Digital TCP/IP Services for OpenVMS

Release Notes

Documentation Comments

If you have comments about or suggestions for this document or any of the Digital TCP/IP Services for OpenVMS (UCX) documentation, mail your comments to the following address:

Digital TCP/IP Services Documentation Group
550 King St. (LKG2-2/Q5)
Littleton, MA 01460

If you have access to the Internet, mail your comments electronically to the following address:

doc_quality@lkg.mts.dec.com

Revision Information:	This is a new document.
Operating System:	OpenVMS Alpha Version 7.0
Software Version:	Digital TCP/IP Services for OpenVMS Version 4.0A

Digital Equipment Corporation
Maynard, Massachusetts

December 1995

Contents

Tables

Preface

The Digital TCP/IP Services for OpenVMS (UCX) product is Digital's implementation of the TCP/IP protocol suite and internet services for OpenVMS Alpha and OpenVMS VAX systems.

This document updates the Digital TCP/IP Services for OpenVMS (UCX) Version 4.0 *Release Notes*.

Intended Audience

These Release Notes are for:

- UCX installers who are using the POLYCENTER Software Installation (PCSI) procedure (for complete information, see the *Digital TCP/IP Services for OpenVMS Installation and Configuration* manual)
- Managers of OpenVMS™ layered products
- OpenVMS system managers
- Network managers (who may or may not be familiar with UNIX)

Documentation Set

For information about Digital TCP/IP Services for OpenVMS Version 4.0A and how to use it, please use the Version 4.0 documentation set, which consists of the following manuals:

- *Digital TCP/IP Services for OpenVMS Installation and Configuration*

  Order Number AA–LU49H–TE
- *Digital TCP/IP Services for OpenVMS User's Guide*

  Order Number AA–PC27G–TE
- *Digital TCP/IP Services for OpenVMS Management*

  Order Number AA–LU50H–TE
- *Digital TCP/IP Services for OpenVMS Management Command Reference*

  Order Number AA-PQQGE-TE
- *Digital TCP/IP Services for OpenVMS Sun RPC Programming*

  Order Number AA–Q06VD–TE
- *Digital TCP/IP Services for OpenVMS System Service and C Socket Programming*

  Order Number AA–LU51H–TE

How To Order Additional Documentation

Use the following table to order additional documentation or information. If you need help deciding which documentation best meets your needs, call 800-DIGITAL (800-344-4825).

Location	Call	Fax	Write
U.S.A.	DECdirect 800-DIGITAL 800-344-4825	Fax: 800-234-2298	Digital Equipment Corporation P.O. Box CS2008 Nashua, NH 03061
Puerto Rico	809-781-0505	Fax: 809-749-8300	Digital Equipment Caribbean, Inc. 3 Digital Plaza, 1st Street, Suite 200 P.O. Box 11038 Metro Office Park San Juan, Puerto Rico 00910-2138
Canada	800-267-6215	Fax: 613-592-1946	Digital Equipment of Canada Ltd. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6 Attn: DECdirect Sales
International			Local Digital subsidiary or approved distributor
Internal Orders	DTN: 264-3030 603-884-3030	Fax: 603-884-3960	U.S. Software Supply Business Digital Equipment Corporation 10 Cotton Road Nashua, NH 03063-1260

1
New Features

This release of the Digital TCP/IP Services for OpenVMS (UCX) product offers the following new software:

- Support for IP multicasting (Section 1.1)
- BIND Server (DNS) enhancements (Section 1.2)
- Management utility — new commands and qualifiers (Section 1.3)
- Sun Remote Procedure calls enhancements (Section 1.4)
- Socket programming enhancements (Section 1.5)
- FTP utility — new command (Section 1.6)
- TELNET command — new qualifiers (Section 1.7)
- Outbound TELNET devices (unsupported) (Section 1.8)
- Remote Copy (RCP) (unsupported) (Section 1.9)
- RLOGIN and RSH (enhanced) (Section 1.10)
- TELNET Print Symbiont (TELNETSYM) (enhanced) (Section 2.1)

1.1 IP Multicasting

IP multicasting, as described in RFC 1112, is now supported. This feature affects only the application programming interfaces; there are no new commands or qualifiers associated with multicasting.

Multicast addresses, which are in the range 224.0.0.0 to 239.255.255.255, are also known as Class D IP addresses. In order to send a multicast datagram, simply specify such an address as the destination address on an IO$_WRITEVBLK QIO or a sendto( ) socket library call.

You may specify a TTL (time-to-live) value other than the default (1), by setting the appropriate socket option:

```
char ttl;
setsockopt (s, IPPROTO_IP, UCX$C_IP_MULTICAST_TTL, &ttl, sizeof(ttl));
```

In order for your application to receive multicast datagrams, it must first join one or more multicast groups. Specify the group to join by using the ip_mreq structure:

```
struct ip_mreq {
   struct in_addr imr_multiaddr;          /* group to join */
   struct in_addr imr_interface;          /* interface on which to join */
}
```

Use network byte order when specifying these addresses. You may specify INADDR_ANY for the interface address if you wish to use the default interface.

To join a multicast group:

```
struct ip_mreq mr;
setsockopt (s, IPPROTO_IP, UCX$C_IP_ADD_MEMBERSHIP, &mr, sizeof(mr));
```

Group memberships are automatically dropped when the associated socket is closed or the application process terminates, but you can also drop them explicitly by specifying the UCX$C_IP_DROP_MEMBERSHIP option in a call otherwise similar to the above.

Once your application has joined a multicast group, it will receive multicast datagrams just as it would receive unicast ones. Note that in order for this to work properly, you must leave the socket's local address unspecified (INADDR_ANY), rather than binding it to the IP address of a particular local interface. Specify only the address family and port number when binding a socket which will receive both unicast and multicast packets.

1.2 BIND Server Enhancements

This release of Digital TCP/IP Services for OpenVMS offers the following new DNS features for the BIND Server.

- Supports cluster load balancing

  The advantages of cluster load balancing and how to set it up are discussed at length in the TCP/IP Services for OpenVMS documentation set. For details, see the *Digital TCP/IP Services for OpenVMS Management* manual.

- Supports well known services in the domain (WKS entries) in server databases

- Round-Robin Scheduling

  The BIND Server now performs round-robin scheduling for TCP/IP. This feature balances client requests by simple rotation. When a multiple resource record satisfies a query, the BIND Server now returns it each time in a round-robin order. For example, in cluster configurations when a particluar "tcpip cluster" is not defined in the "ucx bind configuration database" ("ucx sho config bind" does not list that particluar cluster entry), the resource records of the tcpip cluster gets shuffled in round robin manner. For those tcpip clusters defined in the "ucx bind configuration database", the BIND server performs load balancing.

  Round-Robin scheme is also useful for MX records to share the mail loads among multiple equivelent gateways. Round-Robin scheme, unlike "load balancing," does not take into account the current workload on the cluster hosts and cannot infer the "state" of the cluster hosts (if the cluster hosts are up/down).

1.3 Management Commmands and Qualifiers

This release of UCX offers the following new management commands and qualifiers:

- To support DNS Server cluster load balancing use:

  ```
  UCX> SET CONFIGURATION BIND/CLUSTER=name, /NOCLUSTER=name
  ```

- To support SLIP use:

```
UCX> SET INTERFACE
UCX> SET CONFIGURATION INTERFACE
    /AUTO_START, /NOAUTO_START
    /COMPRESS=option
    /FLOWCONTROL, /NOFLOWCONTROL
    /HOST=host
    /SERIAL_DEVICE=device
```

- To abort services using TCP use:

```
UCX> DISCONNECT DEVICE_SOCKET
```

- To support Troubleshooting use:

```
$ TCPIPTRACE
```

For details, see the *Digital TCP/IP Services for OpenVMS Management* book and the *Digital TCP/IP Services for OpenVMS Management Command Reference*.

The following commands relating to the NFS Server are no longer valid:

- BIND is no longer valid; instead use MAP
- SHOW BIND is no longer valid; instead use SHOW MAP
- UNBIND is no longer valid; instead use NOMAP

For complete command descriptions, see the *Digital TCP/IP Services for OpenVMS Management Command Reference*.

1.4 Sun Remote Procedure Calls (RPC)

For customized development, Digital TCP/IP Services for OpenVMS has these new enhancements:

- Implementation of SUN's Open Network Computing (ONC) Remote Procedure Call (RPC) and External Data Representation (XDR) Protocols (RFC 1790)
- DCL command RPCGEN
- Function library to support RPCGEN
- New location of .H files
- Backward compatibility with older RPC sources

1.4.1 RPCXDR

Digital TCP/IP Services for OpenVMS implements the Sun ONC Version 4.2 RPC Protocol. Digital TCP/IP Services for OpenVMS links this development software with DECC$SHR instead of VAXCRTL. OpenVMS Alpha systems support only DEC C, while OpenVMS VAX systems support DEC C and VAX C. In order to use Digital TCP/IP Services for OpenVMS RPC, you must compile your programs with DEC C.

When you install Digital TCP/IP Services for OpenVMS onto an OpenVMS VAX system, Digital TCP/IP Services for OpenVMS renames your previous copy of UCX$RPCXDR_SHR.EXE to UCX$RPCXDR_SHR.EXE_OLD.

1.4.2 RPCGEN

Digital TCP/IP Services for OpenVMS includes the new RPCGEN command. Before you compile an RPC program, link your RPC programs with either one of these:

- Shareable image—SYS$SHARE:UCX$RPCXDR_SHR.EXE

  Specify the /SHARE qualifier.

- New object library—UCX$RPC:UCX$RPCXDR.OLB

  Specify the /LIBRARY qualifier.

For details, see the *Digital TCP/IP Services for OpenVMS Sun RPC Programming* manual.

1.5 Programming with Sockets

The Application Programming Interfaces (APIs) for development with sockets has thread safety. The socket library I/O routines now allocate event flags dynamically. This prevents process blocking in a DECthreads environment. C Socket programs use DECthreads environments when building application interfaces that use multi-threading.

1.6 FTP

The FTP Client software has the new command SET ERROR_LEVEL, which establishes the default error tolerance level for FTP commands. The optional qualifiers include ERROR, WARNING (default), and SUCCESS.

1.7 TELNET Command

The /CREATE_SESSION and /DELETE_SESSION qualifiers have been added to the TELNET command. These qualifiers take the same parameters and have the same function as the TELNET> CREATE_SESSION and DELETE_SESSION commands that already exist. However, these new qualifiers can be used from within a command procedure. For example:

```
$TELNET
TELNET> CREATE_SESSION TS405 2002 902
```

1.8 Support for Outbound TELNET Devices

You can create a TELNET pseudo-device (virtual terminal) and connect it to a remote listener (port). Use the following TELNET command:

```
TELNET> CREATE_SESSION host port [unit]
```

where:

host	Specifies the remote host name.
port	Specifies the listening port number at the remote end.
unit	Optional. A decimal number specifying the unit number for the pseudo terminal (TN*x*). The unit "0" is the default and has a special meaning allowing the system to pick the next available unit. If the requested unit is already in use, the next available unit is chosen, and the command completes successfully. In any case, the user is notified of the unit number created.

```
TELNET> DELETE_SESSION unit
```

where:

unit	A decimal number specifying the the unit number to be deleted. If the TN specified has been set via CREATE_SESSION, it is disconnected and deleted. Otherwise, an error is returned.

1.9 Remote File Copy (RCP)

Remote Copy (RCP command) copies files between internet hosts.

RCP

Copies files between internet hosts. Issue the RCP command at the DCL prompt. You can copy files:

- From a remote host to your host
- From your host to a remote host
- From one remote host to another

To use RCP, the Remote Shell Service must be enabled.

You have two ways to set up proxy checking:

- On the RCP command line, specify a remote user name and, optionally, a password.
- Add proxies to the Proxy Database.

  For remote UNIX hosts, add an entry in the host equivalence file, /etc/hosts.equiv.

You can specify qualifiers in either OpenVMS-Style format or UNIX-Style format, but do not mix both types on the same command line.

DCL-Style Format

RCP *source_file destination_file*

UNIX-Style Format

rcp */source.file /destination.file*

Parameters

source_file
Required.

Source host and file specification, in the format *[user@]host:file*, where:

- *user@* is the user name on the remote UNIX system.

  Needed only if the UNIX system has the name in its /etc/hosts.equiv file or the UNIX user's .rhosts file.
- *host* is the remote host, followed by a colon (:).
- *file* is the name of the file to copy. A file name without the full path specification defaults to the default (or home) directory. Table 1–1 shows the possible correct formats.

Table 1–1 RCP Command: Specifying the Source File

Host	Possible Formats
UNIX hosts	• Absolute path name, such as /etc/user/hosts, followed by the file name • Path name relative to your default directory, followed by the file name
OpenVMS hosts	• Brackets ([ ]) (indicates your default directory), followed by the file name • Full file specification, such as DKA0:[WILDLIFE.BIRDS.NORTHERN]BIG_BIRDS.TXT To specify a device name, type a colon (:) and then the name. Enclose the entire parameter within quotation marks (" "). • A logical name, such as SYS$LOGIN:ROBIN.DAT, BIRDS$9:[AMERICAN]FINCHES.LIS To specify a logical name, type a colon (:) and then the name. Enclose the entire parameter within quotation marks (" ").

destination
Required.

Destination host and file specification information is of the same form as the source parameter, except:

If the file specification is completely omitted or the file name portion of the file specification is omitted, the default file name used is the same as specified in the source parameter, the directory being the default/home directory of the user.

Qualifiers

/LOG
Optional. Default: no logging.

Logs the files copied to or from the local system.

/PRESERVE (OpenVMS-Style)
-p (UNIX-Style)
Optional.

Preserves the file protection mode and modification date during a copy.

/RECURSIVE (OpenVMS-Style)
-r (UNIX-Style)
Recursively copies each subtree rooted at the directory you specify in the UNIX file specification. For OpenVMS hosts, specify [dir...] (with three trailing ellipses) in the filespec instead of using this qualifier.

/USER=*username*
User on the remote host. Use only if a proxy has not been added for this user on the remote host, or if the remote host's host equivalence file does not contain an entry for this user. If necessary, truncate the user name to the required number of characters using the /TRUNCATE qualifier below. Specifying "username@" with the source or destination parameter is the equivalent UNIX-style method.

/PASSWORD=*password*
Password on the destination system. Use with the /USER qualifier.

/TRUNCATE[=*n*]
Optional. Default: no truncation.

Truncates the user name to the specified number of characters. If you omit *n*, the default is eight characters.

Examples

1. `$ RCP /LOG unixhost:source_file []`

   Copies a file from a remote UNIX system under its home directory to a local file of the same name in the current directory. The log information for the copy is displayed.

2. `$ RCP vmshost:source_file [dir]dest_file`

   Copies a file from a remote VMS system under its home directory to a local file of a different name in the specified directory.

3. `$ RCP /PRESERVE vmshost:[dir1]source_file ":dka300:[dir2]"`

   Copies a file from a remote VMS system from the directory specified to the directory on the device system on the local system into the same file name. The copy preserves the source file's protection mode and modification date.

   Use quotation marks (" ") for specifying the device and directory on the destination.

4. `$ RCP /USER=user1 /PASS=pass1 ":SYS$LOGIN:source_file" unix_host:`

   Copies a file from your login directory of the local system to a remote UNIX system into the user's login directory. The user name and password on the UNIX system are also specified.

5. `$ rcp -r ":dka300:[source_dir...]" "unixhost:/usr/tmp/"`

   Copies all files under the local directory specified to a remote UNIX host's destination directory. If there are subdirectories beneath the source directory specified, all the files in them are copied as well, creating subdirectories on the remote host, as appropriate. Directory hierarchy on a UNIX host is preserved by default.

6. `$ RCP /LOG /RECURSIVE [source_dir...] vmshost:[dest_dir...]`

   Copies the complete local sub-directory tree specified, to a remote VMS host's destination directory while preserving the directory hierarchy and logging each file copy.

7. `$ RCP /LOG /RECURSIVE [source_dir...] vmshost:[dest_dir]`

   Same as Example 6 except that the copy is done to the destination directory itself without preserving directory hierarchy (new sub-directories are not created).

8. ```
   $ RCP /USER=someone /PASSWORD=password -
   _$ /TRUNCATE=4 source_file host:dest_file
   ```

   Copies the local file to a remote user's login directory. Note the truncation of the remote user name. A user name and password are necessary if proxy records or host equivalence file entries are not present.

9. `$ RCP host1:file1 host2:file2`

   Copies a file from one remote host to another. Assumes proxy records or host equivalence files are set up correctly for user executing this command.

10. `$ RCP "user1@host1:sys$dir:file1" "user2@host2:/usr/user2"`

    Copies a file from remote OpenVMS system host1 in the directory pointed to by the logical name SYS$DIR to remote UNIX system host2 in the specified directory. Different user names are used on the two remote systems. Proxy records and host equivalence files must be set up properly because the passwords are not being passed.

11. `$ RCP /USER=username /PASSWORD=password host1:[dir]source_file "host2:/usr"`

    Copies a file from an OpenVMS host to a UNIX host, following the directory specifications. The same user name and password are used on both remote systems.
   ```

1.10 Incoming RLOGIN Proxies

UCX now supports incoming RLOGIN proxies. If you set the proxy flag for the RLOGIN Service, inbound requesters with valid proxies are automatically logged in, thus eliminating "Username:" and "Password:" prompting. (Users without a valid proxy are still prompted.)

To set the proxy flag, issue:

```
UCX> SET SERVICE RLOGIN /FLAGS=(PROXY)
```

To add a proxy for an incoming RLOGIN user to the Proxy Database, issue:

```
UCX> ADD PROXY user /HOST=host /REMOTE_USER=user [ /PERMANENT ]
```

Example: This command adds to the Proxy Database the proxy REMOTE_UGLY_DUCKLING for incoming RLOGIN user cygnet at host swan.

```
UCX> ADD PROXY REMOTE_UGLY_DUCKLING -
_UCX> /REMOTE_USER="cygnet" /HOST="babyswan"
```

2
Corrected Problems

The following problems in previous releases of the Digital TCP/IP Services for OpenVMS (UCX) software are fixed in Version 4.0A.

2.1 TELNETSYM Print Queue Problem

When TELNETSYM is trying to establish a link to a nonresponding printer, if you issue a DCL DELETE /ENTRY command to stop the job, the print queue "hangs."

2.2 Kernel

The CREATE DIRECTORY and CREATE CONTAINER commands now work properly. The container is created correctly, even with a wildcard proxy host specification. In previous releases, this problem occurred with CREATE CONTAINER:

- CREATE fails if you specify the owner of the container, but do not have an explicit host list defined for the proxy entry that matches the /USER_NAME qualifier.
- CREATE also fails if you specify a user name without a host.

```
UCX> SHOW PROXY USER1

VMS User_name     Type      User_ID    Group_ID   Host_name

USER1             OND         19789          18   *

UCX> CREATE CONTAINER [.CFS] /ROOT=775 -
_UCX> /UID=19789 /OWN=USER1 /USER=USER1
%UCX-E-CFSERROR, Error processing UCX filesystem request
-UCX-E-DATAPROB, Error accessing the UCX$PROXY database
-SYSTEM-F-INSFARG, insufficient call arguments


UCX> CREATE CONTAINER [.CFS] /ROOT=775 -
_UCX> /UID=19789 /OWN=USER1 /USER=USER1 /HOST=CRN1
%UCX-E-CFSERROR, Error processing UCX filesystem request
-UCX-E-DATAPROB, Error accessing the UCX$PROXY database
-UCX-W-NORECORD, Information not found
-RMS-E-RNF, record not found
```

After modifying the proxy for USER1:

```
UCX> SHOW PROXY USER1

VMS User_name     Type      User_ID    Group_ID   Host_name

MARYANNE          OND         19789          18   crn1
MARYANNE          OND         19789          18   CRN1
```

```
UCX> CREATE CONTAINER [.CFS] /ROOT=775 -
_UCX> /UID=19789 /OWN=USER1 /USER=USER1 /HOST=CRN1
%UCX-S-CONTACREATE, Container   created
-UCX-S-DIRCREATE, Directory ROOT created
-UCX-I-VMSUSER, VMS user: USER1
-UCX-I-ULTRIXID, UID: 19789, GID: 18
```

2.3 BIND Server

The following BIND Server problems have been corrected:

- The SHOW NAME_SERVICE /STATISTICS and SHOW NAME_SERVICE /QUERY commands now work on both the OpenVMS Alpha and OpenVMS VAX operating systems.
- An undefined BIND server caused ACCVIO on startup. On Alpha systems, UCX$STARTUP ran into ACCVIO when the bind resolver was enabled and no BIND servers were defined. This problem has been corrected in this release.

2.4 NFS

The following NFS problems are fixed:

- NFS Server

  When you export a directory to the world*(host=*)* and then mount it, NFS no longer incorrectly stops writing errlog and OPCOM messages.

- NFS Client
  - Directory updates made locally at the server are no longer delayed in becoming visible at the client regardless of the values for the directory cache and attributes cache timeouts.
  - GRPPRV privilege is no longer required to get access to a file owned by a same-group user with group access allowed.
  - Fixed: The DCL SET DIRECTORY /VERSION_LIMIT command for a container file system creates an ADF, which causes a problem when you delete the directory. The correct way to delete this directory is to also delete its ADF. However, issuing an rmdir (DELETE DIRECTORY) command on both the directory and the ADF caused the ADF deletion to fail.

3

Known Problems and Restrictions

The Digital TCP/IP Services for OpenVMS (UCX) Version 4.0A software has the following problems and restrictions.

3.1 Network Time Protocol (NTP) Service

The default system time is in GMT. The logical name UCX$NTP_TZ stores the UT offset (in seconds). For example, in a location with Daylight Savings Time (EST5EDT4), the UCX$NTP_TZ definition is -040000.

For time-zone offset information, define the logical name UCX$NTP_TZ before you start NTP. Edit the following line in SYS$COMMON:[SYSMGR]UCX$NTPD_STARTUP.COM:

```
$ DEFINE /SYSTEM UCX$NTP_TZ -040000
```

Change the value to your time-zone offset and delete the semi-colon.

If the NTP Service is enabled, turn on your privileges and stop NTP by issuing the following:

```
$ @SYS$MANAGER:UCX$NTPD_SHUTDOWN.COM
```

To restart NTP issue the following:

```
$ @SYS$MANAGER:UCX$NTPD_START.COM
```

3.2 TELNET Print Symbiont (TELNETSYM)

When TELNETSYM is trying to establish a link to a nonresponding printer, if you issue a DCL DELETE /ENTRY command to stop the job, the print queue "hangs."

To reset the queue, type:

```
$ STOP /QUEUE /RESET
```

3.3 SMTP

SMTP does not set the RFC 822 CC: header for outbound mail. SMTP does not work correctly when a user, sending mail to multiple users, mixes local and remote addresses. As a workaround, at the To: prompt, use the following syntax:

```
$ MAIL
MAIL> SEND
To: SMTP%"addr1,addr2,addr3"
```

Each address is a fully qualified SMTP address. For example, mailing a copy of the same file to:

- Users julie and mark on the local system
- User john on remote host beach.ocean.com

- User ramesh on remote host lake.tarn.edu

```
$ MAIL
MAIL> SEND MEMO.TXT
To:   julie,mark,smtp%"john@beach.ocean.com,ramesh@lake.tarn.edu"
Subj: for your review
```

To use SMTP to send to the local recipients, keep the local recipients separate from the non-local recipients. At the To: prompt, issue:

```
To:SMTP%"julie,mark",smtp%"john@beach.ocean.com,ramesh@lake.tarn.edu"
```

3.4 Installation and Configuration

The UCX$CLEANUP procedure states that there are no more questions. Actually, there might be two additional questions after this message.

When you configure NTP, UCX$CONFIG.COM should write out instructions about changing your NTP configuration template file to reflect your actual NTP Server peers. For EFT1, this information is missing. The EFT2 kit will fix this omission. The instructions are now in Section 2.1 of these Release Notes.

3.5 NFS Client and Server - OpenVMS-to-OpenVMS mode

- BACKUP/VERIFY - might issue spurious verification error messages.
- Remotely accessing CMS libraries via NFS - not recommended.
- NFS Client with OpenVMS POSIX - not supported. You cannot MAP remote file systems for use by OpenVMS POSIX applications via Digital TCP/IP Services for OpenVMS's NFS Client.

3.6 RCP (Remote Copy)

How some of the RCP error and status messages appear to you depends on the remote host's implementation. If a remote user supplies incorrect information, the error message he gets is unpredictable. Regardless of the exact text, however, it always informs you that the problem is either an invalid user name or password at the remote system. For example:

```
$ RCP /PRESERVE /LOG /USER=AC LOGIN.COM -
_$ "LORAN:SYS$SYSDEVICE:[ARVIND]AC.DAT"
UCX$RSHD - Permission denied - VMS user:AC
%RCP-E-INVRESP, invalid response from remote system

$ RCP /USER=ARVIND /PASS=AC LOGIN.COM -
_$ "LORAN:SYS$SYSDEVICE:[ARVIND]AC.DAT"
INTERnet ACP AUXS failure  Status = %LOGIN-F-INVPWD

$ RCP /PRESERVE /LOG /USER=AC LOGIN.COM -
_$ "HAGELN:SYS$SYSDEVICE:[ARVIND]AC.DAT"
Login incorrect.

%RCP-E-CONHST, error connecting to remote host HAGELN.UCX.LKG.DEC.COM

$ RCP /USER=ARVIND/PASS=AC LOGIN.COM -
_$ "DOT:SYS$SYSDEVICE:[ARVIND]AC.DAT"
login information not recognized at remote node

%RCP-E-CONHST, error connecting to remote host DOT

$ RCP NEWSYS:LOGIN.COM "arv@loran:abc.dat"
```

```
%RCP-E-INVRESP, invalid response from remote system
INTERnet ACP AUXS failure  Status = %NONAME-E-NOMSG

$ RCP /USER=ARV NEWSYS:LOGIN.COM "LORAN:AC.DAT"

UCX$RSHD - Permission denied - VMS user:ARV
```

When you interactivly invoke RCP, you get the following prompts:

```
From
To
_Option
```

The valid choices are:

```
From -r
To -p
```

If you first answer the _Option prompt, RCP prompts you again (one prompt for -r and the other for -p. You can also answer each prompt by entering Return.

3.7 NFS Client

The Master File Directory (MFD) of an OpenVMS server's disk is not world-readable.

Both UNIX and OpenVMS NFS Clients can mount the MFD. The following are sample Mounts from UNIX and OpenVMS NFS Clients:

- On the NFS Server: `UCX> SHOW MAP Dynamic Filesystem Map Pathname Logical File System/dkb100 DKB100:`
- Mounting from the UNIX Client: `# mount grackle:/dkb100 /mnt`
- Mounting from the OpenVMS Client: `UCX> MOUNT DNFS1: /HOST="nfs_server" /PATH="/dkb100"`

The OpenVMS Client always reads the MFD. However, if the MFD is protected, the operation fails. If the MFD of a disk you are mounting is read-protected against the UIC to which the remote OpenVMS clients are mapped ("destination UIC"):

- Do not mount the MFD.
- Mount the lower-level directories.

  They are readable by the destination UIC. The following are sample directory listings from UNIX and OpenVMS NFS Clients:

  - Getting a listing from the UNIX Client. This command fails (by design) because the MFD is protected: `# ls /mnt`
  - Getting a listing from the OpenVMS Client. This command fails (by design) because the MFD is protected: `$ DIRECTORY DNFS1:`
  - Getting a listing from the UNIX Client. This command executes because the client does not read the MFD: `# ls /mnt/mydir`
  - Getting a listing from the UNIX Client. This command also fails (unintentionally) because the OpenVMS client reads the MFD: `$ DIRECTORY DNFS1:[MYDIR]`

3.8 Digital TCP/IP Services for OpenVMS on VMScluster Systems

The Digital TCP/IP Services for OpenVMS configuration procedure, UCX$CONFIG, creates node-specific data on a VMScluster in SYS$COMMON. UCX$CONFIG does not configure this data in SYS$SPECIFIC on a per node basis.

Node-specific data is identified by the member's node name, which is defined in SCSNODE in SYSGEN. If you change your node name in SYSGEN, follow these steps:

1. Delete the node-specific information.
2. Make the changes in SYSGEN.
3. Rerun the UCX$CONFIG procedure.

Avoid binding a socket to a cluster alias. Because the cluster alias is distributed in time, a socket bound to it has periods during which it is not fully active. A UDP socket may stop to receive data, while a TCP socket may stop both to transmit and receive data.

For receiving data destined to the cluster alias, use the binding to the IP address 0 (UCX$C_INADDR_ANY).

3.9 Management Commands

The management commands have the following restrictions:

- SET NOROUTE command - Partial wildcards are not valid, for example:
  `UCX> SET NOROUTE route*`
- SET SERVICE command - When you modify parameters to a service, disable and re-enable the service for the modifications to take effect.
- DISABLE SERVICE command - Except for TELNET and Remote Login, the DISABLE SERVICE command disables the specified service, but does not stop the current process, if one exists.

  To stop and restart the current process, follow these steps:

  1. Wait until the process exits or stop it with the DCL STOP PROCESS /ID=n command.
  2. Issue ENABLE SERVICE.

3.10 Sun RPC

The Sun RPC programming interface enhanced for Digital TCP/IP Services for OpenVMS Version 4.0 uses the DEC C programming language. You cannot develop software using the new Sun RPC and then compile it with VAX C.