HP Data Protector 6.20 encrypted control communication certificates management

Using custom certificates

Technical white paper

Table of contents

Summary

This document describes how to replace the default Data Protector certificate for establishing encrypted control communication between the clients in Data Protector cell, which is provided during the installation or upgrade, with a custom certificate.

Introduction

Data Protector encrypted control communication helps preventing unauthorized access to clients in Data Protector cell. It is based on Secure Socket Layer (SSL), a cryptographic protocol, which provides network connections and encapsulates existing Data Protector communication protocol.

Using the Data Protector GUI or the CLI, you can remotely enable encrypted control communication for all clients in the Data Protector cell. You must have the correct certificate, private key, and trusted certificate prior to enabling encrypted control communication. Data Protector provides default certificates during the installation or upgrade.

For more information about encrypted control communication or general security considerations in Data Protector cell, see the *HP Data Protector Help* and the *HP Data Protector Concepts Guide*.

For details on using the Data Protector CLI, see the `omnicc` man page or the *HP Data Protector Command Line Interface Reference*.

For general Data Protector procedures and options, see the *HP Data Protector Help*.

Creating and distributing custom certificates

You can use custom keys and certificates to establish encrypted control communication between the clients in Data Protector cell.

This section provides details on how to create custom certificates and configure Data Protector to use them instead of the default ones.

CAUTION The instructions provided in this document contain manual steps. Follow these instructions carefully as they might lead to Data Protector communication issues in case of wrong usage.

Recommendation

Ensure that you have a valid backup of the Data Protector configuration directory on the Cell Manager and on all client systems with the default certificates and configuration files at your disposal.

Prerequisites

- To create a certificate you must obtain appropriate software from www.openssl.org, for example OpenSSL 0.9.8t or a newer update.
- Binaries for Windows systems are available at www.openssl.org/related/binaries.html.

Limitations

- Asymmetric keys are limited to 512 bits.

Notes

- Certificates and keys must be stored in PEM format.
- Avoid using names of existing certificate files for new certificates.

Depending on your needs, you can create custom certificates either by using a certificate authority (see Certificates issued by a certificate authority (CA)) or you can create self-signed certificates (see Self-signed certificates).

Certificates issued by a certificate authority (CA)

To create a unique custom certificate for each client in a Data Protector cell, proceed as follows:

1. On a secured system, create a CA:

   ```
   perl /cygdrive/c/cygwin/usr/ssl/misc/CA.pl -newca
   ```

   Note that you need to specify a different common name for each client system (for example, *ClientName*).

2. On each client system in the Data Protector cell, generate a private/public key pair for a Cell Manager:

   ```
   openssl genrsa -out keys.pem
   ```

3. On each client system in the Data Protector cell, create a certificate request:

   ```
   openssl req -new -key keys.pem -out request.pem
   ```

   Note that you need to specify a different common name for each client system and manually copy the created `request.pem` file to the CA.

4. On the system with the CA installed, sign the certificate request:

```
openssl ca -in request.pem -out certificate.pem
```

Note that you must manually copy the created `certificate.pem` file back to the relevant client system in the Data Protector cell.

5. On the Cell Manager, upload the PEM files (`certificate.pem`, `keys.pem`, and `cacert.pem`) into the certificate repository, located in the directory:

   Windows 7 and Windows Server 2008:
   Data_Protector_program_data`\Config\Server\certificates`

   Other Windows systems: *Data_Protector_home*`\Config\Server\certificates`

   HP-UX, Solaris, and Linux systems: `/etc/opt/omni/server/certificates`

   Other UNIX systems: `/usr/omni/server/certificates`

6. Enable encrypted control communication locally on the Cell Manager:

   ```
   omnicc -encryption -enable ClientName -cert certificate.pem -key
   keys.pem -trust cacert.pem
   ```

   Due to security considerations, note that you need to run the `omnicc -enable` command locally on the Cell Manager.

7. *For each client in the Data Protector cell*, enable encrypted control communication.
   a.) Upload the PEM files (`certificate.pem`, `keys.pem`, and `cacert.pem`) into the certificate repository, located in the directory:

   Windows 7 and Windows Server 2008:
   Data_Protector_program_data`\Config\client\certificates`

   Other Windows systems: *Data_Protector_home*`\Config\client\certificates`

   HP-UX, Solaris, and Linux systems: `/etc/opt/omni/client/certificates`

   Other UNIX systems: `/usr/omni/client/certificates`

   b.) Copy the `config` file from the Cell Manager to all client systems in order to update configuration parameters.
   c.) Edit the `config` file, located in the directory:

   Windows 7 and Windows Server 2008:
   Data_Protector_program_data`\Config\client\config`

   Other Windows systems: *Data_Protector_home*`\Config\client\config`

   HP-UX, Solaris, and Linux systems: `/etc/opt/omni/client/config`

   Other UNIX systems: `/usr/omni/client/config`

   and change configuration parameters in the following 3 lines:

   ```
   certificate_chain_file='/etc/opt/omni/client/certificates/certificate.pem';
   private_key_file='/etc/opt/omni/client/certificates/keys.pem';
   trusted_certificates_file='/etc/opt/omni/client/certificates/cacert.pem';
   ```

8. Modify the client entry in `/etc/opt/omni/server/config` in the Security Exceptions list located on the Cell Manager to following:

   ```
   computer.company.com={
           encryption={
                   enabled=1;
   ```

```
certificate_chain_file='/etc/opt/omni/client/certificates/certificate.pem';
private_key_file='/etc/opt/omni/client/certificates/certificate.pem';
trusted_certificates_file='/etc/opt/omni/client/certificates/certificate.pem';
pkcs12_keystore_filename='/etc/opt/omni/client/certificates/hpdpcert.p12';
pkcs12_keystore_password='hpdpcert';
pkcs12_ca_certificate_filename='/etc/opt/omni/client/certificates/hpdpcert.p12'
;
pkcs12_ca_certificate_password='hpdpcert';
pkcs12_private_key_filename='/etc/opt/omni/client/certificates/hpdpcert.p12';
pkcs12_private_key_password='hpdpcert';
};
};
```

9. Edit `/etc/opt/omni/server/cell/cell_info` to enable encryption icon for the specified client in the Clients context in the GUI , for example:

   ```
   -host "computer.company.com" -os "gpl i686 linux-2.6.18-8.el5" -core
   A.06.20 -da A.06.20 -ma A.06.20 -encryption 1
   ```

   Backup will now run successfully with encryption.

Self-signed certificates

To create a self-signed certificate, proceed as follows:

1. Create `cmprivatekey.pem` using OpenSSL.

   For a Cell Manager, generate a private/public key pair:

   ```
   openssl genrsa -out cmprivatekey.pem
   ```

2. From `cmprivatekey.pem` create `cmcert.pem`.

   Generate a certificate from *keys* (you need to provide validity, for example 20 year):

   ```
   openssl req -new -x509 -key cmprivatekey.pem -out cmcert.pem -days 7200
   ```

3. Concatenate `cmprivatekey.pem` and `cmcert.pem` into `certificate.pem`.

   Windows systems:

   ```
   type cmprivatekey.pem cmcert.pem 2> nul > certificate.pem
   ```

   UNIX systems:

   ```
   cat cmprivatekey.pem cmcert.pem > certificate.pem
   ```

4. Upload the concatenated PEM file into the certificate repository, located on the Cell Manager in the directory:

   Windows 7 and Windows Server 2008:
   Data_Protector_program_data\Config\Server\certificates

   Other Windows systems: *Data_Protector_home*\Config\Server\certificates

   HP-UX, Solaris, and Linux systems: */etc/opt/omni/server/certificates*

   Other UNIX systems: /usr/omni/server/certificates

   IMPORTANT The `omnicc -add_certificate` command will overwrite the existing files in the repository. List certificate repository to find existing names.

Check the names of existing certificates by listing the Cell Manager repository:

```
omnicc -list_certificates
```

A similar output appears:

```
List of known Certificates:
"cert.pem" from 3/16/2012 9:31:14 PM
"hpdpcert.pem" from 11/16/2011 11:22:04 AM
"privkey.pem" from 3/16/2012 9:30:59 PM
```

Upload certificates:

```
omnicc -add_certificate cmkeycert.pem cmkeycert.pem
```

5. Copy `certificate.pem` to the client systems into

   `/etc/opt/omni/client/certificates/.`

6. Enable encrypted control communication.
   - If encrypted control communication *was not already enabled* on the Cell Manager by using the default certificate, run:

     `omnicc -encryption -enable ClientName -cert certificate.pem -key certificate.pem -trust certificate.pem`

     Due to security considerations, note that you need to run the `omnicc -enable` command locally on the Cell Manager.

     Copy the file `/etc/opt/omni/client/config` from the Cell Manager to the client systems into the directory `/etc/opt/omni/client/`.

     The available keywords and their meanings are shown in Client configuration file keywords below.
   - If encrypted control communication on the Cell Manager *was enabled by using the default certificate*, you must edit the `config` file on both the client system and the Cell Manager.

     Edit the `/etc/opt/omni/client/config` file and change `hdpcert.pem` to `certificate.pem` in the following 3 lines:

     ```
     certificate_chain_file='/etc/opt/omni/client/certificates/certificate.pem';
     private_key_file='/etc/opt/omni/client/certificates/certificate.pem';
     trusted_certificates_file='/etc/opt/omni/client/certificates/certificate.pem';
     ```

7. Modify the client entry in `/etc/opt/omni/server/config` from Security Exceptions list located on the Cell Manager to the following:

```
computer.company.com={
        encryption={
                enabled=1;

certificate_chain_file='/etc/opt/omni/client/certificates/certificate.pem';

private_key_file='/etc/opt/omni/client/certificates/certificate.pem';

trusted_certificates_file='/etc/opt/omni/client/certificates/certificate.pem';

pkcs12_keystore_filename='/etc/opt/omni/client/certificates/hpdpcert.p12';
                pkcs12_keystore_password='hpdpcert';

pkcs12_ca_certificate_filename='/etc/opt/omni/client/certificates/hpdpcert.p12'
;
                pkcs12_ca_certificate_password='hpdpcert';

pkcs12_private_key_filename='/etc/opt/omni/client/certificates/hpdpcert.p12';
                pkcs12_private_key_password='hpdpcert';
```

```
                    };
          };
```

8. Edit `/etc/opt/omni/server/cell/cell_info` to enable the display of the encryption icon for the specified client in the Clients context in the GUI , for example:

   ```
   -host "computer.company.com" -os "gpl i686 linux-2.6.18-8.el5" -core
   A.06.20 -da A.06.20 -ma A.06.20 -encryption 1
   ```

   Backups will now run successfully with encryption.

Configuration files reference

Client configuration file

The file is located in:

Windows 7 and Windows Server 2008: *Data_Protector_program_data*`\Config\client\config`
Other Windows systems: *Data_Protector_home*`\Config\client\config`
HP-UX, Solaris, and Linux systems: `/etc/opt/omni/client/config`
Other UNIX systems: `/usr/omni/client/config`

Client configuration file keywords

Keyword	Description
`enabled`	Specifies whether encryption is enabled or not (`0` or `1`). Default: `0`
`trusted_certificates_file`	Specifies the file of trusted certificates. The file should contain multiple certificates in PEM format concatenated together. Default location: ***Windows systems:*** *Data_Protector_home*`\config\client\certificates\hpdpcert.pem` ***UNIX systems:*** `/etc/opt/omni/client/certificates/hpdpcert.pem`
`certificate_chain_file`	Specifies the certificate chain file. A certificate chain is a sequence of certificates, where each certificate in the chain is signed by the subsequent certificate. The last certificate in the chain is normally a self-signed certificate-a certificate that signs itself. Default: same as `trusted_certificates_file` default
`private_key_file`	Specifies the private key file. Default: same as `trusted_certificates_file` default
`pkcs12_keystore_filename`	Specifies the `trusted_certificates_file` converted from PEM to PKCS12 file format. This is used by the Java GUI and is generated automatically when adding a certificate.
`pkcs12_keystore_password`	Password for `pkcs12_keystore_filename`.
`pkcs12_ca_certificate_filename`	Specifies the `certificate_chain_file` converted from PEM to PKCS12 file format. This is used by the Java GUI and is generated automatically when adding a certificate.
`pkcs12_ca_certificate_password`	Password for `pkcs12_ca_certificate_filename`.
`pkcs12_private_key_filename`	Specifies the `private_key_file` converted from PEM to PKCS12 file format. This is used by the Java GUI and is generated automatically when adding a certificate.

pkcs12_private_key_password	Password for pkcs12_private_key_filename.

Example

```
encryption={

       enabled=’1’;

       trusted_certificates_file=’/etc/opt/omni/client/certificates/trusts.pem’;

       certificate_chain_file=’/etc/opt/omni/client/certificates/dpcert.pem’;

       private_key_file=’/etc/opt/omni/client/certificates/dpcert.key’;

       pkcs12_keystore_filename='/etc/opt/omni/client/certificates/hpdpcert.p12';

       pkcs12_keystore_password='hpdpcert';

       pkcs12_ca_certificate_filename='/etc/opt/omni/client/certificates/hpdpcert.
p12';

       pkcs12_ca_certificate_password='hpdpcert';

       pkcs12_private_key_filename='/etc/opt/omni/client/certificates/hpdpcert.p12
';

       pkcs12_private_key_password='hpdpcert';

};
```

Server configuration file

The file is a collection of client configurations. In addition to client configuration options, it has an exception option (=1 or 0).

Example:

```
host1.domain.com={

       encryption={

              enabled=1;

              certificate_chain_file='C:\Program Files\OmniBack\config\client\
\certificates\hpdpcert.pem';

              pkcs12_keystore_filename='C:\Program Files\OmniBack\config\client\
\certificates\hpdpcert.p12';

              pkcs12_keystore_password='********';

              private_key_file='C:\Program Files\OmniBack\config\client\
\certificates\hpdpcert.pem';

              pkcs12_private_key_filename='C:\Program Files\OmniBack\config\client
\\certificates\hpdpcert.p12';

              pkcs12_private_key_password=********';
```

```
            trusted_certificates_file='C:\Program Files\OmniBack\config\client\
\certificates\hpdpcert.pem';

            pkcs12_ca_certificate_filename='C:\Program Files\OmniBack\config\
client\\certificates\hpdpcert.p12';

            pkcs12_ca_certificate_password='********';

      };

};

host2.domain.com={

      encryption={

            exception=1;

      };
```

Security Exceptions list

Specifies clients that are accepted in a plain text mode. The list is stored on the Cell Manager in a `server configuration` file located in:

Windows 7 and Windows Server 2008:
Data_Protector_program_data\Config\server\config
Other Windows systems: *Data_Protector_home*\Config\server\config
HP-UX, Solaris, and Linux systems: /etc/opt/omni/server/config
Other UNIX systems: /usr/omni/server/config

Using `omnicc` to manage your certificates

IMPORTANT This section describes how to manage encryption keys using the `omnicc` command. If you want to avoid distributing the keys using `omnicc`, follow the steps described earlier in Creating and distributing custom certificates.

Prerequisite

You must have the correct certificate (for example, `cmcert.pem`), private key (for example, `cmprivatekey.pem`), and trusted certificate (for example, `cmkeycert.pem`) prior to enabling encrypted control communication.

Encrypting plain (non-encrypted) Cell Manager with non-default certificate

For the Cell Manager and all the clients in the cell

Using the CLI

Run:

```
omnicc -encryption -enable -all -cert cmkeycert.pem -key cmkeycert.pem
-trust cmkeycert.pem
```

A similar output appears:

```
Encryption is enabled for the following hosts:
...
```

Using the GUI

For detailed procedure on how to enable encrypted control communication by using the Data Protector GUI, see the *HP Data Protector Help* index: "encrypted control communication, enabling".

Figure 1: Enabling encrypted control communication for the selected client

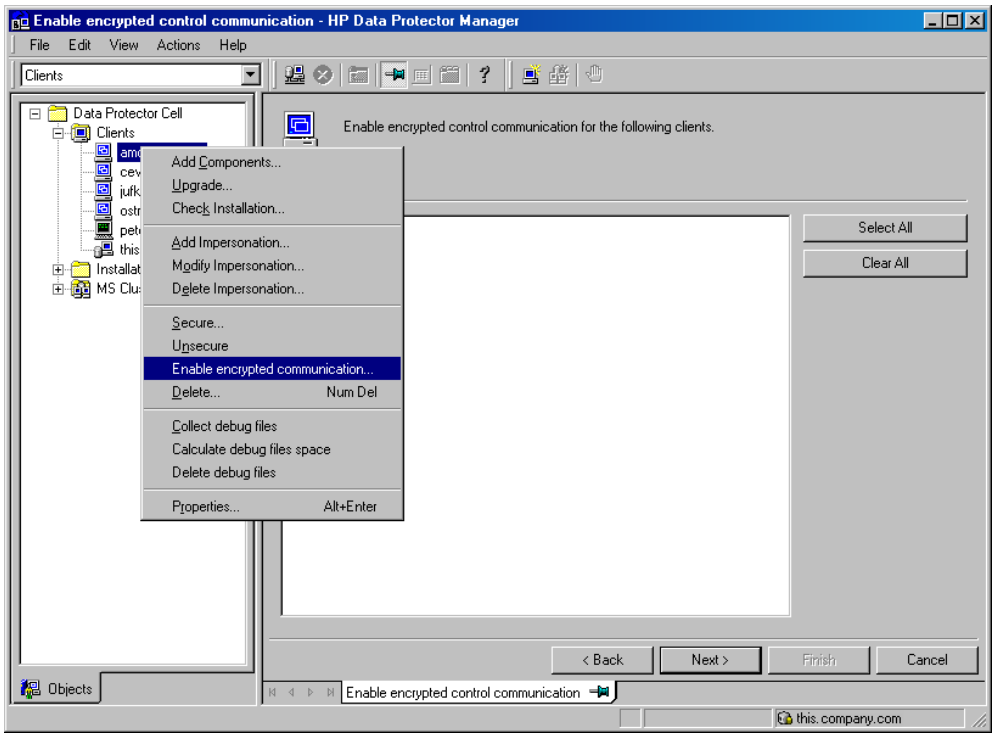


Figure 2: Selecting encrypted control communication for clients in the cell

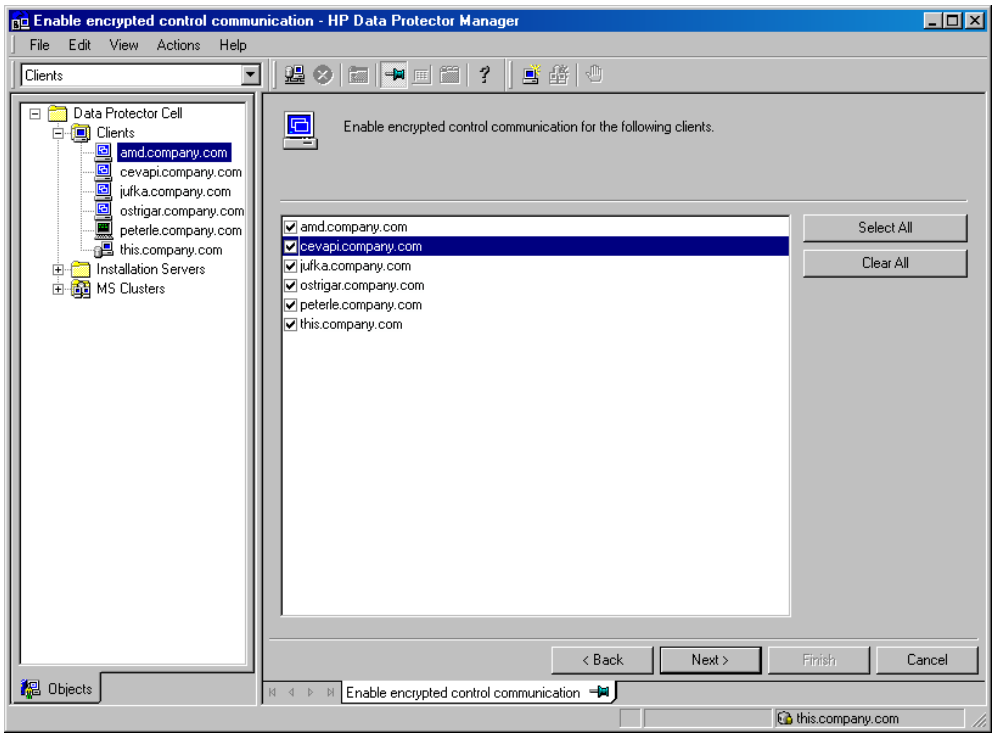
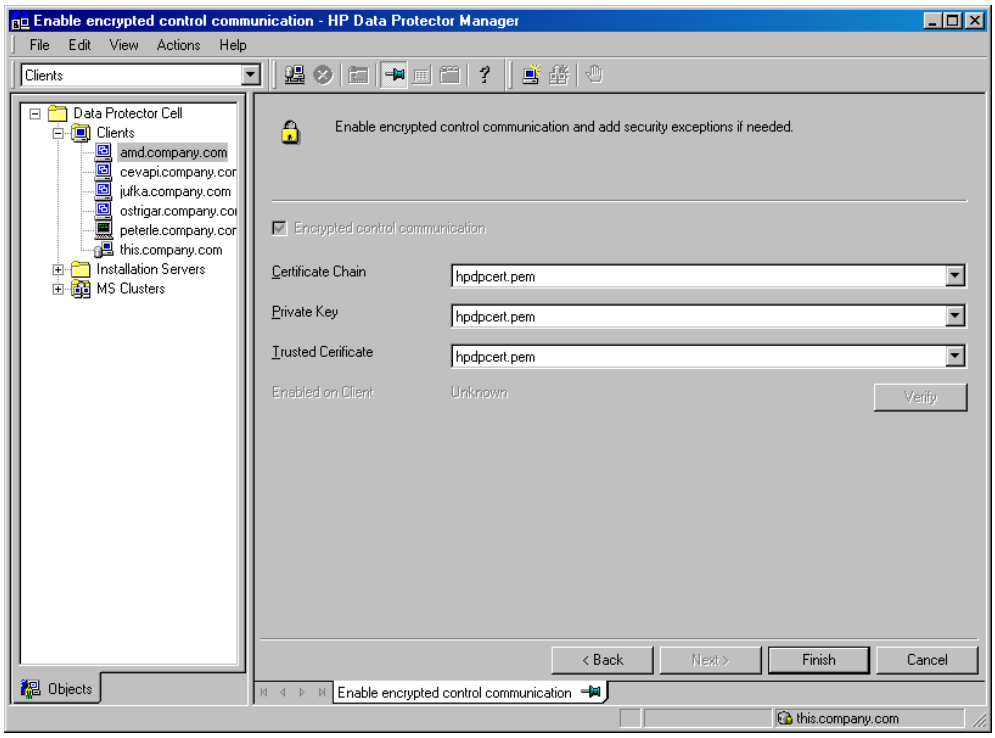


Figure 3: Specifying the certificate, private key, and trusted certificate



For the Cell Manager and specified clients in the cell

Using the CLI

```
omnicc -encryption -enable cmhost host1 host2 -cert cmkeycert.pem -key
cmkeycert.pem -trust cmkeycert.pem
```

A similar output appears:

```
Encryption is enabled for the following hosts:
...
```

Using the GUI

For detailed procedure on how to enable encrypted control communication using the Data Protector GUI, see the *HP Data Protector Help* index: "encrypted control communication, enabling".

Replacing the default or custom existing certificate

In the below example, the following certificates and keys

- `cmkeycert.pem` (keys + certificate)
- `cmprivatekey.pem` (keys)
- `cmcert.pem` (certificate)

are replaced with the new certificates and keys you created

- `cmnewkeycert.pem` (keys + certificate)
- `cmnewprivatekey.pem` (keys)
- `cmnewcert.pem` (certificate)

To replace your old certificate:

1. Concatenate the `cmcert.pem` file and the `cmnewcert.pem` file into a mutually trusted certificate `cmtrustboth.pem`.

2. Upload the `cmtrustboth.pem` certificate to the Cell Manager certificate repository.

3. Set up mutual trust:

   ```
   omnicc -encryption -enable -all -trust cmtrustboth.pem
   ```

4. Replace old certificate with the new one:

   ```
   omnicc -encryption -enable -all -cert cmnewkeycert.pem -key
   cmnewkeycert.pem
   ```

5. Set up trust only for the new certificate:

   ```
   omnicc -encryption -enable -all -trust cmnewkeycert.pem
   ```

Encrypted control communication in a Manager-of-Managers (MoM) environment

The Data Protector Manager-of-Managers concept allows you to manage a large environment, also known as enterprise backup environment, with multiple Data Protector cells centrally from a single point.

For more information and detailed procedures, see the *HP Data Protector Help* index: "MoM".

Setting up the MoM environment from encrypted cells

To establish mutual trust between several cells that are using encrypted control communication, proceed as follows:

1. Concatenate certificates from each cell into the `momtrust.pem` certificate.

2. Upload `momtrust.pem` file into the Cell Manager certificate repository in each cell.

3. In each cell run:
   a. If all clients in the cell have encrypted control communication enabled:

```
omnicc -encryption -enable -all -trust cmtrustboth.pem
```

b. If not all clients in the cell have encrypted control communication enabled, specify only those that have it enabled:

```
omnicc -encryption -enable Hostname1 [Hostname2 … ] -trust
cmtrustboth.pem
```

4. Proceed with the regular procedure for setting up a MoM environment.
5. If there are security exceptions in cells:
   a. For each Cell Manager in MoM collect Security Exceptions list using:

      ```
      omnicc –encryption –list_exceptions
      ```

   b. Merge the lists.
   c. Upload the merged list to each Cell Manager using:

      ```
      omnicc –encryption –add_exception hostname1 [hostname2 ...]
      ```

Setting up the MoM environment from encrypted and plain cells

If your Mom environment consists from plain (non-encrypted) and encrypted cells, proceed as follows:

1. Add unencrypted Cell Managers and their clients to the Security Exceptions list on the encrypted Cell Managers.

   For detailed procedure on adding clients to the Security Exceptions list and location of the *server configuration* file, see the *HP Data Protector Help*.

2. Proceed with the regular procedure for setting up a MoM environment.

Enabling encrypted control communication in the plain MoM environment

To enable encrypted control communication in a MoM environment that consists only from non-encrypted cells, proceed as follows:

1. Create certificate.
2. Upload it into the Cell Manager certificate repository in each cell.
3. Enable encrypted control communication in each cell.
4. If there are security exceptions in cells:
   a. For each Cell Manager in MoM collect exception list using:

      ```
      omnicc –encryption –list_exceptions
      ```

   b. Merge the lists.
   c. Upload the merged list to each Cell Manager using:

      ```
      omnicc –encryption –add_exception hostname1 [hostname2 ...]
      ```

Encrypted control communication and the Data Protector Centralized Media Management Database

In large multi-cell environments with high-end backup devices, you may want to share the devices and media among several cells. This can be achieved by having one Centralized MMDB for all the cells and keeping an individual CDB for each cell. This allows media and device sharing while preserving the security capabilities of the multi-cell structure.

For more information and detailed procedures, see the *HP Data Protector Help* index: "CMMDB".

Setting up the Data Protector Centralized Media Management Database (CMMDB) from encrypted cells

Initial steps are the same as for MoM environment. See Setting up the MoM environment from encrypted cells.

1. Proceed with the regular procedure for configuring a CMMDB.

2. If there are security exceptions in cells:

   a. For each Cell Manager in MoM collect exception list using:

      ```
      omnicc -encryption -list_exceptions
      ```

   b. Merge the lists.

   c. Upload the merged list to each Cell Manager using:

      ```
      omnicc -encryption -add_exception hostname1 [hostname2 ...]
      ```

Setting up CMMDB from encrypted and plain cells

Initial steps are the same as for MoM environment. See Setting up the MoM environment from encrypted and plain cells.

1. Proceed with the regular procedure for configuring a CMMDB.

Enabling encrypted control communication in plain CMMDB

Initial steps are the same as for MoM environment. See Enabling encrypted control communication in the plain MoM environment.

For more information

Visit the following Data Protector online resources to get more information:

http://www.hp.com/go/dataprotector
http://www.hp.com/go/imhub/dataprotector
http://www.hp.com/go/d2d

Call to action

To read more about HP Data Protector, visit www.hp.com/go/dataprotector.

4AA4-1110ENA, Created April 2012