPIP$IMAP.KEY
  ```

To ensure that logical names and configuration options take effect, you should stop the IMAP server before you change them.

1.4 IPv6 Updates and Enhancements

The following sections describe updates and enhancements to IPv6 (Internet Protocol Version 6) functionality.

For additional information about IPv6 changes, see Section 4.7.

1.4.1 IPv6 Configuration Enhancements

The support for IPv6 has been enhanced to include dynamic updates for the ip6.arpa zone and updates to the IPv6 APIs.

1.4.2 Neighbor Discovery Supports Dynamic Update Requests for ip6.arpa DNS Reverse Zone

Neighbor Discovery (TCPIP$ND6HOST process) now supports RFC 3152 and can be configured to send dynamic update requests for the ip6.arpa DNS reverse zone only.

Previously, the Neighbor Discovery daemon would send dynamic update requests for the ip6.int DNS reverse zone only. (The ip6.int reverse zone is being deprecated.)

If you need to support delegations based on the ip6.int zone, make sure that the ip6.int zone gets populated correctly. For more information, refer to Section 3.1.3, Using DNAME To Rename ip6.int, in the *HP TCP/IP Services for OpenVMS Guide to IPv6*.

To update the zone, TCPIP$ND6HOST sends dynamic updates to the primary master name server. The name of the primary master name server is stored in the MNAME field of the SOA record for a zone. To determine the master name server, TCPIP$ND6HOST sends a query for the zone's SOA record to the name server specified in the DNS resolver configuration. To display the DNS resolver configuration information, use the TCP/IP management command SHOW NAME_SERVICE.

To make use of this feature, you must enable dynamic updates. By default, dynamic updates are rejected by DNS servers. For information about allowing dynamic updates, see the BIND Chapter in the *HP TCP/IP Services for OpenVMS Management* guide.

1.4.3 IPv6 Application Programming Interface (API) Updates

The IPv6 programming APIs were updated with TCP/IP Services Version 5.4, and new programming examples were provided.

For more information about using the IPv6 APIs, refer to the *HP TCP/IP Services for OpenVMS Sockets API and System Services Programming* guide.

The following is a list of the specific changes affecting the IPv6 APIs introduced with TCP/IP Services Version 5.4:

- IPv6 Changes:
  - The flag value AI_DEFAULT, which could previously be specified in the ai_flags parameter for a call to the getaddrinfo function, has been deprecated. It will be removed from the NETDB.H file in a future release. To achieve the behavior defined by this flag, specify the logical OR of the flag values AI_V4MAPPED and AI_ADDRCONFIG.
  - The BIND resolver has been updated as described in the following RFC draft:

    ```
    draft-ietf-ipngwg-scoping-arch-04.txt
    ```

This change allows the specification of an IPv6 nonglobal address without ambiguity by also specifying an intended scope zone. The format is as follows:

address%zone_id

The format of the nonglobal address includes the following:

- *address* is a literal IPv6 address
- *zone_id* is a string to identify the zone of the address
- % is a delimiter character to distinguish between the address and zone identifier.

For example, the following specifies a nonglobal address on interface WE0:

```
fe80::1234%WE0
```

- The IPv4 TCP and UDP client and server C socket programming example programs that reside in SYS$COMMON:[SYSHLP.EXAMPLES.TCPIP] have been ported to IPv6. The IPv6 versions of these example programs are located in SYS$COMMON:[SYSHLP.EXAMPLES.TCPIP.IPV6].
- The IPv6 example database and configuration files in SYS$COMMON:[SYSHLP.EXAMPLES.TCPIP.IPV6.BIND] have been updated to reflect current practice.

As noted in the *HP TCP/IP Services for OpenVMS Sockets API and System Services Programming* guide, several programming functions provided in earlier Early Adopter Kits (EAKs) were deprecated. These programming functions will no longer be supported in versions of TCP/IP Services higher than Version 5.5. Do not use these functions if you are developing new applications.

The following table lists the functions and their replacements. If your existing applications use these functions, see the *HP TCP/IP Services for OpenVMS Sockets API and System Services Programming* for changes you should make to your code.

Deprecated Function	Replacement Function
getipnodebyname	getaddrinfo
getipnodebyaddr	getnameinfo
freehostent	freeaddrinfo

1.5 libpcap API Support

The libpcap API (Version 0.8.3) is supported with this release. An example program resides in the directory associated with the logical name TCPIP$LIBPCAP_EXAMPLES. Also included in that directory is a comprehensive documentation file, $$TCPIP$LIBPCAP_DOCUMENTATION.HTML. The libpcap sharable image that implements the libpcap functions, TCPIP$LIBCAP_SHR.EXE, is in the directory associated with the logical name SYS$SHARE.

1.6 Support for Network Time Protocol (NTP) V4.2

This version of TCP/IP Services supports NTP Version 4.2.0. This release retains backward compatibility with NTP Version 3 and NTP Version 2, but not with NTP Version 1. Support for NTP Version 1 has been discontinued because of security vulnerabilities.

This release includes support for the IPv6 address family in addition to support for the IPv4 address family. Either or both families can be used at the same time on the same system.

Configuration options that previously supported the use of the IPv4 address family now accept the IPv6 address family. To use this feature, you must enable IPv6 on TCP/IP Services, as described in the *HP TCP/IP Services for OpenVMS Installation and Configuration* guide.

1.6.1 Cryptography Support

This release supports authentication using symmetric key cryptography. Support for autokey public key cryptography is not available with this release. For more information about symmetric key cryptography, see Section 1.6.7 and the NTP chapter in the *HP TCP/IP Services for OpenVMS Management* guide.

1.6.2 Using NTP Version 4.2.0 with Berkeley Internet Name Domain (BIND)

When using NTP on an IPv6-enabled system, if both IPv4 and IPv6 addresses are associated with the same domain name in the DNS, the BIND resolver uses the IPv6 address for a host specified in TCPIP$NTP.CONF.

1.6.3 Using the NTPDC Utility

Versions of NTPDC provided prior to this release of TCP/IP Services are not IPv6-capable and will only show IPv4 associations when you use the following commands:

- `peers`
- `dmpeers`
- `listpeers`
- `monlist`
- `pstats`
- `reslist`
- `showpeer`

1.6.4 Using the NTPQ Utility

Versions of NTPQ provided prior to this release of TCP/IP Services are not IPv6-capable and will show 0.0.0.0 for IPv6 associations when you use the following commands:

- `peers`
- `lopeers`
- `lpassociations`
- `lpeers`
- `opeers`

- passociations
- pstatus

1.6.5 Using the NTPTRACE Utility

The NTPTRACE utility has not been updated to NTP Version 4.2.0 and works with the IPv4 address family only.

1.6.6 NTP Packet Headers with IPv6

The reference ID field of the NTP packet header changes when operating with IPv6 associations. For IPv4 associations, this field contains the 32-bit IPv4 address of the server. For IPv6 associations, this field contains the first 32 bits of an MD5 hash formed from the address. As a result, when the association is an IPv6 host, the peers command and other similiar commands with NTPQ included in this release will show the refid field containing a random number formatted as an IPv4 address.

1.6.7 NTP_GENKEYS Utility Replaced by NTP_KEYGEN

With this version of TCP/IP Services, the NTP_GENKEYS utility has been replaced by the new NTP_KEYGEN utility. Use the NTP_KEYGEN utility to generate random keys used by NTP Version 3 and NTP Version 4 symmetric key authentication.

Use the -M command line option to have the program generate a TCPIP$NTPKEY_MD5KEY_*hostname.timestamp* file containing 16 random symmetric keys. In the command line, you must enclose the -M in quotation marks to preserve uppercase, as shown in the following example:

```
$ @SYS$MANAGER:TCPIP$DEFINE_COMMANDS.COM
$ ntp_keygen -"M"
```

The host name (*hostname*, as returned by the gethostname() function) and a timestamp are used as part of the file name. Because the algorithm to produce the timestamp is seeded by the system clock, each run of the program produces a different file name.

The TCPIP$NTPKEY_MD5KEY_*hostname.timestamp* file contains 16 MD5 keys. Each key consists of 15 characters selected at random from the ASCII 95-character printing subset. The file is read by the NTP server at the location specified by the keys command in the TCPIP$NTP.CONF configuration file. An additional key consisting of an easily remembered password should be added manually for use with the NTPQ and NTPDC programs. The file must be distributed by secure means to other servers and clients that share the same security compartment. The key identifier for the MD5 program uses only identifiers 1 through 16. The key identifier for each association is specified in the server or peer configuration file command.

1.6.8 NTP Clock Synchronization Enhancement

The NTP slew mechanism for gradually adjusting a clock has been enhanced to facilitate synchronization for offsets of one second or larger. The maximum slew value (the maximum amount that NTP will adjust the clock in one attempt) has been modified to enable the clock to synchronize more quickly for such offsets. NTP now takes only 20 seconds to correct a one-second offset, compared to approximately 30 minutes for earlier versions of NTP.

For clock offsets that are less than one second, the slew mechanism has not been modified.

1.7 SSH Features

The following sections describe new developments in the SSH service.

1.7.1 SSH Upgrade to Version 3.2

The SSH service has been upgraded to Version 3.2. This upgrade introduces changes to the SSH utilities. For more information about the SSH utilities, use the -h flag on the utility command line. For example:

```
$ SSH -h
```

1.7.2 SSH Supports IPv6

The version of SSH in the current release of TCP/IP Services supports IPv6 environments.

In order for SSH to work in the IPv6 environment, the service must be set to IPv6. To display the setting for SSH, enter the following commands:

```
$ TCPIP
TCPIP> SHOW SERVICE SSH /FULL
```

If the IPv6 flag is not included, enter the following command:

```
TCPIP> SET SERVICE SSH /FLAG=IPV6
```

1.7.3 SSH Port Forwarding

SSH for OpenVMS supports UNIX-like port forwarding commands, including the -x and +x flags, as well as the ForwardX11 configuration keyword. For more information about using SSH port forwarding, see:

- Section 4.14.4
- Table 5–2

1.7.4 SSH File Transfers

The maximum file size for SSH file copy operations has been increased from 4 megabytes to 4 gigabytes. In addition, the speed of file transfers has increased significantly, depending on available resources, CPU, network conditions, and so forth. For specific restrictions, see Section 3.11.13.

1.7.5 SSH Batch Jobs

With this version of TCP/IP Services, you can use SSH commands in batch jobs. For specific restrictions in the use of batch jobs for SSH sessions, see Section 3.11.10.

1.8 TCPDUMP Version 3.8.3

This release of TCP/IP Services includes an upgrade to the TCPDUMP utility. Upgraded from Version 2.2 to Version 3.8.3, TCPDUMP uses the libpcap Version 0.8.3 API. For more information about the changes in the new version of TCPDUMP, see the www.tcpdump.org web site, or type TCPIP HELP TCPDUMP to get information about the new version.

The libpcap API is provided for early adopters. For more information, refer to Section 1.5.

1.9 Updated Header Files in TCPIP$EXAMPLES

Several header files that reside in TCPIP$EXAMPLES have been updated with this release of TCP/IP Services. The updates are prompted by:

- Recent changes to Internet Engineering Task Force (IETF) Request for Comments (RFCs)
- Performance considerations (better base alignment of structures)
- Internal changes to TCP/IP Services

Backward compatibility is not assured.

The updated header files are:

- IF.H
- IF_TYPES.H
- IN.H
- IN6.H
- SOCKET.H
- STROPTS.H
- TCP.H
- PCAP.H
- PCAP-PDF.H
- __DECC_INCLUDE_PROLOGUE.H

2

Installation, Configuration, Startup, and Shutdown

This chapter includes notes and changes made to the installation and configuration of TCP/IP Services, as well as startup and shutdown procedures. Use this chapter in conjunction with the *HP TCP/IP Services for OpenVMS Installation and Configuration* manual.

Note

To use TCP/IP Services Version 5.5, you must upgrade to OpenVMS Version 8.2.

2.1 Installing Over V5.3 Early Adopter's Kits (EAKs)

If you have installed one or more of the following V5.3 EAKs, you must use the PCSI REMOVE command to remove the EAKs before you install TCP/IP Services V5.5:

- SSH for OpenVMS EAK
- failSAFE IP EAK

Note

If you install the current TCP/IP Services version after removing the failSAFE IP EAK, you must run TCPIP$CONFIG.COM to reestablish your target and home interfaces.

2.2 Upgrading from TCP/IP Services Version 4.*x*

The following sections describe how to preserve the behavior of the software when you upgrade from an older version of TCP/IP Services (UCX) to the current version.

Note

In the next version of TCP/IP Services, the capability of upgrading directly from any version of TCP/IP Services prior to 5.0 will be removed. Version 5.5 of TCP/IP Services is the last release that includes this capability.

2.2.1 Upgrading LPD

When you merge edits into the system startup command procedure, do not include the commands to start and stop the queue UCX$LPD_QUEUE. This queue has been replaced with TCPIP$LPD_QUEUE. The commands for starting and stopping TCPIP$LPD_QUEUE are in the LPD startup and shutdown command procedures.

After you merge the edits, modify the value of the /PROCESSOR qualifier in the LPD client queue startup commands that you have just appended, replacing UCX$LPD_SMB with TCPIP$LPD_SMB. For example, enter the following command:

```
LSE Command> SUBSTITUTE/ALL "ucx$lpd_smb" "tcpip$lpd_smb"
```

2.2.2 Preserving SNMP Startup and Shutdown Behavior

After you upgrade to the current version of TCP/IP Services, you must perform one of the following actions to ensure correct SNMP startup:

- If SNMP was configured under an old TCP/IP Services installation (UCX) and you want to retain the previous configuration, run the SYS$MANAGER:TCPIP$CONFIG.COM configuration procedure and select the option to automatically convert UCX configuration files.
- After you upgrade to the current version of TCP/IP Services, run the SYS$MANAGER:TCPIP$CONFIG.COM configuration procedure. If SNMP is still enabled, disable SNMP then enable it again. This is necessary for the proper operation of this component.

If you have customized versions of the UCX$SNMP_STARTUP.COM and UCX$SNMP_SHUTDOWN.COM command procedures (used to start and stop extension subagents), save your customized files to a different directory before upgrading to the new version of TCP/IP Services. If you do not perform this step, your customized changes will be lost.

Check for versions of these files in the following locations:

- SYS$MANAGER
- SYS$STARTUP
- SYS$SYSDEVICE:[UCX$SNMP]

After you install TCP/IP Services, manually enter commands into the TCPIP$SNMP_SYSTARTUP.COM and TCPIP$SNMP_SYSHUTDOWN.COM command procedures, as described in the *HP TCP/IP Services for OpenVMS Management* guide.

2.2.3 Customizing SNMP Startup and Shutdown

Enabling SNMP using the TCPIP$CONFIG.COM configuration procedure no longer creates the following files:

- TCPIP$SNMP_SYSTARTUP.COM
- TCPIP$SNMP_SYSHUTDOWN.COM

These command procedures are used for starting and stopping custom SNMP subagents. They will not be affected by installing future versions of TCP/IP Services.

2.2.4 SNMP Messages When You Install TCP/IP Services

For sites where the same version of TCP/IP Services is installed multiple times, informational messages similar to the following may appear in the installation dialog:

```
Do you want to review the options? [NO]

Execution phase starting ...

The following product will be installed to destination:
    DEC AXPVMS TCPIP V5.3-9I               DISK$AXPVMSSYS:[VMS$COMMON.]
The following product will be removed from destination:
    DEC AXPVMS TCPIP V5.3-9H               DISK$AXPVMSSYS:[VMS$COMMON.]
%PCSI-I-RETAIN, file [SYSEXE]TCPIP$ESNMP_SERVER.EXE was not replaced because
file from kit does not have higher generation number
%PCSI-I-RETAIN, file [SYSEXE]TCPIP$HR_MIB.EXE was not replaced because file
from kit does not have higher generation number
%PCSI-I-RETAIN, file [SYSEXE]TCPIP$OS_MIBS.EXE was not replaced because file
from kit does not have higher generation number
%PCSI-I-RETAIN, file [SYSLIB]TCPIP$ESNMP_SHR.EXE was not replaced because file
from kit does not have higher generation number
%PCSI-I-RETAIN, file [SYSLIB]UCX$ESNMP_SHR.EXE was not replaced because file
from kit does not have higher generation number
```

You can ignore these messages.

2.2.5 SNMP Subagent Startup Messages

The SNMP startup procedure can produce the following error messages in subagent log files:

```
25-JUL-2004 14:13:32.47 **ERROR ESNMP_INIT.C line 3777: Could not
connect to master: connection refused
25-JUL-2004 14:13:32.94 WARNING OS_MIBS.C line 942: Master agent
cannot be reached.  Waiting to attempt reconnect.
```

These messages are the result of a timing problem and can be ignored.

2.3 Installation Changes

The following changes have been made to the installation:

- The scalable kernel, which optimizes TCP/IP performance on symmetric multiprocessing (SMP) systems, was optional in the previous release of TCP/IP Services. The scalable kernel now replaces the standard kernel.
- For each of the following components, Version 5.4 of TCP/IP Services provided two images, a conventional image plus an alternate Symmetric MultiProcessing (SMP) image that had a "_PERF" suffix (for example, TCPIP$INTERNET_SERVICES_PERF.EXE).
  - SYS$LOADABLE_IMAGES:TCPIP$BGDRIVER_PERF.EXE
  - SYS$LOADABLE_IMAGES:TCPIP$INTERNET_SERVICES_PERF.EXE
  - SYS$LOADABLE_IMAGES:TCPIP$TNDRIVER_PERF.EXE
  - SYS$SYSTEM:TCPIP$INETACP_PERF.EXE

  The logical name TCPIP$STARTUP_CPU_IMAGES was used to select the alternate Symmetric MultiProcessing (SMP) images.

With TCP/IP Services Version 5.5 there is no need for the alternate (_PERF) images. The logical name TCPIP$STARTUP_CPU_IMAGES is now ignored. HP recommends that you remove the the definition of that logical from the SYS$MANAGER:SYLOGICALS.COM command procedure or any other command procedures.

2.4 Image Identification and Link Dates

Executable images provided by TCP/IP Services typically have an image identification in the format V5.5-*xxaa*, where *xx* is a positive integer, and *aa* is zero or more letters signifying the revision level. In addition, the link dates of images on the kit typically are within a few hours of each other.

Several images on the latest TCP/IP Services kit do not follow this practice. The exceptions are documented here to help you ascertain that your product is correctly installed.

The following images use the identification format V5.5-*xxaa* PF. The "PF" indicates that the image is an improved variant.

- TCPIP$BGDRIVER.EXE
- TCPIP$INTERNET_SERVICES.EXE
- TCPIP$INETACP.EXE

The link dates of these images should be within an hour or so of each other.

With installations on OpenVMS Alpha systems, the following files do not follow the identification and link date conventions, as shown:

```
TCPIP$CFS_SHR         V5.5-6A        27-MAR-2004  SYS$COMMON:[SYSLIB]
TCPIP$NTPTRACE.EXE    V5.5           30-MAR-2004  SYS$COMMON:[SYSEXE]
TCPIP$TELNET_SERVER   V5.4/KRB V2.0   9-JUL-2003  SYS$COMMON:[SYSEXE]
```

With installations on OpenVMS I64 systems, the following files do not follow the identification and link date conventions, as shown:

```
SYS$COMMON:[SYSLIB]TCPIP$CFS_SHR.EXE
"V1.0"
10-MAY-2003 13:12:22.14

SYS$COMMON:[SYSEXE]TCPIP$NTPTRACE.EXE
"V5.5"
30-MAR-2004 23:22:14.46

SYS$COMMON:[SYSEXE]TCPIP$TELNET_SERVER.EXE
"V5.4/KRB V2.0"
 5-DEC-2003 00:21:54.16
```

2.5 Adding a System to an OpenVMS Cluster

The TCPIP$CONFIG.COM configuration procedure for TCP/IP Services Version 5.5 creates OpenVMS accounts using larger system parameter values than in previous versions. Only new accounts get these larger values. These values are useful on OpenVMS Alpha systems but essential on OpenVMS I64 systems.

To have your OpenVMS I64 system join an OpenVMS Cluster as a TCP/IP host, HP recommends adding the system to the cluster before you configure TCP/IP Services. The guidelines in Section 2.5.1 assume you have followed this recommendation.

If you configure TCP/IP Services before you add the system to a cluster, see Section 2.5.2.

2.5.1 Running a Newly Configured Host on the Cluster

The following recommendations assume you are configuring TCP/IP Services on the system after having added the system to the OpenVMS Cluster.

If TCP/IP Services has previously been installed on the cluster and you encounter problems running a TCP/IP component on the system, modify the cluster System Authorization File (SYSUAF) to raise the parameter values for the account used by the affected component. The minimum recommended values are listed in Table 2–1.

Table 2–1 Minimum Values for SYSUAF Parameters

Parameter	Minimum Value
ASTLM	100
BIOLM	400
BYTLM	108000
DIOLM	50
ENQLM	100
FILLM	100
PGFLQUOTA[1]	50000
TQELM	50
WSEXTENT	4000
WSQUOTA	1024

[1] This parameter's value setting is especially critical.

The IMAP, DHCP, and XDM components can exhibit account parameter problems if the value assigned to PGFLQUOTA or to any of the other listed parameters is too low. Use the OpenVMS AUTHORIZE utility to modify SYSUAF parameters. For more information, see *HP OpenVMS System Management Utilities Reference Manual: A-L*.

2.5.2 Configuring TCP/IP Services Before Adding the System to the Cluster

If you configure TCP/IP Services before you add the system to a cluster, when you add the system to the cluster the owning UIC for each of the TCP/IP service SYS$LOGIN directories (TCPIP$*service-name*, where *service-name* is the name of the service) may be incorrect. Use the OpenVMS AUTHORIZE utility to correct these UICs.

2.6 TCPIP$CONFIG.COM Changes

The following sections describe changes to the TCPIP$CONFIG.COM configuration procedure in this release.

2.6.1 Warning Message in TCPIP$CONFIG.COM

If you have run the TCPIP$IP6_SETUP.COM configuration procedure to enable IPv6, then when you run the TCPIP$CONFIG.COM configuration procedure, the following warning message appears when you select the Core environment option:

```
                              WARNING

This node has been configured for IPv6.  If you make any additional
changes to the configuration of the interfaces, you must run
TCPIP$IP6_SETUP again and update your host name information in
BIND/DNS for the changes to take effect.
```

2.6.2 Disabling or Enabling SSH Server

When you use the TCPIP$CONFIG.COM configuration procedure to disable or enable the SSH server, the following prompt is displayed:

```
* Create a new default Server host key? [YES]:
```

Unless you have a specific reason for creating a new default server host key, you should enter "N" at this prompt. If you accept the default, clients with the old key will need to obtain the new key. For more information, see Section 3.11.6.

2.7 SSH Configuration Files Must Be Updated

The SSH client and server on this version of TCP/IP Services cannot use configuration files from previous versions of SSH.

If the SSH client and server detect systemwide configuration files from an older version of SSH, the client and server will fail to start. The client will display the following warning message, and the server will write the following warning message to the SSH_RUN.LOG file:

```
You may have an old style configuration file. Please follow the
instructions in the release notes to use the new configuration
files.
```

If the SSH client detects a user-specific configuration file from an older version of SSH, the SSH client will display the warning and will allow the user to proceed.

To preserve the modifications made to the SSH server configuration file and the SSH client configuration file, you must edit the templates provided with the new version of SSH, as follows:

1. Extract the template files using the following commands:

   ```
   $ LIBRARY/EXTRACT=SSH2_CONFIG SYS$LIBRARY:TCPIP$TEMPLATES.TLB -
   _$ /OUT=TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2]SSH2_CONFIG.
   ```

   ```
   $ LIBRARY/EXTRACT=SSHD2_CONFIG SYS$LIBRARY:TCPIP$TEMPLATES.TLB -
   _$ /OUT=TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2]SSHD2_CONFIG.
   ```

   These commands copy the new template files into the SSH2 configuration directory with a new version number.

2. Copy the modifications made in the old versions of the configuration files to the new versions.

3. Start SSH using the following command:

   ```
   $ @SYS$STARTUP:SSH_STARTUP.COM
   $ @SYS$STARTUP:SSH_CLIENT_STARTUP.COM
   ```

2.8 Troubleshooting SMTP and LPD Shutdown Problems

If SMTP or LPD shutdown generates errors indicating that the queue manager is not running, check your site-specific shutdown command procedure (VMS_SYSHUTDOWN.COM). If this procedure contains the command to stop the queue manager (STOP/QUEUE/MANAGER), make sure this command is after the command that runs the TCPIP$SHUTDOWN.COM command procedure.

Note

You do not have to stop the queue manager explicitly. The queue manager is automatically stopped and started when you restart the system.

3 Restrictions and Limitations

This chapter provides information about problems and restrictions in the current version of TCP/IP Services, and also includes other information specific to a particular command or service, such as changes in command syntax or messages.

3.1 Restrictions on OpenVMS I64 Platforms

The following restrictions apply to OpenVMS I64 platforms only:

- The output of the TCP/IP management command SHOW VERSION/ALL differs from the output seen on OpenVMS Alpha and VAX systems. (The information is listed in one column, and the image name and location are combined.)
- The following components do not work on OpenVMS I64 platforms for this release:
  - NFS server
  - PPP

3.2 NFS Server Does Not Run on I64 Platforms

The NFS server is fully functional on Alpha platforms, but it does not work on I64 platforms for this release. This problem will be fixed in a future update to TCP/IP Services.

On I64 systems, NFS-related components are installed so that the NFS server can be provided in a future release. However, these components will not run. Attempts to start them will fail or result in an error.

You can configure the NFS server on I64 systems, but you cannot successfully start the server. The server will immediately exit. The associated TCP/IP management commands will return errors. For example:

```
TCPIP> SHOW MAP
%LIB-E-KEYNOTFOU, key not found in tree
%TCPIP-E-CFSERROR, error processing TCPIP file system request
-TCPIP-E-NOCFS, error resolving TCPIP$CFS_SHR entry point
-LIB-F-KEYNOTFOU, key not found in tree

TCPIP> MAP "/x" DKA100:
%LIB-E-KEYNOTFOU, key not found in tree
%TCPIP-E-MAPERROR, error processing MAP or UNMAP request
-TCPIP-E-NOCFS, error resolving TCPIP$CFS_SHR entry point
-LIB-F-KEYNOTFOU, key not found in tree
```

Other errors related to the NFS server may also occur.

For information about NFS restrictions on Alpha platforms, refer to Section 3.8.

3.3 PPP Restrictions

The point-to-point protocol (PPP) does not work in this release of TCP/IP Services, on both Alpha and I64 platforms. This problem will be fixed in a future update to TCP/IP Services.

If you try to use PPP, the following type of error can occur:

```
$ PPPD CONN TTA08
%PPPD-I-CONNECTTERM, converting connection on device _TTA0: to a Point-to-Point connection

%LIB-E-ACTIMAGE, error activating image
DKA0:[SYS0.SYSCOMMON.][SYSLIB]TCPIP$PPPD_CALLOUT.EXE;1
-SYSTEM-F-PRIVINSTALL, shareable images must be installed to run privileged image

%PPPD-E-PROTOERR, error initiating network protocol callback routine

%SYSTEM-F-PRIVINSTALL, shareable images must be installed to run privileged image
%PPPD-F-ABORT, fatal error encountered; operation terminated.
```

If you manually install the image TCPIP$PPPD_CALLOUT.EXE, the system will fail when you execute the command in this example. This is not a security problem because privileges are required in order to install images manually.

3.4 SLIP Restrictions

The serial line IP protocol (SLIP) does not work in this release of TCP/IP Services, on both Alpha and I64 platforms. This problem will be fixed on Alpha platforms in a future update to TCP/IP Services.

3.5 Advanced Programming Environment Restrictions and Guidelines

The header files provided in TCPIP$EXAMPLES are provided as part of the advanced TCP/IP programming environment. The following list describes restrictions and guidelines for using them:

- Use of the functions and data structures described in TCPIP$EXAMPLES:RESOLV.H is limited to 32-bit pointers. The underlying implementation will only handle 32-bit pointers. Previously, 64-bit pointers were wrongly accepted, resulting in undefined behavior for the underlying implementation.
- The IP.H and IP6.H header files are incomplete in the OpenVMS environment. They contain include directives for header files that are not provided in this version of TCP/IP Services. Refer to the *HP TCP/IP Services for OpenVMS Sockets API and System Services Programming* for more information.

3.6 BIND/DNS Restrictions

BIND Version 9 has the following restrictions:

- Certain DNS server implementations do not support AAAA (IPv6 address) records. When queried for an AAAA (IPv6) record type by the BIND resolver, these name servers will return an NXDOMAIN status, even if an A (IPv4) record exists for the same domain name. These name servers should be returning NOERROR as the status for such a query. This problem can result in delays during host name resolution.

BIND Version 9.2.1, which is supported with this release of TCP/IP Services, and prior versions of BIND do not exhibit this problem.

- Serving secure zones

  When acting as an authoritative name server, BIND Version 9 includes KEY, SIG, and NXT records in responses as specified in RFC 2535 when the request has the DO flag set in the query.

  Response generation for wildcard records in secure zones is not fully supported. Responses indicating the nonexistence of a name include an NXT record proving the nonexistence of the name itself, but do not include any NXT records to prove the nonexistence of a matching wildcard record. Positive responses resulting from wildcard expansion do not include the NXT records to prove the nonexistence of a non-wildcard match or a more specific wildcard match.

- Secure resolution

  Basic support for validation of DNSSEC signatures in responses has been implemented but should be considered experimental.

  When acting as a caching name server, BIND Version 9 is capable of performing basic DNSSEC validation of positive as well as nonexistence responses. You can enable this functionality by including a `trusted-keys` clause containing the top-level zone key of the DNSSEC tree in the configuration file.

  Validation of wildcard responses is not currently supported. In particular, a “`name does not exist`” response will validate successfully even if the server does not contain the NXT records to prove the nonexistence of a matching wildcard.

  Proof of insecure status for insecure zones delegated from secure zones works when the zones are completely insecure. Privately secured zones delegated from secure zones will not work in all cases, such as when the privately secured zone is served by the same server as an ancestor (but not parent) zone.

  Handling of the CD bit in queries is now fully implemented. Validation is not attempted for recursive queries if CD is set.

- Secure dynamic update

  Dynamic updating of secure zones has been partially implemented. Affected NXT and SIG records are updated by the server when an update occurs. Use the `update-policy` statement in the zone definition for advanced access control.

- Secure zone transfers

  BIND Version 9 does not implement the zone transfer security mechanisms of RFC 2535 because they are considered inferior to the use of TSIG or SIG(0) to ensure the integrity of zone transfers.

3.7 IPv6 Restrictions

The following sections describe restrictions in the use of IPv6.

3.7.1 Mobile IPv6 Restrictions

Mobile IPv6 is not supported in this release.

3.7.2 IPv6 Requires the BIND Resolver

If you are using IPv6, you must enable the BIND resolver. To enable the BIND resolver, use the TCPIP$CONFIG.COM command procedure. From the `Core environment` menu, select BIND Resolver.

You must specify the BIND server to enable the BIND resolver. If you do not have access to a BIND server, specify the node address 127.0.0.1 as your BIND server.

3.8 NFS Restrictions on Alpha Platforms

The following sections describe problems and restrictions with NFS on Alpha platforms.

3.8.1 NFS Server Problems and Restrictions

The NFS server is not supported on I64 platforms for this release. The associated components (lock manager, PCNFS, MOUNT server, `nfsstat`) are also unsupported. For other information specific to the NFS server on OpenVMS I64 systems, see Section 3.1.

The following restrictions apply to the NFS server on OpenVMS Alpha systems:

- Using the `ls` command from a Solaris Version 9 client may hang the OpenVMS server with no error message on either client or server. To avoid this problem, set the `nfs` subsystem attribute `ovms_xqp_plus_enabled` to 7. Refer to the *HP TCP/IP Services for OpenVMS Management* guide for more information about this attribute.
- When performing a mount operation or starting the NFS server with OPCOM enabled, the TCP/IP Services MOUNT server can erroneously display the following message:

  ```
  %TCPIP-E-NFS_BFSCAL, operation MOUNT_POINT failed on file /dev/dir
  ```

  This message appears even when the MOUNT or NFS startup has successfully completed. In the case of a mount operation, if it has actually succeeded, the following message will also be displayed:

  ```
  %TCPIP-S-NFS_MNTSUC, mounted file system /dev/dir
  ```

- If the NFS server and the NFS client are in different domains and unqualified host names are used in requests, the lock server (LOCKD) fails to honor the request and leaves the file unlocked.

  When the server attempts to look up a host using its unqualified host name (for example, `johnws`) instead of the fully qualified host name (for example, `johnws.abc com`), and the host is not in the same domain as the server, the request fails.

  To solve this type of problem, you can do one of the following:

  – When you configure the NFS client, specify the fully qualified host name, including the domain name. This ensures that translation will succeed.

  - Add an entry to the NFS server’s hosts database for the client’s unqualified host name. Only that NFS server will be able to translate this host name. This solution will not work if the client obtains its address dynamically from DHCP.

3.8.2 NFS Client Problems and Restrictions

- To get proper timestamps, when the system time is changed for daylight savings time (DST), dismount all DNFS devices. (The TCP/IP management command SHOW MOUNT should show zero mounted devices.) Then remount the devices.
- The NFS client does not properly handle file names with the semicolon character on ODS-5 disk volumes. (For example, `a^;b.dat;5` is a valid file name.) Such file names are truncated at the semicolon.
- The NFS client included with TCP/IP Services uses the NFS Version 2 protocol only.
- With the NFS Version 2 protocol, the value of the file size is limited to 32 bits.
- The ISO Latin-1 character set is supported. The UCS-2 characters are not supported.
- File names, including file extensions, can be no more than 236 characters long.
- Files containing characters not accepted by ODS-5 on the active OpenVMS version or whose name and extension exceeds 236 characters are truncated to zero length. This makes them invisible to OpenVMS and is consistent with prior OpenVMS NFS client behavior.

3.9 NTP Problems and Restrictions

The NTP server has a stratum limit of 15. The server does not synchronize to any time server that reports a stratum of 15 or greater. This may cause problems if you try to synchronize to a server running the UCX NTP server, if that server has been designated as “free running” (with the `local-master` command). For proper operation, the `local-master` designation must be specified with a stratum no greater than 14.

3.10 SNMP Problems and Restrictions

This section describes restrictions to the SNMP component for this release. For more information about using SNMP, refer to the *HP TCP/IP Services for OpenVMS SNMP Programming and Reference* manual.

3.10.1 Incomplete Restart

When the SNMP master agent and subagents fail or are stopped, TCP/IP Services is often able to restart all processes automatically. However, under certain conditions, subagent processes may not restart. When this happens, the display from the DCL command SHOW SYSTEM does not include TCPIP$OS_MIBS and TCPIP$HR_MIB. If this situation occurs, restart SNMP by entering the following commands:

```
$ @SYS$STARTUP:TCPIP$SNMP_SHUTDOWN.COM
```

```
$ @SYS$STARTUP:TCPIP$SNMP_STARTUP.COM
```

3.10.2 SNMP IVP Error

On slow systems, the SNMP Installation Verification Procedure can fail because a subagent does not respond to the test query. The error messages look like this:

```
   .
   .
   .
Shutting down the SNMP service... done.

Creating temporary read/write community SNMPIVP_153.

Enabling SET operations.

Starting the SNMP service... done.

SNMPIVP: unexpected text in response to SNMP request:
"- no such name - returned for variable 1"
See file SYS$SYSDEVICE:[TCPIP$SNMP]TCPIP$SNMP_REQUEST.DAT for more
details.
sysContact could not be retrieved.  Status = 0
The SNMP IVP has NOT completed successfully.
SNMP IVP request completed.
Press Return to continue ...
```

You can ignore these types of messages in the IVP.

3.10.3 Using Existing MIB Subagent Modules

If an existing subagent does not execute properly, you may need to relink it against the current version of TCP/IP Services to produce a working image. Some subagents (such as those for HP Insight Management Agents for OpenVMS) also require a minimum version of OpenVMS and a minimum version of TCP/IP Services.

The following restrictions apply:

- In general, only executable images linked against the following versions of the eSNMP shareable image are upward compatible with the current version of TCP/IP Services:
  - UCX$ESNMP_SHR.EXE from TCP/IP Services Version 4.2 ECO 4
  - TCPIP$ESNMP_SHR.EXE from TCP/IP Services Version 5.0A ECO 1

  Images built under versions other than these can be relinked with one of the shareable images, or with TCPIP$ESNMP_SHR.EXE in the current version of TCP/IP Services.
- The underlying eSNMP API changed from DPI in TCP/IP Services Version 5.0 to AgentX in later versions of TCP/IP Services. Therefore, executable images linked against older object library versions of the API (*$ESNMP.OLB) must be relinked against either the new object library or the new shareable image. Linking against the shareable image ensures future upward compatibility and results in smaller image sizes.

  Note

  Although images may run without being relinked, backward compatibility is not guaranteed. Such images can result in inaccurate data or run-time problems.

- This version of TCP/IP Services provides an updated version of the UCX$ESNMP_SHR.EXE shareable image to provide compatibility with subagents linked under TCP/IP Services Version 4.2 ECO 4. Do not delete this file.
- The SNMP server responds correctly to SNMP requests directed to a cluster alias. Note, however, that an unexpected host may be reached when querying from a TCP/IP Services Version 4.*x* system that is a member of a cluster group but is not the current impersonator.
- The SNMP master agent and subagents do not start if the value of the logical name TCPIP$INET_HOST does not yield the IP address of a functional interface on the host when used in a DNS query. This problem does not occur if the server host is configured correctly with a permanent network connection (for example, Ethernet or FDDI). The problem can occur when a host is connected through PPP and the IP address used for the PPP connection does not match the IP address associated with the TCPIP$INET_HOST logical name.
- Under certain conditions observed primarily on OpenVMS VAX systems, the master agent or subagent exits with an error from an internal `select()` socket call. In most circumstances, looping does not occur. If looping occurs, you can control the number of iterations by defining the TCPIP$SNMP_SELECT_ERROR_LIMIT logical name.
- The MIB browser provided with TCP/IP Services (TCPIP$SNMP_REQUEST.EXE) supports `getnext` processing of OIDs that include the 32-bit OpenVMS process ID as a component. However, other MIB browsers may not provide this support.

  For example, the following OIDs and values are supported on OpenVMS:

  ```
  1.3.6.1.2.1.25.4.2.1.1.1321206828 = 1321206828
  1.3.6.1.2.1.25.4.2.1.1.1321206829 = 1321206829
  1.3.6.1.2.1.25.4.2.1.1.1321206830 = 1321206830
  ```

  These examples are from `hrSWRunTable`; the `hrSWRunPerfTable` may be affected as well.
- You can ignore the following warning that appears in the log file if a null OID value (0.0) is retrieved in response to a `Get`, `GetNext`, or `GetBulk` request:

  ```
  o_oid; Null oid or oid->elements, or oid->nelem == 0
  ```

3.10.4 Upgrading SNMP

After upgrading to the current version of TCP/IP Services, you must disable and then enable SNMP using the TCPIP$CONFIG.COM command procedure. When prompted for “this node” or “all nodes,” select the option that reflects the previous configuration.

3.10.5 Communication Controller Data Not Fully Updated

When you upgrade TCP/IP Services and then modify an existing communication controller, programs that use the communication controller might not have access to the updated information.

To ensure that programs like the MIB browser (SNMP_REQUEST) have access to the new data about the communication controller, do the following:

1. Delete the communication controller using the TCP/IP management command DELETE COMMUNICATION_CONTROLLER.

2. Reset the communication controller by running the TCPIP$CONFIG.COM command procedure and exiting.
3. Restart the program (such as SNMP) by entering the following commands:

   ```
   $ @SYS$STARTUP:SNMP_SHUTDOWN.COM
   ```

   ```
   $ @SYS$STARTUP:SNMP_STARTUP.COM
   ```

4. Use the TCP/IP management command LIST COMMUNICATION_CONTROLLER to display the information.

3.10.6 SNMP MIB Browser Usage

If you use either the -l (loop mode) or -t (tree mode) flag, you cannot also specify the -m (maximum repetitions) flag or the -n (nonrepeaters) flag. The latter flags are incompatible with loop mode and tree mode.

Incorrect use of the -n and -m flags results in the following types of messages:

```
$ snmp_request mynode.co.com public getbulk -v2c -n 20 -m 10 -t 1.3.6.1.2.1
Warning: -n reset to 0 since -l or -t flag is specified.
Warning: -m reset to 1 since -l or -t flag is specified.
1.3.6.1.2.1.1.1.0 = mynode.company.com
```

3.10.7 Duplicate Subagent Identifiers

With this version of TCP/IP Services, two subagents can have the same identifier parameter. Be aware, however, that having two subagents with the same name makes it difficult to determine the cause of problems reported in the log file.

3.10.8 Community Name Restrictions

The following restrictions on community names are imposed by TCPIP$CONFIG.COM:

- Do not specify community names that include a space character.
- A quotation mark (") specified as part of a community name might be handled incorrectly. Check the validity of the name with the SHOW CONFIGURATION SNMP command, and if necessary, correct the name with the SET CONFIGURATION SNMP command.

3.10.9 eSNMP Programming and Subagent Development

The following notes pertain to eSNMP programming and subagent development.

- In the documentation, the terms “extension subagent”, “custom subagent”, and “user-written subagent” refer to any subagent other than the standard subagents for MIB-II and the Host Resources MIB, which are provided as part of the TCP/IP Services product.
- In the [.SNMP] subdirectory of TCPIP$EXAMPLES, files with the .C, .H, .COM, .MY, and .AWK extensions contain additional comments and documentation.
- The TCPIP$SNMP_REQUEST.EXE, TCPIP$SNMP_TRAPSND.EXE, and TCPIP$SNMP_TRAPSND.EXE programs are useful for testing during extension subagent development.
- For information about prototypes and definitions for the routines in the eSNMP API, see the TCPIP$SNMP:ESNMP.H file.

3.11 SSH Problems and Restrictions

This section contains the following information:

- SSH-related security advisories (Section 3.11.1)
- SSH general notes and restrictions (Section 3.11.2)
- UNIX features that are not supported by SSH (Section 3.11.3)
- SSH command syntax notes and restrictions (Section 3.11.4)
- SSH authentication notes and restrictions (Section 3.11.5)
- SSH keys notes and restrictions (Section 3.11.6)
- SSH session restrictions (Section 3.11.7)
- SSH messages notes and restrictions (Section 3.11.8)
- SSH remote command notes and restrictions (Section 3.11.9)
- SSH batch mode restrictions (Section 3.11.10)
- X11 port forwarding restrictions (Section 3.11.11)
- File transfer restrictions (all file sizes) (Section 3.11.12)
- File transfer restrictions (large files) (Section 3.11.13)

Note

References to SSH, SCP, or SFTP commands also imply SSH2, SCP2, and SFTP2, respectively.

3.11.1 SSH-Related Security Advisories

Computer Emergency Readiness Team (CERT®) advisories are issued by the CERT Coordination Center (CERT/CC), a center of Internet security expertise located at the Software Engineering Institute, a federally-funded research and development center operated by Carnegie Mellon University. CERT advisories are a core component of the Technical Cyber Security Alerts document featured by the United States Computer Emergency Readiness Team (US-CERT), which provides timely information about current security issues, vulnerabilities, and exploits.

CERT and HP Software Security Response Team (SSRT) security advisories might be prompted by SSH activity. CERT advisories are documented at the following CERT/CC web site:

`http://www.cert.org/advisories.`

Table 3–1 provides brief interpretations of several SSH-related advisories:

Table 3–1 CERT/SSRT Network Security Advisories

Advisory	Impact on OpenVMS
CERT CA-2003-24	OpenSSH only; OpenVMS is not vulnerable.

(continued on next page)

Table 3–1 (Cont.) CERT/SSRT Network Security Advisories

Advisory	Impact on OpenVMS
CERT CA-2002-36	A worst case consequence of this vulnerability is a denial of service (DoS) for a single connection of one of the following types: • Server process handling a connection from a malicious client • Client process connecting to a malicious server In either case, a malicious remote host cannot gain access to the OpenVMS host (for example, to execute arbitrary code), and the OpenVMS server is still able to receive a new connection.
CERT-2001-35	OpenVMS is not vulnerable. Affects SSH Version 1 only, which is not supported.
CERT CA-1999-15	RSAREF2 library is not used; OpenVMS is not vulnerable.
SSRT3629A/B	OpenVMS is not vulnerable.

3.11.2 SSH General Notes and Restrictions

This section includes general notes and restrictions that are not specific to a particular SSH application.

- The UNIX path `/etc` is interpreted by the OpenVMS SSH server as TCPIP$SSH_DEVICE:[TCPIP$SSH].
- The following images are not included in this release:
  - TCPIP$SSH_SSH-CERTENROLL2.EXE

    This image provides certificate enrollment.
  - TCPIP$SSH_SSH-DUMMY-SHELL.EXE

    This image provides access to systems where only file transfer functionality is permitted.
  - TCPIP$SSH_SSH-PROBE2.EXE

    This image provides the `ssh-probe2` command, which sends a query packet as a UDP datagram to servers and then displays the address and the SSH version number of the servers that respond to the query.

3.11.3 UNIX Features That are Not Supported by SSH

This section describes features that are expected in a UNIX environment but are not supported by SSH for OpenVMS.

- The server configuration parameter `PermitRootLogin` is not supported.
- The client configuration parameter `EnforceSecureRutils` is not supported.
- There is no automatic mapping from the UNIX ROOT account to the OpenVMS SYSTEM account.
- The SSH1 protocol suite is not supported for terminal sessions, remote command execution, and file transfer operations. Parameters unique to SSH1 in the server and client configuration files are ignored.

3.11.4 SSH Command Syntax

This section includes notes and restrictions pertaining to command syntax.

- From a non-OpenVMS client, if you use OpenVMS syntax for names (such as device names), enclose the names in single quotation marks to prevent certain characters from being interpreted as they would be on a UNIX system.

  For example, in the following command, UNIX interprets the dollar sign ($) as a terminator in the device name SYS$SYSDEVICE:[*user*], resulting in SYS:[*user*].

  ```
  # ssh user@vmssystem directory SYS$SYSDEVICE:[user]
  ```

  To avoid this problem, enter the command using the following format:

  ```
  # ssh user@vmssystem directory 'SYS$SYSDEVICE:[user]'
  ```

3.11.5 SSH Authentication

This section includes notes and restrictions pertaining to SSH authentication.

- This version of SSH does not support Kerberos-based authentication.
- The location of the SHOSTS.EQUIV file has been moved from TCPIP$SSH_DEVICE:[TCPIP$SSH] to TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2].
- If hostbased authentication does not work, the SSH server may have failed to match the host name sent by the client with the one it finds in DNS/BIND. You can check whether this problem exists by comparing the output of the following commands (ignoring differences in case of the output text):
  - On the server host:

    ```
    $ TCPIP
    TCPIP> SHOW HOST client-ip-address
    ```

  - On the client host:

    ```
    $ write sys$output -
    $_ "''f$trnlnm("TCPIP$INET_HOST")'.''f$trnlnm("TCPIP$INET_DOMAIN")'"
    ```

    If the two strings do not match, you should check the host name and domain configuration on the client host. It may be necessary to reconfigure and restart TCP/IP Services on the client host.
- If the user default directory in the SYSUAF user record is specified with angle brackets (for example, <*user-name*>) instead of square brackets ([*user-name*]), hostkey authentication fails. To solve this problem, change the user record to use square brackets.
- The pairing of user name and UIC in the OpenVMS rights database, as displayed by the AUTHORIZE utility's SHOW /IDENTIFIER command, must match the pairing in the SYSUAF record for that user name. If the pairings do not match, the following message error is displayed when the user attempts to establish an SSH session:

  ```
   Received signal 10, SIGBUS: invalid access to memory objects.
  ```

  To solve this, use the AUTHORIZE utility to correct the pairing of user name and UIC value in the OpenVMS rights database.

3.11.6 SSH Keys

This section includes notes and restrictions pertaining to SSH keys.

- SSH client users can copy their own customized version of the SSH2_CONFIG. file and modify the value of the variable `StrictHostKeyChecking`. By setting the value of this variable to "no," the user can enable the client to automatically copy the public key (without being prompted for confirmation) from an SSH server when contacting that server for the first time.

  A system manager can tighten security by setting the `StrictHostKeyChecking` variable to "yes" in the systemwide SSH2_CONFIG. file, and forcing users to use only the systemwide version of the file. In this case, to copy the public key from the server, users (and the system manager) must use another mechanism (for example, a privileged user can manually copy the public key). To enforce this tighter security response, the system manager can perform the following steps:

  1. Edit TCPIP$SSH_DEVICE:[TCPIP$SSH]SSH2_CONFIG. to include the following line:

     ```
     StrictHostKeyChecking  yes
     ```

  2. Restrict
     user access to TCPIP$SSH_DEVICE:[TCPIP$SSH]SSH2_CONFIG.
     For example:

     ```
     $ SET SECURITY/PROTECTION=(G,W) TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2]SSH2_CONFIG.;
     ```

  3. Edit the SYS$STARTUP:TCPIP$SSH_CLIENT_STARTUP.COM command procedure to install the SSH server image with the READALL privilege. In the following example, change the existing line to the replacement line, as indicated:

     ```
        .
        .
        .
     $     image = f$edit("sys$system:tcpip$ssh_ssh2.exe","upcase")
     $!    call install_image 'image' ""          <== existing line
     $     call install_image 'image' "readall"   <== replacement
        .
        .
        .
     ```

  4. Enable the SSH client, as described in the *HP TCP/IP Services for OpenVMS Guide to SSH*.

  Note

  Steps 2 and 3 involve modification of system files. Therefore, it may be necessary to repeat the modifications after a future update of TCP/IP Services.

- If you do not specify the key file in the SSH_ADD command, and SSH_ADD finds no INDENTIFICATION. file, it adds only the first private key it finds in the [*username*.SSH2] directory.

- Do not use the SSH_KEYGEN `-e` option (used to edit the comment or passphrase of the key). This option does not work.

- With this release, the default size of keys generated by the SSH_KEYGEN utility is 2048 bits (for earlier releases, the default size was 1024 bits). Consequently, generation of keys takes longer — sometimes five to ten times longer. On slow systems, or during SSH configuration, key generation may seem to be hanging when it is not. No progress indicator is displayed. During SSH configuration, the following messages indicate the keys are being generated:

```
Creating private key file: TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2]HOSTKEY
Creating public key file: TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2]HOSTKEY.PUB
```

3.11.7 SSH Sessions

This section includes restrictions pertaining to SSH sessions.

- In an SSH session on the OpenVMS server, the originating client host name and the user name or port identification are not available. For example, in a TELNET session, the OpenVMS DCL command SHOW TERMINAL displays the following information about a UNIX client:

```
Remote Port Info: Host: unixsys.myco.com Port:2728
```

  Likewise, information about an OpenVMS client appears as:

```
Remote Port Info: Host: mysys.com Locn:_RTA4:/USER
```

  Neither of these lines are displayed in a similar SSH session.

- Starting SSH sessions recursively (for example, starting one SSH session from within an existing SSH session) creates a layer of sessions. Logging out of the innermost session may return to a layer other than the one from which the session was started.

- Cutting and pasting from SSH terminal sessions on an OpenVMS server can cause data truncation. When this happens, the following error message is displayed:

```
-SYSTEM-W-DATAOVERUN, data overrun
```

- You cannot shut down an OpenVMS system from an SSH session, such as by executing the command:

```
$ @SYS$SYSTEM:SHUTDOWN.COM
```

  The phase of shutdown that stops user processes disconnects the SSH session.

- SSH escape sequences are not fully supported. For example, you may have to enter the Escape . (escape character followed by a space and a period) exit sequence twice for it to take effect. On exit, the terminal is left in NOECHO and PASTHRU mode.

- On certain non-OpenVMS clients, after attempting to exit from an SFTP session, you must press Enter an extra time to return to the operating system prompt.

3.11.8 SSH Messages

This section includes notes and restrictions pertaining to SSH session messages.

- Normally, the translation of the system logical name SYS$ANNOUNCE is displayed after authentication is complete. In this version of SSH, no automated mechanism exists for displaying this text as a prelogin banner.

  To provide a prelogin banner from a text file, create the file SSH_BANNER_MESSAGE. containing the text to be displayed before login.

  To enter multiple lines in the banner text, make sure each line ends with an explicit carriage-return character except the last line.

  Save the banner message file in the TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2] directory, with privileges that allow it to be read by the user account [TCPIP$SSH].

  If you do not use the default file name and location for the message banner file, define them using the BannerMessageFile option in the TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2]SSHD2_CONFIG. file. Specify the location and file name of your banner message file as the argument to the option using one of the following formats:

  ```
  BannerMessageFile   TCPIP$SSH_DEVICE:[TCPIP$SSH]BANNER1.TXT
  ```

  ```
  BannerMessageFile   /TCPIP$SSH_DEVICE/TCPIP$SSH/BANNER2.TXT
  ```

  ```
  BannerMessageFile   /etc/banner3.txt
  ```

  Note that the argument may be in either OpenVMS or UNIX format and is not case sensitive. (If multiple definitions for the same option are included in the configuration file, the last one listed will take effect.)

- Some SSH informational, warning, and error message codes are truncated in the display. For example:

  ```
  %TCPIP-E-SSH_FC_ERR_NO_S, file doesn't exist
  ```

- Some SSH log and trace output messages, and informational, warning, and error messages display file specifications as UNIX path names.

- During certain error conditions or while exiting from an SSH session, SSH displays signal information (as displayed on a UNIX system). For example, pressing Ctrl/C results in the following message:

  ```
    Received signal 2, SIGINT: Interactive attention signal.
  ```

  You can ignore such messages.

- When you log out, the message "Connection to *hostname* closed." may overwrite the last line of the logout message, as in the following example from an SSH session established with host tst1:

  ```
  $ LOGOUT
  Connection to tst1 closed.at  7-AUG-2003 14:37:15.01
  ```

3.11.9 SSH Remote Commands

This section includes notes and restrictions pertaining to SSH remote commands.

- Command lines for remote command execution through SSH are limited to 153 characters.

- After you execute an SSH remote command, you must press the Enter key to get back to the DCL prompt.

- When you execute remote commands on the OpenVMS SSH server, the log file TCPIP$SSH_RCMD.LOG is created in the directory defined by the logical name SYS$LOGIN for your user account. This log file is not purged automatically.
- When you execute remote commands on an OpenVMS SSH client connected to a non-OpenVMS SSH server, output may not be displayed correctly. For example, sequential lines might be offset as if missing a linefeed, as in the following example:

```
$ ssh user@unixhost ls -a
  user’s password:
  Authentication successful.
  .
   ..
     .TTauthority
                 .Xauthority
                            .cshrc
                                  .dt
                                     .dtprofile
```

  To display the output correctly, use the -t option with the command, as in the following command example:

```
$ ssh -t user@unixhost ls -a
```

- Any OpenVMS command that refreshes the display can have unexpected results when executed as a remote SSH command. For example, the following command exhibits this behavior:

```
$ MONITOR PROCESS /TOPCPU
```

  Executed locally, this command displays a bar chart that is continuously updated. When executed as a remote command, it displays each update sequentially. In addition, you cannot terminate the command using Ctrl/C.

3.11.10 SSH Batch Mode

This section includes batch mode restrictions.

- Because the SSH, SFTP, and SCP commands are implemented by code ported from UNIX sources, they do not support all of the standard OpenVMS behaviors for SYS$INPUT, SYS$OUTPUT, and SYS$ERROR in command procedures. For example:
  - SYS$INPUT is not the default batch command procedure.
  - Output written to a batch log file or other SYS$OUTPUT file may have an extra <CR> (ASCII decimal 13) or other explicit formatting characters.
  - You can direct SYS$OUTPUT to a file, as in the following example:

```
$ ASSIGN OUT.DAT SYS$OUTPUT
```

- When you run these commands from an interactive command procedure, you should use the explicit UNIX batch mode flags, as listed in the following table:

For...	Use...
SSH (remote command execution or port forwarding),	-o batchmode yes

For...	Use...
SCP,	`"-B"`
SFTP,	`"-B"` {*batchfile*}

- If you use the SSH command in batch mode with an interactive session (that is, not for remote command execution or setting up port forwarding), the batch job hangs.

  If the `"-S"` option is used in an interactive SSH session, or with an SSH command executed interactively in a DCL command procedure, the terminal session hangs. Ctrl/Y and Ctrl/C will not restore the DCL prompt. To release the hung terminal session, you must restart the SSH client and server.
- For the SFTP command, note the following:
  - If the command is used without a `-b` {*batchfile*} or `"-B"` {*batchfile*} option, SFTP uses the following file by default: SYS$LOGIN:TCPIP$SFTP_BATCHFILE.TXT.
  - Each line of *batchfile*, except the last, must end with a line feed (`<LF>`, ASCII decimal 10).
- When running in batch mode:
  - The SFTP command displays the final state-of-progress indicator; the SCP command does not.
  - The SSH command will not prompt for a password, password update, or passphrase. If one is required, the batch job fails.
  - The SSH command will not cause a new host key to be saved if the value of `StrictHostkeyChecking` is “no;” SSH will not prompt for one if the value is “ask.”

    For other notes and restrictions pertaining to keys, see Section 3.11.6.
  - If an `ls` command is contained in the SFTP batch input, and the interactive output requires input from the keyboard to continue, then some of the output lines might be omitted from the batch log file.

3.11.11 SSH X11 Port Forwarding

This section includes X11 port forwarding restrictions and problems.

- to use X11 forwarding in native mode, the system must be running DECwindows MOTIF Version 1.3 or higher. In addition, the X Authority utility (xauth) is required on the system. The X11 server uses this utility for authenticating host/user connections. For more information on how to use this utility, see the HP DECwindows Motif for OpenVMS documentation.
- To display a remote X11 client application on your X11 server, you must set the display variable on the X11 client to the address of the X11 server the client is connecting to. You can verify that the variable is set correctly by using the following DCL command:

  ```
  $ SHOW LOGICAL DECW$DISPLAY
  ```

  For WSA display devices, use the SHOW DISPLAY command to see the display variable value.

To set the display variable on an OpenVMS client to point to your server, use the SET DISPLAY command as in the following example, where 16.20.176.33 is the server node address:

```
$ SET DISPLAY/CREATE/NODE=16.20.176.33/TRANSPORT=TCPIP
```

SSH on OpenVMS only supports local and TCP/IP transports. If you are using a local transport, you have to be at the system where the display is to appear, and that system must be running the X11 server. For local transport, use the following command to set the display:

```
$ SET DISPLAY/CREATE/TRANSPORT=LOCAL
```

On UNIX systems, use the following command to set the display variable to point to a server node with address 16.20.176.33 and using the TCP/IP transports:

```
>setenv display 16.20.176.33:0.0
```

To use local transport, use the following UNIX command:

```
>setenv display :0.0
```

- To set up a standard port forwarding session on a remote OpenVMS system, HP recommends that you use remote port forwarding; local port forwarding will not work.

3.11.12 SSH File Transfer (All File Sizes)

This section includes SSH restrictions pertaining to file transfer operations.

- On OpenVMS, setting the `ForcePTTYAllocation` keyword to “yes” in the SSH2_CONFIG. file can result in failures when performing file copy operations. (In other implementations of SSH, setting the keyword `ForcePTTYAllocation` to “yes” in the SSH2_CONFIG. file has the same effect as using the `-t` option to the SSH command.)
- Packet-related warnings may appear when using the SFTP and SCP commands on an OpenVMS SSH client to access an OpenSSH server, as in the following example:

```
sftp> ls
.
.bash_logout
.login
Warning: packet length mismatch: expected 27, got 8;  connection to
non-standard server?
```

After a pause, the following message is displayed:

```
sftp> Warning: packet length mismatch: expected 23, got 8;  connection to
non-standard server?
```

The operation on OpenVMS succeeds despite the warnings. You can ignore the warnings. To suppress the warnings, assign the logical name TCPIP$SSH_TOLERANT_PROTOCOL_STATUS systemwide, for example:

```
$ DEFINE/SYSTEM TCPIP$SSH_TOLERANT_PROTOCOL_STATUS 1
```

To retain this assignment through each reboot, add this command to the appropriate startup command procedure.

- File transfer is limited to OpenVMS files with the following record formats (as displayed by the DIRECTORY/FULL command):
  - STREAM_LF
  - Fixed-length 512-byte records
- Not all variants of UNIX path names are supported when referring to files on OpenVMS clients and servers.
- The SCP and SFTP commands from the following Windows clients have been tested and interoperate correctly with the OpenVMS SSH server:
  - PuTTY
  - SSH Communications

  Other versions and other clients may work, depending on protocol implementation and factors such as whether the client can handle OpenVMS-format file specifications.
- When using the SFTP command, pressing Ctrl/C does not display "Cancel" as expected. Also, Ctrl/T does not work as in DCL to display a status line; instead, it switches two adjacent characters, as on UNIX systems. Other problems with character handling have been fixed with this release, as reported in Section 4.14.
- The SFTP `ls` command pauses for an extended time after displaying a page of data and then continues with the next page.
- Using SCP or SFTP command to copy a file back to itself (either in local mode, or by connecting back to the client host) fails with the following error:

  ```
  %TCPIP-E-SSH_FC_ERR_INVA, file record format invalid for copy
  ```
- The SCP command issued from a client using SSH Version 1 will not work with the OpenVMS SSH server. The OpenVMS server does not support SSH Version 1.

3.11.13 SSH Transferring Large Files

This section includes restrictions pertaining to transferring large files:

- The minimum version of DECC$SHR running on your system must be that which was released with OpenVMS Version 8.2.
- You may need to adjust memory parameters (WSDEF, WSQUO, WSEXTENT, and PGFLQUO) to accommodate the memory requirements of the file copy client and server. The exact value depends on system resources and virtual memory configuration. For more information, see Section 2.5.
- Once a file transfer is started using the SCP or SFTP command, you cannot use Ctrl/Y or Ctrl/C to abort the transfer; the only way to stop it is to terminate the client or server process from another session.
  - Stopping an OpenVMS client process:

    On the client, the file transfer server subprocess name is of the format *username_n*, where *username* corresponds to the current user name, and *n* is an integer. When the process is stopped, the following message is displayed on the client:

    ```
    %TCPIP-E-SSH_FC_ERROR, undetermined error within sshfilecopy
    ```

The following messages are displayed on the OpenVMS SSH server:

```
log (TCPIP$SSH_GOME:TCPIP$SSH_RUN.LOG):
Mon 28 13:09:15 INFORMATIONAL: Local disconnected:
   Connection closed.
Mon 28 13:09:15 INFORMATIONAL: connection lost:
   'Connection closed.'
```

– Stopping an OpenVMS server process:

  If you use the OpenVMS DCL command SHOW SYSTEM when a file transfer is active, the command displays two processes relevant to the file transfer. One has a name in the format TCPIP$SSH_*n*, where *n* is an integer. The other has a name in the format TCPIP$*prefix*_BG*n*, where *n* is a BG device number, and *prefix* is S, SS, or SSH. You must stop the BG process; stopping the TCPIP$SSH_*n* process results in the client hanging.

  After the server is stopped, the following messages are displayed on the client:

```
Disconnected; connection lost (Connection closed.)
tcpip$ssh_scp2.exe: warning: child process
(/sys$system/tcpip$ssh_ssh2) exited.
```

```
%TCPIP-E-SSH_FC_ERROR, undetermined error within sshfilecopy
```

3.12 TCPDUMP Restrictions

TCPDUMP works the same way on OpenVMS as it does on UNIX systems, with the following restrictions:

- On UNIX systems, `tcpdump` sets the NIC (network interface controller) into promiscuous mode and everything in the transmission is sent to `tcpdump`.

  On OpenVMS systems, TCPDUMP only sees the packets destined for and sent from the local host. Therefore, TCPDUMP works in copy-all mode. Because it only sees a copy of the packets that are processed by the TCP/IP kernel, TCPDUMP can only trace natively IP, IPv6, and ARP protocols on Ethernet.

  TCPDUMP can format or filter packets that have been traced from another platform running TCPDUMP in promiscuous mode. In this case it will process other protocols, like DECnet.
- Ethernet is the only supported type of NIC. Other types of NICS (such as ATM, FDDI, Token Ring, SLIP, and PPP) are not supported.
- The `-i` option is not supported. On UNIX systems, this option specifies the interface that `tcpdump` is attached to.

  On OpenVMS systems, TCPDUMP obtains packets from the TCP/IP kernel.
- The `-p` option is not supported. On UNIX systems, this option specifies that `tcpdump` stops working in promiscuous mode.

  On OpenVMS, TCPDUMP does not work in promiscuous mode. Therefore, this option is set by default.
- If you are using the Ethereal software to dump IPv6 network traffic, use the following command format to write the data in the correct format:

  ```
  $ TCPDUMP -s 1500 -w filename
  ```
- Only one process at a time can issue traces. This restriction applies to both TCPTRACE and TCPDUMP.

3.13 Determining the TCP/IP Device Name from a Channel Assignment

OpenVMS provides several ways to determine the name of a device on a channel assignment. Using the SYS$GETDVI/SYS$GETDVIW system services, the DVI$_DEVNAM, DVI$_FULLDEVNAM, and DVI$_UNIT items all return information about the device. While the first two items provide the full device name, the DVI$_UNIT item returns only the unit number of the device. To form the complete device name, a program must prefix the unit number (as a string) with the device name and controller information. In the case of the TCP/IP device name, the programmer could add the string `BG` or `BGA`. For example, `BG + 1234` would produce the device name `BG1234:`.

The TCP/IP device name may be altered in a future release. It is good programming practice to use the DVI$_DEVNAM or DVI$_FULLDEVNAM items to obtain the full device-name string. Such programs are not based on the assumption that the TCP/IP device name is BG*nnnn* or BGA*nnnn*, and will not be affected by any future changes to the TCP/IP device name.

3.14 TCP/IP Management Command Restrictions

The following restrictions apply to the TCP/IP management commands:

- TCP/IP Services Version 5.4 introduced failSAFE IP, which obsoletes the IP cluster alias address. Consequently, the following TCP/IP management commands are no longer supported:

  – SET INTERFACE /NOCLUSTER

  – SHOW INTERFACE /CLUSTER

  To display interface addresses, including IP cluster alias addresses, use the following TCP/IP management command:

  ```
  TCPIP> ifconfig -a
  ```

  To delete a cluster alias address from the active system, use a command similar to the following:

  ```
  TCPIP> ifconfig ie0 -alias 10.10.10.1
  ```

  The following TCP/IP management commands continue to be supported:

  – SET INTERFACE/CLUSTER

  – SET CONFIGURATION INTERFACE /CLUSTER

  – SET CONFIGURATION INTERFACE /NOCLUSTER

  – SHOW CONFIGURATION INTERFACE /CLUSTER

- SET NAME_SERVICE /PATH

  This command requires the SYSNAM privilege. If you enter the command without the appropriate privilege at the process level, the command does not work and you are not notified. If you enter the command at the SYSTEM level, the command does not work and you receive an error message.

- SET SERVICE command

  When you modify parameters to a service, disable and reenable the service for the modifications to take effect.

For more information on TCP/IP Services management commands, refer to the *HP TCP/IP Services for OpenVMS Management Command Reference* guide.

4
Corrections

This chapter describes the problems corrected in this version of TCP/IP Services.

4.1 Advanced Programming Environment Problems Fixed in This Release

The following sections describe programming-related problems fixed in this release.

4.1.1 Link Conflicts Occur When Linking to the TCPIP$LIB.OLB Library

Problem:

Link conflicts occur when a program that includes references to the `strdup` or `putenv` function is linked to the TCPIP$LIB.OLB library. The linker produces the %LINK-W-MULDEF warning message, indicating a conflict with functions of the same name in the C RTL library.

Solution:

In earlier versions of TCP/IP Services, the TCPIP$LIB.OLB library included functions that have since been defined in more recent versions of the OpenVMS C RTL library. These TCPIP$LIB.OLB routines, which have the DECC$ prefix, conflict with the routines of the same name in the recent versions of the C RTL library. With this release of TCP/IP Services, the TCPIP$LIB.OLB library has been modified to prevent such conflicts.

4.2 BIND Server Problems Fixed in This Release

The following sections describe BIND server problems fixed in this release.

4.2.1 BIND Slave Refusing Notify Requests

Problem:

A BIND server configured as a slave can refuse notify requests from a master server. The error message written to the TCPIP$BIND_RUN.LOG on the slave includes the text "refused notify from non-master." This problem occurs when the master server has been enabled for IPv6 communication by having the `listen-on-v6` directive specified in the options statement in the TCPIP$BIND.CONF configuration file.

Solution:

This problem is corrected in this release.

4.2.2 The BIND Version 9 Server Process Exits With "Assertion Failure" Error

Problem:

The BIND server process exits with one of the following messages logged in the TCPIP$BIND_RUN.LOG file:

```
REQUIRE((((task) != 0L) && (((const isc__magic_t*)(task))->magic
== ((('T')<< 24 | ('A') << 16 | ('S') << 8 | ('K')))))) failed
Sun 19 03:00:13 CRITICAL: exiting (due to assertion failure)
%SYSTEM-F-OPCCUS, opcode reserved to customer fault at
PC=FFFFFFFF80A6C924, PS=0000001B

REQUIRE(res->item_out == isc_boolean_true) failed
Fri 19 13:12:04 CRITICAL: exiting (due to assertion failure)
%SYSTEM-F-OPCCUS, opcode reserved to customer fault at
PC=FFFFFFFF80E6C924, PS=0000001B
```

Solution:

This problem is corrected in this release.

4.3 failSAFE IP Problems Fixed in This Release

The following sections describe failSAFE IP problems fixed in this release.

4.3.1 failSAFE IP Phantom Failures

Problem:

Phantom failures can occur on systems where failSAFE IP is configured with a single interface address on multiple interfaces. When LAN traffic is infrequent, failSAFE IP can signal a false error.

Solution:

This problem is corrected in this release. failSAFE IP now generates MAC-level broadcast packets, by default. The new configuration parameter GENERATE_TRAFFIC can be set to force failSAFE IP to generate gratuitous ARP packets. You can include the following new configuration parameters in the TCPIP$FAILSAFE.CONF file:

GENERATE_TRAFFIC

Enables failSAFE IP to periodically generate either MAC-level broadcasts or gratuitous ARP packets. You can also configure failSAFE IP to turn off traffic generation.

Default: `mac` (MAC-level broadcast)
Other options: `arp` (gratuitous ARP packets) or `off`

The following is an example line in the configuration file setting the parameter to generate gratuitous ARP packets:

```
GENERATE_TRAFFIC: ARP
```

MAC_PTY

If MAC-level broadcast traffic is being generated, this parameter allows you to specify the MAC protocol type (a two-byte hexadecimal number, such as 6005).

If MAC_PTY is not specified, the MAC broadcast tries each protocol type until an available one is found.

The following is an example line in the configuration file setting the MAC protocol type as 6005:

```
MAC_PTY: 6005
```

For more information about configuring failSAFE IP, see the *HP TCP/IP Services for OpenVMS Management* guide.

4.3.2 Users Cannot Change the Location for the failSAFE IP Log File

Problem:

failSAFE IP log files are always named:

SYS$SYSDEVICE:[TCPIP$FSAFE]TCPIP$FAILSAFE_*node-name*.LOG

Users cannot specify alternate locations on their systems.

Solution:

This problem is corrected in this release. The new configuration parameter LOGFILE allows users to specify a log file location other than the default.

LOGFILE	Specifies the file specification for the log file created by failSAFE IP. The default is SYS$SYSDEVICE:[TCPIP$FSAFE]TCPIP$FAILSAFE_ *node-name*.log. Specify the parameter and location as in the following example: `LOGFILE: DEV1:[STATS]FAILSAFE.LOG`

For more information about configuring failSAFE IP, see the *HP TCP/IP Services for OpenVMS Management* guide.

4.3.3 SHOW INTERFACE Command Does Not Display Pseudointerface Addresses

Problem:

After an interface fails or recovers an alias address, the TCP/IP management command SHOW INTERFACE does not display pseudointerface addresses.

Solution:

This problem is corrected in this release.

4.4 FTP Server Problems Fixed in This Release

The following sections describe FTP server problems fixed in this release.

4.4.1 FTP Does Not Allow IP Address Specification

Problem:

The FTP server does not allow you to specify an IP address other than that of the connected client, or the specification of a privileged port, in the PORT, LPRT, or EPRT commands. Any such commands are rejected with the following error:

```
500 Illegal {PORT|LPRT|EPRT} command.
```

The FTP server and client prevent data connection “theft” by a third party. For the FTP server, this applies to passive-mode connections from an IP address other than the client’s, or from a privileged port. For the FTP client, this applies to active-mode connections from an IP address other than the server’s, or from a port other than port 20.

Solution:

If this software change is not acceptable, you can restore the original behavior by defining the following logical names:

Server	Client
TCPIP$FTPD_ALLOW_ADDR_REDIRECT	TCPIP$FTP_ALLOW_ADDR_REDIRECT
TCPIP$FTPD_ALLOW_PORT_REDIRECT	TCPIP$FTP_ALLOW_PORT_REDIRECT

These logical names allow you to relax the IP address and port checks in the FTP server and the FTP client.

4.4.2 DCL DIRECTORY or UNIX ls Command Returns "Illegal Port Command" Error

Problem:

On an FTP client, if you use a password with an embedded space to log into an OpenVMS FTP server, the following error message is returned in response to the DCL command DIRECTORY or the UNIX command `ls`:

```
500 Illegal PORT command.
```

Solution:

This problem is corrected in this release.

4.5 FTP Client Problems Fixed in This Release

The following sections describe FTP client problems fixed in this release.

4.5.1 FTP Client Fails to Delete Interim Files after GET/MGET Commands

Problem:

After an FTP GET or MGET command entered with wildcard characters completes, the temporary TCPIP$FTP_TEMP*nnnnnnnn*.TMD files created by FTP are supposed to be deleted from the SYS$SCRATCH area. However, if no files match the wildcard criteria, FTP fails to delete any of the temporary files. (If at least one file matches the wildcard criteria, FTP successfully deletes any TCPIP$FTP_TEMP*nnnnnnnn*.TMD files created in SYS$SCRATCH.)

Solution:

This problem is corrected in this release.

4.6 IMAP Problems Fixed in This Release

The following sections describe IMAP problems fixed in this release.

4.6.1 Mail Message Lost after IMAP Move and Purge

Problem:

If you manually move a message out of a folder and then use IMAP to purge the source folder, the mail is lost.

This problem occurs when you:

1. Select a mail file using the IMAP client.
2. Read the message using OpenVMS Mail and move it to another folder.
3. Enter the Expunge command on the selected folder, using the IMAP client.

The message disappears from the destination folder. If the message was copied to a new folder, the folder ceases to exist.

Solution:

This problem is corrected in this release.

4.6.2 IMAP CLOSE Command Does Not Function Properly

Problem:

When a client logs out by issuing the IMAP CLOSE command, the IMAP server does not delete all the messages marked for deletion.

Solution:

This problem is corrected in this release. When you enter the CLOSE command, the IMAP server deletes all the messages marked for deletion.

4.7 IPv6 Problems Fixed in This Release

The following sections describe IPv6 problems fixed in this release.

4.7.1 TCPIP$IP6_SETUP.COM Problems

This section describes TCPIP$IP6_SETUP.COM problems fixed in this release.

- **Problem:**

  The TCPIP$IP6_SETUP.COM command procedure for configuring IPv6 has the following problems:

  - Attempts to configure a 6to4 tunnel fail.
  - All routes required for 6to4 relay router are not configured.
  - The endpoints for automatic tunnels are not configured correctly.
  - IPv6 over IPv6 manual tunnels cannot be configured.
  - Errors are generated in the IPv6 configuration and initialization files during IPv6 host or router configuration.
  - Manual routes cannot be configured.

  Solution:

  The configuration command procedure now enables you to successfully configure 6to4 tunnels, all routes required for a 6to4 relay router, automatic tunnels, IPv6 over IPv6 manual tunnels, and manual routes. (For more information, refer to the *HP TCP/IP Services for OpenVMS Installation and Configuration* guide.)

- **Problem:**

  The TCPIP$IP6_SETUP.COM command procedure requires that TCP/IP Services be started in order to verify specified addresses.

  Solution:

  This problem is corrected in this release. TCP/IP Services no longer needs to be started in order to run TCPIP$IP6_SETUP.COM.

4.7.2 iptunnel create Command Causes BIND Lookups for IPv4 Addresses

Problem:

When invoking an `iptunnel create` command that specifies IPv4 addresses for the tunnel source or end points, numerous DNS name resolution queries are sent to the name server even though resolution is not needed. These queries could result in a delay.

Solution:

This problem is corrected in this release.

4.8 NFS Server Problems Fixed in This Release

The following sections describe NFS server problems fixed in this release.

4.8.1 NFS Server Overwrites Files with Case-Sensitive Lookup

With OpenVMS Version 7.3-1 and higher the /CASE_LOOKUP=BLIND qualifier with the SET PROCESS command causes the case of file names to be ignored during lookups, while /CASE_LOOKUP=SENSITIVE causes the case of file names to be considered. However, if case sensitivity is not enabled on the NFS server, and the NFS client attempts to create both of those files, unexpected results can happen. For example the second file might overwrite the first.

With this release of TCP/IP Services, the TCP/IP management command ADD EXPORT has two new options: CASE_BLIND and CASE_SENSITIVE, which control UNIX-like case sensitivity for NFS server file lookups. For example, when case sensitivity is enabled, NFS preserves the case in the file names AaBBc.TXT and AABBC.TXT, regarding them as two different files.

In general, TCP/IP Services clients (not servers) determine whether lookups are case sensitive because they perform lookups in their local directory cache rather than on the server. However, when a file is being created, the server controls whether case sensitivity is in effect. Make sure that the case-sensitivity options for the server and client match; otherwise, unexpected results can occur.

For more information on the CASE_BLIND and CASE_SENSITIVE options, enter the following command:

```
$ TCPIP HELP ADD EXPORT
```

4.8.2 Directories Created by non-VMS Clients Do Not Inherit Version Limit

Problem:

Newly created directories should inherit the version limit attribute from their parent directory. When a directory is created at the request of an OpenVMS NFS client, the attribute is inherited as expected; however, directories created at the request of non-OpenVMS NFS clients do not inherit this attribute. This is a problem particularly for UNIX clients, because UNIX files only have one version, but the version limit of a new directory is set to zero (no limit).

Solution:

This problem is corrected in this release. Directories created for non-OpenVMS clients now inherit the parent directory's version limit attribute.

4.8.3 NFS Server and netstat Do Not Run Properly on Alpha Systems Not Running EV56 or Later Technologies

Problem:

On Alpha systems predating the EV56 processor, the NFS server and the `netstat` utility either experience excessive instruction time or do not run at all.

Solution:

This problem is corrected in this release.

4.8.4 MOUNT Server Problems Fixed in This Release

The following sections describe MOUNT server problems fixed in this release.

4.8.4.1 Improper Mount Point Verification

Problem:

The MOUNT service exhibits improper verification of mount points for exported file systems.

Solution:

This problem is corrected in this release.

4.8.4.2 Cannot Mount ODS-5 File System

Problem:

When the TYPELESS_DIRECTORIES option is specified in the ADD EXPORT command, you cannot mount an ODS-5 file system even though the export entry contains a directory specification that does not end in .dir.

Solution:

This problem is corrected in this release.

4.8.4.3 Host Name Verification Occurs During Mount and Causes Failure

Problem:

When a client attempts to mount a file system, host name verification is performed even if the `mountd_option_*` `nfs` subsystem attributes were not set. An error or event message on the client may indicate permission denied. The MOUNT server may produce an OPCOM message indicating that the client host name and IP address are not consistent with the hosts database (TCPIP$HOST) or with DNS/BIND information.

Solution:

This problem is corrected in this release.

4.8.4.4 Misleading Mount Server Error

Problem:

The MOUNT server reports a misleading error message when the mount port is already in use.

If the mount port (port 10) is already in use, the mount server reports the following error:

```
ERROR: bind: address already in use
```

This can be mistaken for a BIND/DNS issue when in fact it is the C RTL call `bind()` that is failing.

Solution:

This problem has been corrected in this release. The message has been changed to:

```
ERROR: bind: mount server port(10) already in use
```

4.9 NTP Problems Fixed in This Release

The following sections describe NTP problems fixed in this release.

4.9.1 On High-Performance Alpha Systems NTP Fails to Adjust System Clock

Problem:

When running on certain high-performance Alpha systems, NTP may be unable to adjust the system clock; therefore, NTP will not be able to provide accurate timekeeping. When this happens, the following error message appears in the NTP log file:

```
%SYSTEM-F-BADLOGIC, internal logic error detected
VMS timekeeping is not working as expected - can't proceed
```

Solution:

This problem is corrected in this release.

4.9.2 NTP Creates Lowercase File Names on ODS-5 Disks

Problem:

In previous releases of TCP/IP Services, when the NTP server creates files on ODS-5 disks, it gives them lowercase file names. This causes a file-naming inconsistency with non-ODS-5 disks, which assign uppercase names to files.

Solution:

This problem is corrected in this release. All files are created using uppercase file names.

4.10 RCP Problems Fixed in This Release

The following section describes RCP problems fixed in this release.

4.10.1 RCP File Copy Operation Involving Multiple Files or Directories Fails

Problem:

- Attempts to copy files recursively abort prematurely for no apparent reason, or they fail with a read or write error.
- Attempts to copy files might fail with the following error message:

  ```
  %CONV-F-OPENOUT, error opening !AS as output
  ```

  This occurs when using the /RECURSIVE qualifier or wildcards to copy files located in a directory hierarchy that is greater than eight levels deep.

Solution:

These problems are corrected in this release. RCP now supports copy operations involving directory structures greater than eight levels deep. Directory specifications up to 255 levels are now supported.

4.10.2 OpenVMS-to-OpenVMS File Copy Operations Do Not Preserve File Attributes

Problem:

RCP copy operations between OpenVMS systems do not preserve the file attributes (file organization and structure). Files are automatically converted to STREAM_LF format.

Solution:

With this release, RCP allows users to specify the /VMS qualifier to preserve file attributes (UNIX format: use the -v option).

Note

Specify this qualifier only for file copy operations between two OpenVMS systems; otherwise, the operation will fail.

4.10.3 Attempts to Copy Files Larger than 2GBs Fail

Problem:

Attempts to copy files that are greater than 2 gigabytes in size fail.

Solution:

With this release, RCP can copy files larger than 2 GBs. The file size is limited to 4 gigabytes.

4.11 SMTP Problems Fixed in This Release

The following sections describe SMTP problems fixed in this release.

4.11.1 SMTP Receiver Does Not Check Recipient Deliverability

Problem:

The SMTP receiver does not check to see if the recipient email address in the RCPT TO SMTP protocol command is deliverable (for example, that the user account exists on the system). This check is instead deferred to the processing of the mail message in the SMTP queue by the SMTP symbiont process. By this time, the host has taken responsibility for the message and, if there is a problem delivering the message, must bounce the message itself.

This behavior is more problematic when the system receives SPAM. SPAM arrives on the host for a non-existent user and is bounced by the host's symbiont process to the email address in the SPAM's Return-Path: header. The SPAM's Return-Path: header contains an invalid email address, so the bounced SPAM is in turn bounced back to the host's POSTMASTER account. The POSTMASTER account's mail is forwarded to the SYSTEM account, which means that the SYSTEM user must constantly separate these doubly-bounced SPAMs from their valid email.

Solution:

The SMTP receiver has been changed to check to see if the recipient email address in the RCPT TO SMTP protocol command is deliverable. This solves the problem by not letting the SPAM for the unknown user onto the host in the first place.

The Symbiont-Checks-Deliverability configuration option allows you to turn this feature on and off. Enter this configuration option in the SMTP configuration file (SMTP.CONFIG).

When this option is set to TRUE, the symbiont checks deliverability of RCPT TO recipients. Setting Symbiont-Checks-Deliverability to FALSE (the default) tells the receiver to check the deliverability

4.11.2 SMTP Accepts Mail from Senders Who Should Be Blocked

Problem:

SMTP might accept mail from senders who should be blocked. These are senders listed in the anti-SPAM Reject-Mail-From field of the SMTP.CONFIG file. SMTP fails to block such mail when the entries in the Reject-Mail-From field exceed the 500-character limit for SMTP.CONFIG fields.

Solution:

This problem is corrected in this release. HP has increased the character limit for fields in the SMTP.CONFIG file from 500 to 10,000.

4.11.3 Two Messages Acquire Same Value in Message-ID Header

Problem:

Any two messages composed in the same one-hundredth of a second will acquire the same value in their Message-ID header. This can cause some mail systems to delete the second of the two messages as a duplicate. Message-IDs should be unique.

Solution:

This problem is corrected in this release. Any two messages created in the same one-hundredth of a second will acquire unique values in their Message-ID headers.

4.11.4 Potential Problems Caused by Multiple Addresses in SMTP To: or Cc: Header

Problem:

Multiple addresses in the To: SMTP mail header that is composed in OpenVMS mail are not separated into multiple lines of text but instead appear on one line. For recipients of such messages on OpenVMS, if the length of this To: line exceeds the OpenVMS mail line length limit of 255 characters, the SMTP symbiont breaks the line into multiple lines when delivering the message, but the lines after the first one are not indented (tabbed in). As a result, the lines will appear as malformed headers. This can cause incorrect behavior with some automated programs that read e-mail. The same problem exists for Cc: lines longer than the OpenVMS mail limit.

Solution:

This problem is corrected in this release. When a user composes a mail message, the SMTP software that builds the SMTP To: and Cc: headers ensures that a To: or Cc: header line does not exceed 75 characters. If adding the next recipient address to a header line would cause the line to exceed 75 characters, the SMTP software inserts a line feed and tab into the headers before adding that recipient address.

4.12 SNMP Problems Fixed in This Release

The following sections describe SNMP problems fixed in this release.

4.12.1 TCPIP$CONFIG.COM Refuses SNMP Community Names Containing Special Characters

Problem:

With Versions 5.1 and 5.3 of TCP/IP Services, TCPIP$CONFIG.COM checks for special characters, and disallows community names containing any special character.

Solution:

This release relaxes these restrictions. However, TCPIP$CONFIG.COM does not accept a space in an SNMP community name. In addition, a quotation mark (") specified as part of a community name might not be handled correctly by TCPIP$CONFIG.COM. A message warns the user to check the validity of the name with the SHOW CONFIGURATION SNMP command, and, if necessary, to correct the name with the SET CONFIGURATION SNMP command.

4.13 Sockets API Problems Fixed in This Release

The following sections describe Sockets API problems fixed in this release.

4.13.1 Socket Function getaddrinfo() Hangs

Problem:

Two successive calls to `getaddrinfo()` in the same program cause the second call to hang. This is only true if the `af` parameter is AF_INET6 and the `ai_flags` parameter has not been set to AI_ALL or AI_ADDRCONFIG.

Solution:

This problem is corrected in this release.

4.14 SSH Problems Fixed in This Release

The following sections describe SSH problems fixed in this release.

4.14.1 SSH Server Does Not Allow Password Change

Problem:

The SSH server does not support password change requests for non-VMS clients when account passwords have expired.

Solution:

If the SSH configuration option `AllowNonvmsLoginWith ExpiredPwd` is set to "yes" and the password has expired, the server sends a request to the client to prompt the user for a new password. The user must change the password, or the account will be locked out, and the next attempt to log in will fail.

However, if the OpenVMS account has the `DisForce_Pwd_Change` flag set in the SYSUAF, the server allows the user to log in, displaying the following message:

```
WARNING - Your password has expired; update immediately with SET
PASSWORD!
```

The `DisForce_Pwd_Change` flag must be applied to each OpenVMS account individually.

The default setting for the `AllowNonvmsLoginWith ExpiredPwd` option has been changed to "yes." If the `AllowNonvmsLoginWithExpiredPwd` option is set to "no," the server does not allow password authentication for non-OpenVMS clients when the password has expired. The user does not have the option to change the password. For more information, refer to Section 5.2.

4.14.2 Language Tag Support

Problem:

The password change request that is sent to the SSH client can include a language tag. Some clients do not support the language tag.

Solution:

You can control this feature using the `DisableLanguageTag` configuration option in the SSH server configuration file (SSHD2.CONFIG). By default, OpenVMS password change requests include the language tag. If the client that does not expect the language tab receives it, the client will issue an error message. You can disable sending the language tag by setting the `DisableLanguageTag` option to "yes" in the SSH server configuration file. This prevents the language tag from being included in any password change request.

4.14.3 Accepting Two Passwords

Problem:

The OpenVMS SSH server does not support a secondary password for password authentication.

Solution:

The SSH server detects when a user has a second password. In this case, OpenVMS prompts for the second password. If one password has expired, the user is prompted to change the password. If both passwords have expired, the user is prompted to change the first one, and then is prompted to change the second one.

In order for the SSH client to accept the OpenVMS prompt for the second password, one or both of the following configuration options must be set to 2:

- In the client configuration file (SSH2_CONFIG): `NumberOfPasswordPrompts`
- In the server configuration file (SSHD2_CONFIG): `PasswordGuesses`

Both configuration files may be stored in TCPIP$SSH_DEVICE:[TCPIP$SSH.SSH2]. In addition, the user can have a client configuration file in the user-specific SSH directory ([*username*.SSH2]).

Note

Support for multiple passwords is not specified in any SSH-related RFC.

The second password prompt is enabled by forcing an error situation on OpenVMS for the first password; this is handled by the OpenVMS software internally. However, the message displayed after entering the first password depends on the client software. No intrusion record is created if authentication is enabled. However, if either password is entered incorrectly, an intrusion record is created.

Some clients accept the second password request even if both passwords have expired. However, some clients do not accept the second password request; these clients function correctly when only one of the passwords has expired.

4.14.4 Native-Mode X11 Port Forwarding Does Not Work

Problem:

SSH for OpenVMS does not support the native-mode SSH mechanism for implementing X11 port forwarding (using the `-x` or `+x` SSH command options, or the `ForwardX11` keyword in the client configuration file and the `AllowX11Forwarding` keyword in the server configuration file). SSH only supports standard port forwarding, requiring special setup actions to enable the X11 functionality.

Solution:

This problem is corrected in this release. For more information, see Table 5–2.

4.14.5 SFTP Double Echo and Key-Handling Problems

Problem:

Before using SFTP to connect to a remote system, characters typed at the SFTP prompt (SFTP>) are double echoed. In addition, when connected to the remote system, the left and right arrow keys do not work as expected, as well as the Ctrl/X, Ctrl/W, and Ctrl/C sequences (to erase line, refresh line, and exit, respectively).

Solution:

These problems are corrected in this release. However, pressing Ctrl/C does not display "Cancel" as expected.

4.14.6 SSH, SFTP, and SCP Commands Fail or Do Not Work Properly in Batch Mode

Problem:

The SSH, SCP, and SFTP commands fail or work improperly in batch mode.

Solution:

This problem is corrected in this release.

For restrictions pertaining to batch mode, see Section 3.11.10.

4.14.7 RSA Key Types Not Accepted

Problem:

In prior versions of SSH for OpenVMS, RSA keys are accepted for client authentication to the server, but not accepted for server authentication to the client.

Solution:

Starting with this release of TCP/IP Services, both RSA and DSA key types are accepted for client authentication to the server as well as server authentication to the client.

4.15 SSL Problems Fixed in This Release

The following sections describe SSL problems fixed in this release.

4.15.1 After Installing SSL, POP SSL Ceases to Function

Problem:

After installing the SSL V1.2 kit on TCP/IP Services, POP SSL support ceases to function. The POP server will not listen on its SSL port and, consequently, will not service clients coming in through SSL. The TCPIP$POP_RUN.LOG POP server log file contains these lines:

```
POP server will not listen for SSL connections.
SSL$LIBCRYPTO_SHR32_INIT status: %LIB-E-KEYNOTFOU, key not found in tree
```

Solution:

This problem is corrected in this release.

4.16 TELNET Problems Fixed in This Release

The following sections describe TELNET problems fixed in this release.

4.16.1 TELNET Intrusion Detection Inflexibility

Problem:

In certain circumstances, an intrusion (such as an invalid login) by one user can cause the whole system to be locked out, and with multiport servers such as on a terminal server, all ports could be locked out. The workaround has been to set the TCPIP$TELNET_NO_REM_ID logical. However, this allows the intruding user to log in on another port without being locked out.

Solution:

This problem is corrected in this release. The logical name TCPIP$TELNET_TRUST_LOCATION allows you to specify how to handle TELNET intrusion records. When this logical name is defined, any location string specified by the remote client is included in the intrusion record. For example, many terminal servers provide the physical port number, while OpenVMS clients provide the originating user name and terminal line. Including this information in the intrusion records means that only a particular user or port will be locked out, not the entire remote host (and all user ports).

5

Documentation Update

This chapter describes updates to the information in the TCP/IP Services product documentation.

5.1 Documentation Updated for This Release

The following manuals have been updated for this release:

Table 5–1 Current Documentation Changes

Title	Changes
HP TCP/IP Services for OpenVMS SNMP Programming and Reference	• The trap communities configured for regular SNMP through the TCPIP$CONFIG.COM command procedure, the TCP/IP management command SET CONFIG SNMP, or in the SYS$SYSDEVICE:[TCPIP$SNMP]TCPIP$VMS_SNMP_CONF.DAT file are not used to determine the trap receiver host or community name. The values of the `-c` and `-h` flags to the SNMP_TRAPSND utility are handled as follows: – If no `-c` (community) flag is used, the default name "public" is used in the trap. – If no `-h` (host) flag is used, the trap is sent to LOCALHOST. • The value for the "agent address" field in the SNMPv1 trap PDU is that of the primary interface for the host on which the master agent (TCPIP$ESNMP_SERVER) is running. The value of this address can be verified as follows: 1. Translate the logical name TCPIP$INET_HOSTADDR. 2. Obtain the value of LOCALHOST using the following TCP/IP management command: `$ TCPIP SHOW CONFIGURATION COMMUNICATION` If this value is not in IP address format, determine the IP address using the following command: `$ TCPIP SHOW HOST/LOCAL` *`local-host-name`*

(continued on next page)

Table 5–1 (Cont.) Current Documentation Changes

Title	Changes
HP TCP/IP Services for OpenVMS Sockets API and System Services Programming	• The default setting for the TCPIP_KEEPIDLE option has been corrected. • The new socket options TCP_TSOPTENA, TCP_PAWS, and TCP_SACKENA are documented. • The `accept` routine clearly describes the x-open error return. • Information about how to convert port numbers has been included. • Information about using 64-bit addresses with the `send()` and `receive()` functions has been added. • Information was added to the `getservbyport()` function about converting the port number to network byte order. • More information was added about IOCTL. • All material about the Sockets API was moved from the *HP C Run-Time Library Reference Manual for OpenVMS Systems* to the *HP TCP/IP Services for OpenVMS Sockets API and System Services Programming*. • All material about programming was moved from the *HP TCP/IP Services for OpenVMS Guide to IPv6* to the *HP TCP/IP Services for OpenVMS Sockets API and System Services Programming*. • Information was added about using QIOs in IPv6.
HP TCP/IP Services for OpenVMS ONC RPC Programming	The example in Section 3.5.1 was corrected.
HP TCP/IP Services for OpenVMS Installation and Configuration	• Added information about installing on I64 platforms. • Removed information about installing on VAX platforms. • Added information about using the enhanced IP6_SETUP.COM command procedure to configure IPv6. • Updated scripts of installation and configuration.

In addition, several HELP files have been updated and enhanced, including:

- HELP TCPIP_SERVICES PROGRAMMING_INTERFACES
- HELP TCPIP_SERVICES REMOTE_COMMANDS RCP
- HELP TCPDUMP
- TCPIP HELP IFCONFIG
- TCPIP HELP SYSCONFIG

5.2 Documentation Not Being Updated for This Release

The following manuals are not updated for TCP/IP Services Version 5.5. Documentation changes planned for these manuals are indicated.

Table 5–2 Future Documentation Changes

Title	Changes
Compaq TCP/IP Services for OpenVMS Concepts and Planning	• Information about I64 platforms will be added. • Information about OpenVMS file specifications will be updated.
HP TCP/IP Services for OpenVMS Management	This manual will be enhanced with the following: • Information from Section 1.6, Support for Network Time Protocol (NTP) V4.2 in these release notes will be added. • Information from Section 1.2, failSAFE IP Support for IPv6 in these release notes will be added. In addition, corrections will be made to the description of the TCPIP$FAILSAFE logical name. • Information from Section 1.3, Secure IMAP in these release notes will be added. • Information from Section 4.8.1, NFS Server Overwrites Files with Case-Sensitive Lookup in these release notes will be added. • Information about FTP will include new logical names TCPIP$FTP_COMPAT_REV and TCPIP$FTPD_COMPAT_REV. • Information from Section 4.11.1, SMTP Receiver Does Not Check Recipient Deliverability in these release notes will be added. • Information from the *HP TCP/IP Services for OpenVMS Guide to IPv6* guide will be added. • The discussion of using logical names to specify configuration files will be enhanced to include specifics such as the use of the /SYSTEM and /EXECUTIVE_MODE qualifiers on the DEFINE command, as well as the recommendation to stop the service before changing these logical names.

(continued on next page)

Table 5–2 (Cont.) Future Documentation Changes

Title	Changes

(continued on next page)

Table 5–2 (Cont.) Future Documentation Changes

Title	Changes

HP TCP/IP Services for OpenVMS Guide to SSH	• Information about the changes described in Section 1.7, SSH Features in these release notes will be included.
	• The following information will be added to Chapter 3:
	The location of the Xauthentication executable file can be specified in the SSH client configuration file. Use the `Xauthpat` keyword to specify a device and directory other than the default location (SYS$SYSTEM:DECW$XAUTH.EXE).
	• Chapter 5 will be updated to reflect Section 4.14.4, Native-Mode X11 Port Forwarding Does Not Work in these release notes.
	When X11 port forwarding is enabled on both the SSH client and server, you can use SSH to connect to an SSH server and invoke X11 client programs there, while having them appear on your local display. You can also "chain" port forwarding across multiple systems, even if the intermediate systems are not running the X11 server. For example, from SYSTEM1 you can use SSH to connect to SYSTEM2, and then from SYSTEM2 connect to SYSTEM3. An X11 client application running on SYSTEM3 will be displayed securely on SYSTEM1.
	• The following option will be added to the "Managing Auditing" section in Chapter 4:
	`AllowVmsLoginWithExpiredPw`
	Allowed values: `yes`, `no`
	Default: `yes`
	Description: Controls the behavior when an OpenVMS client attempts to establish an SSH connection to an OpenVMS server account with an expired password. The value `yes` allows the client to interact with the server to update an expired password. The value `no` rejects the login.
	Note that when the `disforce_pwd_change` flag is set in the user's SYSUAF record, the client user is allowed to log in; a warning message is displayed instructing the user to change the password. If the user does not change the password, the account will be locked out and the user will not be allowed to log in again.
	• The following option description will be changed. The default has been changed from "no" to "yes."
	`AllowNonvmsLoginWithExpiredPw`
	Allowed values: `yes`, `no`
	Default: `yes`
	See Section 4.14.1, SSH Server Does Not Allow Password Change in these release notes for detailed information.
	• The examples in the section "Port Forwarding for FTP" will be corrected.
	• Section 6.9.1, Changing the Default Configuration, will be corrected. When specifying multiple hosts, a maximum of three BIND servers will be used.

(continued on next page)

Table 5–2 (Cont.) Future Documentation Changes

Title	Changes
HP TCP/IP Services for OpenVMS User's Guide	• Descriptions of RCP file format and size information will be updated as described in Section 4.10, RCP Problems Fixed in This Release in these release notes. • The new /VMS qualifier for RCP will be added, as documented in Section 4.10.2, OpenVMS-to-OpenVMS File Copy Operations Do Not Preserve File Attributes in these release notes.
HP TCP/IP Services for OpenVMS Tuning and Troubleshooting	• Information from the *HP TCP/IP Services for OpenVMS Guide to IPv6* will be added.
HP TCP/IP Services for OpenVMS Management Command Reference	• The manual will be updated to reflect the information in Section 3.14, TCP/IP Management Command Restrictions in these release notes. • New ADD EXPORT options CASE_BLIND and CASE_SENSITIVE will be added as described in Section 4.8.1, NFS Server Overwrites Files with Case-Sensitive Lookup in these release notes. • IPv6 Neighbor Discovery logical name will be added. To troubleshoot problems with IPv6 Neighbor Discovery, you can define a logical name to obtain debug messages in the log file SYS$MANAGER:TCPIP$ND6HOST.LOG. To set the logical name, enter the following command: `$ DEFINE /SYSTEM TCPIP$ND6HOST_DEBUG 1` Define this logical before you start TCP/IP Services.
HP TCP/IP Services for OpenVMS Guide to IPv6	The `sysconfig` commands in Section 2.6 (Configuring an IPv6 Router) are incorrect. The subsystem parameter on these command lines should be `ipv6`. These commands need not be entered prior to running the IP6_SETUP.COM procedure. The IP6_SETUP.COM sets the appropriate attributes. This manual is deprecated. The information in the new versions of the *HP TCP/IP Services for OpenVMS Installation and Configuration* guide and the *HP TCP/IP Services for OpenVMS Sockets API and System Services Programming* guide has been updated and corrected. The remaining information from the *HP TCP/IP Services for OpenVMS Guide to IPv6* will be included in the *HP TCP/IP Services for OpenVMS Management* guide and the *HP TCP/IP Services for OpenVMS Tuning and Troubleshooting* guide in a future release.

These manuals will be updated in a future release of TCP/IP Services.