Oracle® Health Sciences Information Gateway

Secure Health Email Installation and Configuration Guide

Release 1.2

E22884-01

June 2011

ORACLE®

Oracle Health Sciences Information Gateway Secure Health Email Installation and Configuration Guide, Release 1.2

E22884-01

Contents

Preface

This user's guide introduces you to Oracle Health Sciences Information Gateway (OHIG) Secure Health Email, which provides secure sending and receiving of encrypted email.

Audience

This document is intended for users who need to install and configure OHIG Secure Health Email.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/support/contact.html or visit http://www.oracle.com/accessibility/support.html if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Health Sciences Information Gateway Release 1.2 documentation set:

- *Oracle Health Sciences Information Gateway Release Notes* (Part Number E22758-01)
- *Oracle Health Sciences Information Gateway Installation Guide* (Part Number E22757-01)

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Installing and Configuring OHIG Secure Health Email

Oracle Health Sciences Information Gateway (OHIG) has implemented the National Health Information Network (NHIN) to provide a secure, scalable, standards-based method of sending authenticated and encrypted health information to known and trusted recipients over the internet. OHIG Secure Health Email interacts with Oracle Health Sciences Information Manager (HIM) services, such as Oracle Healthcare Master Person Index (OHMPI), Document Registry, Repository, Policy Engine, and Audit Record Repository (ARR).

This chapter includes the following sections:

- "Understanding OHIG Secure Health Email Components and Templates"
- "Importing the OHIG Secure Health Email Template"
- "Creating the OHIG Secure Health Email VM"
- "Configuring the OHIG Secure Health Email VM"

1.1 Understanding OHIG Secure Health Email Components and Templates

The OHIG Secure Health Email template uses the "Paravirtualized" virtualization method. The template is distributed as a compressed tar file (`*.tgz`). The compressed tar file contains two binary files and a text file. The binary files are the disk images taken from a fully configured and functional VM. The text file is a VM configuration file.

1.1.1 OHIG Secure Health Email Components

The contents of the compressed tar file is listed below:

- Disk Image with Oracle Software

  `/appliance.img`

- Disk Image with Operating System

  `/System.img`

- VM Configuration File

  `/vm.cfg`

1.1.2 OHIG Secure Health Email VM Template

The VM consists of the following pre-installed software:

- Oracle Enterprise Linux 5 (as in `System.img`)

  http://www.oracle.com/technetwork/topics/linux/whatsnew/index.html

- OHIG Secure Health Email specific software (as in `appliance.img`)

  - Apache Ant 1.8.1

    Install directory: `/home/common/ant`

  - Java Development Kit 1.6.0_X

    Install directory: `/home/common/java/latest` *(symbolic link to JDK 1.6.0_X)*

  - Apache James Mail Server 2.3.2

    Install directory: `/home/common/james` *(symbolic link to Apache James Mail Server 2.3.2)*

  - For hiauser *only*:

    * OHIG Secure Health Email Ant Configuration Utility

      Install directory: `/home/hiauser/config`

- VM Memory Settings:

  - 2 GB (2048 MB) of RAM

    > **Note:** The RAM memory setting can be changed after installation in VM Manager.

  - 16 GB of Disk Space

- Linux Users:

  - Root user

    * Username: `root`

    * Linux Group: `root`

    * Password: `ovsroot`

  - OHIG Secure Health Email specific user

    * Username: `hiauser`

    * Linux Group: `hiauser`

    * Password: `hiapass`

    > **Tip:** For security purposes, it is recommended that you change the default passwords after installation.

1.2 Importing the OHIG Secure Health Email Template

1. Copy the OHIG Secure Health Email VM template `.tgz` file to the `/OVS/seed_pool` directory of your Oracle VM Server machine.

2. Uncompress the `.tgz` file:

```
> tar -zxvf <FILENAME>.tgz
```

This step creates a directory with the name of the template.

Example:

```
> cd /OVS/seed_pool
> tar -zxvf /OVS/seed_pool/OVM_HIGV12_X86_EMAIL_PVM.tgz
```

Creates the directory:

```
/OVS/seed_pool/OVM_HIGV12_X86_EMAIL_PVM
```

> **Note:** If you are using 64 bits, you would use `OVM_HIGV12_X86_64_EMAIL_PVM`.

3. Log in to the Oracle VM Manager

> **Note:** The default location for the Oracle VM Manager log in screen is `http://<VM_MANAGER_HOST_NAME>:8888/OVS`.

4. From the Oracle VM Manager console:

   a. Click the **Resources** tab. The Virtual Machine Templates screen is displayed.

   b. Click the **Import** button. The Source screen is displayed.

   c. Choose the **Select from Server Pool (Discover and register)** radio button.

   d. Click **Next**. The General Information screen is displayed.

   Enter or select the following general information:

   - The server pool on which the virtual machine will be located.

   Server Pool Name: *<SERVER_POOL_NAME>*

   - The operating system of the Virtual Machine Operating System:

   `Oracle Enterprise Linux 5`

   - The Oracle VM template to be imported.

   Virtual Machine Template Name: *<VM_TEMPLATE_NAME>*

   - The username used to log in to the Virtual Machine.

   Virtual Machine System Username: `root`

   - The password used to log in to the Virtual Machine.

   Virtual Machine System Password: `ovsroot`

   e. Click **Next**. The Confirm Information screen is displayed.

   f. Click **Confirm**. The Virtual Machine Template screen is displayed with a message to confirm the VM template has been imported.

5. Click the **Resources** tab to see the list of available VM templates.

6. To make the Virtual Machine template available for use, select the Virtual Machine template and click **Approve**, moving the VM template from the "Pending" state to the "Active" state.

The VM template is imported and ready for use in Oracle VM Manager.

1.3 Creating the OHIG Secure Health Email VM

To create the OHIG Secure Health Email VM from the VM template:

1. Create a new VM using the OHIG Secure Health Email VM template just installed by following the instructions in the *VM Manager 2.2 User's Guide* (refer to Section 6.3.1, "Creating Virtual Machine from a Template").

2. To power on the Virtual Machine select the **Virtual Machines** tab, choose the **Virtual Machine Name**, and click **Power On**.

3. In the VM Manager Console ensure that the OHIG Secure Health Email VM is now in the running state (Status=Running).

1.4 Configuring the OHIG Secure Health Email VM

This section provides instructions for configuring the OHIG Secure Health Email VM.

- "How to VNC into a VM"
- "Configuring the VM Network Settings"

1.4.1 How to VNC into a VM

To VNC into a VM:

> **Note:** To enable the VNC Port link in the VM Manager follow the instructions in "Installing OVM Console" at
> http://oss.oracle.com/oraclevm/manager/RPMS/README-console.

Expand the details of the VM by clicking the **+** on **Show**. You can VNC into the box from the VM Manager by clicking on the VNC Port link under the VM details, or you can use a VNC client to log in using the address:

<VM_SERVER_HOST_NAME>:<VM_VNC_PORT>

1.4.2 Configuring the VM Network Settings

To configure the VM to use static IP:

> **Note:** The VM is configured by default to use DHCP to assign an IP address.

If you are using DHCP addressing you can skip the following steps.

1. To configure the VM to use static IP, log in as the root user (default password: ovsroot) and set the IP using the following steps:

   a. Select **System**, **Administration**, and then **Network**.

   b. Choose **Devices**, click **Edit**, select the **Statically Set IP Address** radio button, and then enter the following values:

      - Address: *<VM_IP>*

- Subnet mask: *<SUBNET_MASK>*

- Default Gateway address: *<DEFAULT_GATEWAY_ADDRESS>*

- From the Ethernet Device panel, select the **Hardware Device** tab, and then click the **Probe** button that corresponds to "Bind to MAC address".

This sets the correct MAC address for this machine.

> **Note:** Make certain that you a record the MAC address.

c. Click **OK**.

d. Choose **File** and then click **Save**.

e. Click the **DNS** tab and then enter the following values:

- Hostname: *<VM_HOST_NAME>*

- Primary DNS: *<PRIMARY_DNS>*

- Secondary DNS: *<SECONDARY_DNS>*

- Tertiary DNS: *<TERTIARY_DNS>*

- DNS search path: *<VM_NAME_SUFFIX>*

f. Choose **Next** and click **Save**.

g. Choose the **Hosts** tab, click **New**, and then enter the following values:

- Address: *<VM_IP>*

- Hostname: *<VM_HOST_NAME>*

- Aliases: *<VM_NAME_PREFIX>* hostname

h. Click **Okay**.

i. Choose **File** and then click **Save**.

j. Restart Network Services from a terminal window.

```
> service network restart
```

k. Check the output for *<VM_IP>*.

```
> ifconfig
```

l. Check the output for *<VM_HOST_NAME>*.

```
> hostname
```

m. Check the success of:

```
> ping <VM_IP>
```

n. Check the success of:

```
> ping <VM_HOST_NAME>
```

> **Note:** (Optional) In order to preserve the static IP address when the OVM is powered off, follow below steps, but only if the line
>
> ```
> vif = ['mac=AA:BB:CC:DD:AA:CC,bridge=xenbr0']
> ```
>
> does not match what you have in the `vm.cfg` file (see below).
>
> 1. Power off the Virtual Machine by selecting the **Virtual Machines** tab in the VM Manager, choose the **Virtual Machine Name**, and click **Power Off**.
>
> 2. Edit the `vm.cfg` file that is found on the VM Server under `/OVS/seed_pool/<template_name>` by replacing the line:
>
>    ```
>    vif = ['bridge=xenbr0,type=netfront']
>    ```
>
>   with the MAC corresponding to that virtual machine:
>
>    ```
>    vif = ['mac=AA:BB:CC:DD:AA:CC,bridge=xenbr0']
>    ```
>
>   where `AA:BB:CC:DD:AA:CC` is the MAC corresponding to the created OVM noted above.

2

Configuring OHIG Secure Health Email

This chapter leads you through the steps to install, set up, and configure the James Mail Enterprise Server (James) version 2.3.2 to use with OHIG and OHIM. OHIG Secure Health Email is built on top of the Apache James Mail Server.

This chapter includes the following sections:

- "Preparing the Databases"
- "Configuring the Apache James Mail Server"
- "Managing the Apache James Mail Server"
- "Configuring OHIG Secure Health Email"

> **Note:** For a high-level overview of the Secure Health Email network, see Appendix G, "High-level Network Diagram."

2.1 Preparing the Databases

There are two sections with details about preparing the databases:

- "Preparing the Apache James Mail Server Database"
- "Preparing the OHIG Secure Health Email Database"
- "Configuring OHIG Secure Health Email Properties"

For advanced configuration information, follow the links provided in Appendix H.3, "Apache James Mail Server."

2.1.1 Preparing the Apache James Mail Server Database

To prepare the Apache James Mail Server database tables for Oracle:

> **Note:** Because the Apache James Mail Server Database stores sensitive data, it should be set up with encryption turned on.

1. Copy the files under `/home/hiauser/config/files/database/oracle` to a machine with Oracle SQL*Plus installed.

2. Update the script `create-james-user-oracle.sql` with TABLESPACE parameters matching your environment. Also assign a password for the James database user by assigning a value to variable *JAMES_USER_PASS*. Remember to clear the value after you execute the script.

3. To create the Apache James Mail Server database user load the script
   `create-james-user-oracle.sql` into the database.

   Example:

   ```
   > sqlplus system@<SID>
   SQL> @create-james-user-oracle.sql
   ```

4. To create the Apache James Mail Server database load the script
   `create-james-tables-oracle.sql` into the database.

   Example:

   ```
   > sqlplus <JAMESUSER>@<SID>
   SQL> @create-james-tables-oracle.sql
   ```

2.1.2 Preparing the OHIG Secure Health Email Database

To prepare the OHIG Secure Health Email database tables for Oracle:

1. Copy the files under `/home/hiauser/config/files/database/oracle` to a machine with Oracle SQL*Plus installed.

2. Update the script `create-direct-user-oracle.sql` with TABLESPACE parameters matching your environment. Also assign a password for the OHIG Secure Health Email database user, by assigning a value to the variable *DIRECT_USER_PASS*. Remember to clear the value after you execute the script.

3. To create the OHIG Secure Health Email database user load the script
   `create-direct-user-oracle.sql` into the database.

   Example:

   ```
   > sqlplus system@<SID>
   SQL> @create-direct-user-oracle.sql
   ```

4. To create the OHIG Secure Health Email database load the script
   `create-direct-tables-oracle.sql` into the database.

   Example:

   ```
   > sqlplus <DIRECTUSER>@<SID>
   SQL> @create-direct-tables-oracle.sql
   ```

2.1.3 Setting Up a New Source System in OHMPI for Secure Health Email Server

1. In the OHMPI Oracle Database, create an OHMPI source system for the Secure Health Email Server to create new patients.

   > **Note:** Make sure to enable patient feed from OHMPI to RLS as described in section 1.4.3, "Configuring Assigning Authority Patient Feed - Application Server," in *Oracle Health Sciences Information Manager OHMPI Installation and Configuration Guide* (Part Number E22762-01).

2. Execute SQL (below) in the OHMPI database using OHMPI DB user account.

> **Note:** Record If the value in the "systemcode" column is different than "1.1.1", record the value.

```
INSERT INTO sbyn_systems (systemcode, description, status, id_length,format,
input_mask, value_mask, create_date, create_userid) VALUES ('1.1.1', 'ORCL_
DIRECT', 'A', 23, '[0-9]{23}', 'DDDDDDDDDDDDDDDDDDDDDDDD',
'DDDDDDDDDDDDDDDDDDDDDDDD', sysdate, 'MPI');
```

2.2 Configuring the Apache James Mail Server

This provides the settings for configuring OHIG Secure Health Email, the Apache Mail Server for SSL, and the Remote Manager:

- "Configuring OHIG Secure Health Email Properties"
- "Configuring Apache James Mail Server for SSL"
- "Configuring the Remote Manager"

> **Note:** For advanced configuration information follow the links provided in Section H.3, "Apache James Mail Server".

2.2.1 Loading Initial Data into OHIG Secure Health Email Database

Using the OHIG Secure Health Email Configuration Tool, update the tables with initial data as listed below. See Appendix A, "The OHIG Secure Health Email Configuration Tool," for instructions on tool usage.

> **Note:** A version of Open SSL is available in the VM, and, if needed, you may want to use it.

- Add a domain corresponding to your Secure Health Email Server's host name.

  Example: `ant direct-add-domain -Ddomain_ name=secure.health-enterprise.org`

- Add trusted anchors which could include trusted Certificate Authorities.

  Example: `ant direct-add-anchor -Ddomain_ name=secure.health-enterprise.org -Dcert_ file=certs/oracle-cacert.der`

- Add trusted public certificates associating public certificates with external trusted email addresses.

  Example: `ant direct-add-public-cert -Ddomain_ name=secure.health-enterprise.org -Demail_ address=Patient1@live.com -Dcert_file=certs/patient1-cert.der`

- Add trusted private certificates associating public/private certificate pairs with system secure email addresses. Note The email address used in this step should be used to update config parameter james_init.systemEmailAddress in the next section.

> **Note:** The email address used in this step should be used to update config parameter `james_init.systemEmailAddress` in the next section, "Configuring Apache James Mail Server for SSL".

Example: `ant direct-add-private-cert -Ddomain_ name=secure.health-enterprise.org -Demail_address= direct@secure.health-enterprise.org -Dcert_ file=certs/direct-cert.der -Dkey_ file=certs/private/direct-key.der`

- Add trusted private certificates associating public/private certificate pairs with internal secure email addresses.

  Example: `ant direct-add-private-cert -Ddomain_ name=secure.health-enterprise.org -Demail_address= Dr.John.Doe@secure.health-enterprise.org -Dcert_ file=certs/DrJohnDoe-cert.der -Dkey_ file=certs/private/DrJohnDoe-key.der`

- Add addresses mapping internal secure email addresses to internal corporate email addresses and to a domain.

  Example: `ant direct-add-address -Ddomain_name= secure.health-enterprise.org -Ddisplay_name="Dr. John Doe" -Demail_address=Dr.John.Doe@secure.health-enterprise.org -Dendpoint=Dr.John.Doe@ health-enterprise.org`

2.2.2 Configuring OHIG Secure Health Email Properties

1. `> cd /home/hiauser/config`

2. Edit the `config.properties` file.

> **Note:** Refer to http://download.oracle.com/javase/6/docs/api/java/util/Properties.html for property file formatting rules. This link specifies formatting rules for `config.properties`.

OHIG Secure Health Email Settings

- `james_init.xdsRegistryEndpointUrl`

  XDS registry endpoint URL used to query for submission set documents and metadata.

- `james_init.xdsRepositoryEndpointUrl`

  XDS.b repository URL. In case you are testing against the HIG Adapter Repository service, your URL will look like the following URL:

  `http://<OHIG_ADAPTER_ HOST>:8080/CONNECTAdapter/DocumentRepository_Service`

- `james_init.xdsRepositoryId`

  XDS repository UID.

- `james_init.xdsDocumentOidRoot`

Object ID root to be used while generating new document UUIDs.

- `james_init.xdsSubmissionSetOidRoot`

  Object ID root to be used while generating new submission set UUIDs.

- `james_init.assigningAuthorityId`

  OID of XDS affinity domain assigning authority. Used as "root" of the patient ID in conjunction with the EUID, which is used as "extension". This should match with the Assigning Authority OID configured in the HIG Adapter and OHMPI.

- `james_init.mpiServiceUrl`

  OHMPI's web service endpoint URL.

- `james_init.mpiSystemCode`

  System code setup in OHMPI for use by the OHIG Secure Health Email for creating new patients. The value should be the one noted in the previous step ("Setting Up a New Source System in OHMPI for Secure Health Email Server"), followed to update the OHMPI database.

- `james_init.mpiDelayAfterPatientInsertSecs`

  Wait time in seconds before XDS processing to allow for OHMPI to send patient feed XDS registry.

- `james_init.domain`

  Hostname of OHIG Secure Health Email.

- `james_init.systemEmailAddress`

  OHIG Secure Health Email system email address used to send Message Disposition Notification messages, Notification of Document Availability messages, and error message notifications.

- `james_init.manualEmailAddress`

  Email address which receives error message notifications.

- `james_init.useIheNav`

  Flag determining whether or not to use standard IHE Notification of Document Availability messages.

- `james_init.arrHost`

  Policy Monitor hostname.

- `james_init.arrPort`

  Policy Monitor port number.

Apache James Mail Server Database Settings

- `james_db.driver (Example: oracle.jdbc.driver.OracleDriver)`

  Database driver class.

- `james_db.dburl (Example: jdbc:oracle:thin:@<ORACLE_ HOSTNAME>:1521:orcl)`

  Database connection URL.

- `james_db.username (Example: jamesuser)`

  Database username.

- `james_db.password (Example: jamespass)`

  Database password.

  OHIG Secure Health Email Database Settings

- `direct_db.driver (Example:`
  `oracle.jdbc.driver.OracleDriver)`

  Database driver class.

- `direct_db.dburl (Example: jdbc:oracle:thin:@<ORACLE_`
  `HOSTNAME>:1521:orcl)`

  Database connection URL.

- `direct_db.username (Example: directuser)`

  Database username.

- `direct_db.password (Example: directpass)`

  Database password.

3. `> ant config-james`

For advanced configuration properties, see Appendix D, "Advanced OHIG Secure Health Email Property Reference".

2.2.3 Configuring Apache James Mail Server for SSL

1. `> cd /home/common/james/apps/james/SAR-INF`

   Edit the `config.xml` file.

   a. Search for "pop3server" and uncomment:

   `<!--`

   `<useTLS>true</useTLS>`

   `--!>`

   b. Search for "smtpserver" and uncomment:

   `<!--`

   `<useTLS>true</useTLS>`

   `--!>`

   c. Search for "server-sockets" and ensure the correct values are supplied below after un-commenting the tag *<factory name="ssl"..>* :

   `<factory name="ssl"`
   `class="org.apache.avalon.cornerstone.blocks.sockets.TLSSer`
   `verSocketFactory">`

   `<ssl-factory>`

   `<keystore>`

   `<file>keystore/keystore.jks</file>`

   `<password>changeit</password>`

   `<key-password>changeit</key-password>`

   `<type>JKS</type>`

```
<protocol>SSLv3</protocol>

<algorithm>SunX509</algorithm>

<authenticate-client>false</authenticate-client>

</keystore>

</ssl-factory>

</factory>
```

> **Note:** If connecting to remote SMTP gateway or SMTP server also thru SSL, makes sure to specify `javax.net.ssl.SSLSocketFactory` to use as socket factory by "ExtendedRemoteDelivery" mailet.
>
> For example:
> ```
> <mailet match="RecipientIsRemote" class="ExtendedRemoteDelivery">
>     …
>     …
> <mail.smtp.socketFactory.class>javax.net.ssl.SSLSocketFactory</mail
> .smtp.socketFactory.class>
>     …
>     …
> </mailet>
> ```

2.2.4 Configuring the Remote Manager

1. `> cd /home/common/james/apps/james/SAR-INF`

    Edit the `config.xml` file.

    a. Search for "remotemanager", and edit the following two lines:

    `<port>4555</port>`

    `<account login="root" password="root"/>`

    b. To enable secure telnet, uncomment:

    `<!--`

    `<useTLS>true</useTLS>`

    `--!>`

2.2.5 Configuring Logging

Configuring Apache James Mail Server Logging

`> cd /home/common/james/apps/james/SAR-INF`

Edit the "log-level" settings in the `environment.xml` file.

Configuring Application Code Logging

1. Create a JDK `logging.properties` file in the `/home/common/james/bin` directory.

    Example of a `logging.properties` file:

    `handlers= java.util.logging.ConsoleHandler, java.util.logging.FileHandler`

```
.level= INFO

java.util.logging.ConsoleHandler.level = INFO
java.util.logging.ConsoleHandler.formatter = java.util.logging.SimpleFormatter

#java.util.logging.FileHandler.level = ALL
java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter
java.util.logging.FileHandler.pattern = logs/direct%g.log
java.util.logging.FileHandler.limit = 50000
java.util.logging.FileHandler.count = 10
```

> **Note:** You must create the "logs" directory prior to starting the server.

2. Start the Apache James Mail Server with the system property:

```
-Djava.util.logging.config.file=logging.properties
```

2.3 Managing the Apache James Mail Server

This section provides steps to start the Apache James Mail Server, and to connect to the Remote Manager and then manage the Apache James Mail Server:

- "Starting the Apache James Mail Server"
- "Connecting to the Remote Manager"

2.3.1 Starting the Apache James Mail Server

> **Note:** In order for the default SMTP email ports to open, the James Email Server needs to be started by root user.

Use the UNIX aliases set for `root` and `hiauser` to start and stop James server service.

- Stop the James service running with default configuration by running the alias command "stop".
- Start the James service to run with newly configured parameters, by running the alias command "start".
- Use the alias command "jameslog" to see a running tail-end view of the James console log.

2.3.2 Connecting to the Remote Manager

This section provides details for connecting to the Remote Manager and then managing the Apache James Mail Server. It also provides an example of adding a user.

> `telnet localhost` *<PORT>* (Default: 4555)

JAMES Remote Administration Tool 2.3.2

Please enter your login and password

Login id:

<USERNAME> (Default: root)

Password:

<PASSWORD> (Default: root)

Welcome root. HELP for a list of commands

HELP

Currently implemented commands:

- `help`

  Displays this help.

- `listusers`

  Displays existing accounts.

- `countusers`

  Displays the number of existing accounts.

- `adduser [username] [password]`

  Adds a new user.

- `verify [username]`

  Verifies if a specified user exists.

- `deluser [username]`

  Deletes the existing user.

- `setpassword [username] [password]`

  Sets a user's password.

- `setalias [user] [alias]`

  Locally forwards all email for 'user' to 'alias'.

- `showalias [username]`

  Shows a user's current email alias.

- `unsetalias [user]`

  Unsets an alias for 'user'.

- `setforwarding [username] [emailaddress]`

  Forwards a user's email to another email address.

- `showforwarding [username]`

  Shows a user's current email forwarding.

- `unsetforwarding [username]`

  Removes a forward.

- `user [repositoryname]`

  Changes to another user repository.

- `shutdown [repositoryname]`

  Kills the current JVM (convenient when James is run as a daemon).

- `quit [repositoryname]`

  Closes the connection.

2.3.2.1 Example of Add User

adduser *<USERNAME> <PASSWORD>*

Create system user (`james_init.systemEmailAddress`) and manual processor or error (`james_init.manualEmailAddress`) email user accounts configured earlier.

For example:

adduser direct directpass

adduser error errorpass

2.4 Configuring OHIG Secure Health Email

See Appendix A, "The OHIG Secure Health Email Configuration Tool" for configuration instructions.

2.5 Additional Configuration

Editing the System Email Templates

> cd /home/common/james/bin/templates

Edit the files in the `templates` directory (see Appendix C, "System Email Template Reference").

2.6 Testing OHIG Secure Health Email

Inbound and Outbound Examples

> cd /home/hiauser/config/examples

Follow the instructions in the `README.txt` file.

A

The OHIG Secure Health Email Configuration Tool

This appendix provides a description and examples of the OHIG Secure Health Email script.

- "Using the OHIG Secure Health Email Configuration Tool"

A.1 Using the OHIG Secure Health Email Configuration Tool

This section provides a description of the OHIG Secure Health Email Script, and then provides command line tool examples.

- "Description of the OHIG Secure Health Email Script"

- "Example of OHIG Secure Health Email Commands"

A.1.1 Description of the OHIG Secure Health Email Script

`usage: ant` *`<command>`* `-D`*`<option>`*`*`

Use the above script to configure the OHIG Secure Health Email environment.

A.1.1.1 Commands

- `direct-add-address`

  Associate an address with a OHIG Secure Health Email server domain, mapping an internal secure email address to an internal corporate email address

  - **Options**

    * `domain_name=`*`<HOSTNAME>`*

      The OHIG Secure Health Email server domain name

    * `display_name=`*`<STRING>`*

      The display name for the internal secure email user

    * `email_address=`*`<EMAIL>`*

      An internal secure email address

    * `endpoint=`*`<EMAIL>`*

      An internal corporate email address

- `direct-add-anchor`

  Add a certificate to the list of trusted anchor certificates

- **Options**

    * `domain_name=<HOSTNAME>`

        The direct email server domain name

    * `cert_file=<FILE>`

        A public certificate in `.der` format

- `direct-add-domain`

    The OHIG Secure Health Email server domain name

    - **Options**

        * `domain_name=<HOSTNAME>`

            The OHIG Secure Health Email server domain name

- `direct-add-public-cert`

    Associate a public certificate with a external trusted email address

    - **Options**

        * `email_address=<EMAIL>`

            An external trusted email address

        * `cert_file=<FILE>`

            A public certificate in `.der` format

- `direct-add-private-cert`

    Associate a public/private certificate pair with an internal secure email address

    - **Options**

        * `email_address=<EMAIL>`

            An internal secure email address

        * `cert_file=<FILE>`

            A public certificate in `.der` format

        * `key_file=<FILE>`

            A private certificate in `.der` format

- `direct-add-setting`

    Set a configuration setting

    - **Options**

        * `name=<STRING>`

            The setting name

        * `value=<STRING>`

            The setting value

A.1.2 Example of OHIG Secure Health Email Commands

- `direct-add-address`

    ```
    > ant direct-add-address -Ddomain_name=<HOSTNAME> -Ddisplay_
    name=<STRING> -Demail_address=<EMAIL> -Dendpoint=<EMAIL>
    ```

B

OHIG Secure Health Email Tables

This appendix provides six OHIG Secure Health Email tables that list column names and their data type.

This appendix includes the following section:

- "Using the OHIG Secure Health Email Tables"

B.1 Using the OHIG Secure Health Email Tables

This appendix includes the following OHIG Secure Health Email tables:

- Table B–1, " ADDRESS"
- Table B–2, " ANCHOR"
- Table B–3, " CERTIFICATE"
- Table B–4, " DOMAIN"
- Table B–5, " SEQUENCE"
- Table B–6, " SETTING"

Table B–1 ADDRESS

COLUMN_NAME	DATA_TYPE	COMMENTS
ID	NUMBER(19,0)	Primary key
CREATETIME	TIMESTAMP(6)	Create time
DISPLAYNAME	VARCHAR2(255 BYTE)	Display name
DOMAINID	NUMBER(19,0)	Foreign key to address's DOMAIN
EMAILADDRESS	VARCHAR2(255BYTE)	Internal secure email address
ENDPOINT	VARCHAR2(255 BYTE)	Internal corporate email address
STATUS	NUMBER(10,0)	Object status (0=NEW, 1=ENABLED, 2=DISABLED)
TYPE	VARCHAR2(64 BYTE)	Type of address (should be "XD")
UPDATETIME	TIMESTAMP(6)	Update time

Table B–2   ANCHOR[1]

COLUMN_NAME	DATA_TYPE	COMMENTS
ID	NUMBER(19,0)	Primary key
CERTIFICATEDATA	BLOB	Anchor binary data
CERTIFICATEID	NUMBER(19,0)	<deprecated>
CREATETIME	TIMESTAMP(6)	Create time
FORINCOMING	NUMBER(1,0)	Use anchor for incoming messages (0=NO, 1=YES)
FOROUTGOING	NUMBER(1,0)	Use anchor for outgoing messages (0=NO, 1=YES)
OWNER	VARCHAR2(255 BYTE)	Domain hostname (see DOMAIN) or User name
STATUS	NUMBER(10,0)	Object status (0=NEW, 1=ENABLED, 2=DISABLED)
THUMBPRINT	VARCHAR2(255 BYTE)	Anchor thumbprint
VALIDENDDATE	TIMESTAMP(6)	Anchor expiration date
VALIDSTARTDATE	TIMESTAMP(6)	Anchor start date

[1] Changes to the ANCHOR table require a restart of the application.

Table B–3   CERTIFICATE

COLUMN_NAME	DATA_TYPE	COMMENTS
ID	NUMBER(19,0)	Primary key
CERTIFICATEDATA	BLOB	Certificate binary data
CREATETIME	TIMESTAMP(6)	Create time
OWNER	VARCHAR2(255 BYTE)	Certificate associated email address
PRIVATEKEY	NUMBER(1,0)	Holds public and private key (0=NO public only, 1=YES)
STATUS	NUMBER(10,0)	Object status (0=NEW, 1=ENABLED, 2=DISABLED)
THUMBPRINT	VARCHAR2(255 BYTE)	Certificate thumbprint
VALIDENDDATE	TIMESTAMP(6)	Certificate expiration date
VALIDSTARTDATE	TIMESTAMP(6)	Certificate start date

Table B–4   DOMAIN[1]

COLUMN_NAME	DATA_TYPE	COMMENTS
ID	NUMBER(19,0)	Primary key
CREATETIME	TIMESTAMP(6)	Create time
DOMAINNAME	VARCHAR2(255 BYTE)	Domain hostname
POSTMASTERADDRESSID	NUMBER(19,0)	Foreign key to domain's postmaster ADDRESS
STATUS	NUMBER(10,0)	Object status (0=NEW, 1=ENABLED, 2=DISABLED)
UPDATETIME	TIMESTAMP(6)	Update time

[1] Changes to the DOMAIN table require a restart of the application.

Table B–5   SEQUENCE

COLUMN_NAME	DATA_TYPE	COMMENTS
SEQ_COUNT	NUMBER(38,0)	Sequence count
SEQ_NAME	VARCHAR2(50 BYTE)	Sequence name

Table B–6   SETTING[1]

COLUMN_NAME	DATA_TYPE	COMMENTS
ID	NUMBER(19,0)	Primary key
CREATETIME	TIMESTAMP(6)	Create time
NAME	VARCHAR2(255 BYTE)	Setting name
STATUS	NUMBER(10,0)	Object status (0=NEW, 1=ENABLED, 2=DISABLED)
UPDATETIME	TIMESTAMP(6)	Update time
VALUE	VARCHAR2(1024 BYTE)	Setting value

[1] Changes to the SETTING table require a restart of the application.

Required and Optional SETTINGs

NAME	DEFAULT	VALUE	COMMENT
Required SETTINGs			
AnchorStoreType	n/a	WS	Required to be WS
PublicStoreType	n/a	WS	Required to be WS
PrivateStoreType	n/a	WS	Required to be WS
Optional SETTINGs			

NAME	DEFAULT	VALUE	COMMENT
AnchorResolverType	uniform	uniform, multidomain	uniform ■ `FORINCOMING` anchors, are used for both incoming and outgoing messages multidomain ■ `FORINCOMING` anchors, are used for incoming messages ■ `FORINCOMING` anchors, are used for outgoing messages
BadMessageSaveFolder	null	<directory name>	Test directory for logging bad messages
IncomingMessageSaveFolder	null	<directory name>	Test directory for logging incoming messages
OutgoingMessageSaveFolder	null	<directory name>	Test directory for logging outgoing messages
RawMessageSaveFolder	null	<directory name>	Test directory for logging raw messages

C

System Email Template Reference

This appendix provides the System Email Template properties and comments on them.

This appendix includes the following section:

- "System Email Template Table"

C.1 System Email Template Table

The following table provides the name, file name, property, and comments about the System Email template.

```
> cd /home/common/james/bin/templates
```

Table C–1   System Email Template

NAME	FILENAME	PROPERTY	COMMENT
Header	Header.txt	**now**	The day and time of notification generation in ISO 8601 format.
		hostname	The host name of this system.
		hostaddr	The IP address of this system. (IPv4 or IPv6).
Footer	Footer.txt	**now**	The day and time of notification generation in ISO 8601 format.
		hostname	The host name of this system.
		hostaddr	The IP address of this system. (IPv4 or IPv6).
Error Notification (ERROR)	ErrMessage.txt	**now**	The day and time of notification generation in ISO 8601 format.
		hostname	The host name of this system.
		hostaddr	The IP address of this system. (IPv4 or IPv6).
		subject	The subject of the original message.
		sender	The sender of the received message.
		timeDone	The time of receipt.
		recipients	The recipients for this message.
		recipCount	The number of recipients for this message.

Table C–1   (Cont.) System Email Template

NAME	FILENAME	PROPERTY	COMMENT
Message Disposition Notification- Message (MDN)	MdnMessage.txt	**now**	The day and time of notification generation in ISO 8601 format.
		hostname	The host name of this system.
		hostaddr	The IP address of this system. (IPv4 or IPv6).
		subject	The subject of the original message.
		sender	The sender of the received message.
		timeDone	The time of receipt.
		recipients	The recipients for this message.
		recipCount	The number of recipients for this message.
		action	'processed' or 'error'
Message Disposition Notification- Report	MdnReport.txt	**hostname**	The host name of this system.
		recipients	The recipients of this message.
		messageId	The message ID of the original message.
		system	The system address.
		action	'processed' or 'error'
Notification of Document Availability (NAV)	NavMessage.txt	**now**	The day and time of notification generation in ISO 8601 format.
		hostname	The host name of this system.
		hostaddr	The IP address of this system. (IPv4 or IPv6).
		subject	The subject of the original message.
		sender	The sender of the received message.
		timeDone	The time of receipt.
		recipients	The recipients for this message.
		recipCount	The number of recipients for this message.
		action	'processed' or 'error'

D

Advanced OHIG Secure Health Email Property Reference

This appendix provides the Advanced OHIG Secure Health Email properties and comments on them.

This appendix includes the following section:

- "Advanced OHIG Secure Health Email Properties"

D.1 Advanced OHIG Secure Health Email Properties

The following table provides the property, default value, and comments about the Advanced Secure Health Email properties.

1. `> cd /home/common/james/app/james/SAR-INF`

   Edit the `config.xml` file.

2. Search for "InitMailet".

Table D–1   Advanced Secure Health Email Properties

PROPERTY	DEFAULT	COMMENTS
certStoreCachePolicyMaxItems	1000	Maximum certificate cache size
certStoreCachePolicyTtlSecs	86400 (3600*24=one day)	Time-to-live in seconds for certificates in cache
errSubjectLine	Error Notification	Error notification email subject line
mdnSubjectLine	Message Disposition Notification	Message disposition notification email subject line
navSubjectLine	Document Availability Notification	Document availability notification email subject line
xdmSubjectLine	XDM/1.0/DDM	Cross-enterprise document media interchange email subject line
unknownDocClassCodeCodingScheme	1.3.6.1.4.1.21367.3100.1.2	Unknown document type's coding scheme

Table D–1   (Cont.) Advanced Secure Health Email Properties

PROPERTY	DEFAULT	COMMENTS
unknownDocClassCodeCode	Clinical Data	Unknown document type's code
unknownDocClassCodeCodeDisplayName	Unspecified clinical data transferred via OHIG Secure Health Email	Unknown document type's display name
unknownFacilityCodingScheme	1.3.6.1.4.1.21367.3100.1.2	Unknown healthcare facility type's coding scheme
unknownFacilityCode	Unspecified	Unknown healthcare facility type's code
unknownFacilityCodeDisplayName	Unspecified clinical data transferred via OHIG Secure Health Email	Unknown healthcare facility type's display name
unknownPracticeSettingCodingScheme	1.3.6.1.4.1.21367.3100.1.2	Unknown practice setting type's coding scheme
unknownPracticeSettingCode	Unspecified	Unknown practice setting type's code
unknownPracticeSettingCodeDisplayName	Unspecified clinical data transferred via OHIG Secure Health Email	Unknown practice setting type's display name
unknownConfCodeCodingScheme	Connect-a-thon confidentialityCodes	Unknown confedentiality code's coding scheme
unknownConfCodeCode	N	Unknown confedentiality code
unknownConfCodeCodeDisplayName	Normal	Unknown confedentiality code's display name

E

Inbound and Outbound Email Matrix

This appendix provides an inbound and outbound matrix for OHIG Secure Health Email.

This appendix includes the following section:

- "Inbound and Outbound Email"

E.1 Inbound and Outbound Email

See the following tables for inbound and outbound email requests.

Table E–1   Inbound

Accepted Email Types	Required Content-Type	Outcome
S/MIME Email + one or more CCD Attachments	multipart/mixed "text/xml"	When trusted
S/MIME Email + single XDM Attachment	multipart/mixed "*.zip"	■ On processing success: Sender receives "processed" MDN Recipients receive NAV ■ On processing failure: Sender receives "error" MDN Manual handler receives ERROR When Untrusted ■ Email is dropped

Table E–2   Outbound

Accepted Email Types	Required Content-Type	Outcome
Email + single XDS Registry Request[1] Attachment	multipart/mixed "text/xml"	When trusted
		■ On processing success:
Email + single XDM Attachment	multipart/mixed "*.zip"	Sender receives "processed" MDN
		Recipients receive XDM
		■ On processing failure:
		Sender receives "error" MDN
		Manual handler receives ERROR
		When untrusted
		■ Email is dropped

[1] See Appendix F, "XDS Registry Request XML Schema Reference."

F

XDS Registry Request XML Schema Reference

This appendix provides a reference to the XDS Registry Request XML Schema and an example of an XDS registry request.

This appendix includes the following section:

- "XDS Registry Request XML Schema Reference"

F.1 XDS Registry Request XML Schema

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://hsgbu.oracle.com/direct/XdsRegistry/1"
xmlns:tns="http://hsgbu.oracle.com/direct/XdsRegistry/1"
elementFormDefault="qualified" attributeFormDefault="unqualified">
 <xs:element name="XdsRegistryRequest">
  <xs:complexType>
   <xs:sequence>
    <xs:element name="XdsSubmissionSets" type="tns:XdsSubmissionSetsType"
maxOccurs="1" />
   </xs:sequence>
  </xs:complexType>
 </xs:element>

 <xs:complexType name="XdsSubmissionSetsType">
  <xs:sequence>
   <xs:element name="XdsSubmissionSet" type="tns:XdsSubmissionSetType"
maxOccurs="unbounded" />
  </xs:sequence>
 </xs:complexType>

 <xs:complexType name="XdsSubmissionSetType">
  <xs:sequence>
   <xs:element name="XdsDocumentEntry" type="tns:XdsDocumentEntryType"
minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="id" type="tns:OID" use="required" />
  <xs:attribute name="recommendedRegistry" type="xs:string" use="optional" />
 </xs:complexType>

 <xs:complexType name="XdsDocumentEntryType">
  <xs:attribute name="id" type="tns:OID" use="required" />
 </xs:complexType>
```

```
 <xs:simpleType name="OID">
  <xs:restriction base="xs:string">
   <xs:whiteSpace value="collapse"/>
  </xs:restriction>
 </xs:simpleType>
</xs:schema>
```

Example of XDS Registry Request

```
<?xml version="1.0" encoding="UTF-8"?>
<XdsRegistryRequest xmlns="http://hsgbu.oracle.com/direct/XdsRegistry/1">
 <XdsSubmissionSets>
  <XdsSubmissionSet id="1.3.6.1.4.1.21367.2100.1.2.3.1305228382703.7569" />
 </XdsSubmissionSets>
</XdsRegistryRequest>
```

G

High-level Network Diagram

This appendix provides a high level diagram of the OHIG Secure Health Email network.

This appendix includes the following section:

- "OHIG Secure Health Email Network"

G.1 OHIG Secure Health Email Network

The below figure presents a high-level diagram of the OHIG Secure Health Email network.

Figure G–1   High-level View of the OHIG Secure Health Email Network

H

References
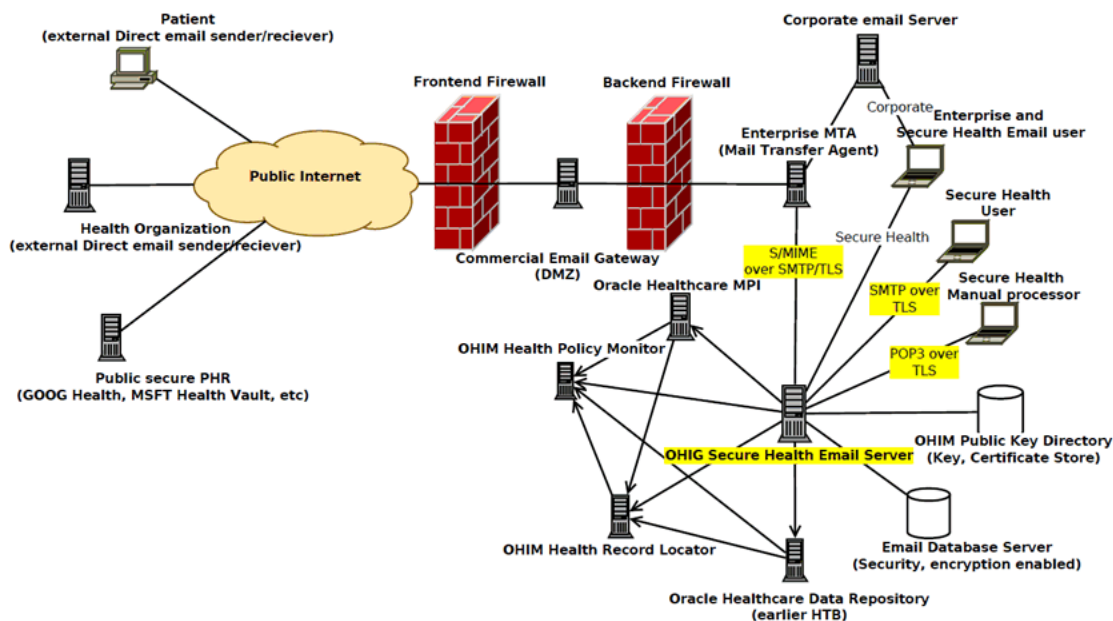
This section provides links to supporting documentation and resources.

H.1 Oracle Virtual Machine

Oracle Virtual Machine (VM) Documentation Index

http://download.oracle.com/docs/cd/E15458_01/index.htm

Oracle VM Manager Release Notes

http://download.oracle.com/docs/cd/E15458_
01/doc.22/e15440/toc.htm

Oracle® VM Manager Installation Guide

Release 2.2, Part Number E15439-01

http://download.oracle.com/docs/cd/E15458_
01/doc.22/e15439/toc.htm

Oracle VM Manager User Guide

Release 2.2, Part Number E15441-02

http://download.oracle.com/docs/cd/E15458_
01/doc.22/e15441/toc.htm

Oracle VM Server Release Notes

http://download.oracle.com/docs/cd/E15458_
01/doc.22/e15443/toc.htm

Oracle® VM Server Installation Guide

Release 2.2, Part Number E15442-01

http://download.oracle.com/docs/cd/E15458_
01/doc.22/e15442/toc.htm

Oracle VM Server User Guide

Release 2.2, Part Number E15444-03

http://download.oracle.com/docs/cd/E15458_
01/doc.22/e15444/toc.htm

Installation of Oracle 11g Database Release 1

Oracle 11g is also available as a VM template

http://www.oracle.com/pls/db111/homepage

Oracle 11g VM Template

http://www.oracle.com/technetwork/server-storage/vm/database-092
479.html

H.2 CONNECT

Please visit the following links for more information about CONNECT software and
documentation:

CONNECT Release Home

http://developer.connectopensource.org/display/NHINR30/Release+3
.0+Home

CONNECT Software Documentation

http://developer.connectopensource.org/display/NHINR30/Software+
Documentation

CONNECT Architecture Overview

http://developer.connectopensource.org/display/NHINR30/Architect
ure+Overview

CONNECT Interface Description Document

http://developer.connectopensource.org/display/NHINR30/Interface
+Description+Document

CONNECT Universal Client GUI User Manual

http://developer.connectopensource.org/display/NHINR30/UC+GUI+Us
er+Manual

Using the CONNECT Solution to support Health Information Exchange

http://developer.connectopensource.org/display/NHINR30/Using+the
+CONNECT+Solution+to+Support+Health+Information+Exchange+(HIE)

CONNECT Specifications

http://www.connectopensource.org/product/connect-nhin-specs

H.3 Apache James Mail Server

Please visit the following links for more information about the Apache James Email
Server and documentation:

James 2.3.2 Documentation

http://james.apache.org/server/2/index.html

Using TLS

http://james.apache.org/server/2/usingTLS.html

I

Acronyms

This section provides a list of commonly used acronyms.

I.1 Acronyms

CCD
Continuity of Care Document

CDA
Clinical Document Architecture

DER
Distinguished Encoding Rules

HIE
Health Information Exchange

HIO
Health Information Organization

HL7
Health Level 7

IHE
Integrating the Healthcare Enterprise

NAV
Notification Of Document Availability

NHIE
Nationwide Health Information Exchange

NHIN
Nationwide Health Information Network

NHIO
Nationwide Health Information Organization

OHIG

Oracle Health Sciences Information Gateway

OHIM

Oracle Health Sciences Information Manager

SAML

Security Assertion Markup Language

VM

Oracle Virtual Machine

WSDL

Web-Service Definition Language

XDM

Cross-Enterprise Document Media Interchange

Glossary

This section provides definitions of commonly used words.

Clinical Document Architecture (CDA)

Clinical Document Architecture (CDA) is a flexible XML-based clinical document architecture that uses the HL7 document markup standard that specifies the structure and semantics for the purpose of exchanging these documents. CDA documents use HL7 v3 Data Types and obtain their machine processable meaning from the HL7 Reference Information Model (RIM). Although the CDA is not a specific document, it can be used to express many types of documents.

CCD, Lab Report (HITSPC37), XDS-MS Discharge Summary (HITSP C48), and History and Physical (HITSP C84) are some of the types of CDA documents. CDA document data sections can be few or numerous and contain narrative text or structured data elements with text or code.

CONNECT

Is a software solution that supports health information exchange that implements Nationwide Health Information Network (NHIN) standards and governance to make sure that health information exchanges are compatible with other exchanges being set up throughout the country. It enables public and private organizations to participate in the NHIN by leveraging their existing health information systems.

CONNECT Adapter

The portion of the CONNECT architecture that encapsulates the components most likely to be customized or replaced by an organization implementing CONNECT.

CONNECT Gateway

The portion of the CONNECT architecture that encapsulates the components most likely to be use as-is by an organization without modification. These components are primarily responsible for orchestrating information exchange with the NHIN.

Continuity of Care Document (CCD)

The Continuity of Care Document (CCD), in accordance with the ASTM E2369-05 Standard Specification for Continuity of Care Record (CCR), describes constraints on the HL7 Clinical Document Architecture, Release 2 (CDA) specification, and is intended as an alternate for the institutions or organizations committed to implementation of the HL7 Clinical Document Architecture specified in the ASTM ADJE2369 implementation. The CCD is just one of numerous types of CDA documents that can contain some of the same CCD sections, but can also contain different sections.

The Continuity of Care Record (CCR) shows one or more patient healthcare encounters, and is the core data set of the most relevant information facts in the patient's health records. It is used to support the patient's continuity of care, and provides a means for a healthcare practitioner, system, or setting to gather together a collection of all of the patient's pertinent data forward it to another practitioner, system, or setting.

Cross-Enterprise Document Media Interchange (XDM)

XDM uses a common file and directory structure over standard media to provide a document interchange that allows patients to carry medical documents using physical media. It also allows person-to-person email for the transfer of medical documents.

Health Information Exchange

Health Information Exchange is an entity that enables the movement of health-related data among entities within a state, a region, or a non-jurisdictional participant group, which might include "classic" regional health information organizations at regional and state levels, Health Information Organization integrated delivery systems and health plans, or health data banks that support health information exchange.

Health Information Organization

Health Information Organization is an organization that enables the movement of health-related data among entities, evolving as a replacement term for health information exchange or HIE. Healthcare Information Technology Standards Panel Or simply HITSP, a cooperative partnership between the public and private sectors formed and supported by ONC for the purpose of harmonizing and integrating standards that will meet clinical and business needs established by AHIC use cases for sharing information among organizations and systems.

Integrating the Healthcare Enterprise

Integrating the Healthcare Enterprise is an initiative by healthcare professionals and industry to improve the way computer systems in healthcare share information, promoting and coordinating the use of established standards such as DICOM and HL7 to address specific clinical need in support of optimal patient care. The Nationwide Health Information Network is being developed by ONC to provide a secure, nationwide, interoperable health information infrastructure that will connect providers, consumers, and others involved in supporting health and healthcare.

Nationwide Health Information Network

Nationwide Health Information Network is a set of standards, services and policies that enable secure health information exchange over the Internet. The network will provide a foundation for the exchange of health information across diverse entities, within communities and across the country, helping to achieve the goals of the HITECH Act. This critical part of the national health IT agenda will enable health information to follow the consumer, be available for clinical decision making, and support appropriate use of healthcare information beyond direct patient care so as to improve population health.

Nationwide Health Information Network Gateway

Within the CONNECT solution, the implementation of the core NHIN services and service interface specifications, comprising the CONNECT gateway and CONNECT adapter. The NHIN health information exchange or NHIE, a health information exchange that implements the NHIN architecture, processes, and procedures, is accredited as a participant of the NHIN.

Oracle Virtual Machine

Oracle Virtual Machine is a platform that provides a fully equipped environment for better leveraging the benefits of virtualization technology. Oracle VM enables you to deploy operating systems and application software within a supported virtualization environment.

Oracle Virtual Machine Manager

Oracle Virtual Machine Manager provides the user interface, which is a standard ADF (Application Development Framework) web application, to manage Oracle VM Servers. It manages virtual machine lifecycle, including creating virtual machines from installation media or from a virtual machine template, deleting, powering off, uploading, deployment and live migration of virtual machines. It manages resources, including ISO files, virtual machine templates, and sharable hard disks.

Oracle Virtual Machine Server

Oracle Virtual Machine Server allows a self-contained virtualization environment designed to provide a lightweight, secure, server-based platform for running virtual machines. Oracle VM Server is based upon an updated version of the underlying Xen hypervisor technology, and includes Oracle VM Agent.

Oracle Virtual Machine Template

Oracle Virtual Machine Template provides an innovative approach to deploying a fully configured software stack by offering pre-installed and pre-configured software images. Use of Oracle VM templates eliminates the installation and configuration costs, and reduces the ongoing maintenance costs helping organizations achieve faster time to market and lower cost of operations.

Security Assertion Markup Language

Security Assertion Markup Language is an XML-based standard for exchanging authentication and authorization data between security domains.

Web Services Description Language

Web Services Description Language is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information.

XML Schema

XML Schema is a means for defining the structure, content, and semantics of XML documents.