HP DCE for OpenVMS Alpha and OpenVMS I64

Release Notes

January 2005

This document contains the release notes for HP Distributed Computing Environment (DCE) for OpenVMS Alpha and OpenVMS I64.

Revision/Update Information:	This is a new manual.
Operating System:	OpenVMS Alpha Version 7.3-2 and higher
	OpenVMS I64 Version 8.2
Software Version:	HP DCE for OpenVMS Version 3.2

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 OpenVMS documentation set is available on CD-ROM.

This document was prepared using VAX DOCUMENT, Version 2.1.

Contents

1 Services HP DCE for OpenVMS Offers 1
1.1 New Features in Version 3.2 2
1.2 New Features in Version 3.1 3
2 Contents of the Kits 4
2.1 Runtime Services Kit 5
2.2 Application Developer’s Kit 6
2.3 CDS Server Kit 6
2.4 Security Server Kit 6
3 Installation/Configuration Prerequisites 7
3.1 DCE Upgrade & Configuration 7
3.2 DCE Installation and Configuration 8
4 Troubleshooting 8
5 Updates to the System Login File 8
6 Sizing for a Large Number of Users 9
7 Support for Applications 9
8 Using TCP/IP Services for OpenVMS (UCX) with DCE 10
8.1 cdsLib Service Definition 10
9 Using MultiNet with DCE 11
10 Using TCPware with DCE 12
11 Kerberos 13
12 Windows NT LAN Manager 13
13 Linking RPC Stub Modules into Shareable Images 13
13.1 Errors Creating a Shareable Image 14
13.2 Errors Linking Against a Shareable Image 15
13.3 Errors Activating Shareable Images 15
14 Restrictions and Known Bugs 17
14.1 Kernel Threads and UPCALLS Support 17
14.2 DCE Applications Need Not Be Relinked 17
14.3 Integrated Login and OpenVMS External Authentication . . . 17
14.4 Minimum Global Pages 18
14.5 RTI (Remote Task Invocation) RPC 18
14.6 Format of X.500 Cell Names 18

14.7 Shutting Down HP DCE for OpenVMS Before Reinstallation 19
14.8 Configuring a CDS Replica Clearinghouse 19
14.9 Reconfiguring a CDS Replica Clearinghouse 19
14.10 Privileged User Refreshing Credentials 20
14.11 Support for Integrated Login Before DCE Startup on OpenVMS Systems 20
14.12 Support for Integrated Login Before DCE Startup on OpenVMS Workstations 20
14.13 32-Character Restriction on DCE Principal Names for Integrated Login 21
14.14 Running DCE IMPORT in Batch Mode Without Password . . 21
14.15 Potential Integrated Login and SYSGEN Problems 21
14.16 Support for Packet Privacy 23
14.17 DCE IDL Compiler and C++ Exceptions 23
14.18 Automatic Registration of Servers 23
14.19 Support for sigwait() 23
14.20 Compiling Stubs on Alpha 24
14.21 Using the -cpp_cmd (/PREPROCESS) IDL Compiler Option on OpenVMS Alpha 24
14.22 POSIX 24
14.23 C RTL Routine Sleep Not Thread Safe 24
14.24 Ordering of System Startup Procedures 24
14.25 Case-Sensitivity of DCE Utilities 25
14.26 CDSCP and DCECP Commands Requiring a Local Server 25
14.27 DCE command line programs fail with SMG error 25
14.28 Dumping the CDS Cache 26
14.29 CDS Clerk Failing on UCX Shutdown 26
14.30 Global Directory Agent Configuration 26
14.31 Changes to RPC Shutdown 28
14.32 IDL error when installing DCE 28
14.33 Port Error during DCE configuration 28
14.34 Problems with Sun Solaris DCE system as CDS master 29
14.35 Compile Warning in Example Programs 29
14.36 "Missing" CXX library 29
14.37 Unknown Ethernet Device on Host System 29
14.38 Public Key routines not supported on OpenVMS 30
14.39 Audit Trail Files Require UNIX-like File Specifications 30
14.40 Installation Warnings 31
14.41 Registration failure of DCE$KERNEL image on I64 31
14.42 The utc_mulftime Factor Argument Type 31
14.43 NTLM RPC Support on I64 31

14.44 DCE IDL C++ Application builds on I64 31
15 New API for G_Float/IEEE_Float Support 31
15.1 RPC_SET_LOCAL_FLOAT_DREP 34
16 New APIs for Authenticated RPC 34
16.1 RPC_WINNT_SET_AUTH_IDENTITY 35
16.2 RPC_WINNT_FREE_AUTH_IDENTITY 36
17 New APIs for Impersonation in DCE 36
17.1 RPC_IMPERSONATE_CLIENT 36
17.2 RPC_REVERT_TO_SELF 37
17.3 RPC_REVERT_TO_SELF_EX 37
17.4 Enhanced RPC Security APIs 37
18 The Routing File 38
18.1 Specifying Filenames in the Routing File 38
18.2 Using the Routing File 38
19 Problems Addressed in DCE Version 3.2 Kit 38
20 Problems Addressed in DCE Version 3.1 Kit 39

RELEASE NOTES

HP DCE for OpenVMS Alpha and I64
Version 3.2

HP DCE for OpenVMS Alpha Version 3.2 replaces HP DCE for OpenVMS Alpha Version 3.1. HP DCE for OpenVMS Industry Standard 64 (I64) Version 3.2 is the first release on OpenVMS I64 platform. Version 3.2 is a complete kit that does not require a previous version of HP DCE for OpenVMS for installation. Version 3.2 can be installed on a new system or can be installed as an update to a previous version of DCE.

Note

HP DCE for OpenVMS V3.2 supports OpenVMS I64 Version 8.2 and OpenVMS Alpha Versions 7.3-2 onwards. It is implicit that, DCE is not supported on unsupported Operating System versions. See Sections 1.1 and 14 for new features and important restrictions and known problems.

1 Services HP DCE for OpenVMS Offers

Version 3.2 of HP DCE for OpenVMS consists of the following services:

- Remote Procedure Call (RPC) service provides connections between individual procedures in an application across heterogeneous systems in a transparent way.
- Interface Definition Language (IDL) compiler (required for developing distributed DCE applications).
- Threads service provides user-mode control and synchronization of multiple operations. Threads is packaged with the base operating system.
- Cell Directory Services (CDS) provides a location-independent method of identifying resources within a cell. A cell is the smallest group of DCE systems that share a common naming and security domain.
- Distributed Time Service (DTS) provides date and time synchronization within a cell.

- DCE Security Services provides authentication and authorization within a cell and is based upon MIT’s Kerberos private key encryption system.

1.1 New Features in Version 3.2

Version 3.2 of HP DCE for OpenVMS includes the following new features:

- I64 Support

  Version 3.2 is the first ported version of HP DCE on OpenVMS I64.

- HP DCE V3.2 for OpenVMS Alpha is an enhanced version of its previous Version 3.1. All the new features of HP DCE V3.1 for OpenVMS Alpha and OpenVMS VAX as described in section 1.2 are also applicable to HP DCE Version 3.2 for OpenVMS Alpha and OpenVMS I64.

- DCE RPC Now Supports IEEE Floating Point Type

  DCE RPC for OpenVMS now supports both G_FLOAT and IEEE floating point types on Alpha and I64 platforms. The default floating-point type on Alpha platform remains G_FLOAT. The default floating-point type on I64 is IEEE_FLOAT. DCE RPC Application developers would need to use rpc_set_local_float_drep call in their application for using the non-default floating point type. DCE RPC on VAX platform only supports G_FLOAT type. Please refer section 15 for more details.

- Support for tuning the buffer size of RPC Sockets

  The RPC runtime created sockets do not depend on the system specified socket buffer size values while setting the buffer quotas. They depend on the internally defined macros that set the buffer size to a default value of 4K bytes. RPC Runtime now provides support for tuning the socket buffer size by means of a logical "RPC_USE_DEFAULT_SOCKET_BUFFER_SIZE". Setting the logical "RPC_USE_DEFAULT_SOCKET_BUFFER_SIZE" will allow RPC Runtime to make use of the system default socket buffer size values. The logical can be defined system wide as follows:

  ```
  $DEFINE /SYSTEM RPC_USE_DEFAULT_SOCKET_BUFFER_SIZE 1
  ```

  To restore the original RPC Runtime behavior the logical will need to be deassigned.

- DCE RPC Supports FAILSafe IP

  DCE RPC has been enhanced to work in a FAILSafe IP environment.

- Fixes in DCE V3.2

  The details of the fixes included in HP DCE V3.2 for OpenVMS are described in the section [19] of this document.

1.2 New Features in Version 3.1

HP DCE for OpenVMS VAX and OpenVMS Alpha Version 3.1 is an enhanced version of its previous Version 3.0. All the new features of Compaq DCE for OpenVMS Version 3.0 are also applicable to DCE for OpenVMS Version 3.1. Please refer the release notes of Compaq DCE for OpenVMS Alpha and OpenVMS VAX Version 3.0 for more details on the new features.

Version 3.1 of HP DCE for OpenVMS VAX and OpenVMS Alpha includes the following new features.

- Support for Time Synchronization with DECnet Time Service
  - The DCE Time Service daemon and command line program DTSCP have been enhanced to support the time synchronization with DECnet time service.
  - The following are the DTSCP commands for DCE DTSS to synchronize with DECnet DTSS:

```
$ dtscp set decnet time source true

$ dtscp set decnet time source false

$ dtscp show decnet time source

$ dtscp show decnet local server

$ dtscp show all

In addition to the existing attributes in HP DCE for OpenVMS
Version 3.0, the DECnet related attributes DECnet Courier Role,
DECnet Acting Courier Role, DECnet Time Source would be displayed
in DCE Version 3.1.

$ dtscp help show

In addition to the existing attributes on HP DCE for OpenVMS
Version 3.0, the DECnet related attributes DECnet Courier Role,
DECnet Acting Courier Role, DECnet Time Source will be displayed
in DCE Version 3.1.

$ dtscp help set

In addition to the existing attributes on HP DCE for OpenVMS Version
3.0, the DECnet related attribute DECnet Time Source will be
displayed in DCE Version 3.1.
```

- CDS Migration
  - DCECP command line control program enhanced to support the CDS migration from 3.0 to 4.0.

  - The following command can be used to upgrade older directory version information to 4.0:

    ```
    dcecp> directory modify <dir name> -upgrade -tree
    ```

    ```
    (or)
    ```

    ```
    $ dcecp -c directory modify <dir name> -upgrade -tree
    ```

- TRY_PE_SITE Option

  The DCE$LOCAL:[ETC.SECURITY]PE_SITE. file facilitates Security Server load balancing and locating of Security Server when the CDS server is down. DCE has been enhanced to first use pe_site file and then the CDS namespace for locating the security server. Defining TRY_PE_SITE logical enables this functionality.

  - If TRY_PE_SITE is set to 1, the client will attempt to locate the security replica using the pe_site file. If it fails, the client will use the traditional method of locating the replica through CDS namespace. If both TRY_PE_SITE and bind_pe_site are set to 1, TRY_PE_SITE behavior takes precedence.
  - If the logicals TRY_PE_SITE and BIND_PE_SITE are not set (or) are set to 0, then client will locate the security replica through CDS namespace.

- Performance Improvement in CDS Clerk

  CDS clerk has been enhanced to use TRY_PE_SITE option. To use this enhancement enable the CDSCLERK_TRY_PE_SITE logical. This can result in better performance for CDS operation.

  ```
  $define/system CDSCLERK_TRY_PE_SITE TRUE
  ```

- Fixes in DCE Version 3.1

  The details of the fixes included in HP DCE for OpenVMS Alpha and OpenVMS VAX version 3.1 are described in the section [20] of this document.

2 Contents of the Kits

HP DCE for OpenVMS has four kits available:

- Runtime Services Kit
- Application Developer’s Kit
- CDS Server Kit
- Security Server Kit

Note that the right to use the Runtime Services Kit is included as part of the OpenVMS license. The other kits each require a separate license.

The following sections list the contents of each of these kits.

2.1 Runtime Services Kit

The Runtime Services Kit provide the basic services required for DCE applications to function. The Runtime Services Kit contains the following:

- NTLM (Windows NT LAN Manager) security (OpenVMS Alpha Version 7.2-1 and higher only).
- Authenticated CDS Advertiser and Client Support.
- CDS Browser.
- DCE Control Program (dcecp).
- CDS Control Program (cdscp).
- Authenticated DCE RPC runtime support (supports DECnet, TCP, and UDP).
- RTI (Remote Task Invocation) RPC for the HP ACMSxp TP product.
- Security Client Support.
- Integrated Login.
- A DCE_LOGIN tool for obtaining credentials.
- A RGY_EDIT tool for registry maintenance functions.
- KINIT, KLIST, and KDESTROY Kerberos tools.
- An ACL_EDIT tool for access control lists (ACLs) for DCE objects.
- RPC Control Program (rpccp).
- Name Services Interface Daemon (nsid); also known as the PC Nameserver Proxy.
- Native Kerberos support.
- XDS Directory Services.
- XDS Object Management.

2.2 Application Developer's Kit

The Application Developer's Kit is used by developers to build DCE applications. The Application Developer's Kit contains the following:

- The above contents of the Runtime Services Kit.
- A mechanism to act as a porting aid in mapping MSRPC calls to DCE RPC calls (OpenVMS Alpha Version 7.2 and higher only).
- Required DCE application development header files.
- Interface Definition Language (IDL) compiler.
- DCE IDL Compiler with C++ Extensions (Object-Oriented RPC).
- Generic Security Service (GSSAPI).
- LSE Templates for IDL.
- UUID Generator.
- .H (Include) files and .IDL files for application development.
- Sample DCE applications.

2.3 CDS Server Kit

The CDS Server provides the naming services necessary for DCE clients to locate DCE server applications. The CDS Server Kit includes the following:

- CDS server (cdsd)
- Global Directory Agent (GDA)

The Global Directory Agent (GDA) lets you link multiple CDS namespaces using the Internet Domain Name System (DNS), X.500, or LDAP.

2.4 Security Server Kit

The Security Server provides security services necessary for authenticated RPC calls between DCE client and server applications to function. The Security Server Kit includes the following:

- Security server (secd)
- Tool used to create the security database (sec_create_db)
- Security server administrative tool (sec_admin)

3 Installation/Configuration Prerequisites

If DCE is being installed on a prior version (DCE Version 3.0 or DCE Version 3.1), you will need to shutdown or clean DCE. Failure to do so can result in inability to start DCE. HP DCE V3.2 is the first release for OpenVMS I64 systems and needs to be installed afresh.

- When you install DCE Version 3.2 on OpenVMS Alpha Version 7.3-2, in order to receive all the Version 3.2 RPC runtime updates you MUST also install the RPC Runtime V4.0 Kit that applies to DCE V3.2.

3.1 DCE Upgrade & Configuration

If you are installing a new version of HP DCE for OpenVMS Alpha over an existing version, you do not have to reconfigure DCE after the installation.

If you are running DCE Version 3.0 or Version 3.1 for OpenVMS, you MUST shutdown or clean DCE making use of the following procedure.

```
$ @SYS$MANAGER:DCE$SETUP STOP
or
$ @SYS$MANAGER:DCE$SETUP CLEAN
```

A DCE clean will also cleanup the temporary database files and is always recommended. Shutting down DCE will also shutdown RPC.

As of OpenVMS Version 7.2, DCE RPC is supplied as part of the OpenVMS operating system, and may be running (RPC Only configuration) without a full DCE kit installed. In this situation, you will need to shut down or clean RPC as follows.

```
$@SYS$MANAGER:DCE$RPC_SHUTDOWN
or
$@SYS$MANAGER:DCE$RPC_SHUTDOWN CLEAN
```

HP DCE V3.2 for OpenVMS Alpha and I64 must be installed by running the DCE$INSTALL.COM procedure. Do not install the product by invoking the POLYCENTER Software Installation utility (PCSI) directly. DCE$INSTALL.COM calls PCSI and performs several pre- installation and post - installation tasks. To install DCE, run the DCE$INSTALL.COM procedure as follows:

```
$ @ddcu:DCE$INSTALL.COM [help] ! optional PCSI help
```

See the HP DCE for OpenVMS Alpha and OpenVMS I64 Installation and Configuration Guide for more information.

Make sure that you run the DCE$INSTALL.COM from a valid directory. Errors may occur during the installation that leaves the default directory invalid.

After the installation, enter the following command to start DCE V3.2:

```
$ @SYS$MANAGER:DCE$SETUP START
```

3.2 DCE Installation and Configuration

If you are installing HP DCE for OpenVMS Version 3.2 for the first time, follow the procedure documented below.

As of OpenVMS Version 7.2, DCE RPC is supplied as part of the HP OpenVMS operating system, and may be running (RPC Only configuration) without a full DCE kit installed. In this situation, you only need to perform the following command to shut down RPC.

```
    $ @SYS$MANAGER:DCE$RPC_SHUTDOWN CLEAN
```

Install the RPC Runtime Patch V4.0 Kit on your system using PRODUCT INSTALL if your running HP OpenVMS Alpha Version 7.3-2.

To install DCE, run the DCE$INSTALL.COM procedure as follows:

```
 $ @ddcu:DCE$INSTALL.COM [help] ! optional PCSI help
```

See the HP DCE for OpenVMS Alpha and OpenVMS I64 Installation and Configuration Guide for more information.

You must configure DCE before starting it. Enter the following command for DCE configuration:

```
    $ @SYS$MANAGER:DCE$SETUP CONFIGURE
```

4 Troubleshooting

A chapter on troubleshooting is part of the HP DCE for OpenVMS Alpha and OpenVMS I64 Product Guide. This chapter includes the following sections:

- General troubleshooting hints
- Time zone and time synchronization problems
- Client/Server Check List

5 Updates to the System Login File

To define foreign commands, have the system manager add the following to your SYLOGIN.COM after the installation:

```
$ If F$SEARCH("SYS$MANAGER:DCE$DEFINE_REQUIRED_COMMANDS.COM")-
.NES. "" THEN @SYS$MANAGER:DCE$DEFINE_REQUIRED_COMMANDS.COM

$ If F$SEARCH("SYS$COMMON:[DCE$LIBRARY]DCE$DEFINE_OPTIONAL_COMMANDS.COM")-
.NES. "" THEN @SYS$COMMON:[DCE$LIBRARY]DCE$DEFINE_OPTIONAL_COMMANDS.COM
```

6 Sizing for a Large Number of Users

The DCE daemons require a number of system resources for each concurrent DCE client or server process. The default number of resources allocated to the daemons is based on a maximum of 70 concurrent users (servers and clients) running on a node. If you are running more than 70 DCE users on a node, you must do the following:

1. Stop DCE if it is running.
2. Define a system-wide logical called DCE$MAX_USERS to the maximum number of users desired. For example, to configure DCE for a maximum of 80 users, enter the following:

   ```
   $ DEFINE/SYSTEM DCE$MAX_USERS 80
   ```

   Add this command to your system startup command file so that it is executed prior to starting DCE.
3. Restart DCE.

7 Support for Applications

The Application Developer's Kit provides support for building DCE applications using DCE Services. It provides Application Programming Interfaces (APIs) to RPC communication services, security services, and CDS name services via the RPC Name Services Interface (NSI). The Application Developer's Kit contains the IDL compiler and Runtime support. The header files and IDL files for developing applications are installed in the following directory:

```
SYS$COMMON:[DCE$LIBRARY]
```

DCE applications must also be linked with the following shareable image:

```
SYS$LIBRARY:DCE$LIB_SHR.EXE
```

This image provides the entry points and global symbol definitions for the DCE API services.

A link options file, SYS$COMMON:[DCE$LIBRARY]DCE.OPT, is also provided. It is recommended that this options file be included when linking your DCE applications. For example:

```
$ LINK PROG,DCE:DCE/OPT
```

Linking applications in this way makes your build procedures more portable between OpenVMS Alpha and OpenVMS I64. It also prevents link environment changes from requiring changes to command files.

To help you port a Microsoft RPC application to the DCE format, a shareable image called SYS$LIBRARY:MSRPC_MAPPING_SHR.EXE can be used to link with the RPC application. This new image provides entry points that map a subset of Microsoft RPC calls to their DCE equivalents. To identify which APIs have been mapped, see the MSRPC_MAPPING.H file. This file must be included in the RPC application.

8 Using TCP/IP Services for OpenVMS (UCX) with DCE

Version 3.2 of HP DCE for OpenVMS Alpha and OpenVMS I64 requires installation of HP TCP/IP Services for OpenVMS Version 5.0 or above.

8.1 cdsLib Service Definition

CDS uses a TCP service definition in the UCX services database. This service defines the port number for CDS client and clerk communication. The DCE$SETUP CONFIGURE operation should properly define this service for you. By default, port number 1234 is used. If your site has another application that has defined a service using port 1234, the CONFIGURE operation will ask you to choose another port number for use with the cdsLib service.

After HP DCE for OpenVMS is configured, should you need to change the port number assigned to the cdsLib service (for example, you want to install an application that needs port 1234), use the following commands:

```
$ UCX SET NOSERVICE "cdsLib"
```

The current service definition is displayed and you are asked if you wish to delete it. Answer YES and enter the following command:

```
$ UCX SET SERVICE "cdsLib" /PORT=nnnn /file=NL: -
  /USER=DCE$SERVER /PROTOCOL=TCP
  /PROCESS=DCE$CDSCLERK
```

Where nnnn is an unused port number to be used by CDS. Note that four additional ports are defined:

- cdsAdver uses port number 1235 for process DCE$CDSADV
- cdsDiag uses port number 1236 for process DCE$CDSD
- kerberos5 uses port number 88 for process DCE$SECD
- ntp uses port number 123 for process DCE$DTS_NTP_PROVIDER

  ```
  $ UCX SHOW SERVICE
  ```

  This command lets you examine the current UCX service definitions.

  The State for all of the DCE services should be Disabled.

Also note that the service definitions in UCX are permanent settings; that is, once defined, they will still be set if UCX is restarted. For this reason, you do not need to put changes to the service definitions in your UCX startup procedure.

9 Using MultiNet with DCE

Version 3.2 of HP DCE for OpenVMS can be used with TGV, Inc.'s MultiNet product in place of HP's TCP/IP Services for OpenVMS. If you want to use MultiNet with HP DCE for OpenVMS, you must contact TGV, Inc. for a copy of MultiNet, which contains support for DCE[1]

Then, follow the installation procedure and choose MULTINET when the installation process prompts you for the specific TCP/IP product you want to use.

Define the following logical for using MultiNet product for TCP/IP communications.

```
$ DEFINE /SYSTEM DCE$INTERNET MULTINET
```

Otherwise, DCE will expect TCP/IP communications to be provided by UCX.

Add or replace the following command to the system startup command procedure (SYS$MANAGER:SYSTARTUP.COM) after the startup commands for the network transports, DECnet and/or HP TCP/IP Services:

```
$ @SYS$STARTUP:DCE$SETUP START
```

To configure DCE with MultiNet, enter the following command:

```
$ @SYS$STARTUP:DCE$SETUP CONFIG
```

The SYSGEN parameter MAXBUF must be set to a value greater than the maximum message size to be transferred between the CDS Clerk and CDS clients. If MAXBUF is not large enough, client processes will hang in an I/O wait state. If this happens, other CDS clients will continue to function and the hung process may be aborted without affecting them. The recommended setting for MAXBUF is 20,000 bytes or greater. (If you have a large CDS database with many directories, you may have to set it even higher.) If DCE processes hang while performing name service requests that transfer larger amounts of data, you probably need to increase the value of MAXBUF as follows:

[1] HP is not responsible for third-party application support. Any issues around third-party IP applications should be directed to those third-party companies and not to HP.

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE ACTIVE
SYSGEN> SET MAXBUF nnnn  ! nnnn = new value for MAXBUF
SYSGEN> WRITE ACTIVE
SYSGEN> USE CURRENT
SYSGEN> SET MAXBUF nnnn  ! nnnn = new value for MAXBUF
SYSGEN> WRITE CURRENT
SYSGEN> EXIT
```

Note that this setting will remain in effect until the next time AUTOGEN is invoked. Make the changes permanent by editing SYS$SYSTEM:MODPARAMS.DAT and adding MIN_MAXBUF = nnnn and then invoking AUTOGEN as described in the installation and configuration guide.

For further information on modifying SYSGEN parameters or on AUTOGEN, refer to the OpenVMS system management documentation.

10 Using TCPware with DCE

Version 3.2 of HP DCE for OpenVMS can also be used with Process Software's TCPware product in place of HP's TCP/IP Services for OpenVMS. If you wish to use TCPware with HP DCE for OpenVMS, you must contact Process Software for a copy of TCPware, which contains support for DCE[1].

Then, follow the installation procedure and choose TCPWARE when the installation process prompts you for the specific TCP/IP product you want to use.

Define the following logical for using TCPWARE product for TCP/IP communications.

```
$ DEFINE /SYSTEM DCE$INTERNET TCPWARE
```

Otherwise, DCE will expect TCP/IP communications to be provided by UCX.

Add the following command to the system startup command procedure (SYS$MANAGER:SYSTARTUP.COM) after the startup commands for the network transports, DECnet and/or DEC TCP/IP Services:

```
$ @SYS$MANAGER:DCE$SETUP START
```

To configure DCE with TCPware, enter the following command:

```
$ @SYS$MANAGER:DCE$SETUP CONFIG
```

11 Kerberos

The DCE Security Server makes UDP port 88 (service name "kerberos5") available for use by native Kerberos clients for authentication.

By default, both DCE Security Server and native Kerberos5 KDC server cannot co-exist in the same system.

Support for native kerberos5 clients has undergone minimal interoperability testing.

12 Windows NT LAN Manager

Another mechanism to provide Authenticated RPC has been added to HP DCE for OpenVMS from Version 3.0 onwards. Support for NTLM (Microsoft's NT LAN manager protocol) has been added in OpenVMS Alpha Version 7.2-1 and higher.

To use Authenticated RPC, a client passes its user security information (credentials) to the client's runtime. The client runtime forwards these credentials to the server runtime through 3-legged protocol exchange. This provides a secure mechanism for authenticating the client, and also allows server impersonation of that client.

To select NTLM security, set the authn_svc parameter of the rpc_binding_set_auth_info call to rpc_c_authn_winnt. More information about manipulation of the data structures involved can be found in Section 17.

13 Linking RPC Stub Modules into Shareable Images

If you build shareable images that contain RPC generated stub modules, you should use a linker options file. PSECT statements in the linker options file are used to resolve differences in the PSECT attributes between the RPC generated object file and the new shareable image. The following sections discuss how to solve problems that can arise when you create, link against, or activate a shareable image that contains RPC generated stub modules.

This section can be summarized as follows:

- Program sections (PSECTs) in shareable images should be SHR,NOWRT or NOSHR,WRT unless the image is installed with privileges.
- Program sections in modules linked against shareable images must match exactly or conflicting PSECT errors will occur.
- Until the program runs, you may have to correct PSECT attributes as far back as the shareable image.

The PSECT attributes of the RPC generated interface specifications (IFspecs) should be set to the following:

```
(GBL,SHR,NOWRT)
```

RPC interface specs usually do not change, so it is rarely required that they be set to a writable PSECT attribute. RPC interface specs are frequently shared. If your shareable image contains more than one cluster and the same interface spec is defined in multiple object modules, these interface specs can be effectively collected into the same global cluster with the GBL PSECT attribute. Note that, in this case, the first module encountered by the linker that defines the IFspec will be used to initialize the value of the IFspec in the shareable image. A map file can help you identify and correct problems with PSECTs and their contents. The contents of any PSECT should be nonzero.

If you find a zero byte PSECT, you may need to explicitly specify the module name in the options file. The module name can be specified directly on its own or as part of the /library/include=( ) statement associated with an object library. PSECTs should not be zero unless they are initialized at runtime, and this presumes that the PSECT is writable (WRT).

13.1 Errors Creating a Shareable Image

The following examples show some of the errors that might occur when you try to create a shareable image with RPC stub object modules.

```
$ link/share/exe=myshr.exe/map=myshr.map -
_$ test1_mgr,test1_sstub,dce:dce.opt/opt
%LINK-I-BASDUERRS, basing image due to errors in relocatable references
%LINK-W-ADRWRTDAT, address data in shareable writeable section in
psect TEST1_V0_0_S_IFSPEC offset %X00000000
in module TEST1_SSTUB file USER:[MY.CODE.DCE]TEST1_SSTUB.OBJ;
$
```

The PSECT name is causing the linker problem. To correct this problem, create an option file including the following line, and place it on your link command line as follows:

```
$ create myopt.opt
PSECT= TEST1_V0_0_S_IFSPEC, shr,nowrt,gbl
ctrl-z
$
$ link/share/exe=myshr.exe/map=myshr.map -
$_ test1_mgr,test1_sstub,dce:dce.opt/opt,myopt.opt/opt
```

This will remove the link problems so that you can create a shareable image. There are still errors in this shareable image whose solutions are shown in the following examples.

13.2 Errors Linking Against a Shareable Image

Once you have a shareable image, you may still see linker problems related to the PSECT attributes between the shareable image and new object files. In the following example, a main routine is linked against the same shareable image from the previous example. The new object module references some of the same variables defined by the RPC stub module.

```
$ link/exec=test1d/map=test1d.map test1_main,sys$input/opt
myshr.exe/share
ctrl-z
%LINK-W-MULPSC, conflicting attributes for psect
TEST1_V0_0_S_IFSPEC
in module TEST1_MAIN file USER:[MY.CODE.DCE]TEST1_MAIN.OBJ;
$
```

If you search the map files of both myshr.map and test1d.map for the PSECT TEST1_V0_0_S_IFSPEC, you will see that the PSECT attributes for this PSECT match; however, the map files are incorrect. The solution to this link problem is to include the PSECT directive in a linker options file for the offending PSECT name. The previous example simply typed in the options from the command line, but you should place these linker statements in a linker option file. The options are typed in from SYS$INPUT in the following example:

```
$ link/exec=test1d/map=test1d.map test1_main,sys$input/opt
PSECT= TEST1_V0_0_S_IFSPEC, shr,nowrt,gbl
myshr.exe/share
ctrl-z
$
```

13.3 Errors Activating Shareable Images

When you run this program, the following results occur:

```
$ run test1d
%DCL-W-ACTIMAGE, error activating image MYSHR
-CLI-E-IMAGEFNF, image file not found SYS$LIBRARY:MYSHR.EXE
$
```

To allow the image activator to check a directory other than SYS$LIBRARY for your new shareable image, you must define a logical name or you must copy your new shareable image into SYS$LIBRARY. In the following example, a logical name is defined and the program is run again with the following results.

```
$ define MYSHR sys$disk:[]myshr.exe;
$
$ run test1d
%DCL-W-ACTIMAGE, error activating image MYSHR
-CLI-E-IMGNAME, image file USER:[MY.CODE.DCE]MYSHR.EXE;
-SYSTEM-F-NOTINSTALL, writable shareable images must be installed
$
```

The problem is in the myshr.exe image: myshr.exe has PSECTs whose PSECT attributes specify both SHR and WRT. The solution is to add the correct PSECT attributes to the myshr.opt options file which is used to build the myshr.exe shareable image. This can be done on the command line, as follows:

```
    $ link/share/exe=myshr.exe/map=myshr.map -
    $_ test1_mgr,test1_sstub,dce:dce.opt/opt,sys$input/opt
    psect= TEST1_V0_0_S_IFSPEC, shr,nowrt,gbl
    psect= RPC_SS_ALLOCATE_IS_SET_UP, noshr,wrt,gbl
    psect= RPC_SS_CONTEXT_IS_SET_UP, noshr,wrt,gbl
    psect= RPC_SS_SERVER_IS_SET_UP, noshr,wrt,gbl
    psect= RPC_SS_THREAD_SUPP_KEY, noshr,wrt,gbl
    psect= RPC_SS_CONTEXT_TABLE_MUTEX,noshr,wrt,gbl
    psect= TEST1_V0_0_C_IFSPEC, shr,nowrt,gbl
    <ctrl-z>
$
```

All of the PSECTs that existed in the myshr.map mapfile that had SHR and WRT attributes were changed so that the PSECT was either SHR,NOWRT or NOSHR,WRT. The choice depends upon your use of the data item. IFspecs are usually shared and nonwritable. The RPC_SS PSECTs are written and not generally shared among program images linked against the shareable image.

The following example tries to relink the main program again, but another problem occurs:

```
$ link/exec=test1d/map=test1d.map test1_main,sys$input/opt
PSECT= TEST1_V0_0_S_IFSPEC, shr,nowrt,gbl
myshr.exe/share
ctrl-z

%LINK-W-MULPSC, conflicting attributes for psect
TEST1_V0_0_C_IFSPEC
in module TEST1_MAIN file USERE:[MY.CODE.DCE]TEST1_MAIN.OBJ
$
```

Because the PSECT attributes of the TEST1_V0_0_S_IFSPEC PSECT were changed in the shareable image, its reference in test1_main.obj is not correct. To solve this problem, add the correct PSECT attribute. For example:

```
$ link/exec=test1d/map=test1d.map test1_main,sys$input/opt
PSECT= TEST1_V0_0_S_IFSPEC, shr,nowrt,gbl
PSECT= TEST1_V0_0_C_IFSPEC, shr,nowrt,gbl
myshr.exe/share
<ctrl-z>
$
```

In the final example, the test1d program is run and the desired results occur.

```
$ run test1d
ncacn_ip_tcp 16.32.0.87 3314
ncacn_dnet_nsp 63.503 RPC270002590001
ncadg_ip_udp 16.32.0.87 1485
```

14 Restrictions and Known Bugs

The following sections provide details on restrictions and known bugs in this version of HP DCE for OpenVMS.

14.1 Kernel Threads and UPCALLS Support

As of OpenVMS V7.2-1, HP DCE for OpenVMS Version 3.0 and above now supports DCE applications on Alpha built with Kernel Threads and Thread Manager upcalls enabled.

The DCE daemons (dced, secd, cdsd, etc.) are shipped with Kernel Threads disabled. Enabling Kernel Threads and Thread Manager upcalls on these images is not currently supported.

14.2 DCE Applications Need Not Be Relinked

All the new APIs implemented in HP DCE for OpenVMS Version 3.0 and 3.1 are also available in HP DCE for OpenVMS Version 3.2. Although, there any many new APIs implemented from DCE Version 3.0 onwards, existing DCE applications need not be re-linked before they can run on this release. However, if the application developer wishes to use any of the new APIs, then it will be necessary to recompile and relink.

14.3 Integrated Login and OpenVMS External Authentication

As of OpenVMS Version 7.1, the operating system provides support for external authentication via PATHWORKS. DCE Integrated Login is incompatible with this functionality. DCE$SETUP.COM will warn the user if external authentication is enabled on the host system. If Integrated Login is enabled in spite of the warning, external authentication will be disabled and applications that are dependent on external authentication may not function as expected.

14.4 Minimum Global Pages

HP DCE for OpenVMS Alpha and OpenVMS I64 Version 3.2 has the following global pages requirements:

- HP DCE for OpenVMS Alpha now requires 7350 global pages before installation. (Previously the requirement was 6000.)
- HP DCE for OpenVMS I64 now requires 17500 global pages before installation.

Note

The Minimum Global Section requirement before installation of DCE V3.2 on I64 is 90 free global sections. The Disk Space requirement for installation of HP DCE V3.2 on OpenVMS I64 is as follows:

Kit	Disk Space
OpenVMS I64 Runtime Kit (RTK)	101000 blocks
OpenVMS I64 Runtime Kit & ADK	113000 blocks

14.5 RTI (Remote Task Invocation) RPC

RTI RPC is a transactional RPC that is provided for use with HP's ACMSxp TP product. RTI RPC requires OSI TP from the OSI Application Developer's Toolkit.

HP DCE V3.2 for Open VMS I64 does not support RTI RPC. The DCE$SOCKSHR_TPS.EXE no longer ships with the I64 Kit. The image has a dependency on OSI TPS Runtime Library. The OSITP has been retired and does not ship with the OSI Application Developers Tool Kit.

14.6 Format of X.500 Cell Names

X.500 cell names have the form c=country/o=organization/ou=organization unit. X.500 cell names can contain spaces or hyphens if they are enclosed in double quotes, but underscores are never allowed, even if they are enclosed in double quotes. For example, the X.500 cell names /c=us /o=hp/ou="excess cell" and /c=us/o=hp/ou="excess- cell" are allowed, but /c=us/o=hp/ou=excess_cell and /c=us/o=hp/ou="excess_cell" are not allowed.

14.7 Shutting Down HP DCE for OpenVMS Before Reinstallation

If you are installing HP DCE for OpenVMS V3.2 over an existing version of DCE on a common system disk in a OpenVMS Cluster environment, be sure to shut down DCE and RPC on all nodes that share the common system disk before the installation. If you do not shut down DCE and RPC, parts of DCE and your OpenVMS cluster may exhibit undesirable characteristics.

If you are reinstalling HP DCE for OpenVMS V3.2 over a previous version kit and you are using Integrated Login, and if you do not shut down DCE on all nodes that share the common system disk, you can cause the LOGINOUT image to fail to run on all of the nodes that share the common system disk.

You can correct this problem by shutting down and restarting DCE on the affected nodes.

However, if LOGINOUT is not running, you cannot log in; therefore, you must reboot the system to correct the problem.

14.8 Configuring a CDS Replica Clearinghouse

Before you configure a CDS replica clearinghouse, make sure that the system clock is synchronized to within seconds of the CDS master server. To validate the time, use the following command:

```
$ dtscp show local servers
```

This shows the skew between the host and all other DTSservers in the cell.

14.9 Reconfiguring a CDS Replica Clearinghouse

If it becomes necessary to reconfigure or rebuild a host that includes a CDS replica clearinghouse, you may find that the creation of the clearinghouse succeeds but the skulk that is executed immediately after fails. If this happens, you will see the following message:

```
*** The creation of the CDS Replica Clearinghouse has succeeded
*** but the namespace has been left in an inconsistent state.
*** This condition will correct itself in a short period of time.
*** Once the command "cdscp set dir /.: to skulk" can be
*** successfully executed the namespace will be consistent and
*** the replica clearinghouse will be fully operational.
*** In the meantime you can replicate directories.
```

This is a known restriction. The situation will clear itself in about an hour; however, you will not be able to create any other clearinghouses until this condition has been corrected.

If you want to correct the problem immediately, you can restart DCE on the master server. You will then be able to skulk the root directory and add additional clearinghouses.

14.10 Privileged User Refreshing Credentials

When a privileged process creates or refreshes credentials, the owner UIC for the files is [DCE$SERVER]. If a privileged process needs to refresh credentials for an unprivileged process, the privileged process should first change its owner UIC to be the same as the unprivileged process and disable its privileges. Otherwise, the owner UIC for the updated credentials will be [DCE$SERVER], and the unprivileged process may no longer be able to read its own credentials.

14.11 Support for Integrated Login Before DCE Startup on OpenVMS Systems

If your OpenVMS system startup allows interactive logins to occur before DCE is started, the interactive logins that occur before DCE is started will not support Integrated Login. If you interactively log in to OpenVMS before DCE is started, you must specify your OpenVMS username and password. You will not be logged in with DCE credentials. (If you log in after DCE is started on systems where Integrated Login is enabled, it is recommended that you specify your DCE principal name and password at the username and password prompts when using Integrated Login.)

14.12 Support for Integrated Login Before DCE Startup on OpenVMS Workstations

If your OpenVMS system startup allows DECwindows Motif to start up and display the DECwindows login box before DCE is fully started, the first DECwindows login will not support Integrated Login. In this case, Integrated Login will not be supported even if the first login occurs after DCE is up and running.

If DECwindows Motif displays the DECwindows login box before DCE is started, you must specify your OpenVMS username and password. You will not be logged in with DCE credentials. (If the DECwindows login box is displayed on your workstation after DCE is started and Integrated Login is enabled, it is recommended that you specify your DCE principal name and password at the username and password prompts when using Integrated Login.)

14.13 32-Character Restriction on DCE Principal Names for Integrated Login

When you log in to an OpenVMS system that has Integrated Login enabled, you can specify either your OpenVMS username or your DCE principal name at the username prompt. However, the DCE principal name you specify can contain no more than 32 characters. If your principal name and cell name combination contains more than 32 characters, specify the OpenVMS username that is associated with your DCE account instead. (This username is entered in the DCE$UAF file.) You should still enter your DCE password to obtain DCE credentials even if you specify your OpenVMS username.

14.14 Running DCE IMPORT in Batch Mode Without Password

If you run DCE IMPORT in batch mode and you do not supply a password for the DCE account on the command line, the password valid flag incorrectly remains set in the DCE registry. Because a password was not supplied, the flag should indicate password not valid and the user should not be allowed to log in. A scan of the DCE account via RGY_EDIT reveals the incorrect flag setting (password valid when actually the password is not valid). However, the user will not be allowed to log in (which is the correct behavior).

14.15 Potential Integrated Login and SYSGEN Problems

The Integrated Login component of HP DCE for OpenVMS uses the SYSGEN parameter LGI_CALLOUTS. LGI_CALLOUTS must be set to 1 only in the ACTIVE SYSGEN parameter set when DCE is running with Integrated Login enabled. LGI_CALLOUTS must never be set to 1 in the CURRENT SYSGEN parameter set - this would prevent all logins from occurring on a subsequent reboot of the system. The following paragraphs discuss the reasons for this restriction and solutions if the problem occurs.

If Integrated Login is enabled on your system, the DCE startup and configuration procedure, DCE$SETUP.COM, sets the SYSGEN parameter LGI_CALLOUTS to 1 in the ACTIVE SYSGEN parameter set when DCE is started and resets the parameter when DCE is shut down. LGI_CALLOUTS must never be set to 1 in the CURRENT SYSGEN parameter set because, in that case, the next time the system is booted the LGI_CALLOUTS parameter is set in the ACTIVE SYSGEN parameter set before DCE is started. This prevents logins from occurring.

If the ACTIVE value of LGI_CALLOUTS is set to 1 when DCE and Integrated Login are not running, the following error is displayed when LOGINOUT attempts to run (for example, for interactive or batch logins):

```
No logical name match
```

Consequently, all users are prevented from logging in to the system.

This problem can occur if, for example, a SYSGEN parameter is modified in the following way while Integrated Login is enabled. This prevents logins because it causes LGI_CALLOUTS to be set to 1 the next time the system is booted.

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> SET param value
SYSGEN> WRITE CURRENT
SYSGEN> EXIT
$
```

The correct way to modify a SYSGEN parameter is to make the change in MODPARAMS.DAT and then run AUTOGEN. If it is essential to modify a SYSGEN parameter without using MODPARAMS.DAT and AUTOGEN, you must ensure that if you use ACTIVE, you write the parameters into ACTIVE only; and if you use CURRENT, you write the parameters into CURRENT only. Do not copy the ACTIVE parameters into CURRENT.

Following are two examples of acceptable ways to modify a SYSGEN parameter:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SET param value
SYSGEN> WRITE CURRENT
SYSGEN> EXIT
$
```

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE ACTIVE     ! optional, default is ACTIVE
SYSGEN> SET param value
SYSGEN> WRITE ACTIVE
SYSGEN> EXIT
$
```

If you cannot log in because LGI_CALLOUTS is set to 1 and DCE is not running, there are two solutions, as follows:

- If you are already logged into the system, use SYSGEN to correct the problem.

  ```
  $ RUN SYS$SYSTEM:SYSGEN
  SYSGEN> SET LGI_CALLOUTS 0
  SYSGEN> WRITE ACTIVE
  SYSGEN> EXIT
  $
  ```

- Reboot the system with a conversational boot and ensure the LGI_CALLOUTS parameter is zero.

  ```
  SYSBOOT> SET LGI_CALLOUTS 0
  SYSBOOT> C
  ```

14.16 Support for Packet Privacy

HP DCE for OpenVMS supports the rpc_c_prot_level_pkt_privacy level of data encryption as of this baselevel. Recent changes in the government’s encryption regulations allow this functionality to be provided in the base DCE kit, as opposed to a separate product (as was done in previous versions of HP DCE for OpenVMS). See the documentation on rpc_binding_set_auth_info for details.

14.17 DCE IDL Compiler and C++ Exceptions

A client using the DCE IDL compiler with C++ extensions invokes methods on objects that causes IDL generated client stub code to be invoked. By default, communications errors or remote faults that occur during the stub’s processing cause exceptions to be raised using the DCE Threads exception handling mechanism. Therefore, C++ code that needs to catch and respond to these exceptions must also use the DCE Threads exception handling mechanism.

Some, but not all, C++ compilers have built-in language support for exceptions. Exceptions are not supported in older versions of the DEC C++ for OpenVMS compilers. C++ application code that processes exceptions returned from DCE IDL stubs should continue to use DCE Threads exceptions if using compilers without exceptions support.

You can avoid the raising of exceptions from DCE IDL stubs by using the [comm_status] and [fault_status] ACF attributes. For more information, see the Guidelines for Error Handling chapter in the DCE Application Development Guide.

14.18 Automatic Registration of Servers

In the IDL compiler, servers are now automatically registered by server stubs. If you call rpc_server_register_if( ), the "already registered" status is returned. (Remove the call to rpc_server_register_if( ) from the server.cxx file before you build the example programs in Chapter 15 of the HP DCE for OpenVMS Alpha and OpenVMS I64 Product Guide.)

14.19 Support for sigwait( )

The DCE Application Guide and DCE Reference Guide include incorrect information about support for sigwait( ).

DECthreads does not support sigwait( ) on the OpenVMS platform.

14.20 Compiling Stubs on Alpha

If a stub is compiled on Alpha with optimization switched on, it may not handle exceptions correctly, depending on the version of DEC C. Therefore, on Alpha, you should compile stubs with optimization switched off, unless you are sure that the version of DEC C that is on your system handles this situation correctly.

14.21 Using the -cpp_cmd (/PREPROCESS) IDL Compiler Option on OpenVMS Alpha

When you specify the -cpp_cmd (/PREPROCESS) option in an IDL command, the IDL compiler preprocesses any IDL or ACF sources by invoking the DEC C compiler with the /PREPROCESS_ONLY qualifier. Because of a bug in some versions of the DEC C compiler on OpenVMS Alpha, the IDL compiler may incorrectly report source line numbers and contents when it reports error messages.

If your IDL and ACF source files do not use C preprocessor directives (such as #define), then you do not need to specify the -cpp_cmd (/PREPROCESS) option. Otherwise, the workaround is to change multiline comments to a series of single line comments.

14.22 POSIX

The OpenVMS POSIX product has been retired, and support for the POSIX command line is not available for HP DCE for OpenVMS Alpha and OpenVMS I64 Version 3.2. The OpenVMS C runtime support for many of the POSIX calls has improved, and most applications should see no change in behavior. Only those applications which require the POSIX command line interface are affected.

14.23 C RTL Routine Sleep Not Thread Safe

The C RTL routine sleep is not thread safe. The sleep call may wake up prematurely if calls to DCE APIs are made at the same time. It is recommended that you use a thread safe mechanism such as pthread_delay_np, pthread_cond_wait, pthread_cond_timedwait, and pthread_cond_signal to delay a thread. For more information on these APIs, please refer to the OSF DCE Application Development Reference Manual.

14.24 Ordering of System Startup Procedures

The order of startup procedures should be as follows: DECnet, TCP/IP software, DCE, then DCE applications.

14.25 Case-Sensitivity of DCE Utilities

Some input to HP DCE for OpenVMS utilities is case-sensitive (for example, CDSCP entity attribute names). Since the DCL command line interface converts all input to uppercase before passing it to a utility, some input to the DCE utilities will need to be enclosed in quotation marks (" ").

When you enter commands directly at DCE utility prompts, you should not use the quotation marks because case-sensitivity is preserved. (Case-sensitivity is not preserved by the Integrated Login utilities DCE$UAF, IMPORT, and EXPORT because these are true native OpenVMS applications.)

14.26 CDSCP and DCECP Commands Requiring a Local Server

There are several CDSCP commands that assume the presence of a CDS server on the local system. These commands will not execute properly in the absence of a local server. At present, CDSCP will return the following error:

```
Failure in routine: cp-xxxxxxx not registered in endpoint map (dce/rpc)

      The affected commands are:

      $ cdscp show server
      $ cdscp disable server
      $ cdscp create clearinghouse <clearinghouse_name>
      $ cdscp delete clearinghouse <clearinghouse_name>
```

DCECP will return the following error:

```
Failure in routine: Error: Binding incomplete (no object ID and no endpoint)

      The affected commands are:

      $ dcecp -c clearinghouse create <clearinghouse_name_list>)
      $ dcecp -c clearinghouse delete <clearinghouse_name_list>)
      $ dcecp -c clearinghouse initiate <clearinghouse_name_list> -checkpoint)
      $ dcecp -c clearinghouse repair <clearinghouse_name_list> -timestamps)
      $ dcecp -c clearinghouse verify <clearinghouse_name_list>)
      $ dcecp -c clearinghouse disable <clearinghouse_name_list>)
```

14.27 DCE command line programs fail with SMG error

If the process has it's UIC set to DCE$SERVER, and does not have the BYPASS privilege set, DCE command line utilities will fail with the following error:

```
error creating SMG virtual keyboard.
%NONAME-E-NOMSG, Message number 00000002
```

The resolution to this problem is to either run under a UIC other than DCE$SERVER, or to set the BYPASS privilege on accounts set to the DCE$SERVER UIC.

This problem does not effect the running of the DCE daemons, only user processes.

14.28 Dumping the CDS Cache

The CDSCP and DCECP commands to examine the CDS cache will fail if CDSCP or DCECP is run under a Process UIC other than [DCE$SERVER].

```
$ cdscp dump clerk cache
Cannot map -1
- check id and protection
An error occured calling a CDS API function. (dce / cds)
```

```
$ dcecp -c cdscache dump
Cannot map -1
- check id and protection
Error: The cache dump failed in an indeterministic mode.
```

To work around this restriction, issue the following DCL command before you invoke CDSCP or DCECP:

```
$ SET UIC [DCE$SERVER]
```

Remember to reset your UIC to its original value after you use this command.

14.29 CDS Clerk Failing on UCX Shutdown

If you issue a SYS$STARTUP:TCPIP$SHUTDOWN command while running DCE, you may get a CDS Clerk failure and an Access Violation. You may then encounter problems restarting the CDS Clerk (and DCE itself) with the DCE$SETUP START command.

The primary problem is that UCX is being shut down while DCE is still active. Since DCE uses UCX, DCE should always be shut down first.

To recover from this problem, you need to shut down DCE first and then restart. Simply trying to restart without first shutting DCE down will not fix the underlying problem. Because temporary files may be left in an indeterminate state, you may also want to perform a DCE$SETUP CLEAN operation before restarting.

14.30 Global Directory Agent Configuration

The Global Directory Agent (GDA) is configured on the OpenVMS node that contains the CDS Master Replica name server. The DNS domain name (for example, zko.dec.com) and the Internet Address of an authoritative DNS Master Bind Server (for example, 16.32.2.11) are required during configuration if you are using DNS Bind style cellnames.

Before access to multiple CDS namespaces is possible, the following are required after the configuration:

1. The Master Bind Server identified during configuration becomes the repository for information the GDA requires to resolve the Internet addresses and binding information needed by CDS to access foreign cell name spaces. This applies to DNS Bind cellnames only. See the Intercell Naming chapter in the HP DCE for OpenVMS Alpha and OpenVMS I64 Product Guide for the binding information content, location, and access.
2. Authenticated access to foreign (intercell) cell name space requires performing the RGY_EDIT cell command. The information needed for the cell command requires coordination with the foreign cell administrator. For more information, see both the Administering a Multicell Environment chapter in the OSF DCE Administration Guide and the Intercell Naming chapter in the HP DCE for OpenVMS Alpha and OpenVMS I64 Product Guide.
3. Before doing the RGY_EDIT cell command, you must first delete the krbtkt account for the foreign cell if one already exists. Similarly, the administrator for the foreign cell must also delete the krbtkt account in the foreign cell’s registry for your cell. For example, if your cell is called first_cell and the foreign cell is called second_cell, then you must run RGY_EDIT on first_cell to delete the account called krbtkt/second_cell, and the administrator on second_cell must delete the registry account called krbtkt/first_cell.

After the cell command, both cell administrators should rerun DCE_LOGIN before attempting authenticated cross-cell requests.

If you are unsuccessful in configuring intercell communication, check for the following:

- The clocks on the systems that are attempting to communicate show times that differ by no more than five minutes. (Use DTS to change the system time once you are running DCE.)
- CDS has the information that should be contained in the CDS_ GDAPointers field in the cell’s root directory. If CDS does not have this information in the cell’s root directory, restart the GDA daemon process (DCE$GDAD) by entering the following commands:

```
$ STOP/ID=xxxxxxxx
$ @sys$manager:dce$setup start
```

where xxxxxxxx is the PID of the DCE$GDAD process.

14.31 Changes to RPC Shutdown

In DCE for OpenVMS Version 1.5, a change was made to disassociate RPC shutdown from DCE shutdown. This was done to allow RPC only applications to remain active while DCE changes were being made.

In DCE Version 1.5, DCE$SETUP stop/clean/clobber did not call the RPC shutdown procedure, and merely gave a warning that RPC would not be shut down. DCE Version 3.0 requires that dced (the new RPC endpoint mapper) be shut down during certain operations. Therefore, the behavior was changed in DCE Version 3.0, and the RPC shutdown procedure is now called from DCE$SETUP.COM. The same is applicable for DCE Version 3.2 as well. This requires the system manager to be aware of any RPC-only applications that may be active at the time of DCE configuration operations.

14.32 IDL error when installing DCE

If installing DCE over an existing implementation, you may see an IDL error if the DCE Application Developer's Kit was previously installed, but is not being installed for the upgrade. The installation is attempting to remove the DCL commands which are associated with the developer's kit from DCLTABLES.EXE, and failing.

This error can safely be ignored - answer NO to the question "Do you want to terminate?".

```
%PCSI-E-MODDELERR, error deleting module IDL_CLD from library
%PCSI-E-OPFAILED, operation failed
Terminating is strongly recommended.  Do you want to terminate?
[YES] n
```

14.33 Port Error during DCE configuration

If the error shown below occurs during DCE configuration, your system has the TCP/IP NTP daemon configured. Since DCE also provides an NTP daemon, you must decide which one you intend to use.

If you choose to use the DCE NTP daemon, then you must disable the TCP/IP NTP daemon via your TCP/IP configuration program before you can enable the DCE one.

If you choose to use the TCP/IP NTP daemon, then you can ignore the following error, and answer "Y" to the question about whether you want to proceed.

```
*************************** ERROR ********************************

     Port number 123 is in use by a service other than "ntp".
     Please check configuration! Service "ntp" must use
     port number 123.

*****************************************************************
Press <RETURN> to continue . . .

Do you want to proceed with this operation  (YES/NO/?) [N]?
```

14.34 Problems with Sun Solaris DCE system as CDS master

There are known problems with Sun Solaris Version 2.6 and Transarc DCE Version 2.1 as the CDS master if you are attempting to configure a split server configuration using HP DCE on OpenVMS, Tru64 UNIX or Windows NT. Solaris Version 2.4 and Transarc DCE Version 1.1 work correctly. Contact your DCE vendor for further information.

14.35 Compile Warning in Example Programs

The CXX example programs may produce the following warning on compilation:

```
IDL_ms.IDL_call_h = (volatile rpc_call_handle_t)IDL_call_h;
...............^
%CXX-W-CASTQUALTYP, type qualifier is meaningless on cast type
at line number 117 in file USER$1:[DCE12.EXAMPLES.RPC.IDLCXX.
ACCOUNT]ACCOUNT_SSTUB.CXX;1
```

This warning can be safely ignored.

14.36 "Missing" CXX library

Some versions of CXX may not include the library SYS$LIBRARY:LIBCXXSTD.OLB. If this is the case, this line may be removed from the options file found in SYS$COMMON:[DCE$LIBRARY]DCE_CXX.OPT.

14.37 Unknown Ethernet Device on Host System

If your system is relatively new, it is possible that DCE might not know about the Ethernet device on the system. DCE uses the Ethernet device to obtain an Ethernet address which is used in the generation of UUIDs. If you see errors such as the following:

```
%UUIDGEN-F-RPC_MESSAGE, Received Error Status: "no IEEE 802
                        hardware address (dce / rpc)"
```

then your Ethernet device is not known by DCE.

You can define one additional Ethernet device in the table used by DCE by defining the logical name DCE$IEEE_802_DEVICE to the name of your Ethernet device as shown in the following example:

```
$ DEFINE/SYSTEM DCE$IEEE_802_DEVICE EWA0
```

This will allow DCE to operate using the Ethernet device named EWA0 (a device type of DE500).

14.38 Public Key routines not supported on OpenVMS

DCE public key technology is not currently supported on OpenVMS. The pkc_* routines and classes (pkc_add_trusted_key, etc.) are not in DCE$LIB_SHR.EXE, and will generate undefined symbols if an application which uses them attempts to link.

The Open Group has stated their intention to replace the existing public key technology in DCE with a non-interoperable replacement, based on X.509v3, in a future release.

"Note that there has been such a high volume of change activity in the IETF relative to Public Key Infrastructure (PKI) and Kerberos that the [RFC 68.3] functionality will not be forward compatible with this Specification. Therefore, current users of DCE 1.2.2-based products with [RFC 68.3] functionality should refrain from deploying the public key based login support."[1]

For this reason, HP is not supplying the obsolete public key functionality in DCE from OpenVMS Version 3.0 onwards. For additional information on the status of public key in DCE, see the Open Group's DCE website at:

```
http://www.opengroup.org/tech/dce/
```

[1] Draft Technical Standard - DCE 1.2.3 Public Key Certificate Login, Draft 0.8, The Open Group, August 1998

14.39 Audit Trail Files Require UNIX-like File Specifications

The command to show the DCE audit trail files requires a UNIX style file specification. For example:

```
$ dcecp -c audtrail show /dcelocal/var/audit/adm/central_trail
```

14.40 Installation Warnings

Some systems may see warnings during DCE installation, as shown below:

```
   The following product will be installed to destination:
DEC AXPVMS DCE V3.2
   DISK$MOOSE2_SYS:[VMS$COMMON.]
   %PCSI-I-RETAIN, file [SYSUPD]DTSS$TIMEZONE_RULES.DAT was not
   replaced because file from kit does not have higher generation
   number
```

These warnings can be safely ignored. They indicate that certain files which may also be provided by OpenVMS are newer than the files in the DCE kit.

14.41 Registration failure of DCE$KERNEL image on I64

DCE startup on I64 reports the following error message.

```
%RPI-F-UNSUPPORTED, Image registration is not supported
in V8.2 on Integrity systems.
```

```
You may safely ignore this message.
```

14.42 The utc_mulftime Factor Argument Type

The input argument, factor, for the DTSS API routine utc_mulftime must be a IEEE_FLOAT type on I64 and G_FLOAT type on Alpha. You can use either CVT$FTOF or CVT$CONVERT_FLOAT to convert the factor argument to appropriate floating point type before calling utc_mulftime.

14.43 NTLM RPC Support on I64

Authenticated RPC over NTLM (Microsoft NT LAN Manager Protocol) has not been tested in this release, as the infrastructure on which DCE RPC depends on is not available on I64.

14.44 DCE IDL C++ Application builds on I64

The C++ Compiler on I64 does not allow compilation of multiple sources separated by a comma list. The IDL C++ Application build procedures compiling multiple sources will need to be modified to build the source files individually.

15 New API for G_Float/IEEE_Float Support

HP DCE V3.2 for OpenVMS now supports both G_FLOAT and IEEE floating point types on I64 and Alpha platforms.

Use the floating point types consistently in a single RPC application. Different RPC applications, each using different floating point types, can be run on a single system.

On I64 Systems:

By default DCE uses IEEE_FLOAT type on I64 systems i.e. DCE applications built for I64 systems would use IEEE_FLOAT floating point types.

Use the following steps for using the G_FLOAT floating point type in RPC applications developed on the C and C++ language:

1. Call the new API function rpc_set_local_float_drep(RPC_APPLICATION_FLOAT_TYPE, &status) before calling any RPC runtime functions. The constant RPC_APPLICATION_FLOAT_TYPE is automatically defined to the floating point type specified on the compiler command line qualifier.
2. Compile the RPC application programs using the compiler qualifier /FLOAT= G_FLOAT.
3. Use the appropriate IDL compile option while building the stubs for:
   - C applications: -CC_CMD "CC/FLOAT=G_FLOAT"
   - C++ applications: -CPP_CMD "CXX/FLOAT=G_FLOAT"
4. Link the RPC applications using the appropriate DCE options file for:
   - C applications: DCE.OPT
   - C++ applications: DCE_CXX.OPT

To use IEEE_FLOAT floating point type in RPC applications developed on the C or C++ language:

1. Compile the RPC application programs using the Compiler Qualifier /FLOAT=IEEE_FLOAT (default option).
2. Link the RPC application with DCE.OPT or with DCE_CXX.OPT.

On Alpha Systems:

By default DCE uses G_FLOAT type on Alpha systems i.e. DCE applications built on Alpha systems would use G_FLOAT floating point types.

The following are the details for using the IEEE_FLOAT floating point type in RPC applications developed on the C and C++ language:

1. Call the new API function rpc_set_local_float_drep(RPC_APPLICATION_FLOAT_TYPE, &status) before calling RPC runtime functions. The constant RPC_APPLICATION_FLOAT_TYPE is automatically defined to the floating point type specified on the compiler command line qualifier.
2. Compile the RPC application programs using the compiler qualifier /FLOAT= IEEE_FLOAT.

3. Use the appropriate IDL compile option while building the stubs for:
   - C applications: -CC_CMD "CC/FLOAT=IEEE_FLOAT"
   - C++ applications: -CPP_CMD "CXX/FLOAT=IEEE_FLOAT"
4. Link the RPC applications using the appropriate DCE options file for:
   - C applications: DCE.OPT
   - C++ applications: DCE_CXX.OPT

   To use G_FLOAT floating point type in RPC applications developed on the C or C++ language:

1. Compile the RPC application programs using the CC or C++ Compiler Qualifier /FLOAT=G_FLOAT (default option).
2. Link the RPC application with DCE.OPT or with DCE_CXX.OPT.

Please also refer the HP C++ Release Notes documentation for any known restrictions or problems with running C++ applications that have been compiled using non-native floating point type.

15.1 RPC_SET_LOCAL_FLOAT_DREP

NAME
rpc_set_local_float_drep: This function sets the float type in
the runtime to the one with which the application is being compiled.

SYNOPSIS:

```
#include <dce/rpc.h>
void rpc_set_local_float_drep (
unsigned8 float_drep,
unsigned32 *status);
```

PARAMETERS

```
 INPUT
  unsigned8  float_drep
 The parameter should always be passed
 as using the macro "RPC_APPLICATION_FLOAT_TYPE".
 This macro will be defined to 0 or 1
 based on the compilation option specified
 for the float type.

 OUTPUT
  unsigned32 *status
 The routine will always return "rpc_s_ok" status.
```

DESCRIPTION:

The routine rpc_set_local_float_drep allows the RPC application to set the floating point type being used by the application. Only G_FLOAT and IEEE_FLOAT floating types are supported. This routine if used, should be placed before any other API calls to the RPC runtime. The first parameter float_drep should be passed using the macro RPC_APPLICATION_FLOAT_TYPE that is defined in IDLBASE.H header file. This macro will be set to appropriate value based on the /FLOAT compilation option.

This routine can be used only on Alpha and I64 and will not be supported on VAX.

RETURN TYPE:
No value is returned.

16 New APIs for Authenticated RPC

The following APIs are included in DCE Version 1.5 and above to manipulate the sec_winnt_auth_identity structure. They are supported on OpenVMS V7.2-1 onwards.

16.1 RPC_WINNT_SET_AUTH_IDENTITY

NAME

```
rpc_winnt_set_auth_identity - This function is called by the
client RPC application to allocate and populate a WINNT
auth_identity structure to be used as a parameter to
rpc_binding_set_auth_info().
The caller must use the rpc_winnt_free_auth_identity()
function to free the WINNT auth_idenity. The strings that are
passed in may be ASCII or Unicode (UCS-4) strings. The input
flag will tell which type of strings they are.
```

SYNOPSIS

```
#include <dce/rpc.h>

PUBLIC void rpc_winnt_set_auth_identity (
rpc_winnt_auth_string_p_t       Username;
rpc_winnt_auth_string_p_t       Password;
rpc_winnt_auth_string_p_t       Domain;
unsigned 32                 CharacterSetFlag;
rpc_auth_identity_handle_t      *auth_identity;
unsigned32                      *stp)
```

PARAMETERS

```
INPUT
username - Pointer to a null terminated string
           containing username.
password - Pointer to a null terminated string
           containing password.
domain   - Pointer to a null terminated string
           containing domain.

CharacterSetFlag

SEC_WINNT_AUTH_IDENTITY_UNICODE
        2 byte Unicode (UCS-2)
SEC_WINNT_AUTH_IDENTITY_ANSI
        regular old ASCII (ISO8859-1)
OUTPUT
auth_identity - Pointer to a pointer to WINNT
                auth_identity structure.
stp           - Pointer to returned status.
```

Note

Be sure to allocate space for three strings (username, password, domain). The string variables will probably be pointers of type unsigned_char_t if the strings are ASCII or pointers of type wchar_t if the strings are Unicode (UCS-2). If the domain string is a valid empty string, then the domain of the computer will be used.

16.2 RPC_WINNT_FREE_AUTH_IDENTITY

```
NAME
rpc_winnt_free_auth_identity - This function is called by the client RPC
application to free a a WINNT auth_identity structure that was previously
allocated by a call to rpc_winnt_set_auth_identity().

SYNOPSIS
#include <dce/rpc.h>

PUBLIC void rpc_winnt_free_auth_identity (
rpc_auth_identity_handle_t  *auth_identity,
unsigned32                       *stp)

PRAMETERS
INPUT
auth_identity - Pointer to a pointer to WINNT auth_identity structure.
On output auth_identity will be set to NULL.
OUTPUT
stp             Pointer to returned status.
```

17 New APIs for Impersonation in DCE

The following APIs are included in DCE Version 1.5 and above to support server impersonation of a client. This means that the server runs with the security credentials of the client, and all of the capabilities of the client belong to the server.

17.1 RPC_IMPERSONATE_CLIENT

```
NAME
rpc_impersonate_client - This function is called by the server application
to allow the current server thread to run with all of the client privileges.

SYNOPSIS

#include <dce/rpc.h>
void rpc_impersonate_client(
rpc_binding_handle_t binding_handle,
unsigned32  *status)

PARAMETERS

INPUT
binding_handle - Specifies a server-side call
   handle for this RPC which represents the
   client to impersonate.

OUTPUT
status - Specifies a pointer to an unsigned 32
  bit integer that holds a status code.
```

17.2 RPC_REVERT_TO_SELF

```
NAME
rpc_revert_to_self -  This function is called by the
server application to revert back to its original
security context after impersonating a client.

SYNOPSIS

#include <dce/rpc.h>
rpc_revert_to_self(*st)

PARAMETERS

INPUT
NONE

OUTPUT
st - Specifies a pointer to an unsigned 32 bit
         integer that holds a status code.
```

17.3 RPC_REVERT_TO_SELF_EX

```
NAME

rpc_revert_to_self_ex - This function is called by the server
application to revert back to its original security context
after impersonating a client.  This acts as a call to
rpc_revert_to_self();

SYNOPSIS

#include <dce/rpc.h>

rpc_revert_to_self_ex(
rpc_binding_handle_t     binding_handle,
unsigned 32             *status)

PARAMETERS

INPUT
call handle - This parameter is ignored.

OUTPUT
status - Specifies a pointer to an unsigned 32 bit
         integer that holds a status code.
```

17.4 Enhanced RPC Security APIs

For more information on existing enhanced RPC security APIs, see the HP DCE for OpenVMS Alpha and OpenVMS I64 Reference Guide.

18 The Routing File

To use routing file services on OpenVMS, you will need to define the following logical name for the process or the system for which logging information is desired:(Syntax is exact for the routing file).

```
$ DEFINE/SYS DCE_SVC_ROUTING_FILE "DCE_LOCAL/VAR/SVC/ROUTING."
```

This will enable DCE applications to find and interpret the routing file and direct any output to the location specified in the routing file.

You can also set the number of buffered writes to perform before data is flushed to the file, as shown below:

```
$ DEFINE/SYS DCE_SVC_FSYNC_FREQ 10
```

The example above will flush the buffer every 10 writes.

18.1 Specifying Filenames in the Routing File

The OpenVMS routing file uses UNIX style filenames when specifying output log files. You can see examples of this in the current routing file that is found in the directory dce$common:[var.svc]routing. The DCE code that reads the routing file uses colons and forward slashes to parse the routing file data lines for output files.

18.2 Using the Routing File

The routing file contains examples of how to set up logging for various components. See the routing file itself for additional information. The routing file can be found in DCE$COMMON:[VAR.SVC]ROUTING.

19 Problems Addressed in DCE Version 3.2 Kit

Following new problems have been addressed in DCE V3.2 for Open VMS.

- IDL Compiler generates "IDL-W-OUTDIRIGN" messages during STDL(IDL) compilation.
- When DCE Setup clean operation is attempted with a wrong password for the "cell_admin" account, the clean operation does not terminate the DCE$DTSD daemon.
- KINIT accepts wrong password.
- DCE startup sometimes fails on a client system, with replica enabled after a clean operation. The startup fails with the below error.

  ```
  Warning: Unable to acquire initial login context: Not registered
  in endpoint map".
  ```

- DCE startup fails while starting up DCED on a OpenVMS system where FAILSafe IP is enabled. An error similar to the following gets reported in the DCE$DCED.OUT.

  ```
  2004-06-19-04:46:56.436+08:00I----- dced FATAL dhd general
  MAIN.C;1 1502 0x7bbe4a78

  Initialization (get bindings) failed, status=0x0e12815a
  ```

- DCE Setup clean does not delete all the temporary files. Some stale files are left over by the DCE$SETUP procedure.
- DCE Install procedure leaves unnecessary temporary files.
- License check for DCE Application Developer's Kit is not robust.
- utc_vmslocaltime does not return the correct time for Australia/WEST timezone.
- When a large number of ACMSxp processing servers are stopped and re-started, DCE goes into a hang state and no DCE commands can be executed.
- DCE Application Clients hang with TCP Connection Closure.
- utc_mulftime API produces invalid output on I64 for scaling factor between -0.5 and +0.5.
- Compilation of DCE Applications with UTC.H as an include will throw "CC-E-REDEFSTRUCT" errors. DCE's version of UTC.H does not include all the checks for structure redefinitions.
- A potential denial of service has been identified in DCE$DCED that may allow a remote initiated buffer overflow.

20 Problems Addressed in DCE Version 3.1 Kit

The following problems have been addressed in the DCE Version 3.1 Kit. This list also includes the problems addressed in DCE Version 3.0 ECO1 and DCE Version 3.0 ECO2 kit.

- The security tickets for SELF-credentials failed to re-authenticate automatically when they expire under a specific condition. "Klist-ef" command displays expired tickets for machines SELF principal.
- DCE CDS Clerk process consumes high CPU time. The looping occurs while cdsclerk was deleting the unused binding handles and marks the inuse handles for future deletion. All these in-used and free (unused) binding handles are part of the dynamically allocated linked list.

- While configuring DCE as a client, the dce$cdsadver process sometimes goes into a hang state on OpenVMS Version 7.3-1.
- The DCE application hangs occasionally with the following error:

  ```
  2002-10-01-21:42:32.552-04:00I0.238 PID#541458706 FATAL rpc
  recv CNRCVR.C;1 563
  ```

  ```
  0x0d563740(rpc__cn_network_receiver) Unexpected exception
  was raised
  ```

- When an illegal state transition detected in CN server call state machine, the DCE/RPC does not report the correct state and event of the state machine.
- When a large number of ACMSxp Processing servers are stopped and re-started several times in a loop, DCE$CDSADVER process aborted with the following error:

  ```
  2002-10-13-23:33:25.440+09:00I----- dce$cdsadver(31600) ERROR
  cds adver ADVER_SOCKET.C;1 130 0x00a69500
  ```

  ```
  %SYSTEM-F-ACCVIO, access violation, reason mask=00, virtual
  address=0000000000000030, PC=FFFFFFFF8118A8E8, PS=0000001B
  ```

- XDS header files shipped with DCE replaces the existing X.500 specific files. The DCE install procedure has been modified to copy into [syslib] area only when the common XDS headers (that shipped with both DCE and X.500 kits) are not present.
- Non-privileged VMS user failed to get all the credential files created during dce_login.
- The DCE specific pre-authentication files in DCE$LOCAL:[VAR.SECURITY.PREAUTH] and dtsd.binding file in DCE$LOCAL:[VAR.ADM.TIME] have been assigned with an in-valid file ownership.
- If a VMS DCE RPC application client calls a VMS DCE RPC server and then cancels the thread in which the RPC was started, the server crashes with the following message:

  ```
  2003-03-20-16:53:31.137+01:00I4.884 PID#551581762 FATAL
  rpc cn_state CNSCLSM.C;1 3123 0x02165b40
  ```

  ```
  Illegal state transition detected in CN server call state machine
  [cur_state: 255, cur_event: 102, call_rep:
  1ec9240]%SYSTEM-F-OPCCUS, opcode reserved to customer
  fault at PC=FFFFFFFF80A3E434, PS=0000001B
  ```

- On a DCE Version 3.0 system the SECD process can report the following messages:

  ```
  krb5kdc: Unknown code DCE:krb 37 (336760869) pa verify failure
  krb5kdc: Unknown code DCE:krb 31 (336760863) pa verify failure
  ```

  One of the causes for the above messages is, mismatch in the passwords stored in the registry database and keytab files for the client and server principals.

  The principal name that is causing "PA verify failure" would be reported in DCE Version 3.1. It is expected the user would take necessary action to correct the password mismatches.

- The "dcecp -c clearinghouse modify <clearinghouse name> -add \{CDS_UpgradeTo 4.0\}" command reports "Error: Unrecognized argument ’4.0}’".

  The correct usage of the command is:

  ```
  $dcecp -c clearinghouse modify <clearinghouse name> -add
  "{CDS_UpgradeTo 4.0}"

  (or)

  dcecp> clearinghouse modify <clearinghouse name> -add
  {CDS_UpgradeTo 4.0}
  ```

- The DCE Security master server ACCVIOs with the following information:

  ```
  %SYSTEM-F-ACCVIO, access violation, reason mask=04, virtual
  address=00000000000000D8, PC=000000007BBB37EC, PS=0000001B
  ```

- When a large number of ACMSxp Processing servers are stopped and re-started several times in a loop, DCE$CDSCLERK process aborted with the following error:

  ```
  2003-04-28-15:30:42.052+09:00I----- dce$cdsclerk(20200985) FATAL rpc
  recv CNRCVR.C;1 566
  0x0418b580 (rpc__cn_network_receiver) Unexpected exception was raised

  %DECthreads bugcheck (version 3.19-059), terminating execution.
  % Reason:  lckMcsLock: deadlock detected, cell = 0x000000000A601580
  % Running on OpenVMS V7.3-1() on AlphaServer GS160 6/731, 0Mb; 8 CPUs,
  pid 538970215
  % The bugcheck occurred at 22-APR-2003 04:25:51.12, running image
  % DSA0:[SYS0.SYSCOMMON.][SYSEXE]DCE$CDSCLERK.EXE;1 in process 20200867
  (named "DCE$CDSCLERK"), under username "SYSTEM". AST delivery is enabled
  for all modes; no ASTs active. Upcalls are disabled. Multiple kernel
  threads are disabled.
  ```

- DCE_SETUP START some times reports the following error.

  ```
  Starting sec_client service (please wait).
  Error: secval service already enabled
  ```

- When the bogus parameter was input as the CDS master server name, there was no condition check in the dce$setup.com file to see whether the input parameter is correct or wrong. Because of this missing condition check there was no error reported even when the input parameter was incorrect.
- Bogus error message is reported when the CDS replica is unavailable: Directory Synchronize command of dcecp throws an error "Unable to communicate with any CDS server" even if one of the CDS server (clearinghouses) is not available on which the replica of the directory being skulked resides. The command response now shows the inaccessible clearinghouse names.

  The output would look like below:

  ```
  dcecp> directory synch /.:
  Skulk failure: /.:
  /.../dce73-cell/ch1_ch inaccessible
  1 replica(s) not updated
  Error: Unable to communicate with any CDS server
  ```

- If the time difference on the client system and DCE security server system is not within the permissible time skew value, the DCE startup on client system aborts without reporting any error to the user.
- Some times the rgy_edit command fails due to failure in gethostbyname() function.
- The DCE daemon dump files are not created in the daemon specific directories. In DCE Version 3.1, the dump file of the daemon would be created in its result directory.

  For example, the dump of the DCED daemon would be available in the directory DCE$LOCAL:[VAR.DCED]
- During the DCE configuration, when prompted for whether the node should accept time from DECdts servers, entering '?' at the prompt, the configuration procedure aborts after displaying the Help as below:

  ```
  Should this node accept time from DECdts servers?
  (YES/NO/?) [N]? ?

  DCE DTS Time servers can accept time synchronization messages
  from DECnet DTSS time servers as well as DCE DTS servers.

  Answering YES to this question will allow this interaction
  to occur.

  %DCL-W-USGOTO, target of GOTO not found - check spelling and
  presence of label \DCE_QUERY_DTS_DECNET\
  ```

- The "Add CDS Replica" option in the "Modify Menu" of the DCE setup procedure becomes "Remove CDS Replica" after the operation of adding the CDS Replica fails.
- Some of the error handling sub-routines in DCE Setup command procedure do not report the exact DCE error message on failure of the DCE Commands and instead report a "CLI-W-ABVERB" error, which is not informative enough to troubleshoot the problem.
- The DCE Security Server aborted with the following error information:

  ```
  2003-07-08-13:39:36.070+08:00I----- PID#2018 FATAL rpc recv
  KRBCLT.C;1 296 0x7bbe46f0
  (rpc__krb_get_tkt) Unexpected exception was raised
  %CMA-F-EXCCOP, exception raised; VMS condition code follows
  -SYSTEM-F-OPCCUS, opcode reserved to customer fault at
  PC=FFFFFFFF80623E54, PS=0000001B
  ```

- When a large number of ACMSxp Processing servers are stopped and re-started several times in a loop for about 2 days, the DCE$DCED process access violated and DCE$DCED.OUT file contains many of the following errors:

  ```
  *** CREATE_NET_CHANNLE_W_MBX FAILURE *** 1b4 errors.
  ```

- A potential denial of service has been identified on OpenVMS systems that have the DCE products installed or that are using the RPC portion of DCE that ships with the OpenVMS operating system. These OpenVMS systems could be vulnerable to a remote initiated Buffer Overflow, resulting in hang.
- DCE Configuration procedure continues operation even after logging an error message when any of the CDSCP commands fail during the CDS replica configuration

  ```
  **********************   ERROR    ****************************
  ***  Could not execute CDSCP command:
  ***  Cdscp create clearinghouse /.:/mr6axp_ch
  ```

  The DCE$CDSD daemon was terminated and dce$setup.com continued its operation of configuring the CDS Replica server instead of aborting.
- "DCE Config" menu does not show the correct option for Auditing during successive enabling & disabling.
- The PE_SITE file in DCE V3.0 does not have RE permission for the world.
- For each execution of the DCOM application, the receiver thread in DCE was unable to free the credentials due to which the persona was getting pilled up for DCOM$RPCSS process and eventually leading to the rundown of DCOM$RPCSS with RMS fatal exception.

- The DCE$CDSD process crashed with SYSTEM-F-STKOVF error in background_activator and back_propogate threads.
- DCECP program access violates when issuing an incomplete DCECP -C ERRTEXT command
- The DCE$RPC_SHUTDOWN.COM did not delete the files in dce$local:[var.dced] directory including EP.DB file. This may result in stale entries in EP.DB file, leading to DCE$DCED daemon crash.
- When Pthread metering is enabled, DCE startup fails while starting up CDSD daemon. The process DCE$CDSD crashes with "%SYSTEM-F-OPCCUS"

```
2003-09-29-05:49:26.215+08:00I----- cedar$dka0:[sys0.syscommon.]
[sy FATAL cds general SERVER_MAIN.C;3 475 0x7bcd0888 Routine
pthread_cond_wait failed : status = -1.
%SYSTEM-F-OPCCUS, opcode reserved to customer fault at
C=FFFFFFFF80A5ED24, PS=0000001B
```

- When insufficient command line arguments are provided to DCE$DCED while invoking DCED daemon from command line in "Endpoint Mapper" mode using "-x " option, the process crashes with "%SYSTEM-F-ACCVIO", access violation

```
$ dced -x
dce73$dka100:[sys0.syscommon.][sysexe]dce$dced.exe;1: option requires an
argument -- x
%SYSTEM-F-ACCVIO, access violation, reason mask=00, virtual
address=0000000000000000, PC=FFFFFFFF806 68324, PS=0000001B
```

- In DCE Version 3.0, the DCE$CONFIG_PROTSEQ symbol is used in conjunction with RPC_SUPPORTED_PROTSEQS logical to configure DCE with only TCP as the RPC supported protocol. It is possible that missing DCE$CONFIG_PROTSEQ symbol could create a DCE configuration issues. In DCE Version 3.1, the DCE$CONFIG_PROTSEQ is made redundant and hence user is not required to define the symbol DCE$CONFIG_PROTSEQ.
- The DCECP command program loops infinitely if a wrong option is entered at the DCL prompt.

```
$ dcecp -x
Error: invalid command name "-X"
Error: invalid command name "-X"
Error: invalid command name "-X"
```

- When stopping DCE$CDSD process using 'cdscp disable server' command, CMA-F-ALERT, error reported in the DCE$CDSD.LOG file. Here is the error message.

  ```
  %CMA-F-EXCCOPLOS, exception raised; some information lost
  -CMA-F-ALERTED, thread execution has been canceled
  ```

- The DCE$CDSD process hangs with the following error

  ```
  2002-10-01-21:42:32.552-04:00I0.238 PID#541458706 FATAL rpc recv
  CNRCVR.C;1 563 0x0d563740(rpc__cn_network_receiver) Unexpected
  exception was raised
  ```

- Wile compiling the DCE applications, the C++ Compiler Version 6.5 throws out the following warning with STUBBASE.H header file.

  ```
  byte_p_t  rpc_ss_mem_alloc   ( rpc_ss_mem_handle *, unsigned )
  ..........^
  %CXX-W-TYPNOLNKFUN, use of a type with no linkage to declare a function
  at line number 988 in file SYS$COMMON:[DCE$LIBRARY]STUBBASE.H;1
  ```

- CDS Checkpoint Migration during the DCE Upgrade to DCE Version 3.1 from Version 1.5:

  When you upgrade the DCE for OpenVMS Version 1.5 to Version 3.1 of DCE, you would need to migrate the CDS checkpoint file that are present in the DCE$LOCAL:[VAR.DIRECTORY.CDS] directory before starting DCE with Version 3.1.

  The CDS checkpoint migration procedure is as below:

  The dce$checkpoint.exe image is delivered through DCE Version 3.1 kit and can be located in SYS$COMMON:[SYSHLP.EXAMPLES.DCE.TOOLS] directory.

  Here are the options available with chkmigrate tool:

  ```
   chkmigrate <FileName> [-cmvo]

   FileName Checkpoint file name
   -c  Check whether the file is proper(default)
   -m  Migrate if file condition is not proper
   -o  Overwrite the existing checkpoint file. Otherwise new
  file name will be <filename>_MIGRATED
   -v  Verbose
  ```

  Here are the details regarding DCE$CHKMIGRATE.EXE usage:

  i $ chkmigrate :== $ device:[directory]DCE$CHKMIGRATE.EXE

  ii $chkmigrate <checkpoint file name>-m

  iii Migrated file name will be <checkpoint file name>_MIGRATED

  iv Rename the original checkpoint file to <checkpoint file name> _OLD v Rename the migrated file checkpoint file name>_MIGRATED to original filename <checkpoint file name>

Please ensure the version numbers of .TLOG* and .CHECKPOINT* are same as the contents of .VERSION file

Here is an example:

Contents of version file

```
$ type DCE$LOCAL:[VAR.DIRECTORY.CDS]MR6AXP-CELL_MIRAGE_CH.VERSION;1
0000000010
```

The corresponding .CHECKPOINT and .TLOG would be

```
MR6AXP-CELL_MIRAGE_CH.CHECKPOINT0000000010
MR6AXP-CELL_MIRAGE_CH.TLOG0000000010
```

- SELF principal not a member of cds-server, on an upgraded system:

  After DCE upgrade to Version 3.1 from Version 1.5, following warning messages would be reported whenever the user restarts DCE Version 3.1

  ```
  2003-05-07-09:29:58.697-04:00I446.824 cdsclerk(13411) WARNING
  cds clerk clerk_bind.c 793 0x140043050

  CDS server principal hosts/HOSTNAME/self is not a member of
  group subsys/dce/cds-server.
  ```

  To eliminate these warning messages, perform the following steps:

  1. Startup DCE.
  2. Log in as cell_admin (dce_login cell_admin)
  3. Execute the following command:

     ```
     $dcecp -c group add /.../<cell name>/subsys/dce/cds-server
     -member /.../<cell name>/hosts/<host name>/self
     ```

     Performing these steps will add the "hosts/<host name>/self" principal to the "subsys/dce/cds-server" group. Subsequently, when DCE starts up, the startup procedure will not generate warning messages.

- The “Times Lookup Paths Broken” counter denotes the number of lookup failures that happened through CDS server while looking for a directory. This counter may get incremented during the DCE startup, which is expected. However, if the counter keeps on incrementing at regular intervals (the default interval is one hour) then there is a possibility of a child pointer of the directory got corrupted (or) it’s not present.

When the counter gets incremented, the directory name of the child pointer is logged in into DCE$CDSD.OUT file. Under this condition, you need to delete and recreate the child pointer of the directory to resolve the lookup failures.

Here is the command to delete and create the child pointer for the directory.

```
dcecp> directory remove <parent_directory_name> \
       -member <child_pointer>
```

```
dcecp> directory add <parent_directory_name> \
-member <child_pointer> \
       -clearinghouse <clearing_house_name>
```

Here is the sample example of the Times_Lookup_Paths_Broken counter incremented:

```
dcecp> cds show /.:
{Creation_Time 2003-09-24-00:23:22.875+08:00I-----}
{Future_Skew_Time 0}
{Read_Operations 9538}
{Write_Operations 49814}
{Skulks_Initiated 14129}
{Skulks_Completed 14128}
{Times_Lookup_Paths_Broken 12}
{Crucial_Replicas 0}
{Child_Update_Failures 0}
{Security_Failures 0}
{Known_Clearinghouses /.../csesup-cell/csesup_ch}
```

The dce$cdsd.out file would contain the message like 2003-09-24-01:27:27.242+08:00I——B_pp_c Dir_name:/.../csesup-cell/dir1, check child_pointer

for this you need to recreate the child pointer using the following command:

```
dcecp> directory add /.../csesup-cell -member dir1  \
clearinghouse /.:/csesup_ch
```

The “Times Lookup Paths Broken” counter may also get incremented if the soft link of the directory is corrupt. In this case, you need to delete the soft link and create it if required.

- When a large number of ACMSxp Processing servers are stopped and re-started several times in a loop, DCE goes into a hang state.
- On cancellation of an RPC-call, the BG devices (Sockets) that were allocated are not released. Several such RPC call cancellations can result in a pileup of BG devices leading to a resource leak.

- Memory leak is encountered in DCE/RPC when DCOM is functional in unauthenticated mode.
- The DCE$DCED process terminates with the following error under load:

  ```
  (socket) rpc_socket_disp_select *** FATAL ERROR at
  SOCKDISPATCH.C;1\3755 ***

  %CMA-F-EXCCOP, exception raised; VMS condition code follows
  SYSTEM-F-OPCCUS,opcode reserved to customer fault at
  PC=00000000007043A8, PS=0 000001B %SYSTEM-F-ABORT, abort
  ```

- The login context specific to application processes expires immediately after starting ACMSxp TP system. The TP System makes use of credentials obtained after a dce_login with "acmsxp_svr" as the principal name.

  Klist fails with the following error:

  No DCE identity available: No currently established network identity for which context exists (dce / sec)

- The DCE$DCED process aborts with SYSTEM-F-ACCVIO when a command procedure containing dce_login and ACMSxp commands is executed in an infinite loop.
- DCE Version 3.0 RPC Only configuration fails on systems with TCPWARE as the IP product. The setup procedure issues UCX commands, which would not work on TCP/IP products other than HP's TCP/IP for OpenVMS. The configuration program terminates abnormally.
- There was a case where a NULL pointer check was not being made in RPC Runtime code because of which the process terminates.
- DCE Client configuration fails in certain cell configurations running HP UNIX DCE Servers. Client configuration aborts with the following error message:

  ```
  Attempting to locate security server

  *** Error: can't locate security server (exiting)
  >>> getcellinfo:  can't obtain handle to security server
  Entry not found (dce / rpc)
  ```

- DCE startup fails after a system reboot. The startup procedure fails while starting the CDS Name Service Advertiser daemon. DCE$SETUP reports the following error message:

  ```
  "Error Starting CDSADVER"
  ```

- Configuration Verification Procedure (CVP) on Version 3.0 fails intermittently with the following error message:

  ```
  %CMA-F-EXCCOPLOS, exception raised; some information lost
  -DCERPC-F-COMMFAILURE, communications failure (dce / rpc)
  DCE for OpenVMS Alpha V3.0 CVP failed.
  ```

- On completion of a CVP run, user was prompted twice to Hit the Return Key for displaying the SETUP menu.

  ```
  Press RETURN> to continue . . .
  Press <RETURN> to continue . . .
  ```

  DCE$SETUP now displays the SETUP Menu with the first Carriage Return itself.

- While configuring LDAP Clients into a LDAP Server Cell, the DCE$SETUP program does not report the ’password validation failure’ message when there is a password mismatch.
- DCE Online Help was not properly structured. Having all the Help topics at the top level of Help clutters up the DCL Help. New DCE DCL Help Library fixes the problem.
- When a call to rpc_binding_from_string_binding fails, the subsequent calls to this function hang forever. New RPC Runtime library fixes the bug in "rpc_binding_from_string_binding".
- When running DCOM applications between Windows 2000 and VMS systems, several RPC_CN_CREATE_AUTH_INFO messages are logged into the DCOM$RPCSS.OUT file leading to exhaustion in disk space.

  User will need to define logical "DCE_DISABLE_LOGGING" to 1 either system or process wide for disabling the error messages on HP DCE 3.1 for OpenVMS.

  ```
  $ DEFINE/SYSTEM/EXEC DCE_DISABLE_LOGGING 1

  (or)

  $ DEFINE DCE_DISABLE_LOGGING 1
  ```

- Failure to detect or ping an active Windows 2000 Client while running Authenticated DCOM between a Windows 2000 and OpenVMS System would cause the DCOM Server applications to timeout and run down after a period of about 10 minutes.

  Several "RPC_CN_AUTH_VFY_CLIENT_REQ" error messages appear in DCOM$RPCSS.OUT file in intervals of 2 minutes

```
2001-10-02-17:01:58.468-04:00I0.629 PID#330 ERROR rpc auth
CNSASSM.C;1 4654 0x01eb9740 RPC_CN_AUTH_VFY_CLIENT_REQ
on server failed: invalid handle (dce /rpc)
```

- Serviceability logging feature does not log the RPC information into the log files, though it creates the file.
- Enabling Kerberos 5 Services on OpenVMS Alpha Version 7.3 and above aborts with the following error message:

```
assert error: expression =context->refcount > 0, in file
DECW$DCERESD:[SECURITY.CLIENT.RCA.SRC]INTERNAL_BINDING.C;1
at line 2706
%SYSTEM-F-OPCCUS, opcode reserved to customer fault at
PC=FFFFFFFF80A5E7F4, PS=0000001B
```

- RPC Fatal Exceptions are reported in DCOM$RPCSS.OUT when running DCOM applications from Windows 2000 after a new login. DCOM$RPCSS process reports the following exception in the OUT file.

```
2002-04-18-15:04:17.604-04:00I0.113 PID#21370 FATAL rpc recv
CNRCVR.C;5 563 0x015a5740 (rpc_cn_network_receiver) Unexpected
exception was raised
```

- When the audit daemon is enabled from the dce_setup command procedure, it fails with the following error message:

```
***Could not execute DCECP command:
***Dcecp -c audfilter create world -at {dce_sec_modify
{failure denial}all}
***%CLI-W-ABVERB, ambiguous command verb - supply more characters
*** Error: dcecp -c audfilter create world -at {dce_sec_modify
{failure denial} all} failed (continuing)

Audit daemon configuration failed
```

- DCE client example programs located in [SYSHLP.EXAMPLES.DCE.RPC] generated using IDL "-trace log_manager" option access violate. The access violation occurs when the symbol RPC_LOG_FILE is defined.

```
%SYSTEM-F-ACCVIO, access violation, reason mask=00, virtual
address=0000000000000000, PC=0000000000239F68, PS=0000001B
```

- DCE Startup Procedure hangs during an upgrade from 1.5 to 3.0 in RPC Only Configuration.
- When an invalid dcecp command is entered, the user is returned to the DCL prompt instead of the dcecp prompt. In the example below, after the error is displayed, the "dcecp>" prompt should be output to allow the user to correct the command and continue.

```
$ dcecp
dcecp> errtext 2350-
Error: The value ’2350-’ is not a valid integer error code.
$
```

- An ACCVIO occurs when configuring Kerberos on pre-V7.2 versions of OpenVMS. The installation should not install MIT Kerberos files on pre-V7.2 Alpha systems and configuration should not ask questions for MIT Kerberos support on systems that do not support the option. Below is a sample of the ACCVIO:

```
$ dce_setup configure

Configuring Kerberos...
%SYSTEM-F-ACCVIO, access violation, reason mask=00, virtual
address=0000000000000001,PC=00000000004D0F50, PS=0000001B
```

- While converting a DTS local server to a DTS global server, the following error is reported:

```
*********************** ERROR ***********************
** An error occurred attempting to log in to DCE with
*** principal name "cell_admin"
Sorry.
Password Validation Failure. - cannot log in with
zero-length password (dce / sec)
```

- DCE processes should be able to produce dumps during abnormal termination. In order to be able to produce dumps, user is granted the option of setting the dump logical ’DCE$DUMP_DAEMONS’. The dump daemons logical can be set using:

```
$DEFINE/SYSTEM/EXEC DCE$DUMP_DAEMONS 1
```

- A DCOM user, running in Unauthenticated mode (no NT security credentials), fails, as follows:

```
$ run sserver
Server: CoRegisterClassObject: (null)(ee1282ca)
Server: Done
$ set message sys$message: DCE$RPC_MSG.EXE
$ exit %xee1282ca
%DCERPC-E-CANTLISTENSOCKE, cannot listen on socket (dce / rpc)
```

- CNSASSM.C is not checking some memory allocation pointers for NULL.

- According to the OSF documentation when a user defined authorization function returns false, the status value from the user supplied function is to be returned to the client. The RPC runtime always returns status "rpc_s_mgmt_op_disallowed" when the user function returns false.

- ETGET and other DCE tools (like ETDMP, ETFMT, etc) should be, but are not installed by the DCE 3.0 upgrade.
- An attempt to start DCE fails after the start of the DCE$SECD daemon. The error "Dcecp server ping was unsuccessful on all endpoints" is output to the terminal. This error message has been changed to:

  Unable to acquire initial login context: Authentication ticket expired. Please restart DCED Which is more descriptive of the real problem and indicates how to fix the problem, i.e. restart DCED
- The DCE security daemon, DCE$SECD, aborts with an ACCVIO. Analysis using the SYS$MANAGER:JPI.COM tool shows the process is continually leaking page file quota.
- DCED can crash with an ACCVIO at VA=0, PC=2C10E4.The DCE$DCED.OUT file shows:

  ```
  Delete DCE$SPECIFIC:[VAR.SECURITY.CREDS]DCECRED_0427F700.NC;1:
  No ticket
  11/06/00 23:29:42 - Starting credentials cleanup
  11/07/00 00:29:43 - Starting credentials cleanup
  11/07/00 01:29:43 - Starting credentials cleanup
  Delete DCE$SPECIFIC:[VAR.SECURITY.CREDS]DCECRED_043ACA21.;2:
  Expired
  Delete DCE$SPECIFIC:[VAR.SECURITY.CREDS]DCECRED_043ACA65.;1:
  Old version
  Delete DCE$SPECIFIC:[VAR.SECURITY.CREDS]DCECRED_043ACA67.;1:
  Empty
  Delete DCE$SPECIFIC:[VAR.SECURITY.CREDS]DCECRED_043ACA68.;1:
  Empty
  11/07/00 02:29:47 - Starting credentials cleanup
  11/07/00 03:29:47 - Starting credentials cleanup
  11/07/00 04:29:47 - Starting credentials cleanup
  %CMA-F-EXCCOP, exception raised; VMS condition code follows
  -SYSTEM-F-ACCVIO, access violation, reason mask=04, virtual
  address=0000000000000000, PC=00000000002C10E4, PS=0000001B
  ```
- A DCE server or client process that needs to maintain a login context crashes with an ACCVIO.For example the DCE$CDSCLERK can fail with the following ACCVIO in the DCE$CDSCLERK.OUT file:

  ```
  %SYSTEM-F-ACCVIO, access violation, reason mask=00, virtual
  address=0000000000000068, PC=0000000000989D50, PS=0000001B
  ```
- Cannot configure and start an RPC only configuration on DCE V3.0. Running uuidgen results in the error:

  ```
  $ run sys$system:dce$uuidgen
  %UUIDGEN-F-RPC_MESSAGE, Received Error Status: "no IEEE 802
  hardware address"
  ```

- kdestroy does not delete the credential files at DCE$LOCAL:[VAR.SECURITY.CREDS]
- During startup, whenever a client application imports the binding information from the name service database using the rpc call rpc_ns_binding_import_next while having less than 13 free event flags the RPC call fails with the ’name service unavailable’ error.
- After 10 to 12 hours DCE$CDSD and DCE$GDAD will ACCVIO. At that moment it is in a process to refresh the Server Identity. However this refresh activity fails with the error "sec_s_login_handle_invalid".
- While modifying DTS configuration from Local Server to Clerk, the DTS entity representing the DTS server still exists in the lan-profile. The server entry should not exist.