Clinical Data Repository / Health Data Repository (CHDR) 2.2

Installation Guide

Document Version 2.2

January 2022

Department of Veterans Affairs (VA)

Office of Information & Technology (OIT)

Federal Information Sharing Technology (FIST)

Revision History

Refer to the SOFTWARE library version of this document to view REDACTED information.

Date	Version	Description	Author
01/2022	2.2	CHDS*2.2*1: Update documentation to reflect the TRM and Fortify compliant release of the CHDR platform and application • Replaced 12c with "TRM compliant" for the WebLogic reference in section **3.4** • Updated log4j jar file version to 2.17.1 in step 3 of section **4.3** • Updated WebLogic link in section **4.4** and **1.3** • Updated section **2** • Updated section **3** • Updated section **3.2** • Linux references updated to Unix references throughout and updated the command in step 2 of section **4.2.2** • Updated Title page, Revision History, Table of Contents, and Footers	Liberty ITS
10/2017	2.1	Update installation guide per decommissioning of the CHDR Admin GUI, CHDR application version number references and WebLogic version number references. Remove all KAAJEE, PSL and VistALink installation references.	READACTED
2/2017	2.0	Updated section 4.9 step 3 to accurately reflect the associated filenames. Updated all references to chdr-2.1.0.8 to chdr-2.1.x.x to indicate this document is applicable to all 2.1 versions of CHDR	READACTED
11/2012	2.0	Updated Pre Installation Considerations, and created steps for the text in the Install Data Sources section. Added Section 4.10, Install CHDR 2.1.x Application. Moved the Backout Procedures, and added an appendix for the Production CHDR Configuration Checklist. Removed Appendices A-C as we now refer to the configuration setup file.	READACTED
1/24/2012	1.0	Technical Writer Review	READACTED
1/19/2012	1.0	Initial Draft	READACTED

Table of Contents

1 Introduction **1**
1.1 Document Purpose 1
1.2 Document Audience 1
1.3 Related Documents 1
2 Pre-Installation Considerations **1**
3 Installation Procedures **3**
3.1 WebLogic Installation and Configuration 3
3.2 Install JDK 3
3.3 Configure CHDR WebLogic Directory Structure 4
3.3.1 Environment Directory Structure 4
3.4 Install and Patch WebLogic Instance 4
3.5 Create and Configure the WebLogic Domain 4
3.6 SSL Certificate Installation 7
3.6.1 Production Keystores 8
3.6.2 SQA and DEV Keystores 8
3.7 WebLogic Managed Servers Start Parameters 8
4 CHDR Domain Configuration **11**
4.1 Install Data Sources 11
4.2 Configure JMS for the CHDR Application 13
4.2.1 Create User for External VIE JMS Access 13
4.2.2 Configure VHIE Persistent Stores 14
4.2.3 Configure VHIE JMS Servers 14
4.2.4 Configure VHIE JMS Module 15
4.3 Install the CHDR Application 15
4.4 Back-Out/Uninstall Procedures 17
5 Post Installation Considerations **18**
6 Troubleshooting **18**
APPENDIX A – Production CHDR Configuration Checklist **19**

1 Introduction

The Department of Defense (DoD) and the Department of Veterans Affairs (VA) in partnership, designed and implemented a Clinical Data Repository/Health Data Repository (CHDR) system that generates standards-based, computable, electronic health records that can be exchanged and shared between the two agency's healthcare systems. By maintaining standardized records, the CHDR system facilitates a seamless transition from military to veteran status and provides for an improved health care delivery system for our Nation's Veterans.

Medical records and patient health care histories are stored and maintained in a centralized repository at each agency. Medical records entered and maintained in the DoD TRICARE system are stored in the Clinical Data Repository (CDR), a component of the Armed Forces Health Longitudinal Technology Application (AHLTA). Similarly, the Health Data Repository (HDR) provides a centralized storage for medical records entered and maintained in the VA VistA, Computerized Patient Record Service (CPRS), and Health*e*Vet systems.

The CHDR system is the link between these two repositories, and includes software components used to exchange clinical records in a real-time, seamless manner and are compliant with the Health Insurance Portability and Accountability Act (HIPAA) and other privacy regulations.

1.1 Document Purpose

The purpose of this document is to provide a detailed description of the procedures and steps necessary for a ground-up installation of the CHDR 2.2 application deployment.

1.2 Document Audience

The intended audience of this manual includes development and production Systems Administrators and Configuration Managers, or any personnel who will be required to install the platform for and/or application of CHDR 2.2. For some descriptions in this document it is assumed that the audience has some level of experience in a Unix system environment and well as some experience in web based application deployment. Training for Unix commands and WebLogic usage is outside the scope of this document.

1.3 Related Documents

- TRM compliant Oracle WebLogic installation, patch and upgrade details can be obtained from: https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/index.html

2 Pre-Installation Considerations

The CHDR application development, sqa/preproduction and production environments are hosted on the Solaris M8 Supercluster - Unix platform at AITC.

All applications in the VA, are required to run on TRM and CRISP compliant platforms. For the CHDR application this includes the M8 Supercluster Unix platform, Oracle JAVA, Oracle WebLogic and Oracle Database. AITC system and WebLogic administrators are responsible for patching the aforementioned components while the development and/or sustainment teams are responsible for testing the applications on any required TRM and CRISP compliant patched components.

The *Production CHDR Configuration Checklist,* included in Appendix A can be used as a guide to ensure you complete all install steps in the deployment of the CHDR application in any environment.

Pre-installation steps to be performed by administrators with appropriate access to each respective environment:

1. Back up all existing directory structures pertaining to the procedures you are attempting. Ensure sure the operating system is patched to the latest allowable patch as prescribed by TRM and CRISP.

2. Contact respective DBA of SQA and Production environments to perform a database comparison to ensure the table names in the SQA environment match the table names in the production environment.

3. Verify that all SSL certificates are installed in the WebLogic managed servers, and or regenerated when updating the domain to a new versions of JAVA and WebLogic.

   Note: If installing a new WebLogic version, it is recommended the new version be installed in a new directory location to ensure a rollback to the old system can be performed without having to restore from backup. This will minimize downtime and maintain an available system should there be issues with the new installation.

4. Verify WebLogic managed server startup parameter `"dgov.va.med.environment.production"` is set to `"true"` in production environment and set to `"false"` in test environment.

5. Verify the amount of wait time between retries of connection is set to 300. Setting can be changed/verified in WebLogic console by setting the Connection Creation Retry Frequency to 300 seconds. See the WebLogic Installation and Administration Guide for details.

6. Ensure that all configuration changes for the timeout value in WebLogic domain JTA tab are set to the correct parameters (from 30 to 3600 seconds for a transaction time out).

3 Installation Procedures

In production, CHDR requires an application server platform capable of load balancing and failover. To support this requirement, the CHDR platform utilizes Oracle WebLogic to deploy multiple managed servers across multiple machines. The current production environment includes two machines, with three managed servers running on each machine. In SQA and DEV environments, only one machine with three managed servers are installed per environment.

The CHDR application requires access to 2 predefined data sources. WebLogic provides the capability to define these data sources and utilize Java Naming Directory Interface (JNDI) technology in the code for reference to these data sources.

Along with these data sources, CHDR also utilizes Java Message Service (JMS) technology to provide patient messages to external interface engines. These interface engines perform both reads and writes to the CHDR JMS queues to perform patient messages transfers to their respective endpoints.

This document includes detailed steps to install the platform and deploy the required applications necessary for CHDR to accomplish its mission in production, SQA, and Development environments. These steps will describe everything necessary to build a CHDR TRM compliant WebLogic platform from the ground up in all three different environments.

3.1 WebLogic Installation and Configuration

AITC Administrators and CHDR development and sustainment personnel with approved elevated permissions work together to perform patching and software installations in each respective environment. JAVA is installed in a 'weblogic' user owned directory for better versatility if/when JAVA updates are required.

To execute any operating system level procedures described in this document, a TRM compliant terminal emulator is used to connect to the respective host. Authentication will take place based on ePAS approved elevated privileges and credentials. The required switch user command to execute elevated permissions is the 'dzdo' command. Only AITC administrators have exclusive permissions to switch to the root user, however, non AITC administrators with approved ePAS permissions have the ability to prefix system level commands with the 'dzdo' command to allow system level maintenance and troubleshooting in the lower environments where and when necessary. Non AITC administrators can also elevate there permissions to be a 'weblogic' or 'oracle' user by utilizing the following commands:

```
dzdo su – weblogic
dzdo su - oracle
```

3.2 Install JDK

Following the CRISP initiative and TRM compliance requirements, system administrators must identify the appropriate version of JAVA and install it in the /u01/app/oracle/java directory. After installation a symbolic link /u01/app/oracle/java/latest should be created and pointed to the desired version of JAVA within the same /u01/app/oracle/java directory.

```
ln -s /u01/app/oracle/java/jdk1.8,0_xxx /u01/app/oracle/java/latest
```

The /u01/app/oracle/java/latest will be used in the WebLogic configurations to minimize impact to JAVA updates and/or changes.

3.3 Configure CHDR WebLogic Directory Structure

3.3.1 Environment Directory Structure

The current directory structure on CHDR machines is as follows:

APPLICATION_HOME = */u01/app/install* --- contains deployable CHDR applications support files.

DOMAIN_HOME = */u01/app/domains/<chdr domain name>* --- contains the WebLogic domain created for the respective environment.

DOMAIN_HOME/applications – contains CHDR application component properties files.

WL HOME = */u01/app/oracle/<installation directory>* --- contains the WebLogic installation and supporting tools to administer WebLogic domains.

JAVA_HOME = */u01/app/oracle/java/latest* --- points to the latest installation of JAVA.

3.4 Install and Patch WebLogic Instance

WebLogic installation details can be obtained from Oracle's documentation site as referenced in the Paragraph 1.3 of this document. The desired TRM compliant WebLogic installation file will be retrieved from production support to ensure proper licensing.

In production, the installation must be performed on both production machines.

Oracle has decommissioned command line "console" installations and AITC has restricted the use of graphical user interface support on the Solaris servers therefore, a silent install is executed using a response file from the command line.

Example of silent install command:

java -jar <*distribution_name.jar*> -silent -responseFile <*response filename*>

Please refer to documentation for more installation details.

The WebLogic middleware should be patched as recommended by TRM, CRISP and Oracle installation requirements. The AITC admins will retrieve the WebLogic installation and patch files from oracle support since they hold the Oracle support contract credentials.

When the WebLogic installation is completed, check for the patches available for the installed version of WebLogic and and follow the online instructions provided in the documentation link in Section 1.3 of this document to complete patch installations.

The new WebLogic server home instance will be located in directory:

WLS_HOME = <*WL_HOME*>*/wlserver*

3.5 Create and Configure the WebLogic Domain

A CHDR domain must be created in each environment. Since the Oracle's console mode is no longer available from Oracle and security restrictions prevent AITC servers from X server configurations, GUI

tools are no longer available. The current approach to creating CHDR domains is to utilize the WebLogic Scripting Tool (WLST) to create a basic domain. A basic domain will provide access to a WebLogic console where the rest of the CHDR components can be installed manually.

A prerequisite to creating the domain is to obtain or create a domain template jar file to be used in the WLST tool.

If an existing domain template cannot be generated from an existing CHDR domain a default template is provided with the WebLogic software installation.

The basic domain template delivered with the WebLogic middleware is:

<WLS_HOME>/common/templates/wls/wls.jar

To create the domain, start WLST by executing the following at the unix command prompt:

```
cd <WL_HOME>/oracle_common/common/bin
./wlst.sh
```

At the wls:/offline> prompt enter the following command:

> createDomain ('*<path to domain_template>*', 'DOMAIN_HOME', '*<weblogic username>*','*<weblogic pwd>*');

After creating the domain, record the weblogic username and password. Typically the default user is 'weblogic' and the password chosen by the installer. Additional administrator names can be added to the WebLogic Security Realm to avoid generic logins to the console.

Once the domain is created and a WebLogic console is accessible the following steps can be performed:

1. Create machines. In the production create 2 machines. In DEV and SQA create one machine per environment.

2. Create managed servers. In production create 6 managed servers: *<chdr domain name>*.ms1 - *<chdr domain name>*.ms6. Assign *<chdr domain name>*.ms1 - *<chdr domain name>*.ms3 to one machine and *<chdr domain name>*.ms4 - *<chdr domain name>*.ms6 to the other machine.

   In the DEV and SQA environments, create 3 managed servers: *<chdr domain name>*`.ms1` – *<chdr domain name>*`.ms3`. for each environment. Only one machine is created on each DEV and SQA domain.

3. Configure each domain with a cluster and assign all respective managed servers to their respective cluster.

4. For each machine configured in a domain, identify the host name and the Node Manager Listener port.

5. In production, create a template of managed servers for machine 2 based on the machines created in the previous steps. Use the WebLogic 'pack' and 'unpack' utilities to enroll the machines in the Nodemanager configuration.

   a. On machine 1, go to directory *WL_HOME/oracle_common/common/bin* and execute the following command:

Note: Command is entered all on one line at command prompt in the UNIX shell.

```
./pack.sh -domain=DOMAIN_HOME  -
template=<APPLICATION_HOME>/chdr.prod.jar    -
template_name="chdr.prod" -managed=true
```

b. Transfer the newly created `chdr.prod.jar` file to the *<APPLICATION_HOME>* directory on machine 2.

On machine 2, go to directory *WL_HOME/oracle_common/bin* and execute the following command:

```
./unpack.sh -template=<APPLICATION_HOME>/chdr.prod.jar -
domain=DOMAIN_HOME/chdr.prod
```

c. Use the following commands to enroll the Node Manager on machine 2 with the Node Manager running on machine 1.

d. On machine 1, start the Node Manager and the WebLogic administrator server:

```
cd <DOMAIN_HOME>/bin
nohup ./startNodeManager.sh &
cd ..
./startWebLogic.sh
```
– ***enter the username and password defined in the domain creation***

e. On machine 2 perform the enrollment utilizing the WebLogic Scripting Tool (WLST) as follows:

```
cd <WLS_HOME>/common/bin
./wlst.sh
connect("weblogic", "<password>", 't3://<machine1
hostname>:<AdminServer     port>')
nmEnroll('<DOMAIN_HOME>','<DOMAIN_HOME>/nodemanger')
exit()
```

f. Start NodeManager on machine 2 as follows:

```
cd <DOMAIN_HOME>/bin
nohup ./startNodeManager.sh &
```

g. Test access to the WebLogic console URL:

http://<machine 1 hostname>:<AdminServer port>/console

h. In production, verify that the Node Manager recognizes both machines as follows: In the left panel of the console screen:

Expand Environments

Click on Machines

Click on machine 2 hostname

Click on the Monitoring tab

Node Manager Status should be Reachable

6. In SQA and DEV environments, start the Node Manager and administration server as follows:

```
cd <DOMAIN_HOME>/bin
nohup ./startNodeManager.sh &
cd ..
./startWebLogic.sh – enter the username and password defined during domain creation
```

7. Test access to the WebLogic console URL:

   http://<hostname>:<AdminServer port>/console

8. In all environments, from the WebLogic console, perform a shutdown of the AdminServer.

9. Create the `boot.properties` security file so that the AdminServer can start without a user intervention for a user ID and password during WebLogic startup.

10. In all environments, perform the following to create the `boot.properties` file.

```
cd DOMAIN_HOME/servers/AdminServer
mkdir security
cd security
```

Note: insert the following lines into a boot.properties file:

```
echo "username=weblogic" > boot.properties
echo "password=<password>" >> boot.properties
```

11. Verify starting the WebLogic AdminServer without user intervention for user ID and password by restarting the respective AdminServer:

```
cd  <DOMAIN_HOME>/bin
nohup  ./startWebLogic.sh &
```

12. Access the console URL `http://<hostname>:<AdminServer port>/console` and log in.

3.6 SSL Certificate Installation

WebLogic is delivered with a DemoIdentity and a DemoTrust keystore. Each of these has a well-known default password and it is highly recommended new keystores are created with new passwords.

The VA and AITC require approved VA Certificate Authority (CA) certificates are installed in production and test environments.

VA CA signed certificates can be obtained with the Assistance of AITC administrators who have access to the VA's Venafi Certificate management systems. Once VA signed certificates are received they can be installed in custom identity and trust keystores for the WebLogic domains.

3.6.1 Production Keystores

The keystores in production environment are maintained by the AITC CHDR WebLogic Admin and configured according to AITC standards and consistent with other project WebLogic installations. The CHDR test environments follow those standards as closely as possible to maintain consistency in deployment platforms across environment boundaries. Generally speaking:

The trust keystore is located in:

WLS_HOME/server/lib/truststorename.jks

And the identity keystore is located in:

DOMAIN_HOME/security/identitystorenam.jks

3.6.2 SQA and DEV Keystores

The keystores used in DEV are:

WLS_HOME/server/lib/05trust.jks

And

DOMAIN_HOME/security/05identity.jks

The keystores used in SQA are:

WLS_HOME/server/lib/<keystore_name>trust.jks

And

DOMAIN_HOME/security/<keystore_name>identity.jks

WebLogic domains will require a Custom Keystore Configuration pointing to the respective keystores and aliases for encrypted connection between the WebLogic domain managed servers and the WebLogic node manager.

3.7 WebLogic Managed Servers Start Parameters

Each managed server in the CHDR domain will require a certain amount of system resources to handle its respective required load. These resources are assigned in the configuration Server Start tab of each managed server.

In production, the following parameters should be entered into the arguments box of each managed Server Start tab:

```
-server -Xms3072m -Xmx3072m -XX:CompileThreshold=8000 -XX:PermSize=16m -
XX:MaxPermSize=512m -XX:SurvivorRatio=12 -XX:+UseConcMarkSweepGC -
XX:+UseParNewGC
```

Also, in production additional parameters are added in order to incorporate system monitoring using the Introscope utility and the following parameters have to be added to the startup parameters:

```
-javaagent:/u01/app/introscope-9.0.7/Agent.jar

-Dcom.wily.introscope.agent.agentName=chdr_prod_msx

-Dcom.wily.introscope.agentProfile=/u01/app/introscope-
9.0.7/IntroscopeAgent.profile
```

AITC Admins have the necessary files for these settings and control the production configuration.

In the SQA and DEV environments the following parameters should be entered into the arguments box of each managed Server Start tab:

```
-server -Xms64m -Xmx896m -XX:CompileThreshold=8000 -XX:PermSize=16m -
XX:MaxPermSize=256m
```

In all environments, for WebLogic, when defining the classpath for managed servers the following must be added to the *Classpath* box in the *Server Start* tab of each managed server:

WL_HOME/server/lib/weblogic_sp.jar:WL_HOME/server/lib/weblogic.jar

Add the following to the *Security Policy File* box in each managed *Server Start* tab.

```
WLS_HOME/server/lib/weblogic.policy
```

The *Server Start* tab can be found by executing the following steps:

1. Log into the respective WebLogic console.
2. In the Domain Structure window expand **Environments.**
3. Click on **Servers.**
4. Click on a server in the Server table to pull up the server configuration.
5. Select the **Server Start tab** and locate the **Arguments** box.
6. In the *Change Center box* in the top left corner of the screen click **Lock & Edit.**
7. Enter the respective parameters described above and click the **Save** button.

For the new domain copy the latest `wllog4j.jar` file from the WebLogic installation directory to the domain lib directory:

cp WLS_HOME/server/lib/wllog4j.jar DOMAIN_HOME/lib

The new domain also needs a modified `setDomainEnv.sh` file for production and test environments.

For all environments add the following text just above the line in `setDomainEnv.sh` containing the text:

```
MEM_DEV_ARGS=""

#@REM CHDR Specific Configuration -- START

USER_MEM_ARGS="-server -Xms64m -Xmx1024m -XX:CompileThreshold=8000 -
XX:PermSize=16m -XX:MaxPermSize=256m"

export USER_MEM_ARGS

#@REM CHDR Specific Configuration -- FINISH
```

For all environments add the following to the *JAVA_PROPERTIES* line in `setDomainEnv.sh`:

```
-Davax.management.bulder.initial=weblogic.management.jmx.mbeanserver.
WLSMBeanServerBuilder
```

Once all of the preceding steps have been taken for installing and configuring a baseline domain, restart all of the servers including the admin server.

The WebLogic platform is now prepared to accept CHDR-specific modifications.

4 CHDR Domain Configuration

The CHDR domain consists of several major components that need to be configured in order to deploy the CHDR application. These include:

- Data Sources
- CHDR specific Java Message System (JMS) module
- The CHDR application and associated properties file

NOTE: In the production environment where two machines exist, all files and directories written to directories *DOMAIN_HOME/applications* and *DOMAIN_HOME/lib*, must be done on both machines. **Those two directories must match in order for WebLogic load balancing and failover to operate correctly.**

4.1 Install Data Sources

The CHDR application includes a CHDR database. The database resides on a separate machine and connections to the database are accomplished via the Java Database Connectivity (JDBC) data sources defined in the WebLogic CHDR domain as referenced in the application. In the current production environment CHDR does not utilize failover data sources and relies on one connection to the CHDR database. SQA and DEV environments are configured with "Multi" data sources and are set up to handle two different database instances in case one fails.

The names of the data sources in production and the *Multi* data sources in SQA and DEV are:

- `DefaultNonXADataSource`
- `DefaultXADataSource`

In WebLogic, perform the following to create the Data Sources:

1. Log into the WebLogic Server console.
2. In the *Domain Structure* window, expand **Services**.
3. Click **Data Sources.**

In production only three data sources are created. In SQA and DEV, nine data sources are created.

In production, failover and/or load balancing for the data sources is not configured:

1. In the Change Center window, click **Lock & Edit**.
2. In the Data Sources table, click the **New** button.
3. Select **Generic Data Source** from the list.
4. In the Name box enter `DefaultNonXADataSource`.
5. In the JNDI Name box enter `vhie.jdbc.DefaultDataSource`.
6. Click the **Next** button.
7. From the Database Driver drop down list, select ***Oracle's Driver (Thin) for Instance connections; Versions:Any**.
8. Click the **Next** button.

9. Accept all defaults in the Transaction Options screen and click **Next**.
10. Enter the Database Name.
11. Enter the Hostname of the Database server.
12. Enter the production Port number for the Database Listener.
13. Enter the production Database User Name and Password.
    a. Confirm Password and click **Next**.
14. Click **Test Configuration**.
15. After testing succeeds, click **Finish**.
16. Click **Save** and Activate Changes.

Repeat the previous steps using:

Name – DefaultXADataSource*JNDI Name* – `vhie.jdbc.DefaultXADataSource`

Database Driver - ***Oracle's Driver (Thin XA) for Instance connections; Versions: Any**

Database Name – same as above

Hostname – same as above

Port – same as above

Database User Name and Password – same as above

In SQA and DEV repeat the WebLogic steps described above using the following information:

Name – `DefaultNonXADataSource`

JNDI Name – `vhie.jdbc.DefaultDataSource1`

Database Driver - ***Oracle's Driver (Thin) for Instance connections; Versions: Any**

Database Name – database instance

Hostname – host for database instance

Port – port for database instance listener

Database User Name and Password – for database instance

Name – `DefaultXADataSource`

JNDI Name – `vhie.jdbc.DefaultXADataSource`

Database Driver - ***Oracle's Driver (Thin XA) for Instance connections; Versions: Any**

Database Name – database instance

Hostname – host for database instance

Port – port for database instance listener

Database User Name and Password – for database instance

4.2 Configure JMS for the CHDR Application

The VHIE JMS system consists of several components:

- Create generic user for external JMS system access
- VHIE Persistent Stores
- JMS VHIE Servers
- JMS VHIE Module

4.2.1 Create User for External VIE JMS Access

1. Log into the WebLogic console.
2. In the *Domain Structure* window click on **Security Realms**.
3. In the *Summary of Security Realms* window, click on **myrealm**.
4. Click on the **Providers** tab.
5. Click on the **Password Validation** tab.
6. Verify **SystemPasswordValidator** exists in the *Password Validation Providers* table.
7. Click on the *SystemPasswordValidator* **provider**.
8. Click on the **Provider Specifics** tab.
9. Verify *Character Policies* are all **0**.
10. In the *Domain Structure* window click on **Security Realms**.
11. In the *Summary of Security Realms* window, click on **myrealm**.
12. In the *Settings for myrealm* window, click the **Users and Groups** tab.
13. Click **New**.
14. Enter **chdroperator** in the *Name* box.
15. Enter **'JMS access user'** in the *Description* box.
16. Select the **DefaultAuthenticator** from the *Provider* drop down list.
17. Enter **'chdroperator'** in the *Password* box.
18. Enter **'chdroperator'** in the *Confirm Password* box.
19. Click **Ok**.

4.2.2 Configure VHIE Persistent Stores

When configuring persistent stores for the CHDR domain in WebLogic, specific directories must be created and associated to each managed server. These directories do not get created during the persistent store creation from the WebLogic console.

Use the following steps to create the persistent store directories for the CHDR domain:

1. Using a preferred terminal emulator, connect to the respective host and log in.
2. Switch user to the WebLogic admin user with the sudo command in Unix.

   ```
   dzdo su - weblogic
   ```

3. For each managed server a **VhieFileStore_auto_x** directory must be created in the *DOMAIN_HOME/servers/<server_name>/data/store* directory.
   Note: x represents the server number in the cluster.
4. In the production environment, this needs to be performed on both servers.

   Example:

   ```
   cd DOMAIN_HOME/servers/<domain_name.ms1>/data/store
   mkdir VhieFileStore_auto_1
   cd DOMAIN_HOME/servers/<domain_name.ms2>/data/store
   mkdir VhieFileStore_auto_2
   …
   ```

5. After the directories have been created, log into the WebLogic console.
6. Click **Lock & Edit**.
7. In the *Domain Structure* window, expand **Services** and select **Persistent Stores.**
8. In the *Persistent Stores* window click **New** and select **FileStore**.
9. Create *Persistent Stores* **VhieFileStore_auto_x** with *Targets* matching the respective managed server number x.
10. In the *Synchronous Write Policy* select **Direct-Write.**
11. Enter the directory **VhieFileStore_auto_x** for each store respective to its managed server.

4.2.3 Configure VHIE JMS Servers

In the *Domain Structure* window:

1. Expand **Services**.
2. Expand **Messaging** and select **JMS Servers**.
3. Click **New** and create **VhieJmsServer_auto_x** for each respective *Persistent File Store* and target each **VhieJmsServer_auto_x** to its respective managed server number.

4.2.4 Configure VHIE JMS Module

Retrieve the CHDR VHIE JMS Module xml file **vhiejmsmodule-jms.xml** and copy it into the DOMAIN_HOME/config/jms directory.

In the *Domain Structure* window:

1. Expand **Services.**
2. Expand **Messaging** and select **JMS Modules**.
3. Click **Next**.
4. Enter **VhieJmsModule** for the *Name*.
5. Enter **vhiejmsmodule-jms.xml** for the *Descriptor File Name*.
6. Click **Next**.
7. Select the cluster as the Target for the module.
8. Click **Next**.
9. Click **Finish**.

In the *Domain Structure* window:

1. Expand **Services.**
2. Expand **Messaging** and select **JMS Modules**.
3. Select the **new VhieJmsModule.**
4. In the *Settings for VhieJmsModule* window click the **Subdeployments** tab.
5. Click **New**.
6. Enter **Vhie_Jms_Module_Subdeployment** for or the *Name*.
7. Select the cluster for the target of the subdeployment.
8. Click **Save**.
9. Click **Activate Changes**.

Restart all managed servers including the Admin Server and check logs for errors.

4.3 Install the CHDR Application

The CHDR deployment consists of deploying the EAR file and the `core.properties` file.

The `core.properties` file exists in three versions, one for each environment: production, SQA and DEV. For the contents of each version of the `core.properties` file.

1. Transfer the CHDR ear to the SQA server /tmp directory where AITC personnel can access it from production.

2. In production, copy the CHDR ear file to the APPLICATIONS_HOME directory for deployment. Create the applications directory and put the core.properties file in it with the following commands.

```
mkdir DOMAIN_HOME/applications/chdr-core
cp core.properties DOMAIN_HOME/applications/chdr-core
```

3. The following files need to be transferred and distributed to the DOMAIN_HOME/lib directory for the CHDR application.

```
CAIP_Keystore.jks
CAIP_JAAS.config
com.bea.core.apache.commons.logging_1.1.0.jar
log4j-2.17.1.jar
log4j.xml
resolver.jar
serializer.jar
wlcommons-logging.jar
xalan-2.7.1.jar
xercesImpl.jar
xml-apis.jar
```

4. Log into the WebLogic console.

5. Click the **Lock & Edit** button.

6. In the *Domain Structure* window select **Deployments**.

7. Click the **Install** button in the *Summary of Deployments* window.

8. Change the path in the *Install Application Assistant* to **APPLICATIONS_HOME**.

9. Click **Next**.

10. Select the chdr-<*version*>**.ear** and click **Next**.

11. Select **Install as Application** radio button and click **Next**.

12. Select the domain cluster for the target and click **Next**.

13. Modify the *Name* to **chdr-<***version***>**

14. Click the **Copy this application onto every target for me** radio button.

15. Click **Next**.

16. Click the **No, I will review the configuration later** radio button.

17. Click **Finish**.
18. Click **Activate Changes**.
19. Select the **chdr-<*version*>** deployment checkbox and click **Start**.
20. Select **Servicing All Requests**.
21. Click **Yes** to start the deployment.
22. Restart the managed servers.

4.4 Back-Out/Uninstall Procedures

If the recommended pre-install procedures were followed, the back-out procedure consists of shutting down and killing the new installation processes and restarting the old installation of the application as described in previous paragraphs of this document. If procedures were not followed, then the old installation can be restored from backup and restarted.

For any uninstall procedures, refer to the Oracle WebLogic installation details can be obtained from Oracle's documentation site at https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/index.html.

5 Post Installation Considerations

Post installation considerations include checking server and deployment status from the WebLogic console and smoke testing the application to ensure the application has been brought online correctly in the respective installation environment. Refer to the CHDR Environment Specs spreadsheet located on the CHDR Team SharePoint repository under the "Current" link. for details in accessing the WebLogic console URL and the application URL.

From the Servers table in the WebLogic console, verify an **OK** *Health* status for each managed server.

From the Deployments table in the WebLogic console, verify an **Active** *State* with an **OK** *Health* status for each deployment.

6 Troubleshooting

Any issues concerning the installation of Oracle's WebLogic middleware or the CHDR application can initially be addressed in the installation, node manager, admin server or managed server log files that are created upon startup of each component. Any system resource or network issues will have to be addressed by AITC. All other issues will have to be addressed depending on the errors at hand. There is no detailed list of possible errors and troubleshooting will have to be accomplished according to any issue reported and/or logged.

If there are no installation or startup errors, but there are still functional issues, refer to the managed server logs for possible indicators of an external dependent system being down or inaccessible. Attempt to identify the problem system and contact the administration group or individual responsible for that system for assistance in resolving that issue. You can refer to the *Systems Management and Security Guide* for possible contacts of each external component.

APPENDIX A – Production CHDR Configuration Checklist

As you deploy and configure CHDR in the production environment, refer to this checklist to make sure that you complete all necessary tasks. The section number of the *CHDR Installation Guide* has been provided as a reference for additional installation instructions.

	Item	Installation Guide Section	Done
1	Java (JDK) is installed.	3.2	☐
2	WebLogic Domain created.	3.5	☐
3	Six managed servers are installed (three servers on each machine).	3.5	☐
4	Cluster is created and contains the correct managed servers.	3.5	☐
5	Node manager listener configuration for each managed server.	3.5	☐
6	SSL is configured for each managed server.	3.6	☐
7	Managed servers start parameters.	3.7	☐
8	Data source configuration.	4.1	☐
12	VHIE JMS security realms configuration.	4.7	☐
13	VHIE Persistent Stores configuration.	4.7	☐
14	VHIE JMS Servers configuration.	4.7	☐
15	VHIE JMS module configuration.	4.7	☐
17	CHDR application deployment.	4.9	☐
18	**OK** Health status for each managed server (from the Servers table in the WebLogic console).	5	☐
19	**Active State** with an **OK** Health status for each deployment (from the Deployments table in the WebLogic console).	5	☐

References:

CHDR Environment Specs: http://REDACTED/CHDR/CHDR%20Documents/Forms/AllItems.aspx