

Frequently Asked Questions

CE Version 3.3.6

July 2021  
Document Release 1.0

|  |
| --- |
| EXIMBILLS © Trade Finance System  Customer Enterprise System Version 3.3.6  Frequently Asked Questions  July 2021 |
| *Copyright 2021 © China Systems Corporation*  *All Rights Reserved*  *This document is protected by United States Copyright Law and may contain Trade Secrets Information which is proprietary to China Systems Corporation. No part of this document may be copied, photocopied, reproduced, translated, distributed, or reduced to any electronic medium or machine-readable form without prior consent in writing from China Systems Corporation. The information in this document may be used only under the terms and conditions of separate China Systems Corporation license agreements.*  *Information is subject to change without notice. China Systems Corporation makes no warranties, either expressed or implied, with respect to the software herein described as to its quality, performance, including, without limitations to, its fitness for any particular purpose.*  *This document may not reflect total system capability at any subsequent date as a result of development. It is also possible that it may contain references to facilities not available on your computer system. Such references should not be construed to mean that these facilities will necessarily be made available on all types of computer hardware or in all user locations.*  *China Systems Corporation accepts no responsibility or liability for any damages or loss of business or revenue due to the use of this document.*  *All trademarks, registered trademarks and trade names mentioned in this document are the sole property of their respective holders.* |
| ORDER MORE EXIMBILLS DOCUMENTATION  Additional copies of Documentation are available for purchase from China Systems Corporation or through your local EXIMBILLS Support Office. |
| csliit  CHINA SYSTEMS CORPORATION  Comments may be addressed to:  [corporatedocs@chinasystems.com](mailto:corporatedocs@chinasystems.com)  China Systems Corporation Ltd.  Corner House, 20 Parliament Street  Hamilton HM12  Bermuda |

Table of Contents

[CE Documentation Library 7](#_Toc75262053)

[CE Documentation Library 8](#_Toc75262054)

[Core System Manuals 8](#_Toc75262055)

[CE Utility Reference Manuals 11](#_Toc75262056)

[Installation Guides 11](#_Toc75262057)

[Introduction 12](#_Toc75262058)

[System Overview 13](#_Toc75262059)

[Manual Overview 14](#_Toc75262060)

[Browser-side Issues 15](#_Toc75262061)

[Creating the Super Officer User Profile 16](#_Toc75262062)

[CECheckCharacterFilterError 17](#_Toc75262063)

[Inbox or Detail Info Error Message 18](#_Toc75262064)

[Inbox Not Working Properly 19](#_Toc75262065)

[Downloading Transaction Data on Weblogic 21](#_Toc75262066)

[Modified LBI Files 22](#_Toc75262067)

[Supporting Multi-Languages 23](#_Toc75262068)

[Session Timeout and Auto Save 24](#_Toc75262069)

[Session Extension and Related Settings 25](#_Toc75262070)

[Cancel Button Issue 40](#_Toc75262071)

[Uploading Files or Images 41](#_Toc75262072)

[Parameter Settings 41](#_Toc75262073)

[Transaction Processing 42](#_Toc75262074)

[Authorization Not Working 44](#_Toc75262075)

[Setting the Date Format 45](#_Toc75262076)

[EC Ledger Error 46](#_Toc75262077)

[Disable Cache 48](#_Toc75262078)

[Manually Stop/Start the WebSphere Service 49](#_Toc75262079)

[Browser Logon Information 52](#_Toc75262080)

[Encrypting User Logon Information 53](#_Toc75262081)

[Chrome Browser Settings 59](#_Toc75262082)

[IE9 Browser Settings 60](#_Toc75262083)

[Displaying the Last Logon Failure 61](#_Toc75262084)

[Browser Navigation Limitations 63](#_Toc75262085)

[JSP and JS File Modifications 63](#_Toc75262086)

[Parameter Settings 69](#_Toc75262087)

[Multiple Authorization Functions in One Product 71](#_Toc75262088)

[Display Error When Printing Historical Records 73](#_Toc75262089)

[Display All Records For Inquire CUBK 77](#_Toc75262090)

[Disallowing the System Login GET Method 83](#_Toc75262091)

[CE Utility and Parameter Issues 86](#_Toc75262092)

[Creating the Super Admin User Profile 87](#_Toc75262093)

[Host Name and Port Number of the Application Server 88](#_Toc75262094)

[Oracle Driver Error 91](#_Toc75262095)

[SQL 2000 SP3 Error 92](#_Toc75262096)

[Catalog Setting Rules 93](#_Toc75262097)

[Maintaining Item Ids 94](#_Toc75262098)

[Deleting Data Source and User Information 95](#_Toc75262099)

[Creating the CE EAR File Backup 96](#_Toc75262100)

[Creating the Backup of CE.EAR Using a Batch File 96](#_Toc75262101)

[Creating the Backup of CE.EAR Using the Command Prompt 99](#_Toc75262102)

[Selecting Multiple Transaction or Security Records in a Catalog 104](#_Toc75262103)

[Selecting Multiple Inbox Records in a Catalog 105](#_Toc75262104)

[CE Utility and CE New Utility Directory Structures 106](#_Toc75262105)

[Security Issues 114](#_Toc75262106)

[Add Admin FAP Issue 115](#_Toc75262107)

[Function Errors on WebLogic 116](#_Toc75262108)

[Supported Security Tokens 117](#_Toc75262109)

[CE DB User Rights in a Production Environment 118](#_Toc75262110)

[Token to Prevent CSRF Attack 119](#_Toc75262111)

[Secure Cookie Attributes 121](#_Toc75262112)

[Hiding Parameters in the Popup Window URL 123](#_Toc75262113)

[Password Validation Rules 126](#_Toc75262114)

[Setting Different Password Policies for Bank and Company Users 133](#_Toc75262115)

[Reactivating Locked Out Users 158](#_Toc75262116)

[Manual Reactivation 158](#_Toc75262117)

[Auto-Reactivation 162](#_Toc75262118)

[Logon and Password Issues 167](#_Toc75262119)

[Logon Information Issues 167](#_Toc75262120)

[User Profile Status Issues 168](#_Toc75262121)

[User Profile Inactive for too Long 168](#_Toc75262122)

[User Profile Not Yet Effective 168](#_Toc75262123)

[User Profile Already Expired 168](#_Toc75262124)

[User Profile Already Logged On 169](#_Toc75262125)

[Password Issues 169](#_Toc75262126)

[Incorrect Password 169](#_Toc75262127)

[Forgotten Password 170](#_Toc75262128)

[Expired Password 170](#_Toc75262129)

[Parameter Checklist for Security Vulnerabilities 171](#_Toc75262130)

[JQuery Upgrade 182](#_Toc75262131)

[Integration Issues 194](#_Toc75262132)

[Updating CE Directly with Back Office Data 195](#_Toc75262133)

[Assigning a Company Group to a Company Sent from EE 196](#_Toc75262134)

[MQ Tool 197](#_Toc75262135)

[MQ Channel Authentication Configuration 209](#_Toc75262136)

[Creating the Queue Manager, Queues, and Channels 210](#_Toc75262137)

[Creating a Local Machine User 222](#_Toc75262138)

[Setting the Channel Authentication Records 224](#_Toc75262139)

[Granting the Local Machine User Access to the Queue 233](#_Toc75262140)

[Configuring the Queues in the CE Utility 237](#_Toc75262141)

[Database Issues 239](#_Toc75262142)

[Copying Tables in DB2 240](#_Toc75262143)

[Assigning the Database Users to the SQL Server Logins 241](#_Toc75262144)

[Importing Existing Tables from the Database 242](#_Toc75262145)

[Manually Stop/Start the Oracle Database 243](#_Toc75262146)

[Backing up the Oracle Database 247](#_Toc75262147)

[Uninstalling the Oracle Database 251](#_Toc75262148)

[Common Oracle Errors 265](#_Toc75262149)

[Database Tools 274](#_Toc75262150)

[Using QTODBC 274](#_Toc75262151)

[Using Squirrel SQL 283](#_Toc75262152)

[Error on Data Retrieval 293](#_Toc75262153)

[Report Issues 294](#_Toc75262154)

[Saving Static Reports 295](#_Toc75262155)

[Special Features 297](#_Toc75262156)

[SMS Notification Feature 298](#_Toc75262157)

[Additional Information 299](#_Toc75262158)

[CE Version 300](#_Toc75262159)

[Manually Maintained Parameters 302](#_Toc75262160)

[Project-Controlled Parameters 303](#_Toc75262161)

[Frequently Used SQL Statements 304](#_Toc75262162)

[Mapping the Parameter and Web Server Drives 308](#_Toc75262163)

[Server-Side Mapping 310](#_Toc75262164)

[Client-Side Mapping 311](#_Toc75262165)

[CE Log Files 315](#_Toc75262166)

[SQL Parser 316](#_Toc75262167)

[Microsoft® Script Editor 318](#_Toc75262168)

[Ultra Edit Text Editor 323](#_Toc75262169)

[Macromedia® Dreamweaver® 330](#_Toc75262170)

[Veracode Scanning 335](#_Toc75262171)

[The CE New Utility 336](#_Toc75262172)

[Glossary 337](#_Toc75262173)

[Glossary 338](#_Toc75262174)

CE Documentation Library

CE Documentation Library

CE Documentation Library

The CE Documentation Library lists all available manuals that serve as references on the use of the Customer Enterprise system. The documents are categorized into three groups: Core System Manuals, CE Utility Reference Manuals, and Installation Guides.

Core System Manuals

TheCE core system manuals provide information on the setup and configuration of various CE parameters, as well as the implementation of supplementary functionalities supported by the system.

Archiving and Recovery

This manual is a reference to the Archive and Recovery functionality of the CE system. Discussions include the required parameter settings for configuring the relevant functions, as well as examples of how this functionality is used in transaction processes.

Building a Product

This manual is a reference to the process of building a product in CE. It provides step-by-step procedures on how to create a basic CE module, transaction function, and product setting; configure parameters; and attach components to generate a working business product.

Data Objects

This manual serves as a reference for creating and utilizing data objects in the CE system. The discussions provide information on how to build data object templates and data object entities, and how to attach these to the transaction function screen.

Frequently Asked Questions

This document addresses commonly-asked questions on the Customer Enterprise system. Issues on the browser-side (transaction processing), CE Utility, Security Module, and database, among others, are addressed in the discussions.

Interfacing CE with CS Eximbills

This document discusses the process of interfacing CE with the CS Eximbills (CSX) back office system. Employing the MQ, FTP, and TCP/IP communication protocols, the interface process using the XML format is detailed in this document.

Interfacing CE with Eximbills Enterprise

This document discusses the process of interfacing CE with the Eximbills Enterprise (EE) back office system. Employing the MQ protocol, the interface process for the transfer of data between CE and EE using the XML format is detailed in this document.

Log Settings

This manual provides information on CE logs as well as general instructions on log configuration in CE. This is especially written for the users who are in charge of maintaining the CE system.

Look and Feel

This manual is a reference guide on designing the CE user interface (i.e., the look and feel of the system). Divided into two parts, the manual provides instructions on creating a) the basic L&F style of CE; and b) the CE L&F style that incorporates widgets.

Multi-Entity

This manual is a reference on implementing the Multi-Entity functionality of the CE system. Discussions cover the implementation and application of CE multi-entity.

Multi-Language

This manual discusses the procedures required in setting up the Multi-Language functionality of CE as it is run by Administrator- and Operator-type users. Furthermore, the configuration of certain system and browser elements as well as the setup of related system parameters is explained in step-by-step procedures.

Reports

This manual is a comprehensive reference guide on the requirements and processes involved in building business products and functions that generate online reports and documents.

Security and System Maintenance Functions

This manual is a comprehensive guide on the security and system maintenance of the CE system. As such, it includes detailed instructions for company and company function management, and user and user function management. Maintenance of key functionalities such as authorization rules, reference numbers, and other services is covered as well. In addition, this document discusses the security concepts in CE to assist users assigned with access rights to the CE Security Module.

Standing Data Functions

This document discusses the functions for the CE standing data that are maintained by operators (e.g., parties, clauses). It includes sections for each function, starting with a brief description of the function, followed by the function input when necessary, and the procedure steps.

Supplementary Functions

This manual is a reference for the CE Utility operator user in configuring parameter, JSP, and transaction function settings to define special or supplementary CE functionalities such as uploading images; sending images and forms to the back-office system; and sending notifications via e-mail, SMS, and widgets.

System Administration Functions

This manual is a reference for the default Super Administrator user of the CE Utility in the configuration and maintenance of the CE environment. It discusses in detail user management and parameter management.

System Reference

This document serves as a quick reference to the following elements that are used when configuring specific parameters in the CE Utility: global system parameters, system parameters, components, XML Generator items, server side system methods, system JS methods, and APIs.

CE Utility Reference Manuals

The CE Utility Reference set of manuals is a guide on the use of the Customer Enterprise Utility Workbench, or simply CE Utility. This reference provides information on every function or feature in the CE Utility and includes instructions and step-by-step procedures on how to operate or use the function in relation to operating and maintaining the CE system and processing a business transaction.

A manual is provided for each function group of the CE Utility:

* *CE Utility Reference: User Manager Functions*, for the functions that belong to the User Manage function group of the CE Utility when accessed by an Administrator or Operator user
* *CE Utility Reference: Parameter Manager Functions*, for the functions that belong to the Parameter Manage function group of the CE Utility when accessed by an Administrator or Operator user
* *CE Utility Reference: System Functions*, for the functions that belong to the System Function group of the CE Utility.
* *CE Utility Reference: Transaction Functions*, for the functions that belong to the Transaction Function group of the CE Utility.
* *CE Utility Reference: Product Functions*, for the functions that belong to the Product Function group of the CE Utility.
* *CE Utility Reference: Maintenance Functions*, for the functions that belong to the Maintenance function group of the CE Utility.

Installation Guides

TheCE installation guides are references on the installation and setup processes of the CE system on different application servers and databases.

Installation Guide WAS 9 - Oracle 19c

This is a reference for installing the CE system on WebSphere Application Server Version 9.0.5.6, with an Oracle 19c database. This includes detailed instructions on configuring the components that are required to successfully run CE.

Introduction

Chapter One

* SYSTEM OVERVIEW
* MANUAL OVERVIEW

System Overview

Customer Enterprise (CE) is a system that provides customers with access to business transactions with more convenience and faster connection. With the help of internet technology, CE is especially designed to allow customers to do business and participate in the business process without being physically present at the bank. Making use of supplementary functionalities, CE may be interfaced with Eximbills Enterprise and CS Eximbills. The system utilizes three main physical components: the database server, the application server, and the web server.

The Customer Enterprise Utility Workbench, or CE Utility, is the main parameter-setting tool of the CE system. It is used to define parameters and rules that are required to successfully carry out business processes and operations, and system tasks. Parameters instruct the system on how to control, manage, and process the business data that it receives. The parameters are written into the background database or saved in the Application and Web servers as XML documents to be used for system operations.

General issues and information about the Customer Enterprise system and the CE Utility are discussed in this reference.

noteNOTE: Refer to the Customer Enterprise documentations for detailed information on the CE system.

Manual Overview

**Purpose**

This document addresses commonly-asked questions on the Customer Enterprise system. Issues on the browser-side (transaction processing), CE Utility, Security Module, and database, among others, are addressed in the discussions.

**Audience**

This manual is a reference specifically for, but not limited to, the following users:

* Users assigned to build modules or troubleshoot parameter errors
* Users assigned to troubleshoot runtime errors
* Users with sufficient knowledge of installing a CE environment, using the different CE and CE Utility functions, and configuring CE parameters

**Prerequisites**

In depth knowledge of the system, as also presented and detailed in the following manuals, is recommended:

* *CE Core Manuals*
* *CE Utility References*
* *CE Installation Guides*

NOTE: Some features discussed in this manual have been tested and documented based on an older system version. Unless otherwise specified, the overall functionality is the same when recreated in the current version.

Browser-side Issues

Chapter Two

* CREATING THE SUPER OFFICER USER PROFILE
* CECHECKCHARACTERFILTER ERROR
* INBOX OR DETAIL INFO ERROR MESSAGE
* INBOX NOT WORKING PROPERLY
* DOWNLOADING TRANSACTION DATA ON WEBLOGIC
* MODIFIED LBI FILES
* SUPPORTING MULTI-LANGUAGES
* SESSION TIMEOUT AND AUTO SAVE
* SESSION EXTENSION AND RELATED SETTINGS
* CANCEL BUTTON ISSUE
* Uploading Files or Images
* AUTHORIZATION NOT WORKING
* SETTING THE DATE FORMAT
* EC LEDGER ERROR
* DISABLE CACHE
* MANUALLY STOP/START THE WEBSPHERE SERVICE
* BROWSER LOGON INFORMATION
* Chrome Browser Settings
* IE9 Browser SEttings
* Displaying the Last Logon Failure
* Browser navigation limitations
* MULTIPLE AUTHORIZATION FUNCTIONS IN ONE PRODUCT
* DISPLAY ERROR WHEN PRINTING HISTORICAL RECORDS
* DISPLAY ALL RECORDS FOR INQUIRE CUBK
* DISALLOWING THE SYSTEM LOGIN GET METHOD

Creating the Super Officer User Profile

To create the default Super Officer user profile, run this SQL Insert statement:

(Start of code)

INSERT INTO CEUSER.SEC\_USER\_INFO ( SEC\_USER\_INFO.C\_BK\_GROUP\_ID, SEC\_USER\_INFO.C\_UNIT\_CODE, SEC\_USER\_INFO.C\_USER\_ID, SEC\_USER\_INFO.C\_USER\_DESC, SEC\_USER\_INFO.C\_PASSWORD, SEC\_USER\_INFO.I\_USER\_TYPE, SEC\_USER\_INFO.I\_AUTH\_LEVEL, SEC\_USER\_INFO.I\_AUTH\_TYPE, SEC\_USER\_INFO.C\_DFLT\_UNIT\_CODE, SEC\_USER\_INFO.C\_DATE\_FORM, SEC\_USER\_INFO.C\_TMPL\_PROTECT, SEC\_USER\_INFO.C\_LOG\_TYPE, SEC\_USER\_INFO.C\_HAVE\_AUTH, SEC\_USER\_INFO.C\_AUTH\_SET, SEC\_USER\_INFO.C\_LOGIN\_RULE, SEC\_USER\_INFO.C\_LOCKED\_FLAG, SEC\_USER\_INFO.C\_FAP\_SET, SEC\_USER\_INFO.C\_CNTY\_CODE, SEC\_USER\_INFO.C\_LANG\_NAME, SEC\_USER\_INFO.C\_LANG\_VAL, SEC\_USER\_INFO.C\_LOCKED\_OP, SEC\_USER\_INFO.T\_LOCKED\_TIME, SEC\_USER\_INFO.EFFECTIVE\_DATE, SEC\_USER\_INFO.EXPIRY\_DATE, SEC\_USER\_INFO.I\_EXPIRY\_TYPE, SEC\_USER\_INFO.I\_MULTI\_LOGIN, SEC\_USER\_INFO.C\_ADD\_FLAG, SEC\_USER\_INFO.C\_TRX\_STATUS, SEC\_USER\_INFO.C\_CREATED\_BU, SEC\_USER\_INFO.C\_CREATED\_BY, SEC\_USER\_INFO.C\_GRP\_CODE, SEC\_USER\_INFO.C\_CURRENCY, SEC\_USER\_INFO.I\_SEC\_TYPE, SEC\_USER\_INFO.D\_CREATED\_DATE, SEC\_USER\_INFO.C\_CHANGE\_PSW, SEC\_USER\_INFO.I\_LOGON\_RETRIES, SEC\_USER\_INFO.C\_INACTIVE, SEC\_USER\_INFO.C\_FIRST\_PWD, SEC\_USER\_INFO.C\_LAST\_PWD, SEC\_USER\_INFO.C\_USER\_MAIL )

VALUES ( 'CSBANK', 'CSOFFICE', 'superofficer', 'The Super Officer', '11111111', 0, 1, 0, NULL, 'yyyy-mm-dd', NULL, NULL, 'F', NULL, NULL, 'F', 'T', 'US', NULL, NULL, 'superofficer', TO\_DATE('2005/12/08 12:51:46', 'YYYY/MM/DD HH24:MI:SS'), NULL, NULL, 1, 0, NULL, NULL, NULL, NULL, 'CSOFFICE', NULL, NULL, TO\_DATE('2007/03/09 00:00:00', 'YYYY/MM/DD HH24:MI:SS'), 'F', 0, NULL, NULL, NULL, NULL )

(End of code)

CECheckCharacterFilterError

Issue

On a new environment, this error occurs when loading a transaction screen: ‘Error 500: Filter [CECheckCharacterFilter]: Filter is unavailable’.

Solution

Add the following parameter in the [CE Parameter Folder]\CE\_SYS folder.

* characterfilter.xml

Inbox or Detail Info Error Message

Issue

The system cannot display the inbox, or the ‘Detail Info’ error message is shown.

|  |
| --- |
|  |

Figure 2. Loading the Inbox

Solution

Start the database first and then the CE application.

Inbox Not Working Properly

Issue

The Inbox does not work properly: the statistics or figures are incorrect

Solution

Check the parameter settings in the CE Utility and make sure that the Key FAP has been properly set. Additionally, the Key FAP selected must be the Item assigned to the relevant product function and not the Item Type.

**EXAMPLE:**

The Key FAP value for Rejected (an Inbox item) is Modify.

|  |
| --- |
|  |

Modify is also the Item assigned to the related product function, ModifyTrx (as opposed to the assigned Item Type – Edit).

|  |
| --- |
|  |

Downloading Transaction Data on Weblogic

Issue

When run on WebLogic, transaction data are not properly downloaded unless the screen is refreshed.

Solution

Add this tag in the transaction JSP file:

<%@ page contentType="text/html; CHARSET=UTF-8" %>

Modified LBI Files

Issue

The transaction screen does not reflect the changes made on an LBI file that is attached to the screen or JSP file.

Solution

If an LBI file is changed and this is attached to the transaction JSP file through the ‘include’ command, save the JSP file again.

**EXAMPLE:**

Refer to the following figure.

The SecuOperator.lbi file is attached to the SecuAdmin.jsp file as shown in this code:

<%@ include file=”SecuOperator.lbi”%>

If the SecuOperator.lbi file is modified, make sure to save the SecuAdmin.jsp file again.

|  |
| --- |
| ° |

Figure 2. Attached LBI File

Supporting Multi-Languages

For the system to support different languages in the system or browser (i.e., special characters are to be accepted), open the WebSphere – Administrative Console and access the application server (e.g., Server1). In the Process Definition – Configuration screen, click on the Java Virtual Machine property. Click on the Custom Properties option and create the following setting:

|  |  |  |
| --- | --- | --- |
| Field |  | Description |

|  |  |  |
| --- | --- | --- |
| **Name** |  | Specify this value: client.encoding.override |
|  |  |  |
| **Value** |  | Specify this value: UTF-8 |
|  |  |  |
| **Description** |  | Specify the language to be supported in this field (e.g., Chinese). |

|  |
| --- |
|  |

Figure 2. Example: Supporting the Chinese Language

Session Timeout and Auto Save

The timeout and auto save functionality for CE transactions are set in CE Utility.

The session timeout is defined in the SESSION\_TIMEOUT system parameter (under the Web Server parameter type). The unit is in minutes, and the value must be greater than 1, as the system prompts a warning message one minute before it has to time out

|  |
| --- |
|  |

Figure 2. SESSION\_TIMEOUT System Parameter

The generated XML file for this system parameter is in:

[Parameter Path]\CSBANK\US\WEB\SYST\sysparam.xml

The Value setting in the SESSION\_TIMEOUT node must be in capitals, ‘VALUE’; otherwise, the system cannot recognize the defined value and defaults it to 30.

Session Extension and Related Settings

During transaction processing, a warning message is shown when a session is about to expire. Through this message, a user may opt to extend the session, otherwise, due to inactivity, the transaction is automatically saved and the user automatically logged off the CE system when the session expires. This warning message may be shown either through a popup window or through a float layer.

Originally, a popup window is displayed to enable users to extend the session. However, this popup window may only be made available in transaction modules. In addition, some browser types block this popup window causing the user to be unaware of session expiry and thus encountering an exception during processing.

In order to fully utilize the extend session feature in transaction, security, standing and system modules, the extend session warning message must be shown through the use of float layer.

NOTE:

When using the extend session feature:

1. In the security, standing, and system modules, transactions are not automatically saved once a session expires.
2. In transaction modules, transactions processed in pending functions are automatically saved once a session expires.

Configuration

To enable the use of float layer for extend session warning messages, the following configurations must be performed:

1. Modify the web.xml file. This file is stored on the following path:

[CEWeb.war folder]\WEB-INF

* Add a new mapping for this filter: CEVerifySessionFilter

|  |
| --- |
| <filter-mapping id="FilterMapping\_1138957604797">  <filter-name>CEVerifySessionFilter</filter-name>  <servlet-name>WSSTrxManager</servlet-name>  </filter-mapping>  <filter-mapping>  <filter-name>CEVerifySessionFilter</filter-name>  <servlet-name>WSGetDataSSScript</servlet-name>  </filter-mapping> |

* Add the WSSessionTimeOutManage servlet and define its mapping.

|  |
| --- |
| <servlet>  <servlet-name>WSSessionTimeOutManage</servlet-name>  <display-name>WSSessionTimeOutManage</display-name>  <description></description>  <servlet-class>com.cs.eximweb.servlets.WSSessionTimeOutManage</servlet-class>  </servlet> |

|  |
| --- |
| <servlet-mapping>  <servlet-name>WSSessionTimeOutManage</servlet-name>  <url-pattern>/servlets/WSSessionTimeOutManage</url-pattern>  </servlet-mapping> |

1. Log on to the CE Utility as a super administrator user and set the value of the EXTEND\_SESSION\_LAYER security system parameter to TRUE.

This parameter is used to control how the extend session notification is to be displayed: through a popup window or a float layer. It may have any of the following values:

* TRUE, in which a float layer is shown as a notification for extending the session. This is made available in transaction, security, standing, and system modules.
* FALSE, in which the standard logic is applied and a popup window is displayed as a notification for extending the session. This is made available in pending transactions only. This is the default value.

|  |
| --- |
|  |

Log on to the CE Utility as an administrator or operator user and generate the xml files for the Security System Parameter. The sys\_para.xml file is generated on the following path:

[Parameter folder]\CE\_SYS\SYST

|  |
| --- |
|  |

1. Log on the CE Utility as an administrator or operator user. In the System Parameter function, make sure that the value of the SESSION\_TIMEOUT Web Server system parameter is less than or equal to the session timeout value of the application server. The value of this parameter is in minutes, and its minimum value is 2.

|  |
| --- |
|  |

To check the session timeout configuration of the application in WebSphere, run the WebSphere Administrative Console and access the following path:

Servers > Server Type > WebSphere Application Servers > [server] > Container Settings Section: Session Management

|  |
| --- |
|  |

Generate the xml file for the Meta System Parameter. The sysparam.xml file is generated on the following path:

[Parameter folder]\[Bank]\[Country]\WEB\SYST

|  |
| --- |
|  |

1. Add the SYS\_ExtendSession.js file in the following path:

[CEWeb.war folder]\JS

File Content:

|  |
| --- |
| /\*\*  \* ADD DORA FOR CEV2-05-12-10  \* Support extend session in security, and change the popup window to div.  \*/  var sessionLife = ""; //session life cycle-1min  var sessionCouter = null; //session counter  var CountdownNum = 30; //count number on layer  var subTimer = null; //session count down  //APPEND EXTEND SESSION LISTENER  function sessionListener(){  var actionUrl = "../servlets/WSTrxManager?\_TRX\_STATUS=SESSION\_LIFE\_CYCLE&MOTION\_TYPE=INIT";  actionUrl = encodeUrl(actionUrl);  sendRequestByAjaxPost(actionUrl, true,iTimeBack);  }  function iTimeBack(xmlhttp){  var iTime = xmlhttp.responseText;  sessionLife = iTime;  if(sessionLife){  sessionCouter = setTimeout("SYS\_DrewExtendLayer(" + sessionLife + ")", sessionLife);  }  }  function SYS\_DrewExtendLayer(sessionTime){  drewExtendSessionLayer();  drewCover();  //SET COUNTDOWN  couterCal();  }  //drew cover layer  function drewCover(){  var oCoverDiv = document.createElement("div");  oCoverDiv.className = "CoverDiv";  oCoverDiv.id = "CoverLayer";  var sStyle = "height:" + getDocHeight() + "px";  oCoverDiv.setAttribute("style",sStyle);  document.body.appendChild(oCoverDiv);  }  //drea extend session layer  function drewExtendSessionLayer(){  rmErrMsg();  //FLOAT LAYER  var floatLayerDIV = document.createElement("div");  floatLayerDIV.id = "floatLayerDIV";  floatLayerDIV.className = "floatLayerDIV";    //HEADER  var layerHeader = document.createElement("div");  layerHeader.id = "layerHeader";  layerHeader.className = "layerHeader";  var headerH3 = document.createElement("h3");  headerH3.innerHTML = "Warning";  layerHeader.appendChild(headerH3);  floatLayerDIV.appendChild(layerHeader);    //HR  var hr = document.createElement("hr");  floatLayerDIV.appendChild(hr);    //MIAN  var layerMain = document.createElement("div");  layerMain.id = "layerMain";  layerMain.className = "layerMain";  if(typeof(SYS\_FUNCTION\_TYPE)!="undefined" && (SYS\_FUNCTION\_TYPE == "MM" || SYS\_FUNCTION\_TYPE == "PM" || SYS\_FUNCTION\_TYPE == "EC" )){  layerMain.innerHTML = "Your session is about to time out. Your transaction details will be saved and you will be logged out in ";  }else{  layerMain.innerHTML = "Your session is about to time out. You will be logged out in ";  }  var countSpan = document.createElement("span");  countSpan.id = "countNum";  countSpan.innerHTML = CountdownNum ;  layerMain.appendChild(countSpan);  var secondSpan = document.createElement("span");  secondSpan.innerHTML = " seconds.";  layerMain.appendChild(secondSpan);    var br1 = document.createElement("br");  layerMain.appendChild(br1);  var br2 = document.createElement("br");  layerMain.appendChild(br2);    var descSpan = document.createElement("span");  descSpan.innerHTML = " To extend your session, please click Extend. ";  layerMain.appendChild(descSpan);  floatLayerDIV.appendChild(layerMain);    //FOOT  var layerFoot = document.createElement("div");  layerFoot.id = "layerFoot";  layerFoot.className = "layerFoot";  // CONTINUE BUTTON  var continueBTN = document.createElement("input");  continueBTN.type = "button";  continueBTN.value = "Extend";  attachObjEvent(continueBTN, "onclick", eval("toExtendSession"));  continueBTN.className = "prim";  layerFoot.appendChild(continueBTN);  floatLayerDIV.appendChild(layerFoot);  document.body.appendChild(floatLayerDIV);  }  //calculate countdown  function couterCal(){  subTimer = setInterval("subTimes($('#countNum').html())", 1000);  }  //decrease count number  function subTimes(curSecond){  if( parseInt(curSecond) > 0 ){  var tempSecond = parseInt(curSecond) - 1;  $("#countNum").html(tempSecond);  }else{  clearInterval(subTimer);  removeLayer();  if(typeof(SYS\_FUNCTION\_TYPE)!="undefined" && (SYS\_FUNCTION\_TYPE == "MM" || SYS\_FUNCTION\_TYPE == "PM" || SYS\_FUNCTION\_TYPE == "EC" )){  \_save\_onclick(true);  }else{  SYS\_CE\_LOGOUT(false);  }  }  }  //RESERT SESSION COUNTER WHEN CONTINUE AND SEND OTHER REQUEST  function restartSessionCounter(){  removeLayer();  if (sessionCouter) {  clearTimeout(sessionCouter);  }  if (sessionLife) {  sessionCouter = setTimeout("SYS\_DrewExtendLayer(" + sessionLife + ")", sessionLife);  }  }  function resetSubTimer(){  if (subTimer) {  clearTimeout(subTimer);  }  if($('#countNum').html()){  subTimer = setInterval("subTimes($('#countNum').html())", 1000);  }  }  //remove extend session float layer  function removeLayer(){  rmNode("CoverLayer");  rmNode("floatLayerDIV");  }  //when click CONTINUE button to extend session  function continueSession(){  clearInterval(subTimer);  restartSessionCounter();  }  function toExtendSession(){  var actionUrl = "../servlets/WSTrxManager?\_TRX\_STATUS=SESSION\_LIFE\_CYCLE&MOTION\_TYPE=EXTEND";  actionUrl = encodeUrl(actionUrl);  sendRequestByAjaxPost(actionUrl, true);  }  function getDocHeight() {  var D = document;  return Math.max(  Math.max(D.body.scrollHeight, D.documentElement.scrollHeight),  Math.max(D.body.offsetHeight, D.documentElement.offsetHeight),  Math.max(D.body.clientHeight, D.documentElement.clientHeight));  }  //END DORA FOR CEV2-05-12-10 |

1. Modify the following JSP and CSS files.

* File: common.jsp

Path: [CEWeb.war folder] \include

|  |
| --- |
| <!-- Project wide JavaScript function -->  <EXIMTAGS:IncludeFile filePath="../JS/Calendar.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/Validation.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/BaseFunc.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/MainMenu.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/Buttons.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/SYS\_ExtendSession.js"/>  <!-- End: include/common.jsp --> |

* File: common.jsp

Path: [CEWeb.war folder] \[Bank]\[Country]\WEB\SCRN\Library

|  |
| --- |
| <!-- Project wide JavaScript function -->  <EXIMTAGS:IncludeFile filePath="../JS/Calendar.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/Validation.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/BaseFunc.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/MainMenu.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/HelpTips.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/Buttons.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/DocumentList.js"/>  <EXIMTAGS:IncludeFile filePath="../JS/SYS\_ExtendSession.js"/>  <!-- End: Library/common.jsp --> |

* File: Secu\_Template.jsp

Path: [CEWeb.war folder] \Templates

|  |
| --- |
| function templateOnInit(){  OnInit();  showMainMenu();  getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  if(this.window == getTopWindow()){  sessionListener();  }  } |

* File: STD\_Template.jsp

Path: [CEWeb.war folder] \Templates

|  |
| --- |
| function templateOnInit(){  \_OnInit();  if (window.showMainMenu&&("A" == LOGIN\_NAME || !window.widgetPopup)){  showMainMenu();  }  getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  if(this.window == getTopWindow()){  sessionListener();  }  } |

* File: Trx\_Template.jsp

Path: [CEWeb.war folder] \Templates

|  |
| --- |
| function templateOnInit(){  \_OnInit();  if window.showMainMenu&&!isDO) {  showMainMenu();  }  getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  $.CSWidget.includePageStyles();  if(this.window == getTopWindow()){  sessionListener();  }  } |

* File: SYS\_Template.jsp

Path: [CEWeb.war folder] \Templates

|  |
| --- |
| function templateOnInit(){  OnInit();  if (window.showMainMenu&&("A" == LOGIN\_NAME || !window.widgetPopup)){  showMainMenu();  }  getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  if(this.window == getTopWindow()){  sessionListener();  }  } |

* File: WGT\_Template.jsp

Path: [CEWeb.war folder] \Templates

|  |
| --- |
| <script language="javascript" type="text/javascript">  if("T" == "<CETAGS:SysField fldName="gpm.ENABLED\_CLIENT\_COOKIE"/>"){  window.jSessionId = "<%= escapedSessionId %>";  }  window.widgetPopup = <%= actPopup%>;  window.parallelProcess = <%=parallel %>;  if(window.widgetPopup){  if(window.parallelProcess){  window.trxSessionId = "<CETAGS:SysField fldName='reqA.TrxSessionId' defaultVal=''/>";  }  }  window.refreshToken = <%=longToken%>;//CEV2-01-19-02(CEV2-0784) Cash 2014-04-15  function widgetOnInit(){  if(this.window == getTopWindow()){  sessionListener();  }  }  </script>  </head>  <body onLoad="widgetOnInit()"> |

* File: Standard.css

Path: [CEWeb.war folder] \theme

|  |
| --- |
| /\* ADD DORA FOR CEV2-05-12-10 \*/  .floatLayerDIV{  width:550px;  height:230px;  box-shadow:#262525 0px 0px 20px;  border-radius: 12px;  z-index:111000;  position:absolute;  top:30%;  left:30%;  background:#FFF;  padding:5px;  }  #CoverLayer{  opacity: 0.3;  z-index: 110000;  }  .layerHeader{  width: 90%;  height: 16%;  }  .layerHeader h3{  float: left;  font-size: 14px;  color: #2b3772;  margin: 15px 0 0 10px;  display: inline;  }  .floatLayerDIV hr{  height: 1px;  border : none;  border-top : 1px dashed #aba7a7;  }  #countNum{  color:#0064B2;  font-weight:bold;  }  .layerMain{  padding-top: 10px;  padding-left: 8px;  height: 45%;  font-size: 13px;  }  .layerFoot{  height: 20%;  text-align: center;  cursor: pointer;  }  /\* END DORA FOR CEV2-05-12-10 \*/ |

Transaction Processing

|  |
| --- |
| Example |

|  |  |  |
| --- | --- | --- |
| 1. Process a transaction in the Apply for Import LC function and leave it idle. |  |  |
|  |  |  |
| 2. Before the session expires, a warning message is shown through a float layer.  To extend the session, click on the Extend button before the countdown (in seconds) ends.  In this example, leave the session idle until the countdown ends.  **NOTE:**  i. This feature is also applied in security, standing, and system modules.  ii. A session expires based on the number of minutes indicated in the SESSION\_TIMEOUT web server system parameter. |  |  |
|  |  |  |
| The user is automatically logged off the CE system.  Re-log on the CE system using the relevant credentials. |  |  |
|  |  |  |
| 3. Run the Modify Transaction function. |  |  |
|  |  |  |
| 4. The transaction being processed before the session expired was automatically saved in this function.  **NOTE:**  This is only applicable to transaction modules. Transactions processed in security, standing, and system modules are not automatically saved when a session expires. |  |  |
|  |  |
|  | C:\Users\Joahna\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Image39.jpg |

Cancel Button Issue

Issue

When the Cancel button is clicked, the system stays in the current page.

Solution

Currently, CE only supports the following functions for canceling a transaction.

|  |  |  |
| --- | --- | --- |
| Function |  | Action |

|  |  |  |
| --- | --- | --- |
| **\_Cancel\_ToMainMenu()** |  | The system returns to the main menu. |
|  |  |  |
| **\_Cancel\_ToPrevious()** |  | The system returns to the previous screen. |

Uploading Files or Images

Files or images can be attached, edited, deleted, and viewed when running a transaction.

NOTE: Refer to the *CE Supplementary Functions* manual for additional details on images.

Parameter Settings

The supported file types, maximum file size, and the path where uploaded files are to be stored can be defined in the Image Control function of the Transaction Function group in the CE Utility.

|  |
| --- |
|  |

Figure 2. Image Control Function

On the other hand, the total size of all uploaded images can be controlled through the ALL\_IMAGE\_LENGTH parameter. This setting is defined in the System Parameter function of the Parameter Manage function group.

|  |
| --- |
|  |

Figure 2. ALL\_IMAGE\_LENGTH System Parameter

Transaction Processing

During transaction processing, clicking the Upload Image button on the transaction function screen opens the Image(s) Information window. Images and other files can be uploaded and managed in this window.

|  |
| --- |
|  |

Figure 2. Uploading an Image

A warning message is displayed when the file being uploaded has an unsupported file type. The upload process is then terminated.

|  |
| --- |
|  |

Figure 2. Invalid File Type Warning Message

If the size of the file being uploaded exceeds the maximum size allowed, the upload is stopped and a warning message is displayed as well.

|  |
| --- |
|  |

Figure 2. Invalid File Size Warning Message

Authorization Not Working

Issue

The authorization rule is not followed.

A function, with an AP type and with the Release flag unchecked, is run and the transaction is confirmed. However, the transaction is not queued to the authorization function.

Solution

Note the following points when setting an authorization rule:

* An Amount field in the screen must be linked to this authorization rule (through the Product Authorize function in CE Utility);
* The value of this amount field in the transaction screen must not be null and must be greater than zero.
* The Amount/Rate format for the related pending function has to be set.

Setting the Date Format

When a transaction is run, the process flows from the application server (AP), to the web server (Web), and back to the application server.

When retrieving data from AP to display it on Web (AP to Web), CE converts the date format from the system format (YYYY-MM-DD) to the user-defined one using the SYS\_CONVERT\_DISPLAY\_DATE() system JS function. The SYS\_DateFormat.js file is the system level js function that is used to format date value.

When the transaction is submitted (Web to AP), CE converts the user-defined format to the system format using the SYS\_CONVERT\_SYSTEM\_DATE() system JS function.

The following conditions must be satisfied in order to set up a date format:

* The two JS functions can convert the format automatically as long as the CLASS attribute of a field begins with a ‘DATE\_’.
* Before a date-type field is converted, it must first have the proper date format: if the date value is to be changed from the system format to the user-defined one, it must first be in system format; conversely, a date value must first be in the user-defined format if it is to be changed to the system format.

noteNOTE: The Date Format setting (i.e., the C\_DATE\_FORM field), is set in the company group level (using the Add Company Group function in the Security Module).

EC Ledger Error

Issue

The ‘EC ledger’ error occurs when a transaction for a Trx Manager EC-type function is confirmed.

|  |
| --- |
|  |

Figure 2. EC Ledger Error Message

Solution

For functions assigned with the Trx Manager EC main program (e.g., Modify LC), it is not required to set any function attribute. The system retrieves the attributes of the preceding or original function of the selected record via function Id (e.g., Apply for LC).

Note the value of the EVENT parameter (e.g., <EVENT>MASTER</EVENT>); this is used to retrieve the table name and to get value of I\_EVENT\_TIMES. If the EVENT parameter is incorrect, the value of I\_EVENT\_TIMES becomes null and, consequently, the EC Ledger error occurs.

To correct this error, check the Attribute component of the original function and make sure that the Event Name property is set to MASTER.

|  |
| --- |
|  |

Figure 2. Function Attribute – Event Name

Disable Cache

Cache Manage can be disabled in CE, open the WebSphere – Administrative Console and access the application server (e.g., Server1). In the Process Definition – Configuration screen, click on the Java Virtual Machine property. Click on the Custom Properties option and create the following setting:

|  |  |  |
| --- | --- | --- |
| Field |  | Description |

|  |  |  |
| --- | --- | --- |
| **Name** |  | Specify this value: ce.cache.disabled |
|  |  |  |
| **Value** |  | Specify this value: true |

|  |
| --- |
|  |

Figure 2. Example: Disable Cache

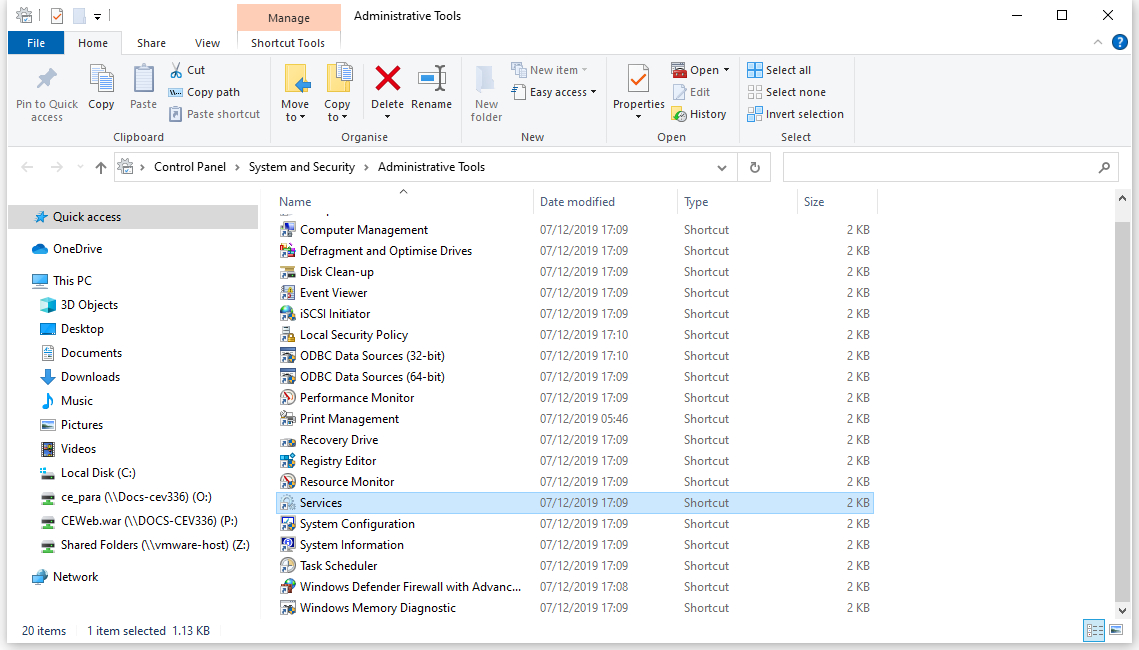
Manually Stop/Start the WebSphere Service

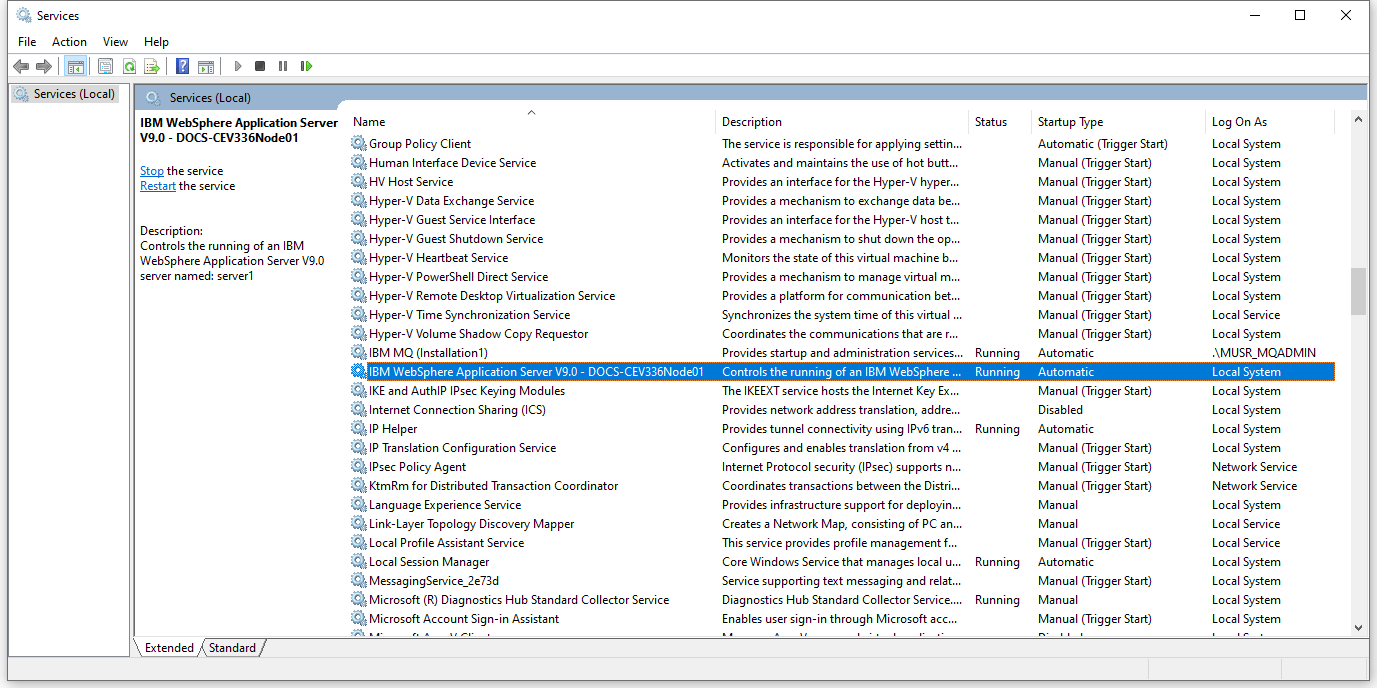
The following instances require the manual running of the stop/start process for the WAS service.

* After installing the CE.ear file;
* After modifying the parameters of CE in the WAS console;
* After modifying the setting of JDBC provider and data source; and
* After replacing the JAR files and the class files of CE.

There are three ways to run the stop/start process for the WebSphere service:

1. In the Windows Control Panel, run the Services option from the Administrative Tools folder. In the Services window, select the IBM WebSphere application and click on the stop, start or restart options for the service.





1. Access the IBM WebSphere Application Server menu from the Windows Programs group. Click on the Stop/Start the Server option from the relevant profile sub-menu of the Profile folder.

|  |
| --- |
|  |

1. In the Enterprise Applications page of the WebSphere Administrative console, select the CE program and click on the stop/start service option.

|  |
| --- |
|  |

Browser Logon Information

In the Browser Logon window, the Company Code field is the value of the Company Group code. If a customer has no specified company group, the system designates the company code value as the company group value.

|  |
| --- |
|  |

Figure 2. Browser Logon Window

Encrypting User Logon Information

|  |
| --- |
|  |

Figure 2. Encrypted User Logon Information

In order to avoid showing sensitive information in clear text, the user’s company code, logon Id, and password can be encrypted upon logging on to the system. Enabling this feature includes the following steps:

* Adding the DECRYPT\_FIELDS system parameter;
* Modifying the LegalRequest.properties file;
* Adding new servlets to the web.xml file; and
* Modifying JS files.

Add the DECRYPT\_FIELDS System Parameter

In the System Parameters function of the CE Utility, add the DECRYPT\_FIELDS system parameter. This is used to define the fields to be encrypted when logging on to CE.

|  |  |  |
| --- | --- | --- |
| Field |  | Value |

|  |  |  |
| --- | --- | --- |
| **Parameter Name** |  | DECRYPT\_FIELDS |
|  |  |  |
| **Parameter Attribute** |  | Common |
|  |  |  |
| **Parameter Value** |  | Field Ids of the fields to be encrypted (separated by semicolons)  **EXAMPLE:**  C\_PASSWORD;C\_USER\_ID;C\_BUSINESS\_UNIT |
|  |  |  |
| **CDATA** |  | Unmarked |
|  |  |  |
| **Parameter Desc** |  | Decrypt Fields |

Run the XML Generator function and generate the XML files for the System Parameter attribute.

Modify the LegalRequest.properties File

Manually modify the LegalRequest.properties file located in the CE\_SYS folder of the parameter drive. Add the following paths:

* /CEWeb/JS/rsaencrypt/Publickey.js
* /CEWeb/JS/rsaencrypt/jsencrypt.min.js
* /CEWeb/SYS\_JS/jQuery/jquery.min.js

|  |
| --- |
|  |

Figure 2. LegalRequest.properties File

Add New Servlets

Manually modify the web.xml file located in the WEB-INF folder of the CEWeb.war directory. Add the following codes in red to define the new servlets:

**Code 1:**

(Start code)

<!-- CEV2-11-01-28 bruce 2014-09-22 S -->

<filter>

<filter-name>CECharsetFilter</filter-name>

<display-name>CECharsetFilter</display-name>

<filter-class>com.cs.eximweb.servlets.CECharsetFilter</filter-class>

</filter>

<!-- CEV2-11-01-28 bruce 2014-09-22 E -->

<!-- CEV2-14-15-04 drew 2018-03-22 S -->

<filter>

<filter-name>CEDecryptFilter</filter-name>

<display-name>CEDecryptFilter</display-name>

<filter-class>com.cs.ceweb.servlets.CEDecryptFilter</filter-class>

</filter>

<!-- CEV2-14-15-04 drew 2018-03-22 E -->

(End code)

**Code 2:**

(Start code)

<!-- CEV2-11-01-28 bruce 2014-09-22 S -->

<filter-mapping>

<filter-name>CECharsetFilter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

<!-- CEV2-11-01-28 bruce 2014-09-22 E -->

<!-- CEV2-14-15-04 drew 2018-03-22 S -->

<filter-mapping>

<filter-name>CEDecryptFilter</filter-name>

<url-pattern>/screen/SECindex.htm</url-pattern>

</filter-mapping>

<!-- CEV2-14-15-04 drew 2018-03-22 E -->

(End code)

Modify JS Files

Make sure that the following JS files are in the [CEWeb.war folder]:\JS\rsaencrypt folder:

* Publickey.js
* jsencrypt.min.js

NOTE: These files are bundled with the CE EAR file.

Modify the SYS\_index.htm file in CEWeb.war drive by adding the following codes in red:

**Code 1:**

(Start code)

<script language="javascript" type="text/javascript" src="SYS\_JS/jQuery/jquery.min.js"></script>

<script language="javascript" type="text/javascript" src="JS/rsaencrypt/jsencrypt.min.js"></script>

<script language="javascript" type="text/javascript" src="JS/rsaencrypt/Publickey.js"></script>

<script language="javascript" type="text/javascript">

(End code)

**Code 2:**

(Start code)

function Login()

{

var C\_BUSINESS\_UNIT = document.LOGIN.C\_BUSINESS\_UNIT.value;

var C\_USER\_ID = document.LOGIN.C\_USER\_ID.value;

var C\_PASSWORD = document.LOGIN.C\_PASSWORD.value;

var sErrMsg = "";

document.LOGIN.C\_BUSINESS\_UNIT.value = rsaEncript(C\_BUSINESS\_UNIT,publicKey);

document.LOGIN.C\_USER\_ID.value = rsaEncript(C\_USER\_ID,publicKey);

document.LOGIN.C\_PASSWORD.value = rsaEncript(C\_PASSWORD,publicKey);

if (C\_PASSWORD.length == 0) {

sErrMsg = document.LOGIN.C\_PASSWORD.title + " is empty.";

}

if (C\_USER\_ID.length == 0) {

sErrMsg = document.LOGIN.C\_USER\_ID.title + " is empty.";

}

if (C\_BUSINESS\_UNIT.length == 0) {

sErrMsg = document.LOGIN.C\_BUSINESS\_UNIT.title + " is empty.";

}

if (sErrMsg.length > 0) {

alert(sErrMsg);

} else {

var act = document.LOGIN.action;

document.LOGIN.action = getAbsoluteURL("screen/SECindex.htm");

document.LOGIN.submit();

}

}

(End code)

**Code 3:**

(Start code)

function rsaEncript(content,publicKey){

var encrypt = new JSEncrypt();

encrypt.setPublicKey(publicKey);

var encrypted = encrypt.encrypt(content);

return encrypted;

}

(End code)

Chrome Browser Settings

Issue

A blank page is shown when executing View History and Compare Event in catalog using the Chrome Browser. This is caused by the following code:

var trxWin = openWin("", "trxview", wStyle);

Solution

Include the HTM file Sys\_Dummy.htm for the URL in the relevant JS code.

**EXAMPLE:**

openWin("Sys\_Dummy.htm ", "trxview", wStyle);

IE9 Browser Settings

Issue

A JS error messages is displayed when an input is made in any field when IE9 browser is used.

|  |
| --- |
|  |

Figure 2. Example: JS Error Message

Solution

1. Include the <EXIMTAGS:IncludeFile filePath="../SYS\_JS/SYS\_CrossBrowser.js"/> in the common.jsp file found in following paths:

* [Parameter Folder]\include
* [Parameter Folder]\[Bank]\[Country]\WEB\SCRN\[Library]
  1. Remove the following codes in SYS\_Crossbrowser.js:

//Start: Detect browser type

var isMozilla = (typeof(document.implementation) != 'undefined') && (typeof(document.implementation.createDocument) != 'undefined');

var isMSIE = (typeof(window.ActiveXObject) != 'undefined');

//End: Detect browser type

* 1. Clear the WAS temporary files located in the application server directory.

**EXAMPLE:**

C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\temp\DOC- CEV28Node01\server1\CE

Displaying the Last Logon Failure

Issue

CE displays the last logon failure time in the screen the next successful logon to the system.

Solution

1. Add the following codes to BankBanner.lbi file.

<tr id="LOGFAILTIME">

<td colspan="3" align="left"><span class="logtime" id="LOGFAILTIMEDESC"></span></td>

</tr>

<script language="javascript" type="text/javascript">

var logFailDesc = "<MLG:Desc name='sys.incl.BKNAFAIL.TIME'>Your last failure logon was </MLG:Desc>";

var logFail = "<CETAGS:GETLOGFAILTIME timeFormat="hh:mm" />";

if(logFail!=""){

document.getElementById("LOGFAILTIMEDESC").innerHTML = logFailDesc + logFail;

}else{

document.getElementById("LOGFAILTIME").style.display="none";

}

</ script>

1. Update the Widget parameter with the following codes.

<iw:itemSet id="labels">

<iw:item id="labelWel" alias="" lang="" readOnly="false" value="Welcome">

</iw:item>

<iw:item id="labelLastTime" alias="" lang=""

readOnly="false" value="Your last logon time was">

</iw:item>

<iw:item id="labelFailTime" alias="" lang=""

readOnly="false" value="Your last failure logon time was">

</iw:item>

<iw:item id="labelToday" alias="" lang=""

readOnly="false" value="Your current logon time is">

</iw:item>

</iw:itemSet>

Browser Navigation Limitations

The user’s navigation through the CE browser may be limited depending on the specified business requirements. The one or more of the following restrictions may be applied:

* Disallowing refreshing the page in the main transaction window
* Disallowing the use of multiple windows and tabs
* Allowing refreshing the page in certain popup windows (e.g., Inquire Clause window) only

When the user performs an action prohibited by the set limitations, the system clears the session and logs the user out of CE. An error message is displayed requiring the user to login to the system again.

JSP and JS File Modifications

To apply the aforementioned limitations in browser navigation, the following JSP and JS files in CEWeb.war must be modified.

noteNOTE: The navigation limitations and the code changes to be applied may vary depending on the business requirements.

1. \WEB-INF\web.xml

|  |
| --- |
| Add a new filter. |
| <filter>  <filter-name>CEPreventRefreshFilter</filter-name>  <display-name>CEPreventRefreshFilter</display-name>  <filter-class>com.cs.ceweb.servlets.CEPreventRefreshFilter</filter-class>  </filter> |
| Add the filter mapping under CERequestFilter. |
| <filter-mapping>  <filter-name>CERequestFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping>  <filter-mapping>  <filter-name>CEPreventRefreshFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping> |

1. \include\header.jsp

|  |
| --- |
| Add this statement. |
| <%  String refreshToken = (String)WSSessionManager.getObject(request,"REFRESH\_TOKEN");  long longToken = 0;  if(refreshToken != null){  longToken = Long.parseLong(refreshToken);  }  %> |

1. \screen\CE\_Add\_op\_func.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language="javascript" type="text/javascript">  <!--  window.refreshToken = <%=longToken%>;  function \_submit\_onclick()  {  //CEV2-martin-s  if(parent.document.MAINFORM.FUN\_DESC){  document.IFRAMEMAINFORM.FUN\_DESC.value = parent.document.MAINFORM.FUN\_DESC.value;  }  //CEV2-martin-e  if(getXMLDOM()){  //CEV2-0536 CEV2.1.1 2010-01-05 Frank Modify Start  var actionUrl = document.IFRAMEMAINFORM.action;  actionUrl=encodeUrl(actionUrl);  document.IFRAMEMAINFORM.action = actionUrl;  //Frank Modify End  document.IFRAMEMAINFORM.submit();  }  }  -->  </script> |

1. \screen\CE\_Edit\_op\_func.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language="javascript" type="text/javascript">  <!--  window.refreshToken = <%=longToken%>;  function OnInit()  {  //CEV2-martin-s  if(parent.document.MAINFORM.FUN\_DESC){  document.IFRAMEMAINFORM.FUN\_DESC.value = parent.document.MAINFORM.FUN\_DESC.value;  }  //CEV2-martin-e  initInq();  ClickFirst();  }  ……………………………………………..  -->  </script> |

1. \screen\errorPage.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language = "javascript" src="../SYS\_JS/SYS\_BaseFunc\_CAL.js"></script>  <script language = "javascript" src="../SYS\_JS/SYS\_Common.js"></script> |

1. \screen\SYS\_main\_menu.jsp

|  |
| --- |
| Add the statement in blue. |
| <%  //CEV2-14-04-01 Edward edit begin for Veracode 2012-10-11  String pswExpireNotify = (String)WSSessionManager.getObject(request, "PSW\_EXPIRE\_NOTIFY");  pswExpireNotify = CEEscapeHelper.escapeJavaScript(pswExpireNotify);  //CEV2-14-04-01 Edward edit end for Veracode 2012-10-11  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 S  String globalRefreshToken = (String)WSSessionManager.getObject(request, "GLOBAL\_REFRESH\_TOKEN");  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 E  %>  <script language="javascript" type="text/javascript">  <!--  //CEV2-05-08-01 2011-9-9(CEV2-0548) Louis bound Start  var PSW\_EXPIRE\_NOTIFY = "<%=pswExpireNotify%>";  //CEV2-05-08-01 2011-9-9(CEV2-0548) Louis bound End  var PswNotify = document.getElementById("PswNotify");  if (PSW\_EXPIRE\_NOTIFY != "null"){  PswNotify.innerHTML = PSW\_EXPIRE\_NOTIFY;  document.getElementById("PswNotifyTr").style.display = "";  }  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 S  var globalRefreshToken = <%=globalRefreshToken%>;  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 E  -->  </script> |

1. \screen\SYS\_security\_menu.jsp

|  |
| --- |
| Add the statement in blue. |
| <%  //CEV2-14-04-01 Edward edit begin for Veracode 2012-10-11  String pswExpireNotify = (String)WSSessionManager.getObject(request, "PSW\_EXPIRE\_NOTIFY");  pswExpireNotify = CEEscapeHelper.escapeJavaScript(pswExpireNotify);  //CEV2-14-04-01 Edward edit end for Veracode 2012-10-11  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 S  String globalRefreshToken = (String)WSSessionManager.getObject(request, "GLOBAL\_REFRESH\_TOKEN");  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 E  %>  <script language="javascript" type="text/javascript">  <!--  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 S  var globalRefreshToken = <%=globalRefreshToken%>;  //CEV2-01-19-02(CEV2-0784) john.zhong 2014-02-27 E  function OnInit()  {  //nothing to do.  }  -->  </script> |

1. \Templates\Secu\_Template.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language="javascript" type="text/javascript">  <!--  ………………………………………  window.refreshToken = <%=longToken%>;  if(module!=null && product!="" && module!="null" && product!="null"){  setFuncName();  }  function templateOnInit(){  OnInit();  showMainMenu();  getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  }  -->  </script> |

1. \Templates\STD\_Template.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language="javascript" type="text/javascript">  <!--  ………………………………………….  window.refreshToken = <%=longToken%>;  if(module!=null && product!="" && module!="null" && product!="null"){  setFuncName();  }  function templateOnInit(){  \_OnInit();  if (window.showMainMenu&&("A" == LOGIN\_NAME || !window.widgetPopup)){  showMainMenu();  }    getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  }-->  </script> |

1. \Templates\SYS\_Tempalte.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language="javascript" type="text/javascript">  <!--  ……………………………………………………  window.refreshToken = <%=longToken%>;  if(selFuncDesc==null||selFuncDesc.length<=0){  selFuncDesc = "<%=escapedFuncName%>";  }  if(module!=null && product!="" && module!="null" && product!="null"){  setFuncName();  }  function templateOnInit(){  OnInit();  if (window.showMainMenu&&("A" == LOGIN\_NAME || !window.widgetPopup)){  showMainMenu();  }  getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  }-->  </script> |

1. \Templates\Trx\_Template.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language="javascript" type="text/javascript">  <!—  …………………………………….  window.refreshToken = <%=longToken%>;  if(module!=null && product!="" && module!="null" && product!="null"){  setFuncName();  }  function templateOnInit(){  \_OnInit();  if (window.showMainMenu&&!isDO) {  showMainMenu();  }  getHeader("headerName", selFuncDesc); //set trx name  $.CSPage.initPage();  }-->  </script> |

1. \Templates\WGT\_Template.jsp

|  |
| --- |
| Add the statement in blue. |
| <script language="javascript" type="text/javascript">  if("T" == "<CETAGS:SysField fldName="gpm.ENABLED\_CLIENT\_COOKIE"/>"){  window.jSessionId = "<%= escapedSessionId %>";  }  window.parallelProcess = <%=parallel %>;  if(<%= parallel %>){  window.trxSessionId = "<%= escapedTrxSessionId %>";  }  //CEV2-04-01-09 Cash 2014-2-13 s  window.widgetPopup = <%= actPopup%>;  //CEV2-04-01-09 Cash 2014-2-13 e  window.refreshToken = <%=longToken%>;  </script> |

Parameter Settings

The following parameter settings must also be defined for browser navigations limitations to be applied.

1. [Parameter Drive]\CE\_SYS\refreshSetting.xml

|  |
| --- |
| Add a new parameter file. This file is manually maintained. |
| <root>  <exceptions>  <exception-uri>  <uri></uri>  </exception-uri>  <exception-svt>  <svt></svt>  </exception-svt>  <exception-req>  <params>  <param name="\_TRX\_STATUS">RESEND\_GAPI</param>  </params>  <params>  <param name="\_TRX\_STATUS">INQTRX\_UNITCODE</param>  </params>  </exception-req>  </exceptions>  <no-exceptions>  <no-exception-svt>/screen/SECindex.htm</no-exception-svt>  </no-exceptions>  </root> |

1. CE Utility – Error Handling Function

|  |
| --- |
| Define the new error code 000469 through the Error Handling function of the Maintenance function group in the CE Utility. |
|  |

1. [Parameter Drive]\CE\_SYS\SYST\sys\_external\_session\_obj.xml

|  |
| --- |
| Add the statement in blue. |
| <root>  <fixed>  <INBOX\_DOM/>  <REFRESH\_TOKEN/>  <PSW\_EXPIRE\_NOTIFY/>  <WIDGET\_ATTRS/>  </fixed>  <regexp>  </regexp>  </root> |

Multiple Authorization Functions in One Product

When there are multiple authorization functions that are included in one product, the SYS\_PdtMultiAuthFuncMapping.xml file must be configured to specify which authorization function must be used of each specific product function. This file is located on the following path:

[Parameter Folder]\CE\_SYS

Example:

1. In their corresponding function\_root.xml files, two authorization functions have the following details:

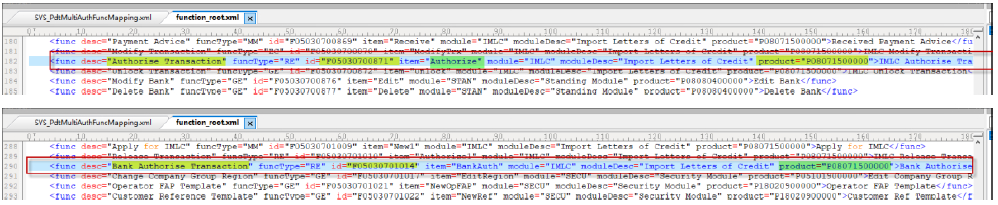
Authorise Transaction

* Function ID – F05030700871
* Product ID – P08071500000

Bank Authorise Transaction

* Function ID – F05030701014
* Product ID – P08071500000

Both functions are included in the same product.



1. In their corresponding function\_root.xml files, three product functions have the following details:

Apply for Import LC

* Function ID – F05030700829
* Product ID – P08071500000

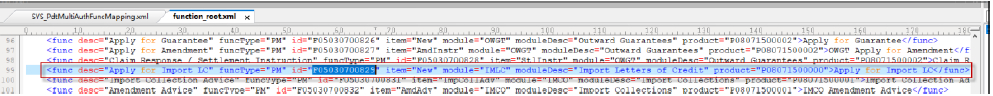
Apply for Amendment

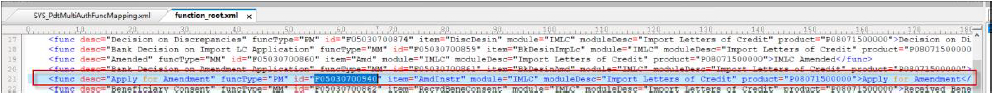
* Function ID – F05030700940
* Product ID – P08071500000

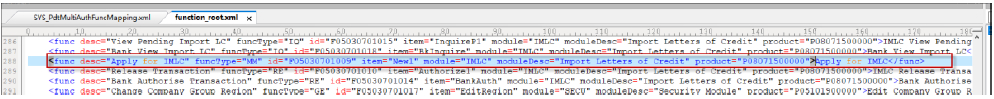
Apply for IMLC

* Function ID – F05030701009
* Product ID – P08071500000

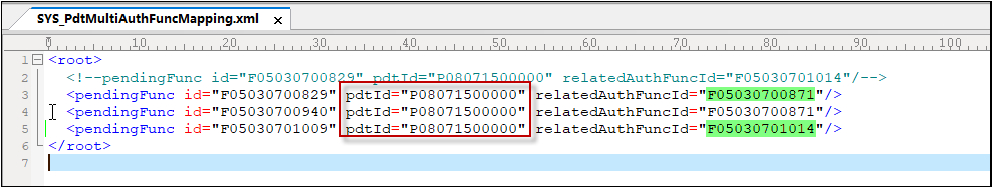
All these product functions are included in the same product.







1. In the SYS\_PdtMultiAuthFuncMapping.xml file, configure the same product ID for all three product functions and indicate the ID of the related authorization function that is to be used to authorize transactions from each product function.



Display Error When Printing Historical Records

When printing historical records, the style and fields may be improperly displayed on the preview screen.

Sample Procedure and Solution:

1. Log on to the CE system as an operator user and run the View Import LC function.

|  |
| --- |
|  |

1. Select the relevant record. In the dropdown list for More Actions, select the View Historical option and click on the Go button.

|  |
| --- |
|  |

1. In the screen that is displayed, select the record and click on the Select button.

|  |
| --- |
|  |

1. Details of the record are shown. Click on the Print button.

|  |
| --- |
|  |

1. The preview screen displays improper style and placement of fields. Log off the CE system.

|  |
| --- |
|  |

1. Open the Trx\_Template.jsp file. This file is stored on the following path:

[CEWeb.war folder]\Templates

Remove or disable the following script:

<EXIMTAGS:IncludeFile filePath="../JS/SYS\_InqCUBK.js"/>

|  |
| --- |
|  |

After updating this JSP file, generate the XML file for the Screen parameter in the CE Utility.

1. Log on to the CE system. View and print the same transaction. The preview screen then displays the proper style and placement of fields.

|  |
| --- |
|  |

Display All Records For Inquire CUBK

During transaction processing, search buttons may be used to inquire on the available CUBK records for the relevant field. However, once a record is selected and the search button is again clicked on, only the already selected record is displayed on the inquiry list. All available records are only displayed once the reset button is clicked on.

Sample Processing and Solution:

1. In the CE Utility, the CUBK rule is defined. This rule is then attached to the relevant search button on the screen (e.g., for Beneficiary Bank).

|  |
| --- |
|  |

|  |
| --- |
|  |

1. Log on to the CE system and run the Apply for Import LC function. Click on the relevant Search button to inquire on the available beneficiary banks.

|  |
| --- |
|  |

1. A window is displayed listing the available banks. Select a record by clicking on the Bank ID (e.g., BANK001).

|  |
| --- |
|  |

1. The values are retrieved in the relevant fields. Click on the Search button for beneficiary bank again.

|  |
| --- |
|  |

1. Only the details of the already selected record (e.g., BANK001) are displayed on the list.

|  |
| --- |
|  |

**NOTE:**

The Reset link must first be clicked on to remove the already selected record in order to display all available bank records in the inquire list.

|  |
| --- |
|  |

Log off the CE system.

1. Modify the Tautocomplete.js file. This file is stored on the following path:

[CEWeb.war folder]\JS\autoComplete

|  |
| --- |
| autocomplete key press  textBox.keyup(function (e) {  if(textBox.val() === '' || e.keyCode === keys.ESC || e.keyCode === keys.TAB) {  hideDropDown();  return;  }  // return if up/down/return key  if ((e.keyCode < 46 || (e.keyCode > 111 && e.keyCode < 186)) && (e.keyCode != 8)) {  e.preventDefault();  return;  }  currentPage = 1;  var callBack = settings.loadData;  var param = {};  var result = {};  param.isAutoComplete = true;  param.callBack = ajaxSuccessMethod.call(result);  param.currentPage = currentPage;  param.pageSize = pageSize;  //CEV2-02-17-17 martin.yin add for enhance cubk 20200824S  param.autoField = getEvtTarget(e).id;  //End  if(settings.columns.length === 0) {  param.needTableHead = true;  }else{  param.needTableHead = false;  }  callBack(param);  settings.data = result.data;  if(result.columns && result.columns.length > 0) {  settings.columns = result.columns;  settings.keyMap = result.keyMap;  settings.aliasMap = result.aliasMap;  }  if(result.errMsg) {  el.ddTableCaption.html(result.errMsg);  }  totalRecordNum = result.totalRecordNum;  currentPage = result.currentPage;  pageSize = result.pageSize;  CSRFTOKEN = result.CSRFTOKEN; |

1. Log on to the CE system and run the Apply Import LC function. The inquire window lists all available records when the Search button is clicked on even after a record has already been previously selected.

|  |
| --- |
|  |

Disallowing the System Login GET Method

When logging on to the CE system, the Post method is used to submit the form in the CE login page. However, the Get method is also allowed by the system, which is incorrect.

**Example:**

In the SYS\_index.htm file – located in the CEWeb.war folder – and the Post and GET methods are both allowed to be used for submitting the form.

* Post method:

|  |
| --- |
|  |

* Get method:

|  |
| --- |
|  |

**Configuration**

In order to disallow the Get method from being used during system login, the following configuration must be made in the Error Handling function:

* Log on to the CE Utility as an operator user and run the Error Handling function.

Add the following error details:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Error Key | Error Code | Error Message | Recommended Action | Error Level | Variable |
| SYSTEM | 000559 | GET method is not allowed | / | ERROR | N/A |

|  |
| --- |
|  |

Generate the xml file for the Error Handling parameter.

**Sample Procedure:**

1. In the SYS\_index.htm file, the Get method is used. This file is stored on the CEWeb.war folder.

|  |
| --- |
|  |

1. Log on to the CE system. The added error message is displayed and the user is not allowed to log on to the CE system.

|  |
| --- |
|  |

CE Utility and Parameter Issues

* CREATING THE SUPER ADMIN USER PROFILE

Chapter Three

* HOST NAME AND PORT NUMBER OF THE APPLICATION SERVER
* ORACLE DRIVER ERROR
* SQL 2000 SP3 ERROR
* CATALOG SETTING RULES
* MAINTAINING ITEM IDS
* DELETING DATA SOURCE AND USER INFORMATION
* creating the ce ear file backup
* Selecting Multiple Transaction or Security Records in a Catalog
* Selecting Multiple Inbox Records in a Catalog
* CE UTILITY AND CE NEW UTILITY DIRECTORY STRUCTURES

Creating the Super Admin User Profile

To create the default Super Admin user profile in CE Utility, run this SQL Insert statement:

(Start of code)

INSERT INTO "CEUSER"."SEC\_PARAMGR\_HDER" (C\_USER\_ID,C\_PASSWORD,C\_USER\_NAME,C\_USER\_DESC,C\_ASSIGNED\_BY,I\_USER\_TYPE,C\_PW\_HIST\_NUMS,C\_PW\_HIST,I\_MAX\_PW\_AGE,I\_MIN\_PW\_AGE,C\_REC\_PW\_DATE,I\_ACC\_LOCK,C\_MAC\_TYPE) VALUES ('superadmin','6c34629835d389f3cc4b449f2e551f5c','Super Admin','Super Administrator','owner',1,0,'6c34629835d389f3cc4b449f2e551f5c',0,0,TO\_DATE( '06-27-2006 00:00:00', 'MM-DD-YYYY HH24:MI:SS' ),0,'MD5');

(End of code)

Host Name and Port Number of the Application Server

Several CE Utility functions such as AP Server require the host name and port number of the application server.

Using WebSphere 9.0

To determine the Host and Port Values:

|  |
| --- |
| Do the following… |

|  |  |  |
| --- | --- | --- |
| 1. Run the WebSphere Administrative Console. |  |  |
|  |  |  |
| 2. In the Servers menu, click on the WebSphere Application Servers option. |  |  |
|  |  |  |
| 3. Select the application server name in the list that is shown (e.g., server1). |  |  |
|  |  |  |
| 4. The Configuration screen is displayed.  Click on the Ports option in the Communications section. |  |  |
|  |  |  |
| 5. The Ports screen is displayed.  The Host and Port values are specified in the BOOTSTRAP\_ADDRESS item. |  |  |
|  |  |  |
| 6. Use the Host and Port values in the relevant CE Utility function. |  |  |

Using WebLogic 8.1

Access the domain assigned to CE (e.g., cedomain), double- click on the Services folder, and click on the SNMP subfolder. Among the details displayed on the right section of the WebLogic screen is the Listen Port number.

|  |
| --- |
|  |

Figure 3. WebLogic Setting

Oracle Driver Error

Issue

This error occurs when logging on CE Utility: ‘Cannot Find User Information’ when Oracle 11g is installed.

Solution

The Oracle 11g driver is currently used. Change the CE Utility database profile: set the database version to Oracle 10g even if the version used is Oracle 11g.

|  |
| --- |
|  |

Figure 3. Database Information Dialog Box

SQL 2000 SP3 Error

Issue

When running CE Utility on Windows 2003 and the database version used is SQL2000 SP3, this error occurs: ‘SQLServer 2000 Driver for JDBC Unsupported VM encoding MS936’.

NOTE: This error usually occurs if the version used is J2RE\_1\_4\_2\_03.

Solution

Update the Java 2 Runtime Environment program to the J2RE\_1\_4\_2\_12 multi-language version.

Catalog Setting Rules

A few rules on catalog settings (thru the Product Catalog and Transaction Function - Component functions):

1. The ORDER BY relation must be placed at the end of the catalog clause, except for the next condition –
2. For field conditions that do not use specific values, use RELATION to connect these. Fixed conditions must be placed at the end of the catalog clause, but before the ORDER BY relation.

**EXAMPLE:**

As ‘C\_TRX\_STATUS = 'M’’ is a fixed condition, it is placed at the end of the clause.

C\_MAIN\_REF [OPERAND] 'VALUE' [RELATION] C\_USER\_ID [OPERAND] 'VALUE' AND C\_TRX\_STATUS = 'M'

1. Examples for comparing dates:

* …And (days(DATE\_FLD1) – days(DATE\_FLD2)) > 180
* …And (days(current date) - days(EXPIRY\_DT)) > 180
* …And (days(current date) – days (date(DATE\_FLD))) > 180
* …And (days(SYS\_BUSI\_DATE) – days ((DATE\_FLD)) > 180

Maintaining Item Ids

An Item Id can be manually modified and maintained. The Item Ids are defined in this file:

[CE Utility folder]\ce\_params\Script\_XML\product\_item\_prar.xml

|  |
| --- |
|  |

Figure 3. Item Ids

Deleting Data Source and User Information

Information on the data sources and users, shown in the CE Utility logon window, is maintained in the UserInfo.xml file. This file is placed in the main CEUtility folder.

|  |
| --- |
|  |

Figure 3. User Info.xml File

Creating the CE EAR File Backup

To ensure that the latest parameter data are kept for purposes of contingency, the Application server parameters may be bundled into an EAR file. The following are two ways to backup an EAR file:

* Use the Batch file
* Use the Command Prompt

Creating the Backup of CE.EAR Using a Batch File

The CE.EAR file can be created by using batch file. These commands can be created via notepad, however, the file extension must be saved in .BAT format.

|  |
| --- |
|  |

Figure 3. Sample Batch file to Back Up the CE.EAR file

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| The following are the contents of the batch file that is to be used to create an EAR file:   |  |  |  | | --- | --- | --- | | Concept |  | Description |  |  |  |  | | --- | --- | --- | | setlocal  for /f "tokens=1-4 delims=/ " %%a in (  'date /t'  ) do set month=%%b&set day=%%c&set year=%%d  set exportdate=%year%%day%%month% |  | This command is used to get the current date. | |  |  |  | | set EARExpander="C:\Program Files\IBM\WebSphere\AppServer\bin\EARExpander.bat" |  | This command is used to expand the contents of the EAR file that is to be exported. | |  |  |  | | set CEDIR="C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\installedApps\DOCS-CEV336Node01Cell\CE.ear" |  | This command is used to get the absolute path of the folder that is to be exported. | |  |  |  | | set exportdir=C:\CE\_EAR\_BACKUP\%exportdate% |  | This command is used to define the directory/location where the EAR file is to be exported/created. | |  |  |  | | %EARExpander% -ear %exportdir%\CE\_%exportdate%.ear -operationDir %CEdir% -operation collapse -expansionFlags all |  | This command is used to create the EAR file. | |  |  |  | |

|  |
| --- |
| To Create the CE.EAR Backup Using a Batch File |

|  |  |  |
| --- | --- | --- |
| 1. Run the batch file to create the backup copy of the EAR file. |  |  |
|  |  |  |
| 2. The process of creating backup for the EAR file is initialized. |  | Image28 |
|  |  |  |
| 3. The created backup copy of the EAR file is displayed in the directory that is defined in the batch file. |  |  |

Creating the Backup of CE.EAR Using the Command Prompt

The CE.EAR file can be created via Command Prompt by defining the JAVA\_HOME environment variable and setting the path to the EAR file to be created.

NOTE: Windows 10 is used to simulate this procedure.

|  |
| --- |
| To Create the CE.EAR Backup Using the Command Prompt |

|  |  |  |
| --- | --- | --- |
| 1. To define the JAVA\_HOME environment variable, access the following path:  Control Panel > System and Security > System > Advanced System Settings |  |  |
|  |  |  |
| 2. The System Properties window is displayed.  In the Advanced tab, click on the Environment Variables button. |  |  |
|  |  |  |
| 3. Click on the New button at the upper section of the screen in the Environment Variables window to define the variable. |  |  |
|  |  |  |
| 4. Define the settings for the JAVA\_HOME environment variable. |  |  |
|  |  |  |
| 5. Access the Command Prompt. |  |  |
|  |  |  |
| 6. Insert the following command to set the path to the JAVA\_HOME variable:  Set path = %JAVA\_HOME%\bin |  |  |
|  |  |  |
| 7. Change the directory into the path to which the CE.ear file is installed. |  |  |
|  |  |  |
| 8. Insert the following command to initialize the start of the backup process:  jar –cvf [Path to which the created ear file is to be stored]\[filename of the EAR file to be created] \* |  |  |
|  |  |  |
| 9. The creation of the backup EAR file is initialized. |  |  |
|  |  |  |
| 10. The newly created EAR file is displayed on the relevant path that is defined. |  |  |

Selecting Multiple Transaction or Security Records in a Catalog

In this system version, a flag is added to control whether or not to enable the multiple selection in a catalog.

Add the following codes to relevant function used for transaction catalog:

(Start of code)

function multiCheck(){

//enable multi selection for special functions

if (FUNC\_ID == "F05030700193") {

return true;

}

return false;

}

(End of code)

Selecting Multiple Inbox Records in a Catalog

In this system version, a flag is added to control whether or not to enable the multiple selection in a catalog.

Add the following codes to the relevant function used for inbox catalog:

(Start of code)

function isInboxMultiSel(){

try {

if (ITEM\_ID == "Awaiting Authorisation") {

return true;

}

}

catch (e) {

showExcpt("SYS\_MultiCatalog", e);

}

}

(End of code)

CE Utility and CE New Utility Directory Structures

To define which parameter settings are to be utilized by the CE system, the value of the global system parameter PARA\_FILE\_FORMAT must be set. This parameter may have any of the following values:

* NU, in which the parameters generated by the CE New Utility are used by the CE system.
* LEGACY, in which the parameters generated in the old CE Utility (e.g., CSBANK folder) are used by the CE system.

If this parameter is set to LEGACY, the CE core system is not able to use the parameters that are currently in CE NU.

If the CE New Utility is to be used, this global system parameter must be set to NU, and the following additional parameter and file changes must be further configured.

NOTE:

i. For added details on the CE New Utility, refer to [The CE New Utility](#The_CE_New_Utility_CH9) discussion in Chapter 9.

ii. Refer to the CE New Utility documentation set for more details on using the CE New Utility.

Configuration

To configure the directory structure for the CE New Utility, the following steps must be performed.

**Parameter Setting**

1. Log on to the CE Utility as a super administrator user and run the System Parameters function.

Add the PARA\_FILE\_FORMAT global system parameter and set its value to NU in order to support the CE New Utility.

Log on to the CE Utility as an operator user and generate the XML file for the Security System Parameter.

|  |
| --- |
|  |

1. Manually modify the following XML files.

* sys\_screens.xml

Path: [Parameter folder]\CE\_SYS\SYST

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <SCREENS>  ……  <SCREEN>  <ALIAS>SecuCatalog</ALIAS>  <DESC>Security standard catalog</DESC>  <JSPFILE>../Default/CATA/SYS\_SecuCatalog.jsp</JSPFILE >  <TMPLFILE>../Templates/SYS\_Template.jsp</TMPLFILE>  </SCREEN>  ……  </SCREENS> |

* DO\_D00000000XXX.xml

Path: [Parameter folder]\CE\_SYS\DO

**NOTE:**

This applies to the xml files of all data objects set in the DB Dictionary function of the CE Utility.

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <DO DATATYPE="2005" Desc="[Field Description]" Field="[Field Name]" Max="n" Min="0"  Name="[DO Name]"  PendingStore="N" Screen="..\Default\SCRN\DO\[DO JSP Screen Name]" StoreInto="D" StoreType="X">  <Elements>  ……  </Elements>  </DO> |

1. Manually add the schemaType.xml file into the following path:

[Parameter folder]\CE\_SYS\SYST

schemaType.xml File Content

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <root>  <schemainfos>  <schemainfo desc="Security schema" type="Security">CEUSER</schemainfo>  <schemainfo desc="Transaction schema" type="Transaction">CETRX</schemainfo>  </schemainfos>  </root> |

1. The following files are maintained manually. Thus, add these files in their corresponding folders according to the project.

* [Parameter folder]\CE\_SYS\BIRT.properties
* [Parameter folder]\CE\_SYS\characterfilter.xml
* [Parameter folder]\CE\_SYS\criteria\_mapping.xml
* [Parameter folder]\CE\_SYS\Cache\_Config.xml
* [Parameter folder]\CE\_SYS\CE\_Log\_Config.xml
* [Parameter folder]\CE\_SYS\CustomPolicy.xml
* [Parameter folder]\CE\_SYS\ejbserver.properties
* [Parameter folder]\CE\_SYS\EXPORT folder
* [Parameter folder]\CE\_SYS\FTL folder
* [Parameter folder]\CE\_SYS\LegalRequest.properties
* [Parameter folder]\CE\_SYS\password\_config.xml
* [Parameter folder]\CE\_SYS\password\_verification.xml
* [Parameter folder]\CE\_SYS\password\_verification\_type.xml
* [Parameter folder]\CE\_SYS\refreshSetting.xml
* [Parameter folder]\CE\_SYS\release\_ext\_mapping.xml
* [Parameter folder]\CE\_SYS\RPTEngine.xml
* [Parameter folder]\CE\_SYS\RSAKEY.properties
* [Parameter folder]\CE\_SYS\sys.char.fld.xml
* [Parameter folder]\CE\_SYS\sys.char.list.xml
* [Parameter folder]\CE\_SYS\SBRFieldMapping.xml
* [Parameter folder]\CE\_SYS\SYS\_PdtMultiAuthFuncMapping.xml
* [Parameter folder]\CE\_SYS\SYST\criteria.char.xml
* [Parameter folder]\CE\_SYS\SYST\sys\_batch.xml
* [Parameter folder]\CE\_SYS\SYST\sys\_cache\_data.xml
* [Parameter folder]\CE\_SYS\SYST\sys\_extend\_session\_fields.xml
* [Parameter folder]\CE\_SYS\SYST\sys\_external\_session\_obj.xml
* [Parameter folder]\CE\_SYS\SYST\sys\_screens.xml
* [Parameter folder]\CE\_SYS\WIDGETS\framework\cs-iwidget.xml

**CEWeb.war Modification**

1. Modify the file path structure.
2. In the CEWeb.war folder, add a new folder named Default.

|  |
| --- |
|  |

1. Access the following folder ─

[CEWeb.war folder]\[Bank]\[Country]\WEB

─ and move all the contents of this folder into the newly created Default folder in step A.

|  |
| --- |
|  |

1. Delete the following folder:

[CEWeb.war folder]\[Bank]

|  |
| --- |
|  |

1. Update the path for the following files with the path to the newly created Default folder

* [CEWeb.war folder]\Default\SCRN\BatchUpload.jsp
* [CEWeb.war folder]\JS\Buttons.js
* [CEWeb.war folder]\JS\SYS\_userprofile\_OpFunc\_Template.js
* [CEWeb.war folder]\screen\SYS\_CE\_Template.jsp
* [CEWeb.war folder]\screen\SYS\_DoMain.jsp
* [CEWeb.war folder]\Templates\STD\_SIMPLE\_DO\_Template.jsp
* [CEWeb.war folder]\Templates\STD\_Template.jsp
* [CEWeb.war folder]\Templates\STD\_Tree\_DO\_Template.jsp
* [CEWeb.war folder]\Templates\SYS\_Tree\_DO\_Template.jsp
* [CEWeb.war folder]\Templates\Trx\_Template.jsp

Example:

|  |
| --- |
|  |

1. Update the include file path parameter in the following JSP files according to project. Change the root path from “../../../../” into “../../”.

* [CEWeb.war folder]\Default\SCRN\EXCO\_ViewExco.jsp
* [CEWeb.war folder]\Default\SCRN\EXLC\_ViewExptLc.jsp
* [CEWeb.war folder]\Default\SCRN\IMCO\_ViewImpColl.jsp
* [CEWeb.war folder]\Default\SCRN\IMLC\_ViewArchive.jsp
* [CEWeb.war folder]\Default\SCRN\IMLC\_ViewImpLc.jsp
* [CEWeb.war folder]\Default\SCRN\IWGT\_ViewGtee.jsp
* [CEWeb.war folder]\Default\SCRN\OWGT\_ViewGtee.jsp
* [CEWeb.war folder]\Default\STD\Std\_Country\_Edit.jsp
* [CEWeb.war folder]\Default\STD\Std\_Currency\_Edit.jsp
* [CEWeb.war folder]\Default\STD\Std\_Currency\_Release.jsp
* [CEWeb.war folder]\Default\STD\Std\_Lov\_Edit.jsp

**NOTE:**

This is an example list. Update the necessary files based on project.

Example:

|  |
| --- |
|  |

Security Issues

* ADD ADMIN FAP ISSUE

Chapter Four

* FUNCTION ERRORS ON WEBLOGIC
* SUPPORTED SECURITY TOKENS
* CE DB User Rights in a Production Environment
* Token to Prevent CSRF Attack
* Secure Cookie Attributes
* Hiding Parameters in the Popup Window URL
* Password Validation Rules
* Reactivating locked out users
* logon and password issues
* PARAMETER CHECKLIST FOR SECURITY VULNERABILITIES
* JQUERY UPGRADE

Add Admin FAP Issue

Issue

The system cannot display the proper Add Admin FAP screen. Instead, the following screen is shown.

|  |
| --- |
|  |

Figure 4. Add Admin FAP

Solution

Run the following SQL statement to insert the required data into the CEUSER.SEC\_UNIT\_FUNC\_LOCK table.

(Start of code)

INSERT INTO CEUSER.SEC\_UNIT\_FUNC\_LOCK ( C\_UNIT\_CODE, C\_BK\_GROUP\_ID, C\_REFUSE\_REASON, C\_LOCKED\_FLAG, C\_TRX\_STATUS, C\_ADD\_FLAG, I\_FUNC\_TYPE, C\_CREATED\_BY, T\_LOCKED\_TIME, C\_CREATED\_BU, C\_LOCKED\_OP, C\_GRP\_CODE, C\_STATUS\_NAME, C\_LOCKED\_BU )

VALUES ('CSOFFICE', 'CSBANK', NULL, 'F', NULL, NULL, 0, NULL, '11/07/2019 01:01:38.467 PM', NULL,'superofficer', NULL, NULL, NULL)

Go

(End of code)

Function Errors on WebLogic

Issue

When running the Security Module on WebLogic, some functions cannot work properly.

**EXAMPLES OF ERRORS:**

* Run the Edit Operator FAP function, open a record and then cancel it. Run the Inquiry function and select the same user or record; all the fields (modules) are enabled, similar with the Edit screen.
* Run the Inquiry function first. When the Edit function is run next, the fields are disabled.

Solution

Modify the Sec\_Edit\_Op\_Func.jsp file in the screen folder and add the following codes:

(Start of code)

<%

response.setHeader("Pragma","No-cache");

response.setHeader("Cache-Control","no-cache");

response.setDateHeader("Expires", -1);

%>

(End of code)

Supported Security Tokens

CE currently supports the following security tokens –

* VASCO tokens : for re-authentication and non-repudiation
* eThales tokens: for multi-factor authentication, re-authentication, and non-repudiation
* Safeword tokens: for re-authentication.
* RSA Secure Tokens: for multi- factor authentication and re-authentication.

CE DB User Rights in a Production Environment

CE currently uses three types of users to access the database during production. Each user is granted with certain rights in the database.

* CEMETA - this is used to maintain the parameter configuration in the CE utility.
* CEUSER - this user is assigned with SELECT, INSERT, UPDATE, DELETE rights to all the tables that belong to ceuser and cetrx.
* CETRX - this user is assigned with SELECT, INSERT, UPDATE and DELETE rights to all tables that belong to cetrx and to the ceuser.TRX\_SYS\_ERR\_LOG table.

Token to Prevent CSRF Attack

Check that the following codes exist in the system's main pages:

* Print Token tag

This tag can be added to the Template jsp files in this path:[CEWeb.war Folder]/Templates.

<CETAGS:PrintToken/>

* Hidden Fields Codes

Check that the following codes for hidden fields exist in the system by viewing the page source after logging on to the CE system:

<input type="hidden" name=" CSRFTOKEN” id=" CSRFTOKEN" value="8AA7E5D93B6366EC013B6393640D0134">

* Error Code

Check that the following error code exists in the system:

|  |
| --- |
|  |

Figure 4. Edit Error Message Window

Secure Cookie Attributes

The following security attributes in the CE system are used to provide secure transmission of information from client to server:

* Secure Attribute - This is only used when the transmission of information is done via HTTPS. This prevents attacks and thefts through eavesdropping.
* HttpOnly Attribute - This is used when the transmission of information is done via HTTP or HTTPS. This restricts access from non-HTTP APIs such as JavaScript.

Parameter Setting

The global system parameter COOKIE\_SECURITY is used to specify the cookie security attribute to be applied in the system. This parameter is configured through the System Parameter function in CE Utility when logged on as a Super Administrator user.

|  |
| --- |
|  |

Figure 4. Parameter Config Window

|  |  |  |
| --- | --- | --- |
| Parameter Value |  | Description |

|  |  |  |
| --- | --- | --- |
| HttpOnly |  | This restricts access from other non-HTTP APIs such as JavaScript. |
|  |  |  |
| SECURE |  | This ensures that the cookie is always encrypted when transmitting information from client to server. |
|  |  |  |
| HttpOnly;SECURE |  | This is a combination of the two attributes - HttpOnly and SECURE. |

Hiding Parameters in the Popup Window URL

Issue

The parameters are shown in the URL of the popup window.

Solution

Modify the following files:

|  |  |  |
| --- | --- | --- |
| **File** | **Original Code** | **New Code** |
| SysViewImage.jsp | window.open(url,"win","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); | openWin(url,"win","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); |
| Button.js | var trxWin = window.open(actionUrl, "ceImgView", wStyle); | var trxWin = openWin(actionUrl, "ceImgView", wStyle); |
| openWin(url, winID, wStyle); var winID1= addWinIDSuffix(winID); var objNewWin = window.open(url, winID1, wStyle); arChildWin.push(objNewWin); | openWin(url, winID, wStyle); //var winID1= addWinIDSuffix(winID); //var objNewWin = window.open(url, winID1, wStyle); //arChildWin.push(objNewWin); |
| var wOpen = window.open("", "\_ViewGAPI", wStyle); oForm.target = "\_ViewGAPI"; | //var wOpen = window.open("", "\_ViewGAPI", wStyle); var wOpen = openWin("../screen/SYS\_Dummy.htm", "\_ViewGAPI", wStyle); oForm.target = addWinIDSuffix("\_ViewGAPI"); //oForm.target = "\_ViewGAPI"; |

|  |  |  |
| --- | --- | --- |
| DocumentList.js | var win = "newWin";  //openWin(url,"win","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow");  window.open(url, "win", "toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); | //var win = "newWin";  openWin(url,"win","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow");  //window.open(url, "win", "toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); |
| SECU\_SecuCompanyGrp.js | window.open(url); | openWin(url); |
| SHGT\_ApplyForGtee\_Function.js | window.open(url,"window",wStyle); | openWin(url,"window",wStyle); |
| SYS\_ResultSuccess.js | var win = "newWin"; window.open(url, "win", "toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); //openWin(url, "win", "toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); | //var win = "newWin"; //window.open(url, "win", "toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); openWin(url, "win", "toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); |
| SYS\_userprofile\_OpFunc.js | window.open(actionUrl); | openWin(actionUrl); |
| SYS\_userprofile\_View.js | window.open("21\_02.htm"); | openWin("21\_02.htm"); |
| error.jsp | window.open("../SYS\_index.htm","wind",wStyle); | openWin("../SYS\_index.htm","wind",wStyle); |
| loginfailPage.jsp | window.open('','\_top'); | openWin("../screen/SYS\_Dummy.htm","\_top"); |
| mailReport.jsp | var win ="newWin"; //openWin(url,"win","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); window.open(url,"\_blank","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); | //var win ="newWin"; openWin(url,"win","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); //window.open(url,"\_blank","toolbar=0,menubar=0,resizable=1,scrollbars=1,status=1,fullwindow"); |
| SYS\_CE\_Locked\_Message.jsp | window.open("../SYS\_index.htm","wind",wStyle); | openWin("../SYS\_index.htm","wind",wStyle); |
| SYS\_GetSecuItem.jsp | document.forms[0].target = vTar; \_login\_win = window.open("", vTar, wStyle); | //document.forms[0].target = vTar; document.forms[0].target = addWinIDSuffix(vTar); //\_login\_win = window.open("", vTar, wStyle); \_login\_win = openWin("../screen/SYS\_Dummy.htm", vTar, wStyle); |
| SYS\_ViewImagingDetail.jsp | window.open(actionUrl,"Img",wStyle); | openWin(actionUrl,"Img",wStyle); |

Password Validation Rules

In CE, the following password rules apply:

* The password is case sensitive.
* The minimum and maximum length of password can be defined.
* The space character is not allowed.
* The sequential characters are not allowed.
* The repeating characters are not allowed.
* The password must be a combination of either numbers, letters, or special characters.
* Other details used in the system cannot be used as password.
* The old passwords cannot be reused.
* The password is encrypted.
* The user account is to be deactivated after a specified number of unsuccessful logon attempts.

Before these can be utilized, the following actions are required:

* Add the password\_verification.xml file in the path:

[CE Parameter Folder]/CE\_SYS

The contents of this file are as follows:

(Start of code)

<root>

<regexp-lib>

<item name="includeRepeatChar"><![CDATA[(.{2,20})\1]]></item>

<item name="number"><![CDATA[^\d+$]]></item>

<item name="letter"><![CDATA[^[a-zA-Z]+$]]></item>

<item name="char"><![CDATA[^[-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+$]]></item>

<item name="number\_letter"><![CDATA[^(?![a-zA-Z]+$)(?![0-9]+$)[a-zA-Z0-9]+$]]></item>

<item name="number\_char"><![CDATA[^(?![\d]+$)(?![-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+$)[\d-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+$]]></item>

<item name="letter\_char"><![CDATA[^(?![a-zA-Z]+$)(?![-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+$)[a-zA-Z-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+$]]></item>

<item name="number\_letter\_char"><![CDATA[^(?!(?:[^a-zA-Z]|\D|[a-zA-Z0-9])$).+$]]></item>

</regexp-lib>

<strength>

<level value="1">

<regexp-item item="number"/>

<regexp-item item="letter"/>

<regexp-item item="includeRepeatChar"/>

<rule name="CECheckConsecutivePassword"/>

</level>

<level value="2">

<regexp-item item="number\_letter"/>

<regexp-item item="number\_char"/>

<regexp-item item="letter\_char"/>

</level> <level value="3">

<regexp-item item="number\_letter\_char"/>

</level> </strength>

<valid case="userprofile">

<passlevel>2</passlevel>

<min-length>4</min-length>

<max-length>16</max-length>

<rule name="CECheckEqualityFlds">

<param name="userDesc">usr.C\_USER\_DESC</param>

<param name="userId">usr.C\_USER\_ID</param>

</rule>

<rule name="CECheckRepeatChangePassword"/>

</valid>

<valid case="webcheck">

<rule name="CECheckEqualityFlds">

<param name="userId">usr.C\_USER\_ID</param>

<param name="userDesc">usr.C\_USER\_DESC</param>

<param name="unitCode">usr.C\_UNIT\_CODE</param>

<param name="deltUnitCode">usr.C\_DFLT\_UNIT\_CODE</param>

<param name="BKGrpId">usr.C\_BK\_GROUP\_ID</param>

</rule>

<rule name="CECheckRepeatChangePassword"/>

</valid>

</root>

(End of code)

* Add the following error codes.

|  |  |  |
| --- | --- | --- |
| **Error Key** | **Error Level** | **Error Message** |
| SYSTEM | ERROR | Old password is incorrect! |
| SYSTEM | ERROR | The password level is too weak. |
| SYSTEM | ERROR | The password is checked invalid according to customer rule. |
| SYSTEM | ERROR | Invalid rule for checking password. |

* Add the following function in the Basefunc.js file.

(Start of code)

function checkPwdStrength(){

try {

var password = document.getElementById("NEWPASSWORD").value;

if((typeof(password)=="undefined") || password == ""){

return;

}

password = encodeURIComponent(password);

var actionUrl = "../servlets/WSTrxManager?\_TRX\_STATUS=PWD\_CHECK&C\_PASSWORD="+password;

actionUrl = encodeUrl(actionUrl);

sendRequestByAjaxPost(actionUrl, true, genPwdStrength);

} catch (e) {

showExcpt("BaseFunc", e);

}

}

function genPwdStrength(xmlhttp) {

try {

var errorMsg = getSysErrorFromRespXml(xmlhttp);

if(errorMsg!=null) {

checkPwdStrengthFail(errorMsg);

return;

}

var rtnDom = xmlhttp.responseXML;

if (typeof(rtnDom) == "object") {

var root = rtnDom.documentElement;

if (root == null) {

return null;

}

var status = XMLManager.getChildNodeValue(root,"Status",true);

if(status=="N"){

var warnMsg = XMLManager.getChildNodeValue(root,"Warning",true);

checkPwdStrengthFail(warnMsg);

document.getElementById("pwd-strength").style.display="none";

return;

}

var level = XMLManager.getChildNodeValue(root,"PasswordLevel",true);

var iLevel = parseInt(level);

var pwd\_ul = document.getElementById("pwd-strength");

var liArray = pwd\_ul.getElementsByTagName("li");

for (var i = 0; i < liArray.length; i++) {

liArray[i].className="";

}

document.getElementById("pwd-strength").style.display="block";

for (var j = iLevel; j > 0; j--) {

document.getElementById("pwdstrength\_"+j).className="pwdstrength\_"+level;

}

var curr = document.getElementById("pwdstrength\_"+level);

curr.className = curr.className + " currs";

}

} catch (e) {

showExcpt("BaseFunc", e);

}

}

function checkPwdStrengthFail(errMsg){

\_SYS\_Display\_ERROR(errMsg);

}

(End of code)

* Check that the following codes exists in the SYS\_UserProfile.jsp file.

(Start of code)

<tr>

<td class="FldLabel" nowrap="nowrap">New Password </td>

<td><input name="NEWPASSWORD" type="password" class="CHAR\_M" id="NEWPASSWORD" title="New Password" size="10" maxlength="10" onblur="checkPwdStrength()"></td>

<td>

<ul id="pwd-strength" class="pwd-strength">

<li id="pwdstrength\_1">Weak</li>

<li id="pwdstrength\_2">Medium</li>

<li id="pwdstrength\_3">Strong</li>

</ul>

</td>

(End of code)

* Check that the following codes exist in the Basefunc.js file.

(Start of code)

function Change\_User(wStyle) {

try {

//2010-12-15 Frank modify start

var actionUrl = "../servlets/WSTrxManager";

if(typeof(window.jSessionId)!="undefined"){

actionUrl += window.jSessionId;

}

actionUrl+="?\_TRX\_STATUS=USERPROFILE&\_ACTION=GETUSERINFO";

//Frank modify end

//var url = "../servlets/WSTrxManager?\_TRX\_STATUS=USERPROFILE&\_ACTION=GETUSERINFO";

var winID = "UserProfile";

if(typeof(wStyle) == "undefined"){

wStyle = new OpenWinStyle();

wStyle.setWidth(340);

wStyle.setHeight(350);

}

// if (wStyle == undefined) {

// var wStyle = "menubar=no,toolbar=no,scrollbars=no,resizable=no,width=330,height=210";

// }

openWin(actionUrl, winID, wStyle);

} catch (e) {

DisExcpt("BaseFunc", e);

}

}

(End of code)

* Add the following codes in the Standard.css file.

(Start of code)

.pwd-strength{

margin: 5px 0 0 10px;

padding: 0;

display: none; 400px;

}

.success{

margin: 5px 0 0 170px;

padding: 0;

display: none;

}

#pwd-strength li{

list-style-type: none;

float: left;

border-top: 6px #DCDCDC solid;

color: #DCDCDC;

width: 50px;

}

#pwd-strength li.pwdstrength\_1{

border-top: 6px #E70403 solid;

width: 50px;

color: #DCDCDC;

}

#pwd-strength li.pwdstrength\_2{

border-top: 6px #FFAB16 solid;

width: 50px;

color: #DCDCDC;

}

#pwd-strength li.pwdstrength\_3{

border-top: 6px #32C040 solid;

width: 50px;

color: #DCDCDC;

}

#pwd-strength li.currs {

color: #000000;

}

(End of code)

Setting Different Password Policies for Bank and Company Users

By default, the same password policy is applied to all CE users. This includes the same password complexity levels and password expiry dates. If required, the system can be configured in order to implement different password policies for bank and company users. These segregated password policies include the following, among others:

* Different password verification rules;
* Different password expiration settings;
* Different password expiration notification settings;
* Different maximum number of logon attempts;
* Different maximum number of days inactive; and
* Different password history number settings.

To implement different password policies for bank and company users, the following files must be modified:

* CustomPolicy.xml
* password\_verification\_type.xml
* SYS\_UserProfile.js
* SYS\_ResetPassword.js

CustomPolicy.xml

Manually add a new policy to the CustomPolicy.xml file, which is stored in the following path:

[CE Parameter Folder]:\CE\_SYS

**EXAMPLE:**

(Start of Code)

<object name="PwdVerificationType" extClass="com.cs.core.policy.PwdVerificationRuleTypeIML">

<type name="pwdVerification"></type>

<type name="checkPwdExpireDays"></type>

</object>

(End of Code)

NOTE: If the above mentioned policy is not defined in the CustomPollicy.xml file, the global password verification rule (i.e., default settings) is implemented to all users. This rule is configured in the password\_verification.xml file stored in the path: [CE Parameter Folder]:\CE\_SYS.

Password\_verification\_type.xml

Manually add the pwdverification-rule parameters for company and bank users, which are indicated as NotBank and Bank respectively, to the password\_verification.xml file located in the path:

[CE Parameter Folder]:\CE\_SYS

**EXAMPLE:**

(Start of Code)

<pwdverification-rule type="NotBank">

<PSW\_EXPIRE\_DAYS attr="C">60</PSW\_EXPIRE\_DAYS>

<PSW\_EXPIRED\_CHANGE attr="C">TRUE</PSW\_EXPIRED\_CHANGE>

<PSW\_EXPIRE\_NOTIFY\_DAYS attr="C">10</PSW\_EXPIRE\_NOTIFY\_DAYS>

<PSW\_HISTORY\_NUM attr="C">3</PSW\_HISTORY\_NUM>

<MAX\_LOGON\_RETRIES attr="C">4</MAX\_LOGON\_RETRIES>

<MAX\_INACTIVE\_DAYS attr="C">30</MAX\_INACTIVE\_DAYS>

<regexp-lib>

<item name="includeRepeatChar"><![CDATA[(.{2,20})\1]]></item>

<item name="number"><![CDATA[^.\*\d+.\*$]]></item>

<item name="letter"><![CDATA[^.\*[a-zA-Z]+.\*$]]></item>

<item name="char"><![CDATA[^.\*[-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+.\*$]]></item>

<item name="lowercase" ignorecase="false"><![CDATA[^.\*[a-z].\*$]]></item>

<item name="uppercase" ignorecase="false"><![CDATA[^.\*[A-Z].\*$]]></item>

</regexp-lib>

<strength>

<level value="1">

<rule name="PasswordHybridRegExpValidation">

<param name="or">number,letter</param>

</rule>

</level>

<level value="2">

<rule name="PasswordHybridRegExpValidation">

<param name="and">number,letter</param>

</rule>

</level>

<level value="3">

<rule name="PasswordHybridRegExpValidation">

<param name="and">lowercase,uppercase,number</param>

</rule>

</level>

</strength>

<valid case="userprofile">

<passlevel>3</passlevel>

<min-length>8</min-length>

<max-length>16</max-length>

<rule name="CECheckEqualityFlds">

<param name="userDesc">usr.C\_USER\_DESC</param>

<param name="userId">usr.C\_USER\_ID</param>

</rule>

<rule name="CECheckRepeatChangePassword"/>

<rule name="CECheckConsecutivePassword">

<param name="intervalSize">1</param>

<param name="illegalLen">3</param>

</rule>

</valid>

<valid case="webcheck">

<min-length>8</min-length>

<max-length>16</max-length>

<rule name="CECheckEqualityFlds">

<param name="userId">usr.C\_USER\_ID</param>

<param name="userDesc">usr.C\_USER\_DESC</param>

<param name="unitCode">usr.C\_UNIT\_CODE</param>

<param name="BKGrpId">usr.C\_BK\_GROUP\_ID</param>

</rule>

<rule name="CECheckRepeatChangePassword"/>

<rule name="CECheckConsecutivePassword">

<param name="intervalSize">1</param>

<param name="illegalLen">3</param>

</rule>

</valid>

</pwdverification-rule>

<pwdverification-rule type="Bank">

<PSW\_EXPIRE\_DAYS attr="C">60</PSW\_EXPIRE\_DAYS>

<PSW\_EXPIRED\_CHANGE attr="C">TRUE</PSW\_EXPIRED\_CHANGE>

<PSW\_EXPIRE\_NOTIFY\_DAYS attr="C">10</PSW\_EXPIRE\_NOTIFY\_DAYS>

<PSW\_HISTORY\_NUM attr="C">3</PSW\_HISTORY\_NUM>

<MAX\_LOGON\_RETRIES attr="C">4</MAX\_LOGON\_RETRIES>

<MAX\_INACTIVE\_DAYS attr="C">30</MAX\_INACTIVE\_DAYS>

<regexp-lib>

<item name="includeRepeatChar"><![CDATA[(.{2,20})\1]]></item>

<item name="number"><![CDATA[^.\*\d+.\*$]]></item>

<item name="letter"><![CDATA[^.\*[a-zA-Z]+.\*$]]></item>

<item name="char"><![CDATA[^.\*[-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+.\*$]]></item>

<item name="lowercase" ignorecase="false"><![CDATA[^.\*[a-z].\*$]]></item>

<item name="uppercase" ignorecase="false"><![CDATA[^.\*[A-Z].\*$]]></item>

</regexp-lib>

<strength>

<level value="1">

<rule name="PasswordHybridRegExpValidation">

<param name="or">number,letter</param>

</rule>

</level>

<level value="2">

<rule name="PasswordHybridRegExpValidation">

<param name="and">number,letter</param>

</rule>

</level>

<level value="3">

<rule name="PasswordHybridRegExpValidation">

<param name="and">lowercase,uppercase,number</param>

</rule>

</level>

</strength>

<valid case="userprofile">

<passlevel>3</passlevel>

<min-length>8</min-length>

<max-length>16</max-length>

<rule name="CECheckEqualityFlds">

<param name="userDesc">usr.C\_USER\_DESC</param>

<param name="userId">usr.C\_USER\_ID</param>

</rule>

<rule name="CECheckRepeatChangePassword"/>

<rule name="CECheckConsecutivePassword">

<param name="intervalSize">1</param>

<param name="illegalLen">3</param>

</rule>

</valid>

<valid case="webcheck">

<min-length>8</min-length>

<max-length>16</max-length>

<rule name="CECheckEqualityFlds">

<param name="userId">usr.C\_USER\_ID</param>

<param name="userDesc">usr.C\_USER\_DESC</param>

<param name="unitCode">usr.C\_UNIT\_CODE</param>

<param name="BKGrpId">usr.C\_BK\_GROUP\_ID</param>

</rule>

<rule name="CECheckRepeatChangePassword"/>

<rule name="CECheckConsecutivePassword">

<param name="intervalSize">1</param>

<param name="illegalLen">3</param>

</rule>

</valid>

</pwdverification-rule>

(End of Code)

The pwdverification-rule parameter for the company and bank users have the same structure. The values of the elements within the parameters can be defined differently between the two user types.

|  |  |  |
| --- | --- | --- |
| Element |  | Description |

|  |  |  |
| --- | --- | --- |
| **<pwdverification- rule>** |  | This element defines the password policy for different types of users. The Type attribute may have any of the following values:   * NotBank, which indicates that the password policy is for company users; or * Bank, which indicates that the password policy is for bank users. |
|  |  |  |
| **PSW\_EXPIRE\_ DAYS** |  | This refers to the number of days the password is valid for logging on to the system. The system uses the value of SEC\_USER\_INFO.D\_PSW\_DATE as the start day for calculating the expiry date. After the specified number of days, the password expires and has to be changed. |
|  |  |  |
| **PSW\_EXPIRE\_ NOTIFY\_DAYS** |  | This element refers to the day when the password expiration notification is to expire. This day is calculated by subtracting the value of the PSW\_EXPIRE\_NOTIFY\_DAYS from the value of the PSW\_EXPIRE\_DAYS.  **EXAMPLE:**  If the PSW\_EXPIRE\_DAYS is set to 10 and the PSW\_EXPIRE\_NOTIFY\_DAYS is set to 2, the password expiration notification is shown on the 8th day.  The value of this element must be less than the value of the PSW\_EXPIRE\_DAYS. |
|  |  |  |
| **PSW\_EXPIRED\_ CHANGE** |  | The value of this element determines whether or not the user is allowed to reset the password once it expires. This can have any of the following values:   * True, which indicates that the user is allowed to reset the password. Upon logging on to the system, the Change Password screen is displayed. * False, which indicates that the user is not allowed to reset the password. Upon logging on to the system, an error message is displayed. This is the default value. |
|  |  |  |
| **PSW\_HISTORY\_ NUM** |  | The value of this element determines how many times the password has to be reset to a new and different one before the same password can be used again.  **EXAMPLE:**  A user’s password is LetMeIn01! currently. If the PSW\_HISTORY\_NUM is set to 3, the user can use the password LetMeIn01! after 3 password resets.  When resetting an expired password, the system checks the password history in the following cases:   * Resetting the password in Change User Profile; and * Resetting the password from the link in the Forget Password e-mail |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **PASSWORD\_ VERIFY** |  | The value of this element determines whether or not the system validates against the user password policy during the following cases:   * Resetting the password in Change User Profile; and * Resetting the password from the link in the Forget Password e-mail. |
|  |  |  |
| **MAX\_LOGON\_ RETRIES** |  | This element refers to the maximum number of failed logon attempts before the user account is locked by the system. |
|  |  |  |
| **MAX\_INACTIVE\_ DAYS** |  | This element refers to the maximum number of consecutive days a user account may remain inactive before the system deactivates it. |
|  |  |  |
| **<regexp-lib>** |  | This element is divided into several <item> elements with each one defining a regular expression. These regular expressions are used to set the password complexity rules in the <strength> element.  The <item> element supports the following attributes:   * name: This is a mandatory attribute and must be unique in the <regexp-lib> element. * ignorecase: This is an optional attribute used to indicate the case sensitivity. The default value is true.   **NOTE:**  If there are no letters in the expression (e.g., [a-z]), set the ignorecase attribute to false. This helps improve system performance.  **EXAMPLE:**  <regexp-lib>  <item name="includeRepeatChar"><![CDATA[(.{2,20})\1]]></item>  <item name="number"><![CDATA[^.\*\d+.\*$]]></item>  <item name="letter"><![CDATA[^.\*[a-zA-Z]+.\*$]]></item>  <item name="char"><![CDATA[^.\*[-`=\\\[\];',./~!@#$%^&\*()\_+|{}:"<>?]+.\*$]]></item>  <item name="lowercase" ignorecase="false"><![CDATA[^.\*[a-z].\*$]]></item>  <item name="uppercase" ignorecase="false"><![CDATA[^.\*[A-Z].\*$]]></item>  </regexp-lib> |
|  |  |  |
| **<strength>** |  | This element contains the password complexity rules (i.e., strength of the password). It is divided into three <level> elements corresponding to the three tiers of password complexity:   * <level value=“1”> for weak complexity; * <level value=“2”> for medium complexity; and * <level value=“3”> for strong complexity.   Each <level> has only one <rule> element with a name attribute hardcoded as PasswordHybridRegExpValidation.  Each <rule> can have several <param> elements to define the password complexity rule. The value of a <param> is one or more regular expression names defined in the <regexp-lib> element. Each regular expression name must be separated by a comma ( , ).  The relationship between the regular expressions is determined by the value of the <param> element’s name attribute. It can have any of the following values:   * and, which indicates that the password must meet the conditions of all the regular expressions in the <param> element; * or, which indicates that the password must meet the conditions of any of the regular expressions in the <param> element; * or\_X (with X being an integer from two and above); which is used when there are several <param> elements; and   **EXAMPLE:**  <strength>  <level value="1">  <rule name="PasswordHybridRegExpValidation">  <param name="or">number,letter</param>  </rule>  </level>  <level value="2">  <rule name="PasswordHybridRegExpValidation">  <param name="and">number,letter</param>  </rule>  </level>  <level value="3">  <rule name="PasswordHybridRegExpValidation">  <param name="and">lowercase,uppercase,letter</param>  <param name="or\_2">number,char</param>  </rule>  </level>  </strength>  In this example, if a password is a combination of numbers OR letters, it has a weak complexity (level 1). If it is a combination of numbers AND letters, it has a medium complexity (level 2). If it is a combination of lowercase AND uppercase letters, AND numbers OR special characters, it has a strong complexity (level 3).  An exclamation mark ( ! ) can be used to indicate that the password cannot have any character defined in a regular expression. It is placed at the beginning of the regular expression that is to be excluded. When this is used, the name attribute is not omitted from the <param> element.  **EXAMPLE:**  <rule name="PasswordHybridRegExpValidation">  <param>!char</param>  </rule>  This example means that the password cannot have any special character. |
|  |  |  |
| **<valid>** |  | This element refers to the password validation rules for different cases. It has a case attribute, which can have one of the following values:   * userprofile, which refers to the password validation performed upon resetting the password in Change User Profile. The system uses Ajax to check the password. * webcheck, which refers to the validation performed for one-time passwords. JavaScript is used to invoke the onBlur or onChange events. The system then performs the validation for the whole password.   Each <valid> element or case has the following elements to define the password validation rules:   * <passlevel>, which refers to the required password complexity (e.g., level 1); * <min-length>, which refers to the minimum password length (default is 1); * <max-length>, which refers to the maximum password length (default is 20); and * <rule>, which refers to a password validation rule.   A case can have more than one <rule> element. Each <rule> comes with a name attribute, which determines the type of validation that the rule performs.  CECheckEqualityFlds  If a <rule> element’s name attribute is set to CECheckEqualityFlds, the password cannot contain the values of the fields defined in the <rule>. The <rule> can have one or more <param> elements, each with a name attribute. To define a field, set the unique field name as the value of the name attribute, and the field Id as the value of the <param> element.  **EXAMPLE:**  <rule name="CECheckEqualityFlds">  <param name=“userDesc”>usr.C\_USER\_DESC</param>  <param name=“userId”>usr.C\_USER\_ID</param>  </rule>  This example means that the password cannot contain the values of the C\_USER\_DESC and C\_USER\_ID fields.  CECheckRepeatChangePassword  If a <rule> element’s name attribute is set to CECheckRepeatChangePassword, the system checks the password history based on the value of the <PSW\_HISTORY\_NUM> element.  CECheckConsecutivePassword  If a <rule> element’s name attribute is set to CECheckConsecutivePassword, the system checks the sequence interval of consecutive characters in the password according to numerical and alphabetical order (e.g., ABC or 123). This <rule> contains two <param> elements, each with a name attribute. The values of these name attributes are:   * intervalSize: This refers to the minimum sequence interval in the alphabetical and numerical order allowed for the password’s consecutive characters. The value of this <param> element must be a positive integer. The default value is 1. * illegalLen: This refers to the maximum length of consecutive sequential characters allowed in a password. The value of this <param> element must be a positive integer. The default value is 3.   **EXAMPLE:**  <rule name="CECheckConsecutivePassword">  <param name=“interval size”>1</param>  <param name=“illegalLen”>3</param>  </rule>  In this example, the following passwords are not allowed:   * 123ABqwe because 123 are 3 consecutive characters with a sequence interval of 1 in the numerical order. * cbaa1354 because cba are 3 consecutive characters with a sequence interval of 1 in the reverse alphabetical order. * 654abdeg because 654 are 3 consecutive characters with a sequence interval of 1 in the reverse numerical order.   <rule name="CECheckConsecutivePassword">  <param name=“interval size”>0</param>  <param name=“illegalLen”>2</param>  </rule>  In this example, the following passwords are not allowed:   * 11234ab because 11 are 2 consecutive characters with a sequence interval of 0 in the numerical order. * abcde4cc because cc are 2 consecutive characters with a sequence interval of 0 in the alphabetical order. * 655432 because 55 are 2 consecutive characters with a sequence interval of 0 in the numerical order. |

NOTE: The following elements are optional:

* PSW\_EXIRE\_DAYS
* PSW\_EXPIRE\_NOTIFY\_DAYS
* PSW\_EXPIRED\_CHANGE
* PSW\_HISTORY\_NUM
* PASSWORD\_VERIFY
* MAX\_LOGON\_RETRIES
* MAX\_INACTIVE\_DAYS

If not defined, the system uses the corresponding settings in the sys\_para.xml file, which is stored in this path: [CE Parameter Folder]:\CE\_SYS\SYST

SYS\_UserProfile.js

The SYS\_UserProfile.js file is located on this path: [CEWeb.war Folder]\JS. This file must be updated with the following new version codes:

|  |  |  |
| --- | --- | --- |
| Function |  | Code |

|  |  |  |
| --- | --- | --- |
| **function Confirm()** |  | **Old Version Code:**  function Confirm(){  try {  if (ChkPassword()) {  Submit();  }  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  }  **New Version Code:**  function Confirm(){  try {  if (ChkPassword()) {  //Submit();  SubmitByAjax();  }  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  } |
|  |  |  |
| **function Submit()** |  | **Old Version Code:**  function Submit(){  try {  var oForm = document.userprofile;  //CEV2-0536 CEV2.1.1 2010-01-07 Frank Modify Start  var actionUrl = "../servlets/WSTrxManager";  //CEV2-01-20-01 cash 20130114 s  //actionUrl = encodeUrl(actionUrl);  //Frank modify end  //oForm.action = actionUrl;  //oForm.submit();  submitForm(oForm,actionUrl);  //e  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  }  **New Version Code:**  function Submit(){  try {  var oForm = document.userprofile;  //CEV2-0536 CEV2.1.1 2010-01-07 Frank Modify Start  //var actionUrl = "../servlets/WSTrxManager";  //CEV2-01-20-01 cash 20130114 s  //actionUrl = encodeUrl(actionUrl);  //Frank modify end  //oForm.action = actionUrl;  //oForm.submit();  //submitForm(oForm,actionUrl);  //e  var actionUrl = "../servlets/WSTrxManager?CHANGE\_PROFILE\_STATUS=AJAXSUCCESS";  submitForm(oForm,actionUrl);  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  } |
|  |  |  |
| **function ChkPassword()** |  | **Old Version Code:**  function ChkPassword(){  try {  if (beInt(nConfTimes) >= 3) {  showMsg("Max confirm times is 3, please retry it later.");  window.close();  return false;  }  var invalid = " "; // Invalid character is a space  var minLength = 4; // Minimum length  var OLDPASSWORD = document.userprofile.OLDPASSWORD.value;  var NEWPASSWORD = document.userprofile.NEWPASSWORD.value;  var CONFPASSWORD = document.userprofile.CONFPASSWORD.value;  var charPwd = NEWPASSWORD.charCodeAt(0);  var isConsecutive = true;  var iFirstInterval = charPwd - NEWPASSWORD.charCodeAt(1);  var sErrMsg = "";  if (iFirstInterval != 1 && iFirstInterval != -1) {  isConsecutive = false;  }  var iInterval = 0;  for (var i = 1; i < NEWPASSWORD.length; i++) {  if (charPwd == NEWPASSWORD.charCodeAt(i)) {  sErrMsg = 90009;  }  if (i > 1) {  if (isConsecutive) {  iInterval = charPwd - NEWPASSWORD.charCodeAt(i);  if ((iInterval != 1 && iInterval != -1) || iFirstInterval != iInterval)  isConsecutive = false;  }  }  charPwd = NEWPASSWORD.charCodeAt(i);  }  if (isConsecutive) {  sErrMsg = 90010;  }  if(sErrMsg!=""){  var msg = \_getClientMessage(sErrMsg);  \_SYS\_Display\_WARN(msg);  return;  }  if (CONFPASSWORD.length == 0) {  sErrMsg = "Confirm Password is empty.";  document.userprofile.CONFPASSWORD.focus();  }  if (NEWPASSWORD.length == 0) {  sErrMsg = "New Password is empty.";  document.userprofile.NEWPASSWORD.focus();  }  //CEV2-01-01-09 2015-06-15 hank Modify s  //if(!chkNewPwd(NEWPASSWORD)){  //document.userprofile.NEWPASSWORD.focus();  //return false;  //}  if(!chkConfPwd(NEWPASSWORD, CONFPASSWORD)){  document.userprofile.CONFPASSWORD.focus();  return false;  }  //CEV2-01-01-09 2015-06-15 hank Modify e  if (OLDPASSWORD.length == 0) {  sErrMsg = "Old Password is empty.";  document.userprofile.OLDPASSWORD.focus();  }  if ((NEWPASSWORD.length > 0 && CONFPASSWORD.length > 0) &&  (NEWPASSWORD != CONFPASSWORD)) {  sErrMsg = "Confirm Password and New Password should be the same.";  document.userprofile.NEWPASSWORD.focus();  }  if ((OLDPASSWORD.length > 0 && NEWPASSWORD.length > 0) &&  (OLDPASSWORD == NEWPASSWORD)) {  sErrMsg = "The New Password cannot the same with Old Password.";  document.userprofile.NEWPASSWORD.focus();  }  nConfTimes++;  if (sErrMsg.length == 0) {  return true;  }  else {  showMsg(sErrMsg);  return false;  }  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  }  **New Version Code:**  function ChkPassword(){  try {  if (beInt(nConfTimes) >= 3) {  showMsg("Max confirm times is 3, please retry it later.");  window.close();  return false;  }  var invalid = " "; // Invalid character is a space  var minLength = 4; // Minimum length  var OLDPASSWORD = document.userprofile.OLDPASSWORD.value;  var NEWPASSWORD = document.userprofile.NEWPASSWORD.value;  var CONFPASSWORD = document.userprofile.CONFPASSWORD.value;  var charPwd = NEWPASSWORD.charCodeAt(0);  var isConsecutive = true;  var iFirstInterval = charPwd - NEWPASSWORD.charCodeAt(1);  var sErrMsg = "";  if (iFirstInterval != 1 && iFirstInterval != -1) {  isConsecutive = false;  }  var iInterval = 0;  for (var i = 1; i < NEWPASSWORD.length; i++) {  //if (charPwd == NEWPASSWORD.charCodeAt(i)) {  // sErrMsg = 90009;  //}  if (i > 1) {  if (isConsecutive) {  iInterval = charPwd - NEWPASSWORD.charCodeAt(i);  if ((iInterval != 1 && iInterval != -1) || iFirstInterval != iInterval)  isConsecutive = false;  }  }  charPwd = NEWPASSWORD.charCodeAt(i);  }  //if (isConsecutive) {  // sErrMsg = 90010;  //}  if(sErrMsg!=""){  var msg = \_getClientMessage(sErrMsg);  \_SYS\_Display\_WARN(msg);  return;  }  if (CONFPASSWORD.length == 0) {  sErrMsg = "Confirm Password is empty.";  document.userprofile.CONFPASSWORD.focus();  }  if (NEWPASSWORD.length == 0) {  sErrMsg = "New Password is empty.";  document.userprofile.NEWPASSWORD.focus();  }  //CEV2-01-01-09 2015-06-15 hank Modify s  //if(!chkNewPwd(NEWPASSWORD)){  //document.userprofile.NEWPASSWORD.focus();  //return false;  //}  if(!chkConfPwd(NEWPASSWORD, CONFPASSWORD)){  document.userprofile.CONFPASSWORD.focus();  return false;  }  //CEV2-01-01-09 2015-06-15 hank Modify e  if (OLDPASSWORD.length == 0) {  sErrMsg = "Old Password is empty.";  document.userprofile.OLDPASSWORD.focus();  }  if ((NEWPASSWORD.length > 0 && CONFPASSWORD.length > 0) &&  (NEWPASSWORD != CONFPASSWORD)) {  sErrMsg = "Confirm Password and New Password should be the same.";  document.userprofile.NEWPASSWORD.focus();  }  if ((OLDPASSWORD.length > 0 && NEWPASSWORD.length > 0) &&  (OLDPASSWORD == NEWPASSWORD)) {  sErrMsg = "The New Password cannot the same with Old Password.";  document.userprofile.NEWPASSWORD.focus();  }  nConfTimes++;  if (sErrMsg.length == 0) {  return true;  }  else {  showMsg(sErrMsg);  return false;  }  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  } |
|  |  |  |
| **function confirmEmail()** |  | **Old Version Code:**  function confirmEmail(){  try {  var C\_PRIVATE\_MAIL = document.getElementById("C\_PRIVATE\_MAIL");  var \_ACTION = document.getElementById("\_ACTION");  \_ACTION.value = "CHANGEBASEPROFILE";  if(\_TYPE == "PM"){  Confirm();  }  if(C\_PRIVATE\_MAIL.value == ""){  showEmailErrMsg('Email address can not be empty!');  return false;  }else if(!/^(\w)+(\.\w+)\*@(\w)+((\.\w{2,3}){1,3})$/.test(C\_PRIVATE\_MAIL.value)){  showEmailErrMsg('Email address is not in the right format!');  return false;  }else{  Submit();  }  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  }  **New Version Code:**  function confirmEmail(){  try {  var C\_PRIVATE\_MAIL = document.getElementById("C\_PRIVATE\_MAIL");  var \_ACTION = document.getElementById("\_ACTION");  \_ACTION.value = "CHANGEBASEPROFILE";  if(\_TYPE == "PM"){  Confirm();  }  if(C\_PRIVATE\_MAIL.value == ""){  showEmailErrMsg('Email address can not be empty!');  return false;  }else if(!/^(\w)+(\.\w+)\*@(\w)+((\.\w{2,3}){1,3})$/.test(C\_PRIVATE\_MAIL.value)){  showEmailErrMsg('Email address is not in the right format!');  return false;  }else{  //Submit();  SubmitByAjax();  }  }  catch (e) {  showExcpt("SYS\_UserProfile", e);  }  } |

Add the following new functions to the SYS\_UserProfile.js file.

|  |  |  |
| --- | --- | --- |
| Function |  | Code |

|  |  |  |
| --- | --- | --- |
| **function SubmitByAjax()** |  | function SubmitByAjax(){  var oForm = document.getElementById("userprofile");  var actionUrl="../servlets/WSTrxManager";  oForm.action = actionUrl;  oForm.method = "post";  postFormByAjax(document.userprofile, \_callback);  } |
|  |  |  |
| **function \_callback(xmlhttp)** |  | function \_callback(xmlhttp){  var rtnDom = xmlhttp.responseXML;  if (typeof(rtnDom) == "object") {  var root = rtnDom.documentElement;  if (root == null) {  return null;  }  var status = XMLManager.getChildNodeValue(root,"Status",true);  var changeType = XMLManager.getChildNodeValue(root,"ChangeUserType",true);  if(status=="N"){  var errorMsg = XMLManager.getChildNodeValue(root,"ErrorMsg",true);  if(changeType == "CHANGEPASSWORD"){  loadPwdErrorPage(errorMsg);  }  if(changeType == "CHANGEUSER"){  showEmailErrMsg(errorMsg);  }  return;  }  if(status=="Y"){  Submit();  }  }  } |
|  |  |  |
| **function loadPwdErrorPage(errorMsg)** |  | function loadPwdErrorPage(errorMsg){  \_SYS\_Display\_ERROR(errorMsg);  document.getElementById("NEWPASSWORD").value = "";  document.getElementById("CONFPASSWORD").value = "";  document.getElementById("pwd-strength").style.display="none";  document.getElementById("confPWDIcon").style.display="none";  } |

SYS\_Reset\_Password.js

The SYS\_Reset\_Password.js located on this path: [CEWeb.war Folder]\JS. Update this file with the following new version codes:

|  |  |  |
| --- | --- | --- |
| Function |  | Code |

|  |  |  |
| --- | --- | --- |
| **function \_ResetPwd\_Confirm()** |  | **Old Version Code:**  function \_ResetPwd\_Confirm(){  try {  if (chkNewPwd(NEWPASSWORD.value) && chkConfPwd(NEWPASSWORD.value, CONFPASSWORD.value)) {  // if (chkConfPwd(NEWPASSWORD.value, CONFPASSWORD.value)) {  // var isConsecutiveOK = consecutiveCheckOk(NEWPASSWORD.value,3);  // var sErrMsg="";  // if (!isConsecutiveOK) {  // sErrMsg = 90010;  //  // }  // var isDuplicateOK = duplicateCheckOk(NEWPASSWORD.value,3);  // if(isDuplicateOK==1){  // sErrMsg = 90009;  // }  // if(isDuplicateOK==2){  // sErrMsg = 90009;  // }  // if(sErrMsg!=""){  // var msg = \_getClientMessage(sErrMsg);  // \_SYS\_Display\_WARN(msg);  // // showMsg(sErrMsg);  // return;  // }  var oForm = document.getElementById("MAINFORM");  var actionUrl = "../servlets/WSTrxManager";  submitForm(oForm,actionUrl);  }  }  catch (e) {  showExcpt("SYS\_Reset\_Password", e);  }  }  }  **New Version Code:**  function \_ResetPwd\_Confirm(){  try {  //if (chkNewPwd(NEWPASSWORD.value) && chkConfPwd(NEWPASSWORD.value, CONFPASSWORD.value)) {  if(CheckNewConfirmPassword()){  // if (chkConfPwd(NEWPASSWORD.value, CONFPASSWORD.value)) {  // var isConsecutiveOK = consecutiveCheckOk(NEWPASSWORD.value,3);  // var sErrMsg="";  // if (!isConsecutiveOK) {  // sErrMsg = 90010;  //  // }  // var isDuplicateOK = duplicateCheckOk(NEWPASSWORD.value,3);  // if(isDuplicateOK==1){  // sErrMsg = 90009;  // }  // if(isDuplicateOK==2){  // sErrMsg = 90009;  // }  // if(sErrMsg!=""){  // var msg = \_getClientMessage(sErrMsg);  // \_SYS\_Display\_WARN(msg);  // // showMsg(sErrMsg);  // return;  // }  // var oForm = document.getElementById("MAINFORM");  // var actionUrl = "../servlets/WSTrxManager";  // submitForm(oForm,actionUrl);  ajaxSubmit();  }  }  catch (e) {  showExcpt("SYS\_Reset\_Password", e);  }  }  }  } |

Add the following new functions to the SYS\_Reset\_Password.js file.

|  |  |  |
| --- | --- | --- |
| Function |  | Code |

|  |  |  |
| --- | --- | --- |
| **function ajaxSubmit()** |  | function ajaxSubmit(){  var oForm = document.getElementById("MAINFORM");  var actionUrl="../servlets/WSTrxManager";  oForm.action = actionUrl;  oForm.method = "post";  postFormByAjax(document.MAINFORM, \_ajaxSuccess);  } |
|  |  |  |
| **function \_ajaxSuccess(xmlhttp)** |  | function \_ajaxSuccess(xmlhttp){  var rtnDom = xmlhttp.responseXML;  if (typeof(rtnDom) == "object") {  var root = rtnDom.documentElement;  if (root == null) {  return null;  }  var status = XMLManager.getChildNodeValue(root,"Status",true);  if(status=="N"){  var errorMsg = XMLManager.getChildNodeValue(root,"ErrorMsg",true);  loadErrorInfoPage(errorMsg);  return;  }  if(status=="Y"){  formSubmit();  }  }  } |
|  |  |  |
| **function formSubmit()** |  | function formSubmit(){  try {  var oForm = document.MAINFORM;  var actionUrl = "../servlets/WSTrxManager?CHANGE\_PROFILE\_STATUS=AJAXSUCCESS";  submitForm(oForm,actionUrl);  }  catch (e) {  showExcpt("SYS\_Reset\_Password", e);  }  } |
|  |  |  |
| **function loadErrorInfoPage(errorMsg)** |  | function loadErrorInfoPage(errorMsg){  \_SYS\_Display\_ERROR(errorMsg);  document.getElementById("NEWPASSWORD").value = "";  document.getElementById("CONFPASSWORD").value = "";  document.getElementById("newPwd\_status\_tip").style.display="none";  document.getElementById("confPwd\_status\_tip").style.display="none";  } |
|  |  |  |
| **function CheckNewConfirmPassword()** |  | function CheckNewConfirmPassword(){  try {  var invalid = " "; // Invalid character is a space  var minLength = 4; // Minimum length  var NEWPASSWORD = document.MAINFORM.NEWPASSWORD.value;  var CONFPASSWORD = document.MAINFORM.CONFPASSWORD.value;  var charPwd = NEWPASSWORD.charCodeAt(0);  var isConsecutive = true;  var iFirstInterval = charPwd - NEWPASSWORD.charCodeAt(1);  var sErrMsg = "";  if (iFirstInterval != 1 && iFirstInterval != -1) {  isConsecutive = false;  }  var iInterval = 0;  for (var i = 1; i < NEWPASSWORD.length; i++) {  if (i > 1) {  if (isConsecutive) {  iInterval = charPwd - NEWPASSWORD.charCodeAt(i);  if ((iInterval != 1 && iInterval != -1) || iFirstInterval != iInterval)  isConsecutive = false;  }  }  charPwd = NEWPASSWORD.charCodeAt(i);  }  if(sErrMsg!=""){  var msg = \_getClientMessage(sErrMsg);  \_SYS\_Display\_WARN(msg);  return;  }  if (CONFPASSWORD.length == 0) {  sErrMsg = "Confirm Password is empty.";  document.MAINFORM.CONFPASSWORD.focus();  }  if (NEWPASSWORD.length == 0) {  sErrMsg = "New Password is empty.";  document.MAINFORM.NEWPASSWORD.focus();  }  if(!chkConfPwd(NEWPASSWORD, CONFPASSWORD)){  document.MAINFORM.CONFPASSWORD.focus();  return false;  }  if ((NEWPASSWORD.length > 0 && CONFPASSWORD.length > 0) &&  (NEWPASSWORD != CONFPASSWORD)) {  sErrMsg = "Confirm Password and New Password should be the same.";  document.MAINFORM.NEWPASSWORD.focus();  }  if (sErrMsg.length == 0) {  return true;  }  else {  showMsg(sErrMsg);  return false;  }  }  catch (e) {  showExcpt("SYS\_Reset\_Password", e);  }  } |

Reactivating Locked Out Users

When the maximum number of failed logon attempts to the CE application has been exceeded by a user, the user account gets locked out (i.e., the user is not allowed to log on to the system). The reactivation process depends on the system’s configuration. Two processes are available:

* Manual Reactivation
* Auto-reactivation

NOTE: Manual reactivation is always available. On the other hand, auto-reactivation is available only when it is enabled.

Manual Reactivation

The system’s default settings require the locked out user to be manually reactivated by a user that has access to the Security module.

|  |
| --- |
| EXAMPLE |

|  |  |  |
| --- | --- | --- |
| 1. When an Operator user uses incorrect logon credentials to access the CE system, the error 000337 is displayed. |  |  |
|  |  |  |
| 2. When the maximum number of failed logon attempts is exceeded, the account gets locked out and error 000310 is displayed.  The user must contact the bank administrator to request for reactivation. |  |  |
|  |  |  |
| 3. Log on to the CE system as a user with access rights to the Security module and run the Activate/Inactivate Operator function in the Operator Maintain group. |  |  |
|  |  |  |
| 4. The function catalog screen is displayed. Select the relevant user and click on the Active User button.  **NOTE:**  i. The value on the Logon Retries column refers to the number of failed logon attempts by the relevant user account.  ii. The maximum number of logon retries is defined through the MAX\_LOGON\_RETRIES global system parameter. |  |  |
|  |  |  |
| 5. In the Activate/Inactivate Operator function screen that is displayed, click on the Active User button. |  |  |
|  |  |  |
| 6. A confirmation message is displayed. Click on the OK button to continue. |  |  |
|  |  |  |
| 7. The Instruction Completed screen is shown. |  |  |
|  |  |  |
| 8. Release the transaction through the Authorise Operator function.  **NOTE:**  The Logon Retries field value is reset to 0. |  |  |
|  |  |  |
| 9. The Operator user can attempt to log on to the system again. |  |  |
|  |  |  |
| 10. When correct logon credentials are used, the Operator user is able to access the system. |  |  |

Auto-Reactivation

The system can be configured to allow the auto-reactivation of locked out user accounts. To enable this feature, the following parameters must be defined or modified in the CE Utility:

* UNLOCKTIME global system parameter
* Error code 000310

NOTE: The system can only automatically reactivate user accounts that are deactivated by exceeding its maximum number of failed logon attempts. Accounts that are locked out for any other reason are not automatically reactivated.

Parameter Setting

To enable auto-reactivation, the global system parameter UNLOCKTIME must be defined. This parameter is used to specify the time (in minutes) that the user account remains locked out before it is automatically reactivated. During this time, the user cannot log on to the system even when using the correct logon details. When the specified time has elapsed, the user can try to log on to the system again.

To configure this system parameter, log on to the CE Utility as Super Administrator user and access the System Parameters function in the Parameter Manage group.

|  |
| --- |
|  |

Figure 4. UNLOCKTIME System Parameter

The UNLOCKTIME global system parameter may have any of the following values:

|  |  |  |
| --- | --- | --- |
| Parameter Value |  | Description |

|  |  |  |
| --- | --- | --- |
| **0** |  | The auto-reactivate feature is not enabled. The user must contact the bank administrator to reactivate the account. This is the default value.  **NOTE:**  If this system parameter is not defined, the auto-reactivate feature is not enabled as well. |
|  |  |  |
| **Any positive integer** |  | The system calculates the number of minutes from the last failed logon attempt. If the calculated time is less than this parameter value, the user is not allowed to log on to the system and the account remains deactivated. Otherwise, the account is reactivated and the user is able to log on. |

The system error message 000310 must also be modified. When the Auto-reactivation feature is disabled, this error message instructs the locked out user to contact the bank administrator for reactivation. If the Auto-reactivation feature is enabled, error message 000310 must be configured to instruct the user to wait for a specified amount of time before the account gets reactivated automatically.

NOTE: The error message 000135 is the default error message for unexpected exception. It provides a general message to the user, which can apply to all exceptions.

Modify the configuration of the error message 000310 through the Error Handling function in the Maintenance group of the CE Utility.

|  |
| --- |
|  |

Figure 4. Error Message 000310 Configuration

|  |  |  |
| --- | --- | --- |
| Field |  | Value |

|  |  |  |
| --- | --- | --- |
| **Error Code** |  | 000310 |
|  |  |  |
| **Error Key** |  | SYSTEM |
|  |  |  |
| **Error Level** |  | ERROR |
|  |  |  |
| **Error Message** |  | You have exceeded your maximum logon retries. |
|  |  |  |
| **Recommended Action** |  | Please try again after [%1 Hour %2 Minute %3 Second]  **NOTE:** The %1, %2, and %3 variables refer to the hours, minutes, and seconds, respectively. |

Example

|  |
| --- |
| EXAMPLE |

|  |  |  |
| --- | --- | --- |
| 1. When an Operator user uses incorrect logon credentials to access the CE application, error 000337 is displayed. |  |  |
|  |  |  |
| 2. When the maximum number of failed logon attempts is exceeded, the account gets locked out and error 000310 is displayed.  The error message displays the amount of time the user must wait before logging on to the system again. |  |  |
|  |  |  |
| **NOTE:**  If a locked out user attempts to log on to the system during the lockout time, the timer is not restarted. |  |  |
|  |  |  |
| 3. When the specified lockout time has elapsed, the system automatically reactivates the account. The user can attempt to log on to the system again. |  |  |
|  |  |  |
| 4. When correct logon credentials are used, the Operator user is able to access the system. |  |  |

Logon and Password Issues

There are several reasons why a CE user’s logon attempt to the CE system may fail. The usual logon issues that may be encountered are often of the following types:

* Logon Information Issues;
* User Profile Status Issues; and
* Password Issues

NOTE:

i. For the complete information on CE security settings and the Security module, refer to the *CE Security and System Maintenance Functions* manual.

ii. The system can be configured in such a way that password verification rules (e.g., limit for the period of inactivity) for bank users are different from that of company users. For details on this configuration, refer to the [Password Validation Rules](#password_validation_rules) topic of this chapter.

Logon Information Issues

In the CE Logon screen, the user profile’s company code, logon Id, and password are required to be specified to access the system. If any of the logon information provided is incorrect, an error message is displayed and the user is denied access. The same goes when the specified company code or logon Id does not exist in the database. The user can return to the Logon screen and provide the correct logon information.

Password issues are discussed separately in this topic.

User Profile Status Issues

Access to the system is denied to a user profile that is inactive for too long, not yet effective, already expired, or currently logged on in a different machine.

User Profile Inactive for too Long

A user profile automatically gets suspended when the account has not been logged on the CE system for a period longer than the maximum number of days allowed for inactivity. When the details of the suspended user profile are viewed in the Security module through the Inquire SSO, Inquire Admin, or Inquire Operator functions, the Inactive/ Suspended flag of the Advanced Security Information section is marked. Unmarking this flag grants the corresponding user access to the system again.

NOTE: The limit for the period of inactivity is defined through the MAX\_INACTIVE\_DAYS security system parameter.

User Profile Not Yet Effective

When a user is created in the Security module, it is mandatory to specify the value of the Effective Date field, which refers to the date when the new user profile can be used. If a new user attempts to log on to CE before this date, an error message is displayed and the system denies the user access.

User Profile Already Expired

Once a user profile expires, it cannot be used to log on to CE anymore. The expiry date is specified in the Expiry Date field when the profile is created or modified in the Security module. This information can be viewed through the Inquire SSO, Inquire Admin, or Inquire Operator functions.

The affected user can contact the bank administrator to request for the reactivation of the expired account. The administrator can then set the expiry date at a later date and reactivate the account through the Activate/Inactivate SSO, Activate/Inactivate Admin, or Activate/Inactivate Operator functions of the Security module.

User Profile Already Logged On

Whether or not a user profile can be logged on the system from multiple machines at the same time depends on its Multi Login field settings in the Security module. The value of this field is specified during the creation or modification of a user profile. It can be set to either Yes or No.

If the Mutli Login field is set to No, the user profile can be used to log on to CE through one machine only. Until the account is logged off, all attempts to access the same account through other machines are denied.

Password Issues

The common password issues encountered when attempting to log on to the CE system are:

* Incorrect password;
* Forgotten password; and
* Expired password.

Incorrect Password

A user is automatically denied access to the system when the specified password on the CE Logon screen is incorrect. The user must simply try logging on again with a correct password. However, if the maximum number of failed logon attempts is reached, the account is automatically suspended. The error message displayed contains instructions on how to regain access to CE. These instructions depend on the system configuration.

If the Auto-reactivate feature is enabled, the error message displays the amount of time that the user must wait before the account becomes reactivated. Once the specified time has elapsed, the user can try to log on again.

If the Auto-reactivate feature is disabled, the error message instructs the user to contact the bank administrator for manual reactivation. The Bank administrator can then reactivate the suspended account through the Activate/Inactivate SSO, Activate/Inactivate Admin, or Activate/Inactivate Operator functions of the Security module.

NOTE:

i. The maximum number of failed logon attempts is defined through the MAX\_LOGON\_RETRIES security system parameter

ii. For details on how to enable or disable the Auto-reactivate feature, refer to the preceding [Reactivating Locked Out Users](#reactivating_lockedout_users) topic of this chapter.

Forgotten Password

In case of a forgotten account password, the I Forgot My Password link is provided in the CE Logon screen. Clicking on this link initiates the password reset process.

NOTE: For details on the password reset process through the I Forgot My Password Link, refer to the *CE Security and System Maintenance Functions* manual.

Expired Password

Once a user profile’s password expires, the user is denied access to CE. Whether or not the user can reset the account password upon expiration depends on the setting of the PSW\_EXPIRED\_CHANGE security system parameter. If this parameter is set to True, the user is prompted to reset the account password upon logon. If it is set to False, the user is not allowed to reset the password upon logon. The bank administrator must be contacted for further processing.

NOTE: The number of days the password is valid for logging on to the system is defined through the PSW\_EXPIRE\_DAYS security system parameter.

Parameter Checklist for Security Vulnerabilities

To prevent the potential for compromised integrity in the CE system, certain parameters must be configured to avoid security vulnerabilities.

Cookie Settings for HTTP Session

It is recommended to set the complete Domain and Secure cookie attributes in the Set-Cookie HTTP response header option. The Path cookie attribute must also be set to the context root (i.e. /CEWeb/).

To configure the Cookie settings:

**IBM WebSphere**

1. Run the WebSphere Administrative console. In the Applications menu, expand the Application Types sub-menu and select the WebSphere enterprise applications option.

|  |
| --- |
|  |

1. Click on the CE application link.

|  |
| --- |
|  |

1. The configuration page for the CE application is shown. Select the Session Management option in the Web Module Properties section.

|  |
| --- |
|  |

1. The configuration page for session manager properties is shown. In the Session tracking mechanism section, mark the corresponding Enable cookies option. To set the cookie path and attribute, click on the Enable cookies link.

|  |
| --- |
|  |

1. The Cookie settings configuration page is displayed. Mark the following items to enable these options:

* Restrict cookies to HTTPS sessions
* Set session to HTTP Only to help prevent cross-site scripting attacks

In the Cookie path section, mark the following option to set the cookie path to the context root.

* Use the context root

Click on the OK button.

|  |
| --- |
|  |

1. Click on the Save link to store the settings to the master configuration.

|  |
| --- |
|  |

**Weblogic**

Update the weblogic.xml file that is stored on this path:

.ear\CEWeb.war\WEB-INF

|  |
| --- |
|  |

**JBOSS**

Update the web.xml file that is stored on this path:

.ear\CEWeb.war\WEB-INF

|  |
| --- |
|  |

Global System Parameter Settings

The values of certain global system parameters must be defined to prevent security vulnerabilities.

Cross Site Request Forgery (CSRF) Attack

|  |  |  |
| --- | --- | --- |
| Parameter |  | Definition |

|  |  |  |
| --- | --- | --- |
| **DISABLE\_CSRFTOKEN** |  | This controls whether or not the system checks the token to prevent CSRF attacks.  This security system parameter may have either one of the following values:   * TRUE, in which the token is not checked * FALSE, in which the token is checked to prevent CSRF attacks. This is the default value. |
|  |  |  |
| **NO\_TOKEN\_CHECK\_REQ** |  | This is used for defining the parameter from the page or action URL that is not to be checked by the system. Multiple values can be defined, each separated by a semi-colon (;).  **EXAMPLE:**  If this system parameter is set to PWD\_CHECK;USERPROFILE, then the system does not check the PWD\_CHECK and USERPROFILE values on the URL. |
|  |  |  |
| **NO\_TOKEN\_CHECK\_SVT** |  | This is used to define unchecked servlet settings. The value is the url-pattern value that is set in the web.xml file. The values to be defined are separated by semi-colon (;). In addition, the servlet can simply be added as a value of this parameter.  **EXAMPLE:**  screen/SYS\_UserProfile.jsp;/servlets/CELoginAuthorizationManager;/servlets/WSSGetUserProfile |
|  |  |  |
| **NO\_TOKEN\_CHECK\_URI** |  | This is used to define unchecked file request. If the URI ends with a value that is defined in the setting, the request does not check the token. The values to be defined are separated by a semi-colon (;).  The following are the file extensions that may be used:.js, .htm, .html, .css, .jpeg, .png, .gif, .ico, .bmp, .jpg, .tiff, .pcx, .tga, .exif, .fpx, .svg, .psd, .cdr, .pcd, .dxf, .ufo, .eps, .ai, .raw  **EXAMPLE:**  <NO\_TOKEN\_CHECK\_URI attr="C">.JS;.CSS</ NO\_TOKEN\_CHECK\_URI > |

Host Header Poisoning Vulnerability

|  |  |  |
| --- | --- | --- |
| Parameter |  | Definition |

|  |  |  |
| --- | --- | --- |
| **DISABLE\_HTTPHOSTCHECK** |  | This is used to control whether or not to apply the HTTP host attack check.  This security system parameter may have either one of the following values:   * TRUE, in which the HTTP host check is disabled. * FALSE, in which the HTTP host check is enabled, using the configuration specified in the HTTPHOST\_LIST security system parameter. |
|  |  |  |
| **HTTPHOST\_LIST** |  | This is used to configure the legal host to prevent Host Header Poisoning Vulnerability. A semicolon can be used to separate multiple hosts.  To enable this setting, the DISABLE\_HTTPHOSTCHECK must first be set to FALSE.  **EXAMPLE:**  <HTTPHOST\_LIST attr="C">10.39.104.39:39083; 127.0.0.1</HTTPHOST\_LIST> |

NOTE: When defining the URL, the IP address and port number of the AP server must be used. For the bank’s client users, the domain must also be defined.

Example:

- Internal URL: http://10.39.104.39:9080/CEWeb/SYS\_index.htm

- External URL: <http://www.neweximbills.com/CEWeb/SYS_index.htm>

If only the external access is to be allowed, the value of the HTTPHOST\_LIST parameter must be set to [www.neweximbills.com](http://www.neweximbills.com).

If both the internal and external access are to be allowed, the value of the HTTPHOST\_LIST parameter must be set to www.neweximbills.com;10.39.104.39:9080.

HSTS Header in HTTP Response

|  |  |  |
| --- | --- | --- |
| Parameter |  | Definition |

|  |  |  |
| --- | --- | --- |
| **SECU\_EXT\_RESPONSE\_HEADER** |  | This is used to define the security attributes in the HTTP response header. More than one attribute can be defined. Separate multiple attributes by using \r as in the example below.  **EXAMPLE:**  Strict-Transport-Security:max-age=16070400;includeSubDomains\rX-XSS-Protection:1;mode=block\rContent-Security-Policy:font-src http://192.168.1.9:9080; img-src 'self' data: base64; default-src 'unsafe-inline'; 'unsafe-eval' http://192.168.1.9:9080 |

NOTE: When defining the URL, the IP address and port number of the AP server must be used. For the bank’s client users, the domain must also be defined.

The value of the SECU\_EXT\_RESPONSE\_HEADER parameter is defined through the CE Utility. However, if this parameter is manually configured by modifying the sys\_para.xml file, its value must be defined by replacing the single quotations with &apos;.

Example:

Through the CE Utility:

Strict-Transport-Security:max-age=16070400;includeSubDomains\rX-XSS-Protection:1;mode=block\rContent-Security-Policy:font-src http://10.39.104.39:9080; img-src 'self' data: base64; default-src 'unsafe-inline'; 'unsafe-eval' http://10.39.104.39:9080

By Manual Configuration:

Strict-Transport-Security:max-age=16070400;includeSubDomains\rX-XSS-Protection:1;mode=block\rContent-Security-Policy:font-src http://10.39.104.39:9080;default-src &apos;unsafe-inline&apos; &apos;unsafe-eval&apos; http://10.39.104.39:9080

The sys\_para.xml file is stored on the path: [Parameter Folder]\CE\_SYS\SYST

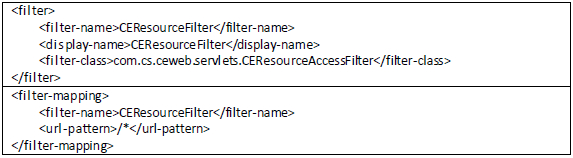
Web Resource Access Control

The CE application must prevent any web resource access for illegal client users.

* Web.xml File

Make sure that filter and filter mapping are configured in the web.xml file. This file is stored on the following path:

[CEWeb.war Folder]\WEB-INF

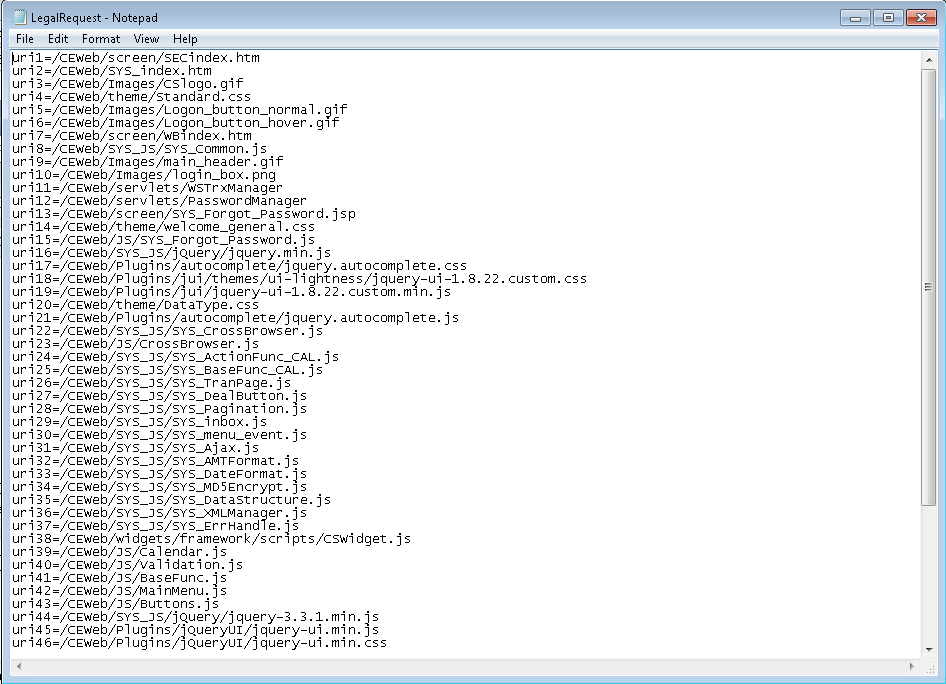


* LegalRequest.properties File

The CEWeb context root must be updated in the LegalRequest.properties file. This file is stored on the following path:

[Parameter Folder]\CE\_SYS

NOTE: In projects, files used when logging on to the system must be updated with a unique URI ID.

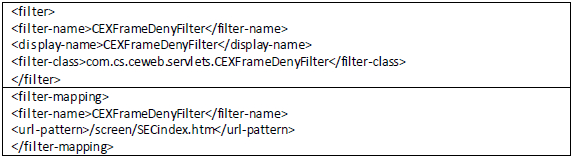


X-Frame Attack

* Web.xml Configuration

To prevent X-Frame attacks, make sure that filter and filter mapping are configured in the web.xml file. This file is stored on the following path:

[CEWeb.war Folder]\WEB-INF



* HTTP Server Configuration

**Apache Server**

Configure the X-Frame options on the Apache server.

|  |
| --- |
| Header always append X-Frame-Options [value] |

One of following values may be specified:

* DENY, in which the page cannot be displayed in a frame, regardless of the site’s attempts.
* SAMEORIGIN, in which the page can only be displayed in a frame on the same origin as the page itself.
* ALLOW\_FROM, in which the page can only be displayed in a frame on the specified origin.

**IHS (IBM HTTP Server)**

The X-Frame-Options header may be added in IHS. To do this:

1. Ensure that the headers module (i.e., mod\_headers module) is loaded. To load this module, uncomment the line and restart the web server.

|  |
| --- |
| LoadModule headers\_module modules/mod\_headers.so |

1. In the http.conf file, add the X-Frame-Options header using the Header set.

|  |
| --- |
| Header set X-Frame-Options [value] |

One of following values may be specified:

* DENY, in which the page cannot be displayed in a frame, regardless of the site’s attempts.
* SAMEORIGIN, in which the page can only be displayed in a frame on the same origin as the page itself.
* ALLOW\_FROM, in which the page can only be displayed in a frame on the specified origin.

JQuery Upgrade

Using outdated versions of the JavaScript jQuery libraries may cause security vulnerabilities such as cross site scripting. To avoid vulnerabilities, the following items must be updated:

* jQuery upgrade from v1.7.1 to v3.3.0
* jQuery UI upgrade from v1.8.22 to v1.12.0

 NOTE: Internet Explorer versions 6, 7, and 8 are not supported after the jQuery upgrade.

Procedure:

1. Modify the LegalRequest.properties file. This file is located in the following path:

[CE Parameter folder]\CE\_SYS

|  |
| --- |
|  |

1. Delete the following folder and file:

* Folder: jui folder

Path: [CEWeb.war folder]\Plugins

* File: jquery.min.js

Path: [CEWeb.war folder]\SYS\_JS\jQuery

1. Create a new folder:

* Folder: jQueryUI
* Path: [CEWeb.war folder]\Plugins

1. The old jQuery methods must be updated in order to support these methods once the jQuery upgrade is applied.

In the [CEWeb.war folder]\JS path, search for the following files –

* CE\_AddUserInfo.js
* SECU\_SecuAdmin.js
* SECU\_SecuOperator.js
* SYS\_MultiCatalog.js
* SYS\_userprofile\_OpFunc.js
* Secu\_OPFAP\_Template.js
* SecuAuthRuleTemplate\_Add.js
* Secu\_ADFAP\_Template.js
* SYS\_userprofile\_OpFunc\_Template.js
* Secu\_AuthRule\_Template.js
* Secu\_Template\_Info.js
* SECU\_SecuAdmin-Martin.js

--- Update these files with the following new version codes:

|  |  |
| --- | --- |
| **Old Version Code** | **New Version Code** |
| .attr("checked",false") | .prop('checked',false) |
| .attr("checked","checked") | .prop('checked',true) |
| .attr("checked","true") | .prop('checked',true) |
| .attr('checked','true') |
| .removeAttr('checked') | .prop('checked',false) |
| .attr('checked')=="checked" | .prop('checked') |
| .attr('checked') == 'checked' |
| .attr('checked')!="checked" | !$("").prop('checked') |
| $("input[type=checkbox].REGION[checked]") | $("input[type=checkbox].REGION") |

NOTE: The files to be modified may differ in every project.

1. The following JSP and JS files require specific changes:

* common.jsp
* Secu\_ADFAP\_Template.js
* Secu\_ OPFAP\_Template.js
* search.js
* excratetabs.css
* Personalize.js
* Plugins.js
* mainmenu.js
* quickaccess.js
* recentfunc.js
* summary.js
* welcome.js

noteNOTE:

1. After modifying the JSP files, make sure to generate the relevant Screen XML files.
2. For detailed information on the changes, refer to the related material that accompanies the CE Version 3.3.0 release.

|  |  |
| --- | --- |
| **FILE** | **UPDATE** |
| \CEWeb.war\CSBANK\US\WEB\SCRN  \Library\common.jsp |  |
|  |
| \CEWeb.war\JS\Secu\_ADFAP\_Template.js |  |
| \CEWeb.war\JS\Se cu\_ADFAP\_Templa te.js |  |
| \CEWeb.war\JS\Se cu\_OPFAP\_Templa te.js |  |
|  |
|  |
| \CEWeb.war\widg ets\ExchangeRate\ excratetabs.css |  |
| \CEWeb.war\widg ets\framework\per sonalize\Personaliz e.js |  |
| \CEWeb.war\widg ets\framework\per sonalize\Personaliz e.js |  |
| \CEWeb.war\widg ets\framework\plu gins\Plugins.js |  |
| \CEWeb.war\widg ets\MainMenu\ma inmenu.js |  |
| \CEWeb.war\widgets\CriteriaSearch\  search.js | Replace:    With: |
| \CEWeb.war\widg ets\QuickAccess\q  uickaccess.js |
| \CEWeb.war\widg ets\RecentFunctio  n\recentfunc.js |
| \CEWeb.war\widg ets\TrxSummary\s  ummary.js |
| \CEWeb.war\widg ets\WelcomeBox\  welcome.js |

Integration Issues

* UPDATING CE DIRECTLY WITH BACK OFFICE DATA

Chapter Five

* ASSIGNING A COMPANY GROUP TO A COMPANY SENT FROM EE
* MQ TOOL
* MQ Channel Authentication Configuration

Updating CE Directly with Back Office Data

CE can be directly updated with data sent by the back office system (e.g., CS Eximbills, Eximbills Enterprise): the CE user is not required to select the relevant record and release it. Specifically, both the SEC\_BUSINESS\_UNIT and SEC\_BUSINESS\_UNIT\_TEMP tables in CE are automatically updated with this data from the back office system.

In the STP transaction component of the receiving CE function, include the following setting:

stp.updateFieldValue(“L\_DUAL\_CTRL”, “F”);

With this setting, the master file is updated directly and regardless of the company profile (e.g., Authorize Transaction by Self).

Assigning a Company Group to a Company Sent from EE

The parent company or company group is manually assigned to a child company (record) that is sent by Eximbills Enterprise.

In the STP mapping rule related to the receiving function in CE, assign the actual company group name to the C\_PARENT\_ID tag via Value option in the STPs Receive Message dialog box.

|  |
| --- |
|  |

Figure 5. STPs Mapping Function: C\_PARENT\_ID

Alternatively, this assignment can be hard-coded in the XML file of the STP mapping rule:

stp.updateFieldValue(“C\_PARENT\_ID”, “CS\_Company\_Group”);

MQ Tool

The MQ Message Toolkit, also known as the MQ Tool, is an external program that allows consultants to validate the GAPI parameters set in the CE Utility. This tool directly communicates with the IBM WebSphere MQ to perform interface scenarios such as verify the GAPI Parameters for outgoing CE messages, and verifying the GAPI and STP Parameters for demerging messages into CE transaction tables. In addition, the MQ Tool also has features that can be used to encrypt or decrypt data.

Before this tool can be utilized, the following settings are required:

* The Java program must be installed in the system, and the JAVA\_HOME environment variable must be defined.
* The MQ tool must establish a connection with the WebSphere MQ.

**NOTE:** Windows 10 is used in the succeeding sample procedure.

To use the MQ Tool:

|  |
| --- |
| Do the following … |

|  |  |  |
| --- | --- | --- |
| 1. Once the Java program has been installed, the JAVA\_HOME environment variable must be defined.  Access the following path:  Control Panel > System and Security > System > Advanced System Settings |  |  |
|  |  |  |
| 2. The System Properties window is displayed.  In the Advanced tab, click on the Environment Variables button. |  |  |
|  |  |  |
| 3. Click on the New button at the upper section of the screen in the Environment Variables window to define the variable. |  |  |
|  |  |  |
| 4. Define the settings for the JAVA\_HOME environment variable. |  |  |
|  |  |  |
| 5. The tool must then establish a connection with the WebSphere MQ. To do this, run the MQ Tool batch file. |  |  |
|  |  |  |
| 6. In the MQ Message Toolkit window, specify the following MQ details:   * Host Name: Specify the name of the host machine on which the IBM WebSphere MQ is installed. * Queue Mgr: Specify the name of the Queue Manager defined in the MQ. * Send Queue: Specify the name of the Local Queue that is used to store the messages that are sent by the MQ tool. * Recv Queue: Specify the name of the Local Queue that is used to store messages that are received by the MQ tool. * Channel: Specify the channel defined in the MQ. Unless a specific channel is specified, the default channel SYSTEM.DEF.SVRCONN may be used. * CCSID: It is the Coded Character Set Id. Specify its value which is defined under the General properties of the Queue Manager. * Port: It refers to the TCP Port. Specify its value, which can be seen under the TCP properties of the Queue Manager. * MQ: Mark this option button to specify an MQ connection.   Click on the Connect button to establish connection. |  |  |
|  |  |  |
| 7. Once connected, a message is displayed confirming the connection. |  |  |
|  |  |  |
| 8. To disconnect from the MQ, click on the Disconnect button. A message is displayed confirming the disconnection. |  |  |
|  |  |  |
| The flow of sending and receiving of MQ messages may be tested with the use of the Send Message and Receive Message tabs in the MQ tool.  For checking the GAPI Parameters for Outgoing Messages, use the Receive Message tab to test if the MQ have received the GAPI Message from CE. Click the Receive button to retrieve the message and validate the GAPI Parameters. |  |  |
|  |  |  |
| For checking the STP/GAPI Parameters for Incoming Messages, use the Send Message tab to send a test message to the MQ. Enter a sample message in the text area then, click the Send button. From then, the message must already be handled by the STP/GAPI Parameters demerging them in the corresponding transaction table. |  |  |

|  |
| --- |
| To Encrypt/Decrypt Data Using the MQ Tool |

|  |  |  |
| --- | --- | --- |
| 1. In the MQ Tool window, access the Text Toolkit tab. |  |  |
|  |  |  |
| 2. Select the From Input option and specify the data to be encrypted in the text area at the upper section of the window.  **NOTE:**  The From File option may also be used to specify the source file from which to encrypt data. |  |  |
|  |  |  |
| 3. Click on the Encrypt button to encrypt the data. The result is displayed at the lower section of the window. |  |  |
|  |  |  |
| 4. To decrypt, place the data to be decrypted in the text area at the upper section of the window. |  |  |
|  |  |  |
| 5. Click on the Decrypt button (<->), beside the Encrypt button, to decrypt the data. The result is displayed at the lower section of the window. |  |  |
|  |  |  |
| 6. Data may also be encrypted or decrypted in Base64 form. Place the data to be encrypted in the text area at the upper section of the window.  **NOTE:**  Base-64 encoding is a way of taking binary data and turning it into text that are easily transmitted in HTML form data. This is used when data that are required to be stored and transmitted over media are designed to deal with textual data. This is to ensure that the data remains intact without modification during transport. |  |  |
|  |  |  |
| 7. Click on the Base64 button to encrypt the data in this form. The result is displayed at the lower section of the window. |  |  |
|  |  |  |
| 8. To decrypt data in Base64 form, place the data to be decrypted in the text area at the upper section of the window. |  |  |
|  |  |  |
| 9. Click on the Decrypt button (<->), beside the Base64 button, to decrypt the data. The result is displayed at the lower section of the window. |  |  |
|  |  |  |
| 10. Another way to encrypt or decrypt data is through Zip+Base64. To encrypt data using Zip+Base64, place the data to be encrypted in the text area at the upper section of the window. |  |  |
|  |  |  |
| 11. Click on the Zip+Base64 button to encrypt the data. The result is displayed at the lower section of the window. |  |  |
|  |  |  |
| 12. To decrypt data in Zip+Base64 form, place the data to be decrypted in the text area at the upper section of the window. |  |  |
|  |  |  |
| 13. Click on the Decrypt button (<->), beside the Zip+Base64 button, to decrypt the data. The result is displayed at the lower section of the window. |  |  |
|  |  |  |
| 14. If a user attempts to use Zip+Base64 to decrypt data with incorrect format, the Result Content section displays the error message, which indicates that the data are not in GZIP format. |  |  |

MQ Channel Authentication Configuration

If the CE system is to connect to IBM WebSphere MQ version 7.5 or higher, the local machine username and password are required to be specified when defining queues in the Queue Manager function of the CE Utility. To ensure that the system connects to WebSphere MQ properly, the following configurations must be done:

1. Create the queue manager, queue, and channels;
2. Create a local machine user;
3. Set the channel authentication records;
4. Grant the local machine user access to the queue; and
5. Configure the queues in the CE Utility.

NOTE: When connecting to lower versions of WebSphere MQ, the local machine username and password are not required.

Creating the Queue Manager, Queues, and Channels

To create the queue manager, queue, and channels in WebSphere MQ:

|  |
| --- |
| Do the following… |

|  |  |  |
| --- | --- | --- |
| 1. From the Start menu, run the IBM MQ Explorer from the IBM MQ program folder. |  |  |
|  |  |  |
| 2. In the IBM MQ Explorer window, right-click on Queue Managers, and then select the following options to create a new Queue Manager:  Queue Managers > New > Queue Manager |  |  |
|  |  |  |
| 3. The Create Queue Manager window is displayed. Specify the name and basic information of the queue manager.  Click on the Next button. |  |  |
|  |  |  |
| 4. Specify the data and log details for the queue manager.  Click on the Next button. |  |  |
|  |  |  |
| 5. Specify the queue manager settings and click on the Next button. |  |  |
|  |  |  |
| 6. Specify the listener information and click on the Next button. |  |  |
|  |  |  |
| 7. Configure the MQ Explorer settings and click on the Finish button. |  |  |
|  |  |  |
| 8. MQ proceeds to create the Queue Managetr. |  |  |
|  |  |  |
| 9. The newly created queue manager (e.g., QM\_CE) node is displayed in the Queue Managers list. Expand this node. |  |  |
|  |  |  |
| 10. Right-click on Queues, and then select the following options to create a new Local Queue:  Queues > New > Local Queue |  |  |
|  |  |  |
| 11. The New Local Queue window is displayed.  Specify the relevant details for the local queue.  Click on the Next button. |  |  |
|  |  |  |
| 12. In the Change Properties window that is displayed, specify the relevant details.  Click on the Finish button. |  |  |
|  |  |  |
| 13. A confirmation message is shown once the queue is successfully created. |  |  |
|  |  |  |
| 14. Repeat steps 10 to 13 to create the another local queue (e.g., Q\_BANK).  The new queues are displayed on the grid. |  |  |
|  |  |  |
| 15. Right-click on the Channels option to create a new server-connection channel. |  |  |
|  |  |  |
| 16. The New Server-connection Channel window is displayed.  Specify the relevant channel details.  Click on the Next button. |  |  |
|  |  |  |
| 17. In the Change Properties window, specify the relevant details and click on the Finish button. |  |  |
|  |  |  |
| 18. A confirmation message is shown once the server-connection channel is successfully created. |  |  |
|  |  |  |
| 19. The new server-connection channel is displayed on the grid.  **NOTE:**  The channel status and icon are changed to active once the channel starts to receive or send messages. |  |  |

Creating a Local Machine User

Create a local machine user that is a regular user (i.e., the user is not an administrator and is not part of any group) with a password.

NOTE: When connecting to WebSphere MQ version 7.0 or lower, the local machine user is not required to be a regular user.

|  |
| --- |
| Do the following . . . |

|  |  |  |
| --- | --- | --- |
| 1. Create a local computer user. This user must not belong to any group.  Access the following path:  Control Panel > System and Security > Administrative Tools > Computer Management |  |  |
|  |  |  |
| 2. In the Computer Management window, expand the Local Users and Groups node. Right-click on the Users folder and select the New User option from the dropdown list that is displayed. |  |  |
|  |  |  |
| 3. The New User window is shown. Specify the details of the user. Click on the Create button. |  |  |
|  |  |  |
| 4. The created user is added in the list of users. |  |  |

Setting the Channel Authentication Records

To set the channel authentication records:

|  |
| --- |
| Do the following . . . |

|  |  |  |
| --- | --- | --- |
| 1. In the IBM MQ Explorer, right-click on the corresponding server-connection channel and select Properties.  On the Left panel, click on the MCA option. Specify the ID of the created local user. Click on the Apply button, and then the OK button. |  |  |
|  |  |  |
| 2. In the IBM MQ Explorer, expand the Channels node for the relevant queue manager (e.g., QM\_CE).  Right-click on the Channel Authentication Records option and create a new Channel Authentication Record. |  |  |
|  |  |  |
| 3. The Create a Channel Authentication Record window is displayed.  Select the Allow Access rule type and click on the Next button. |  |  |
|  |  |  |
| 4. In the Match Part of the Identity window, select the Client Application User Id option and click on the Next button. |  |  |
|  |  |  |
| 5. In the Matching the Channels window, specify the created channel profile (i.e., CHANNEL1) and click on the Show Matching Channels button.  Click on the Next button. |  |  |
|  |  |  |
| 6. In the Matching a Remote Client User Id window, specify the created local machine user ID and click on the Next button. |  |  |
|  |  |  |
| 7. In the Authorization User Id window, select the Fixed User Id option. Specify the created local machine user ID and click on the Next button. |  |  |
|  |  |  |
| 8. In the Authentication window, select the As Queue Manager option and click on the Next button. |  |  |
|  |  |  |
| 9. In the Optional Attributes window, specify the rule description and click on the Next button. |  |  |
|  |  |  |
| 10. A summary of the defined settings is displayed.  Click on the Finish button. |  |  |
|  |  |  |
| 11. The created rule is shown on the Channel Authentication Records grid. |  |  |

Granting the Local Machine User Access to the Queue

To grant the local machine user with access to the queue:

|  |
| --- |
| Do the following . . . |

|  |  |  |
| --- | --- | --- |
| 1. In the IBM MQ Explorer window, right-click on the relevant queue manager (e.g., QM\_CE).]  Select the following options:  Queue Managers > Object Authorities > Add Role Based Authorities |  |  |
|  |  |  |
| 2. In the Add Role Based Authorities window that is displayed, specify the following details:   * User: specify the local machine user name. * Mark the following options: * Full administrative access * Permit reading of messages on queues   Click on the OK button. |  |  |
|  |  |  |
| 3. A confirmation message is displayed.  Click on the Yes button. |  |  |
|  |  |  |
| 4. A message confirming that the user was successfully granted authority is displayed.  Click on the OK button. |  |  |
|  |  |  |
| 5. In the IBM MQ Explorer window, right-click on the Queues node of the relevant queue manager (e.g., QM\_CE).  Select the following options:  Queues > Object Authorities > Manage Authority Records |  |  |
|  |  |  |
| 6. In the window that is displayed, select the generic profile that is to act on all queues, and access the Users tab.  Select the relevant user record and click on the Edit button. |  |  |
|  |  |  |
| 7. In the Edit Authorities window, select all options in the Authorities section and click on the OK button. |  |  |
|  |  |  |
| **NOTE:**  If there are no existing authority records in Generic Profiles, right-click on the Generic Profiles node to create a new User Authority Using New Profile record. |  |  |
|  |  |  |
| In the window that is displayed, specify the following details:   * Entity Name: [local machine user Id]@[computer name] * Profile Name: \*\* * Authorities: select all options   Click on the OK button.  The authority record becomes available for selection. |  |  |

Configuring the Queues in the CE Utility

In the Queue Manager function of the CE Utility:

|  |
| --- |
| Do the following … |

|  |  |  |
| --- | --- | --- |
| 1. Run the Queue Manager function in the System Function group.  Define the Q\_CE queue.  Specify the username and password of the local machine user that was created.  Click on the Save button. |  |  |
|  |  |  |
| 2. Define the Q\_BANK queue.  Specify the username and password of the local machine user that was created.  Click on the Save button. |  |  |
|  |  |  |
| 3. Run the XML Generator function and generate the XML files for the Queue Manager parameter.  Select the relevant queue Ids in the Queue Manager tab. |  |  |
|  |  |  |
| 4. In the generated XML files, the local machine username and password are defined. |  |  |

Database Issues

* COPYING TABLES IN DB2

Chapter Six

* ASSIGNING THE DATABASE USERS TO THE SQL SERVER LOGINS
* IMPORTING EXISTING TABLES FROM THE DATABASE
* MANUALLY STOP/START THE ORACLE DATABASE
* backing up the oracle database
* uninstalling the oracle database
* common oracle errors
* DATABASE TOOLS
* Error on Data Retrieval

Copying Tables in DB2

Run the following SQL command to copy a table in DB2:

(Start of code)

create table New\_Table like Old\_Table;

insert into New\_Table (select \* from Old\_Table);

(End of code)

Assigning the Database Users to the SQL Server Logins

To link each of the CE SQL users with the relevant login, run the following commands in the New Query tool of MS SQL:

(Start of code)

exec sp\_change\_users\_login 'Update\_One','CETRX','CETRX'

exec sp\_change\_users\_login 'Update\_One','CEUSER','CEUSER'

exec sp\_change\_users\_login 'Update\_One','CEMETA','CEMETA'

(End of code)

|  |
| --- |
|  |

Figure 6. SQL Query

Importing Existing Tables from the Database

Issue

When the import process is run from the DB Dictionary function, an error occurs and the system prompts the following message:

|  |
| --- |
|  |

Figure 6. Error Message: Importing Tables

Solution

This error occurs when the physical database is not matched with the DB Dictionary. For instance, there are tables that are deleted from the CE Utility, but still exist in the database.

These tables must be manually deleted from the database to successfully complete the import process. The Check Table option may be used to specify the differences between the physical database and the DB Dictionary, to be able to correct the differences accordingly.

Manually Stop/Start the Oracle Database

There are three ways to run the stop/start process for the Oracle service:

1. Using the Command prompt
2. Using the Services option in the Windows Control Panel
3. Using the Oracle Enterprise Manager Database Control

NOTE: The Oracle version used in the following procedures is Oracle 12c.

|  |
| --- |
| Using the Command Prompt |

|  |  |  |
| --- | --- | --- |
| 1. To start the database, specify the following commands:   * >sqlplus/nolog * SQL>connect/as sysdba * SQL>startup |  |  |
|  |  |  |
| 2. To stop the database, specify the following commands:   * >sqlplus/nolog * SQL>connect/as sysdba * SQL>shutdown immediate |  |  |

|  |
| --- |
| Using Windows Control Panel |

|  |  |  |
| --- | --- | --- |
| 1. Access the Services window through the following path:  Control Panel > System and Security > Administrative Tools > Services |  |  |
|  |  |  |
| 2. In the Services window, select the Oracle service and click on the stop, start, pause or restart options for the service. |  |  |

|  |
| --- |
| Using the Oracle Administration Assistant for Windows |

|  |  |  |
| --- | --- | --- |
| 1. From the Start menu, access the Oracle home menu from the Windows programs group. In the Configuration and Migration Tools folder, select the Administration Assistant for Windows option.  **NOTE:**  The Administration Assistant for Windows is only available for Oracle versions 12c and lower. |  |  |
|  |  |  |
| 2. In the window that is displayed, expand the nodes to view the relevant Oracle database SID. (e.g., CEDB). |  |  |
|  |  |  |
| 3. Right-click on the relevant SID and select the option to start or stop the database service. |  |  |

Backing up the Oracle Database

Data backup is essential to prevent data loss in the event of primary data failure. To ensure that the latest data are kept, the CE database may be exported into a dump file to serve as backup.

Exporting a dump file may be done through the use of a batch file containing the following codes:

set dbbakdir=[dump file directory]

exp userid=[user id]/[password]@[database SID]

file=%dbbakdir%\[dumpfile name].dmp

log= %dbbakdir%\[logfile name].log

owner=[database owner]

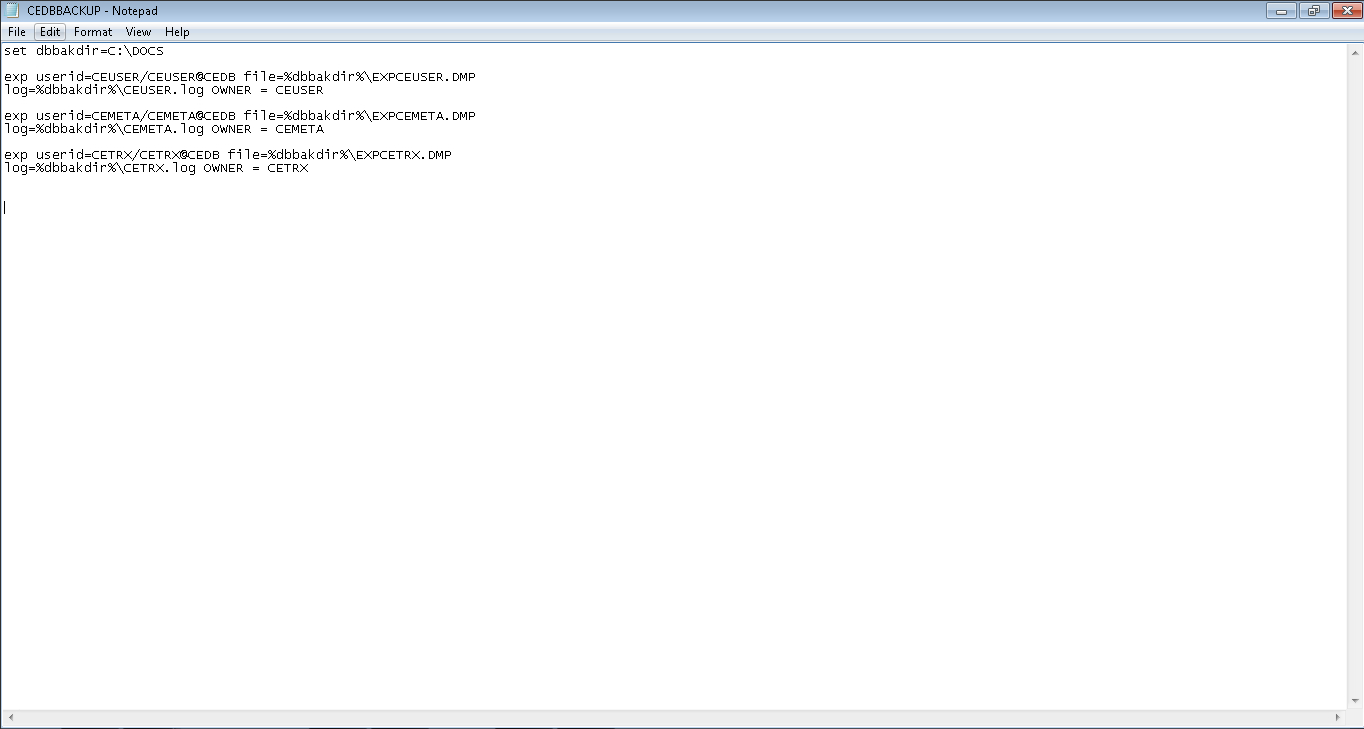


Figure 6. Sample Batch file to Backup the Database

|  |  |  |
| --- | --- | --- |
| Code |  | Description |

|  |  |  |
| --- | --- | --- |
| **set dbbakdir=[dump file directory]** |  | This sets the directory on which the backup dump files and logs are to be generated. |
|  |  |  |
| **exp userid=[user id]\[password]@[database SID]** |  | This contains the export command and the login credentials of the database system administrator. |
|  |  |  |
| **file=%dbbakdir%\[dumpfile name].dmp** |  | This contains the directory of the backup files previously set, and the name to be assigned to the dump file to be exported. |
|  |  |  |
| **log= %dbbakdir%\[logfile name].log** |  | This contains the directory previously set, and the name to be assigned to the backup log file to be generated. |
|  |  |  |
| **owner=[database owner]** |  | This specifies the database owner. |

|  |
| --- |
| To Create a Database Backup… |

|  |  |  |
| --- | --- | --- |
| 1. Run the batch file to create the backup copy of the database. |  |  |
|  |  |  |
| 2. The process of creating backup for the database is started. |  |  |
|  |  |  |
| 3. The created backup copy of the database and the corresponding logs are generated in the directory that is defined in the batch file. (e.g. C:\DOCS). |  |  |

Uninstalling the Oracle Database

NOTE:

i. For more details and examples on uninstalling Oracle database, refer to the official documentation on the Oracle site.

ii. Refer to the CE installation guides for details on completely uninstalling the CE system.

**For Oracle 10g and 12c**

|  |
| --- |
| Through Oracle Universal Installer |

|  |  |  |
| --- | --- | --- |
| 1. Uninstall all Oracle components using the Oracle Universal Installer (OUI). |  |  |
|  |  |  |
| 2. Run the Registry Editor. |  |  |
|  |  |  |
| 3. In the Registry Editor window, delete the registry entries for all Oracle products in the following registry location:  HKEY\_LOCAL\_MACHINE/Software/Oracle. |  |  |
|  |  |  |
| 4. Delete any references to Oracle services/components and key entries that starts with ‘Ora’ which are evidently related to Oracle, in the following registry location:  HKEY\_LOCAL\_MACHINE/SYSTEM/CurrentControlSet/Services  Reboot the Workstation once done. |  |  |
|  |  |  |
| 5. Delete the Oracle Base directory. |  |  |
|  |  |  |
| 6. Empty the Temp directory and the recycle bin. |  |  |

|  |
| --- |
| Through deinstall.bat |

|  |  |  |
| --- | --- | --- |
| 1. Run the deinstall.bat file as an Administrator user. This file is located on the following path:  ORACLE\_HOME]\deinstall |  |  |
|  |  |  |
| 2. The deinstaller program is initialized.  Verify the Listener name and click Enter. |  |  |
|  |  |  |
| 3. Verify the database SID and click Enter. |  |  |
|  |  |  |
| 4. The details of the details are automatically discovered by the system.  Input y and then click Enter. |  |  |
|  |  |  |
| 5. Specify the type of the database to be deleted.  Input the corresponding number and then click Enter. |  |  |
|  |  |  |
| 6. Specify the diagnostic destination location of the database.  Either accept the default value or specify the relevant location.  Click Enter. |  |  |
|  |  |  |
| 7. Specify the storage type used by the database.  Either accept the default value or specify the relevant storage type.  Click Enter. |  |  |
|  |  |  |
| 8. Specify the database directories.  Either accept the default value or specify the relevant directories.  Click Enter. |  |  |
|  |  |  |
| 9. Specify the fast recovery area location for the database.  Either accept the default value (blank) or specify the relevant location.  Click Enter. |  |  |
|  |  |  |
| 10. Specify the location of the spfile.  Either accept the default value or specify the relevant location.  Click Enter. |  |  |
|  |  |  |
| 11. A summary of the deinstallation settings is displayed.  Input y and click Enter to continue. |  |  |
|  |  |  |
| 12. A confirmation message is displayed once the deinstallation process is started.  The command window automatically closes once the deinstallation process is completed. |  |  |
|  |  |  |
| 13. The deinstaller automatically removes all directories from the ORACLE\_HOME.  However, the following steps must be performed manually:   * Manually delete the app directory from the File Explorer. * Manually remove any user account that belongs to the database. |  |  |

**For Oracle 19c**

|  |
| --- |
| Do the following… |

|  |  |  |
| --- | --- | --- |
|  |  | ***Delete Oracle Home Environment Variable*** |
|  |  |  |
| 1. Open the following folder:  Control Panel > System and Security > System > Advanced System Settings |  |  |
|  |  |  |
| 2. In the System Properties window, click on the Environment Variables option. |  |  |
|  |  |  |
| 3. The Environment Variables window is displayed. In the System variables section, double click on the Path variable. |  |  |
|  |  |  |
| 4. In the list of variables that is shown, select the Oracle Home path and click on the Delete button.  Click on the OK button once done. |  |  |
|  |  |  |
|  |  | ***Delete Oracle 19c Registries*** |
|  |  |  |
| 5. Run the Registry Editor. |  |  |
|  |  |  |
| 6. In the Registry Editor window, access the following node:  HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services  Under this node, delete the following registries:   * OracleJobScheduler[SID] * OracleOraDB19Home1MTSRecoveryService * OracleOraDB19Home1TNSListener * OracleRemExecServiceV2 * OracleService[SID] * OracleVssWriter[SID]   **NOTE:**  If there are other software from Oracle installed in the system, consult the system administrator before deleting any registries. It is also advisable to create backup copies of registries before deletion. |  |  |
|  |  |  |
| 7. Access the following node:  \HKEY\_LOCAL\_MACHINE\SOFTWARE  Under this node, delete the following registry and its contents that are associated with the Oracle database 19c software:   * Oracle   Once done, restart the machine to apply the changes made. |  |  |
|  |  |  |
|  |  | ***Delete Home Users and Groups*** |
|  |  |  |
| 8. During Oracle installation, user and group profiles were automatically created. To access these profiles, run the lusrmgr.msc file. |  |  |
|  |  |  |
| 9. In the Users folder, delete the Oracle user that was created during Oracle installation. |  |  |
|  |  |  |
| 10. In the Groups folder, delete the following Oracle groups:   * ORA\_ASMADMIN * ORA\_ASMDBA * ORA\_ASMOPER * ORA\_CLIENT\_LISTENERS * ORA\_DBA * ORA\_DBSVCACCTS * ORA\_GRID\_LISTENERS * ORA\_INSTALL * ORA\_OPER * ORA\_OraDB19Home1\_DBA * ORA\_OraDB19Home1\_OPER * ORA\_OraDB19Home1\_SYSBACKUP * ORA\_OraDB19Home1\_SYSDG * ORA\_OraDB19Home1\_SYSKM |  |  |
|  |  |  |
|  |  | ***Delete Oracle Database 19c Files and Directories*** |
|  |  |  |
| 11. Delete the following Oracle folders. These folders were created and defined during Oracle installation:   * Oracle Home * Oracle Base   **NOTE:**  Some cases when these folders cannot be deleted may be caused by a pending system restart. Thus, restart the system first before deleting. Otherwise, reboot Windows in safe mode before deleting these folders. |  |  |
|  |  |
|  |  |
|  |  |  |
| 12. Delete the Oracle folder created by the installer during installation. This folder is created inside the Program Files folder. |  |  |

Common Oracle Errors

|  |  |  |
| --- | --- | --- |
| Error |  | Description |

|  |  |  |
| --- | --- | --- |
| **ORA Error** |  | CAUSE:  This error is caused by an attempt to make a basic peer-to-peer (single protocol network) connection.  SOLUTION:   1. Verify that the database is running. 2. Perform a loopback test. 3. Check base connectivity for underlying network transport. 4. Check that the client computer has the tnsnames.ora and the sqlnet.ora files in the correct locations. 5. Test the Net8 foundation layer. |
|  |  |  |
| **ORA ErrorORA-00001: unique constraint ( constraint\_name ) violated** |  | CAUSE:  When executing an INSERT or UPDATE statement, it has created a duplicate value in a field restricted by a unique index.  SOLUTION:   1. Drop the unique constraint 2. Change the constraint to allow duplicate values 3. Modify the SQL so that a duplicate value is not created |
|  |  |  |
| **ORA-00054: resource busy and acquire with NOWAIT specified** |  | CAUSE:  When executing a Lock Table or Select For Update command with the No Wait keyword, the resource was unavailable.  SOLUTION:   1. Re-execute the statement(s). 2. Execute the command without the No Wait keyword. |
|  |  |  |
| **ORA-00060: deadlock detected while waiting for resource** |  | CAUSE:  While executing a statement, the session was deadlocked because another session had the same resource locked. The statement(s) that were executed have been rolled back.  SOLUTION:   1. Re-execute the statement(s) that were rolled back. 2. The user may execute a rollback and re-execute all statements since the last commit was executed. |
|  |  |  |
| **ORA-00200: controlfile could not be created** |  | CAUSE:  It was not possible to create the control file.  SOLUTION:  Check that there is sufficient disk space and that there are no conflicts in filenames. |
|  |  |  |
| **ORA-00312: online log string thread string: `string`** |  | CAUSE:  This message reports the filename for details of another message.  SOLUTION:  View the associated messages for the appropriate action to take. |
|  |  |  |
| **ORA-00235: control file fixed table inconsistent due to concurrent update** |  | CAUSE:  Concurrent update activity on a control file caused a query on a control file fixed table to read inconsistent information.  SOLUTION:  Re-execute the operation. |
|  |  |  |
| **ORA-00313: open failed for members of log group string of thread string** |  | CAUSE:  The online log cannot be opened, and the file cannot be found.  SOLUTION:  Check the accompanying errors and make the log available. |
|  |  |  |
| **ORA-00320: cannot read file header from log string of thread string** |  | CAUSE:  The file is not available.  SOLUTION:  Restore the log file. |
|  |  |  |
| **ORA-00321: log string of thread string, cannot update log file header** |  | CAUSE:  Cannot write to the log file.  SOLUTION:  Restore the access to the file. |
|  |  |  |
| **ORA-00396: error string required fallback to single-pass recovery** |  | CAUSE:  The indicated error caused two-pass instance or crash recovery to fail. Recovery was retried with an alternate (slower) method to avoid the error.  SOLUTION:  Correct the cause of the indicated error (also recorded) so that future instance or crash recovery can succeed with the two-pass algorithm. This usually requires making more main memory available to the recovery process. |
|  |  |  |
| **ORA-12154: TNS:could not resolve the connect identifier specified** |  | CAUSE:  When connecting to Oracle, the service name is either missing from the TNSNAMES.ORA file or is incorrectly defined.  SOLUTION:   1. Make sure that the TNSNAMES.ORA file exists and is in the proper directory. 2. Make sure that the service name that is being accessed is included in the TNSNAMES.ORA file and that it is correctly defined. 3. Make sure that there are no syntax errors in the TNSNAMES.ORA file. For example, if there are unmatched brackets in the file, the file is to be rendered unusable. |
|  |  |  |
| **ORA-06512: at stringline string** |  | CAUSE:  This error is caused by back trace message, as the stack is unwound by unhandled exceptions.  SOLUTION:   1. Fix the problem causing the exception or write an exception handler for this condition. 2. Contact the application administrator or DBA. |
|  |  |  |
| **ORA-00904: invalid column name** |  | CAUSE:  Executing an SQL statement that included an invalid or missing column name.  SOLUTION:   1. The column name must begin with a letter. 2. The column name cannot be longer than 30 characters. 3. The column name must be made up of alphanumeric characters or the special characters:  $, \_, and #. If the column name uses any other characters, it must be enclosed in double quotation marks. 4. The column name cannot be a reserved word. |
|  |  |  |
| **ORA-01722: invalid number** |  | CAUSE:  Executing an SQL statement that tried to convert a string to a number, but was unsuccessful.  SOLUTION:   1. Only numeric fields or character fields that contain numeric values can be used in arithmetic operations. Make sure that all expressions evaluate to numbers. 2. When adding or subtracting from dates, make sure that the numeric value is added or subtracted from the date. |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **ORA-00600: internal error code; arguments: [string], [string], [string], [string], [string], [string], [string], [string]** |  | CAUSE:  This is the generic internal error number for Oracle program exceptions. This indicates that a process has encountered an exceptional condition.  SOLUTION:  Report as a bug - the first argument is the internal error number |
|  |  |  |
| **ORA-00936: missing expression** |  | CAUSE:  Executing an SQL statement, but omitting a part of the syntax.  SOLUTION:  Execute the SELECT statement which includes the complete list of the columns. |
|  |  |  |
| **ORA-01034: Oracle not available** |  | CAUSE:  This error is caused by Oracle that is not started up. It is either caused by the SGA requiring more space than was allocated or the operating-system variable pointing to the instance is improperly defined.  SOLUTION:   1. Refer to accompanying messages for possible causes and correct the problem mentioned in the other messages. 2. If Oracle has been initialized, verify that Oracle was linked correctly. |
|  |  |  |
| **ORA-12545: Connect failed because target host or object does not exist** |  | CAUSE:  The address specified is not valid, or the program being connected to does not exist.  SOLUTION:   1. Ensure that the Address parameters are entered correctly. 2. Ensure that the executable for the server exists. 3. If the protocol is TCP/IP, edit the TNSNAMES.ORA file to change the host name to a numeric IP address. |
|  |  |  |
| **ORA-00942: table or view does not exist** |  | CAUSE:   1. An SQL statement is executed that references a table or view that does not exist. 2. The user does not have access to the table or view, or the table or view belongs to another schema and was not referenced by the schema name.   SOLUTION:   1. If the table or view does not exist, create the table or view. 2. Appropriate privileges must be given if the user does not have access to the table or view. 3. If the table or view belongs to another schema and was not referenced by the schema name, rewrite the SQL to include the schema name. |
|  |  |  |
| **ORA-03113: end-of-file on communication channel** |  | CAUSE:  An unexpected end-of-file on the communication channel was encountered.  SOLUTION:   1. Check for network problems and review the SQL\*Net setup. 2. Test to see whether the server process is disabled and whether a trace file was generated at failure time. |
|  |  |  |
| **ORA-06502: PL/SQL: numeric or value error** |  | CAUSE:  The executed statement resulted in an arithmetic, numeric, string, conversion, or constraint error.  SOLUTION:  Change the data, how it is manipulated, or how it is declared, so that values do not violate constraints. |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **ORA-04031: unable to allocate num bytes of shared memory num, num, num** |  | CAUSE:  The SGA memory has been exhausted.  SOLUTION:   1. Reduce the use of shared memory. 2. Increase the SHARED\_POOL\_SIZE initialization parameter in the initialization file. |
|  |  |  |
| **ORA-01756: quoted string not properly terminated** |  | CAUSE:  A quoted string is not terminated with a single quote mark.  SOLUTION:  Insert the closing quote and re-execute the statement. |
|  |  |  |
| **ORA-29283: invalid file operation** |  | CAUSE:  An attempt was made to read from a file or directory that does not exist, or the file or directory access was denied by the operating system.  SOLUTION:  Verify file and directory access privileges on the file system, and verify that the file exists. |
|  |  |  |
| **ORA-00020: maximum number of processes num exceeded** |  | CAUSE:  All process state objects are in use.  SOLUTION:   1. Re-execute the statement(s). 2. Restart Oracle. |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **ORA-12203: TNS:unable to connect to destination** |  | CAUSE:   1. An Invalid address was specified or the destination address is not listening. 2. This error can also occur because of underlying network or network transport problems.   SOLUTION:   1. Verify that the net service name entered was correct. 2. Verify that the address portion of the connect descriptor which corresponds to the net service name is correct. 3. Ensure that the destination process is running at the remote node. |
|  |  |  |
| **ORA-01017: invalid username/password; logon denied** |  | CAUSE:  Logging onto Oracle with an invalid username and password combination.  SOLUTION:  Enter a valid username and password combination in the proper format. |
|  |  |  |
| **ORA-01403: no data found** |  | CAUSE:   1. Executing a SELECT INTO statement and no rows were returned. 2. Referencing an uninitialized row in a table.   SOLUTION:  Terminate the processing of the data. |
|  |  |  |
| **ORA-01033: ORACLE initialization or shutdown in progress** |  | CAUSE:  An attempt was made to log on while Oracle is being started up or shutdown  SOLUTION:  Retry the operation. |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **ORA-01830: date format picture ends before converting entire input string** |  | CAUSE:  When entering a date value, the date entered did not match the date format  SOLUTION:  Enter a date value using the Oracle default date format DD-MMM-YYYY.  **NOTE:**  Refer to the [Setting the Date Format](#setting_date_format) discussion. |

Database Tools

There are certain debugging tools available to assist consultants to debug errors easier and faster during parameterization work in the system. The following tools may be used to view tables, fields and records in the database:

* QTODBC
* Squirrel SQL Client

Using QTODBC

The Query tool using ODBC (QTODBC) is a Universal Data Access (UDA) tool that allows the user to query ODBC data sources, author SQL scripts and queries, execute multiple SQL Scripts and return query results to a grid or free-form text. This tool is used in debugging data inconsistencies or inaccuracies, if any, in CE.

noteNOTE:

1. To be able to use this tool, the user must ensure proper connectivity to the database server.
2. The following sample procedure is based on the Oracle 10G database.

To use the QTODBC tool, the Data Source must first be set up.

|  |
| --- |
| To Create a Data Source |

|  |  |  |
| --- | --- | --- |
| 1. To use the QTODBC tool in debugging, the Data Source must first be set up. To do this, run the QTODBC program from the QTODBC home directory. |  |  |
|  |  |  |
| 2. The Select Data Source window is displayed. Access the Machine Data Source tab and click on the New button. |  |  |
|  |  |  |
| 3. In the Create New Data Source window, select the type of data source to be applied then click on the Next button. |  |  |
|  |  |  |
| 4. In the drivers list that is shown, select the driver from which the data source is to be set up. (e.g., OraDb10g\_home1).  Click on the Next button when done. |  |  |
|  |  |  |
| 5. The summary of the newly defined settings is displayed. Review the details and click on the Finish button. |  |  |
|  |  |  |
| 6. The Oracle ODBC Driver Configuration screen is displayed. The following details must be specified:   * Data Source Name: specify the data source name, which is the service name defined in Oracle (e.g., CEDB). * Description: specify a brief description for the new data source. * TNS Service Name: Specify the service name defined during database installation. * User Id: Specify a valid username to access the database. The users defined during the installation of the database are to be used.   Click on the Test Connection button when done to test the proper communication to the database. |  |  |
|  |  |  |
| 7. In the Oracle ODBC Driver Connect dialog box, specify a valid User Name and Password and click on the OK button. |  |  |
|  |  |  |
| 8. A confirmation message is displayed indicating a successful connection. Click on the Ok button. |  |  |
|  |  |  |
| 9. In the Oracle ODBC Driver Configuration window, click on the Ok button to complete the process of creating the new data source. |  |  |
|  |  |  |
| 10. The newly created data source is displayed under the Machine Data Source tab. Click on the Ok button to start using the QTODBC tool. |  |  |

|  |
| --- |
| To use the QTODBC tool |

|  |  |  |
| --- | --- | --- |
| 1. Run the QTODBC program from the QTODBC home directory. |  |  |
|  |  |  |
| 2. The Select Data Source window is displayed. In the Machine Data Source tab, select the data source to be used then click on the Ok button. |  |  |
|  |  |  |
| 3. In the Oracle ODBC Driver Connect dialog box, specify the relevant User Name and Password.  Click on the Ok button when done.  **NOTE:**  The Database access depends on the access rights of the user |  |  |
|  |  |  |
| 4. The main page of the QTODBC tool program is displayed. |  |  |
|  |  |  |
| 5. To start debugging, SQL commands may be specified in the condition text area.  Click on the Execute Query button to execute the command. Alternatively, the F5 button on the toolbar may also be used.  The result of the query is displayed at the lower section of the screen.  **NOTE:**  The QTODBC tool is not case sensitive. |  |  |
|  |  |  |
| 6. A list of the existing Tables from the database is also available at the left panel of the screen. This may also be used for inquiry by double-clicking on the relevant table to be inquired.  The result of the query is then displayed at the lower section of the screen. |  |  |

Using Squirrel SQL

The Squirrel SQL Client is a program written in Java that allows the user to view the contents of a database, issue SQL commands, and perform a number of other functions. The graphical front end is built to support JDBC-compliant databases.

noteNOTE:

i. To be able to use this tool, the user must ensure proper connectivity to the database server.

ii. The following sample procedure is based on the Oracle 10G database.

|  |
| --- |
| Do the following… |

|  |  |  |
| --- | --- | --- |
| 1. Run the squirrel-sql.bat file from the Squirrel SQL home directory. |  |  |
|  |  |  |
| 2. The Squirrel SQL Client splash screen is displayed. |  |  |
|  |  |  |
| 3. In the Squirrel main window, select the Drivers option. |  |  |
|  |  |  |
| 4. The Drivers window displays the different databases that the client supports. Squirrel detects if the JDBC driver for a given database is in the classpath; if not, an X is marked next to the driver. |  |  |
|  |  |  |
| 5. To interact with databases using Squirrel, an Alias must be created. Click on the Aliases option. |  |  |
|  |  |  |
| 6. In the Aliases option from the toolbar, select New Alias. |  |  |
|  |  |  |
| 7. The Add Alias window is displayed. Specify the following:   * Name: Specify the service name defined in Oracle (e.g., cedb). * Driver: Select the relevant database. * URL: Specify the relevant URL of the service. * User Name/Password: Specify a valid username and password to access the database. The users defined during the installation of the database are to be used.   Click on the Test button to test the connection to the database. |  |  |
|  |  |  |
| 8. In the Connect to: [database name] window, click on Connect.  A message prompts confirming a successful connection. |  |  |
|  |  |  |
| 9. Click on the Ok button in the Add Alias window. |  |  |
|  |  |  |
| 10. The Connect to: [database name] window is again displayed.  Click on the Connect button to start using the Squirrel tool. |  |  |
|  |  |  |
| 11. The Squirrel main window is displayed. The left window panel displays the different Schemas that are associated with the specified database. Double-click on a Schema. |  |  |
|  |  |  |
| 12. Double-click on the Table option. |  |  |
|  |  |  |
| 13. The tables that are included in the selected schema are displayed. Select the relevant table. |  |  |
|  |  |  |
| 14. The Content tab displays the rows that are included in the selected table. |  |  |
|  |  |  |
| 15. Click on the Row Count tab to view the row count of the selected table. |  |  |
|  |  |  |
| 16. The Primary key tab lists the primary keys of the selected table. |  |  |
|  |  |  |
| 17. To issue queries using SQL statements, access the SQL tab.  Specify the SQL statement in the text area and click on the Run SQL icon. |  |  |
|  |  |  |
| 18. The Results tab displays the result of the query. |  |  |
|  |  |  |
| 19. The result of the query may be copied to the clipboard in HTML form. To do this, right-click on the results section and select the Copy as HTML option from the popup menu that is displayed. |  |  |
|  |  |  |
| 20. The Meta Data tab displays the appropriate Java data types that map to a given database table's columns. (i.e., java.lang.String, java.math.BigDecimal) |  |  |
|  |  |  |
| 21. Right-click on the relevant table to access the popup menu that provides access to advanced script generation and graphing features. |  |  |
|  |  |  |
| 22. The Add a graph option allows the user to create [table charts](http://www.squirrelsql.org/screenshots/graph.png). |  |  |

Error on Data Retrieval

Issue

After selecting a record from the catalog, the details are not populated on the corresponding fields in the screen.

Solution

1. Modify the value of SYS\_FILTER\_DB parameter to FALSE in sys\_para.xml file.

<SYS\_FILTER\_DB attr="C">FALSE</SYS\_FILTER\_DB>

2. Restart the application server.

Report Issues

* SAVING STATIC REPORTS

Chapter Seven

Saving Static Reports

Issue

The system saves the generated static report as a file with WSSGenerateReport...xls file name instead of the report template name.

**EXAMPLE:**

WSSGenerateReport96fd7baa.xls

Solution

Add the code ?OUTPUT\_TYPE=export after the WSSGenerateReport code in all the relevant JS functions.

**EXAMPLE:**

Add the ?OUTPUT\_TYPE=export code in function \_confirm\_onclick()

(Start of code)

function \_confirm\_onclick(){

try {

var C\_REPORT\_TNAME = getFldValue("C\_REPORT\_TNAME");

if (C\_REPORT\_TNAME == 'ConfUnconfLCAdvisedDuringPeriod' || C\_REPORT\_TNAME == 'ExportLCAdvisedDuringPeriod' || C\_REPORT\_TNAME == 'ExportLCCancelledDuringPeriod' || C\_REPORT\_TNAME == 'ExportLCsAmendedDuringPeriod' || C\_REPORT\_TNAME == 'ExportLCTransactionsDuringPeriod' || C\_REPORT\_TNAME == 'ExportBillsDiscountedDuringPeriod' || C\_REPORT\_TNAME == 'ExportDocumentsMaturityDuringPeriod' || C\_REPORT\_TNAME == 'ExportDocsUnpaidOnMaturityDuringPeriod' || C\_REPORT\_TNAME == 'ExportLCDocsDiscountedDuringPeriod' || C\_REPORT\_TNAME == 'OutstandingEXLCAtMaturityDuringPeriod'){

if (!checkNull("ISSUE\_DT")) return false;

if (!checkNull("EXPIRY\_DT")) return false;

}

var vForm = document.MAINFORM;

vForm.action = "../servlets/WSSGenerateReport**?OUTPUT\_TYPE=export**";

vForm.method = "POST";

vForm.target = "\_blank";

vForm.submit();

vForm.target = "";

}

catch (e) {

showExcpt("EXLC\_Reports", e);

}

}

(End of code)

Special Features

* SMS notification Feature

Chapter Eight

SMS Notification Feature

The CE system provides the facility to send notifications to concerned recipients via SMS in order to advise them of business transactions such as purchase orders. To generate these notifications, templates and rules for automatic sending are used.

To fully set up the system to generate SMS notifications, the following must be configured:

* JSP and JS files;
* SMS template settings; and
* Functions settings.

For the complete information on the configuration of this feature, refer to the *CE Supplementary Functions* manual.

NOTE: Generation of the SMS content is done in CE. However, the actual settings and processes to interface CE with an SMS service center are project-specific.

Additional Information

* CE VERSION

Chapter Nine

* MANUALLY MAINTAINED PARAMETERS
* PROJECT-CONTROLLED PARAMETERS
* FREQUENTLY USED SQL STATEMENTS
* MAPPING THE PARAMETER AND WEB SERVER DRIVES
* CE LOG FILES
* SQL PARSER
* MICROSOFT SCRIPT EDITOR
* ULTRA EDIT TEXT EDITOR
* MACROMEDIA DREAMWEAVER
* veracode scanning
* THE CE NEW UTILITY

CE Version

There are two ways to determine the CE version used.

1. Run the CE Utility, open the Help menu, and click on the About Customer Enterprise option.

|  |
| --- |
|  |

The About Customer Enterprise window is displayed, showing the details of the CE Utility version.

|  |
| --- |
|  |

NOTE: The CE system version indicated in the CE Utility is the same as the CE core version.

1. Check the Version node in this file: [CEWeb.war folder]\ceversion\CEAbout.xml

|  |
| --- |
|  |

Manually Maintained Parameters

The following parameters, found in the parameter directory, cannot be generated by the CE Utility. These have to be manually maintained.

* [CE Parameter Folder]\CE\_SYS\BIRT.properties
* [CE Parameter Folder]\CE\_SYS\characterfilter.xml
* [CE Parameter Folder]\CE\_SYS\criteria\_mapping.xml
* [CE Parameter Folder]\CE\_SYS\Cache\_Config.xml
* [CE Parameter Folder]\CE\_SYS\CE\_Log\_Config.xml
* [CE Parameter Folder]\CE\_SYS\CustomPolicy.xml
* [CE Parameter Folder]\CE\_SYS\ejbserver.properties
* [CE Parameter Folder]\CE\_SYS\LegalRequest.properties
* [CE Parameter Folder]\CE\_SYS\password\_config.xml
* [CE Parameter Folder]\CE\_SYS\password\_verification.xml
* [CE Parameter Folder]\CE\_SYS\refreshSetting.xml
* [CE Parameter Folder]\CE\_SYS\release\_ext\_mapping.xml
* [CE Parameter Folder]\CE\_SYS\RPTEngine.xml
* [CE Parameter Folder]\CE\_SYS\RSAKEY.properties
* [CE Parameter Folder]\CE\_SYS\sys.char.fld.xml
* [CE Parameter Folder]\CE\_SYS\sys.char.list.xml
* [CE Parameter Folder]\CE\_SYS\SBRFieldMapping.xml
* [CE Parameter Folder]\CE\_SYS\SYS\_PdtMultiAuthFuncMapping.xml
* [CE Parameter Folder]\CE\_SYS\SYST\criteria.char.xml
* [CE Parameter Folder]\CE\_SYS\SYST\sys\_batch.xml
* [CE Parameter Folder]\CE\_SYS\SYST\sys\_cache\_data.xml
* [CE Parameter Folder]\CE\_SYS\SYST\sys\_extend\_session\_fields.xml
* [CE Parameter Folder]\CE\_SYS\SYST\sys\_external\_session\_obj.xml
* [CE Parameter Folder]\CE\_SYS\SYST\sys\_screens.xml
* [CE Parameter Folder]\CE\_SYS\WIDGETS\framework\cs-iwidget.xml
* [CE Parameter Folder]\CE\_SYS\EXPORT folder
* [CE Parameter Folder]\CE\_SYS\FTL folder

Project-Controlled Parameters

Most parameter files are controlled and maintained in the Nanjing Development Center.

Only the following parameters can be modified and controlled by the project team:

* [CE Utility Folder]\DataDic.xml
* [CE Utility Folder]\UserInfo.xml
* [CE Utility Folder]\ce\_params\Script\_XML\product\_item\_prar.xml

Frequently Used SQL Statements

The following are the frequently used SQL statements.

1. Select Statement
   * To display all the details included in a specific table:

Select \* from [Schema].[Table Name] where [Field Name] = “[Field Value]”

* + To filter the fields that are to be displayed:

Select [Field Name 1],Field Name 2],[Field Name 3] from [Schema].[Table Name] where [Field Name] = “[Field Value]”

* + To return all records in a table but display only field 1 and field 2:

Select [Field Name 1],[Field Name 2] from [Schema].[Table Name]

* + To return field 1 for all records in a table with a specific value for field 2:

Select [Field Name 1] from [Schema].[Table Name] where [Field Name 2] = “[Field Value]”

* + To return all records in a table where field 1 is one of the three possible values:

Select \* from [Schema].[Table Name] where [Field Name 1] in [value1,value2,value3]

* + To return the number of records in a table:

Select count (\*) from [Schema].[Table Name]

* + To return the number of records in a table with a specific value for field 1:

Select count (\*) from [Schema].[Table Name] where [Field Name 1] = “[Field value]”

* + To select all unique values for field 1 from a table together with the number of records with that unique value:

Select [Field Name 1], count (\*) from [Schema].[Table Name] group field 1

* + To select all unique values for combinations of field 1 and field 2 from a table, together with the number of records with that combination:

Select [Field Name 1], [Field Name 2], count (\*) from [Schema].[Table Name] group by [Field Name 1], [Field Name 2]

* + To select the number of unique values:

Select count (distinct field1) from [Schema].[Table Name]

* + To select all duplicate records in a table, where two (or more) records are considered duplicates if they share a common value for a single field:

Select field, count (field) from [Schema].[Table Name] group by field having count (\*) > 1

* + To select all duplicate records in a table, where two (or more) records are considered duplicates if they share common values for a pair of fields:

Select [Field Name 1], [Field Name 2], count (\*) from [Schema].[Table Name] group by [Field Name 1], [Field Name 2] having count (\*) > 1

1. Update Statement
   * To modify the tables

Update [Schema].[Table Name] set [Field Name] = “[Current field value]” where [Field Name] = ‘[New field value]’

* + To update a particular field from all records in a table:

Update [Schema].[Table Name] set [Field Name 1]= ’[Field Value]’

* + To update specific records in a table in reference to a field value:

Update [Schema].[Table Name] set [Field Name 1]=’[Field Value 2]’ where [Field Name 1]=’[Field Value 1]’

* + To update more than one field at the same time:

Update [Schema].[Table Name] set [Field Name 1]=’[Field Value 1]’, [Field Name 2]=’[Field Value 2]’

1. Delete Statement
   * To delete records

Delete from [Schema].[Table Name] where [Field Name] = “[Field Value]”

1. Create Statement

* To Create an Event Master table

CREATE TABLE CETRX.[Module Name]\_EM\_ISSUE (ADD\_AMT\_COVR VARCHAR2(146),APPL\_ID VARCHAR2(20),APPL\_NM VARCHAR2(35),AVAL\_BY VARCHAR2(128),BENE\_ADD1 VARCHAR2(35),BENE\_ADD2 VARCHAR2(35),BENE\_ADD3 VARCHAR2(35),BENE\_BK\_ADD1 VARCHAR2(35),BENE\_BK\_ADD2 VARCHAR2(35),BENE\_BK\_ADD3 VARCHAR2(35),BENE\_BK\_ID VARCHAR2(16),BENE\_BK\_NM VARCHAR2(35),BENE\_BK\_SW\_ADD VARCHAR2(11),BENE\_ID VARCHAR2(20),BENE\_NM VARCHAR2(35),BK\_MAIN\_REF VARCHAR2(16),CURRNT\_STATUS VARCHAR2(35),CUST\_NO VARCHAR2(16),C\_BK\_GROUP\_ID VARCHAR2(12),C\_CP\_UNIT\_CODE VARCHAR2(12),C\_CUST\_ID VARCHAR2(12),C\_EVENT\_NAME VARCHAR2(20),C\_EVENT\_STATUS VARCHAR2(4),C\_FUNC\_SHORT\_NAME VARCHAR2(16),C\_LOCKED\_BU VARCHAR2(32),C\_LOCKED\_FLAG VARCHAR2(1),C\_LOCKED\_OP VARCHAR2(32),C\_LOCK\_FUNC\_ID VARCHAR2(12),C\_MAIN\_REF VARCHAR2(20) NOT NULL,C\_NOW\_EVENT VARCHAR2(20),C\_PRODUCT\_ID VARCHAR2(16),C\_PRODUCT\_NAME VARCHAR2(30),C\_RELE\_BY VARCHAR2(16),C\_TRX\_REF VARCHAR2(24) NOT NULL,C\_TRX\_STATUS VARCHAR2(4),C\_UNIT\_CODE VARCHAR2(12) NOT NULL,DOC\_PRES CLOB,DRAFTS\_AT VARCHAR2(109),D\_CREA\_DATE DATE,D\_RELE\_DATE DATE,D\_SYS\_OP\_DATE DATE,D\_SYS\_REL\_DATE DATE,EXPIRY\_DT DATE,EXPIRY\_PLC VARCHAR2(35),FORM\_OF\_LC VARCHAR2(128),GOODS\_DESC CLOB,I\_CANCEL\_ETIMES INTEGER,I\_EVENT\_TIMES INTEGER,LC\_AMT DECIMAL(18,3),LC\_CCY VARCHAR2(3),LC\_NO VARCHAR2(16),LTST\_SHIP\_DT DATE,MIX\_PMT\_DETL VARCHAR2(146),NEG\_TOL INT,NXT\_STATUS VARCHAR2(35),PARENT\_MAIN\_REF VARCHAR2(20),PARTIAL\_SHIP VARCHAR2(128),POS\_TOL INT,SHIP\_PRD VARCHAR2(400),TENOR\_DAYS INTEGER,TENOR\_TYPE VARCHAR2(128),TNSHIP VARCHAR2(128),TOL\_SPEC VARCHAR2(128),TRX\_DT DATE,TTL\_EXPOE DECIMAL(18,3),T\_LOCKED\_TIME DATE,T\_RELE\_TIME DATE,T\_SYS\_OP\_TIME DATE,T\_SYS\_REL\_TIME DATE, PRIMARY KEY (C\_MAIN\_REF,C\_UNIT\_CODE,C\_TRX\_REF))

* To create a Master table

CREATE TABLE CETRX.[Module Name]\_MASTER (AVAL\_BY VARCHAR2(128),BK\_MAIN\_REF VARCHAR2(16),CURRNT\_STATUS VARCHAR2(35),C\_AUTH\_BU VARCHAR2(20),C\_BK\_GROUP\_ID VARCHAR2(12),C\_CP\_UNIT\_CODE VARCHAR2(12),C\_CUST\_ID VARCHAR2(12),C\_EVENT\_NAME VARCHAR2(20),C\_FUNC\_SHORT\_NAME VARCHAR2(16),C\_LOCKED\_BU VARCHAR2(32),C\_LOCKED\_FLAG VARCHAR2(1),C\_LOCKED\_OP VARCHAR2(32),C\_LOCK\_FUNC\_ID VARCHAR2(12),C\_MAIN\_REF VARCHAR2(20) NOT NULL,C\_NOW\_EVENT VARCHAR2(20),C\_PRODUCT\_ID VARCHAR2(16),C\_PRODUCT\_NAME VARCHAR2(30),C\_RELE\_BY VARCHAR2(16),C\_TRX\_REF VARCHAR2(24),C\_TRX\_STATUS VARCHAR2(4),C\_UNIT\_CODE VARCHAR2(12) NOT NULL,D\_CREA\_DATE DATE,D\_RELE\_DATE DATE,D\_SYS\_OP\_DATE DATE,D\_SYS\_REL\_DATE DATE,I\_CANCEL\_ETIMES INTEGER,I\_EVENT\_TIMES INTEGER,LC\_AMT DECIMAL(18,3),LC\_CCY VARCHAR2(3),LC\_NO VARCHAR2(16),NXT\_STATUS VARCHAR2(35),PARENT\_MAIN\_REF VARCHAR2(20),SYS\_IMG\_TYPE VARCHAR2(128),TRX\_DT DATE,T\_LOCKED\_TIME DATE,T\_SYS\_OP\_TIME DATE,T\_SYS\_REL\_TIME DATE, PRIMARY KEY (C\_MAIN\_REF,C\_UNIT\_CODE))

* To create an Auth table

CREATE TABLE CETRX.[Module Name]\_AUTH (C\_AUTH\_STAT VARCHAR2(4),C\_AUTH\_USER VARCHAR2(16),C\_MAIN\_REF VARCHAR2(20),C\_PRODUCT\_NAME VARCHAR2(30),C\_SIGN\_DATA VARCHAR2(256),C\_STAT\_INFO VARCHAR2(256),C\_TRX\_DATA CLOB,C\_TRX\_REF VARCHAR2(24),C\_UNIT\_CODE VARCHAR2(12),D\_AUTH\_DATE DATE,I\_AUTH\_LEVEL NUMERIC(22,0),I\_EVENT\_TIMES INTEGER,T\_AUTH\_TIME DATE)

* To create a Ledger table

CREATE TABLE CETRX.[Module Name]\_LEDGER (CURRNT\_STATUS VARCHAR2(35),C\_AUTH\_BU VARCHAR2(20),C\_AUTH\_STATUS VARCHAR2(10),C\_BK\_GROUP\_ID VARCHAR2(12),C\_CP\_UNIT\_CODE VARCHAR2(12),C\_CREA\_BY VARCHAR2(16),C\_CUST\_ID VARCHAR2(12),C\_DATA\_CHANGED CLOB,C\_EVENT\_DESC VARCHAR2(40),C\_EVENT\_NAME VARCHAR2(20),C\_EVENT\_STATUS VARCHAR2(4),C\_FUNC\_ID VARCHAR2(12),C\_FUNC\_SHORT\_NAME VARCHAR2(16),C\_LOCKED\_BU VARCHAR2(32),C\_LOCKED\_FLAG VARCHAR2(1),C\_LOCKED\_OP VARCHAR2(32),C\_LOCK\_FUNC\_ID VARCHAR2(12),C\_MAIN\_REF VARCHAR2(20) NOT NULL,C\_OUTPUT\_INFO VARCHAR2(20),C\_PRODUCT\_ID VARCHAR2(16),C\_PRODUCT\_NAME VARCHAR2(30),C\_REFUSE\_REASON CLOB,C\_RELE\_BY VARCHAR2(16),C\_TEMP\_DATA CLOB,C\_TRX\_REF VARCHAR2(24),C\_UNIT\_CODE VARCHAR2(12) NOT NULL,D\_CREA\_DATE DATE,D\_RELE\_DATE DATE,D\_SYS\_OP\_DATE DATE,D\_SYS\_REL\_DATE DATE,I\_EVENT\_TIMES INTEGER NOT NULL,I\_PREV\_ETIMES INTEGER,LC\_NO VARCHAR2(16),NXT\_STATUS VARCHAR2(35),PARENT\_MAIN\_REF VARCHAR2(20),T\_CREA\_TIME DATE,T\_LOCKED\_TIME DATE,T\_RELE\_TIME DATE,T\_SYS\_OP\_TIME DATE,T\_SYS\_REL\_TIME DATE, PRIMARY KEY (C\_MAIN\_REF,C\_UNIT\_CODE,I\_EVENT\_TIMES))

Mapping the Parameter and Web Server Drives

The two main folders of a CE environment are the directories that contain the CE business parameters in XML format and the original JS or JSP; these pertain to the parameter server and web server, respectively. Both must be mapped to specific drives.

The parameter and web servers are mapped to ensure that any parameter definition or modification created in the CE Utility is saved and reflected in the system, and that the generated XML files are stored in proper directories.

* GEN\_XML\_ROOTPATH
* GEN\_WEB\_ROOTPATH

|  |
| --- |
|  |

Figure 9. System Parameter Settings in the CE Utility

The parameter and web servers may be mapped to any available drive of the host machine, for example, the o: and the p: drives.

|  |
| --- |
|  |

Figure 9. Sample Mapped Parameter Server and Web Server Drives

Server-Side Mapping

There are two ways by which the drives may be mapped. Depending on the environment setup (i.e., the number of users involved), any of these types of mapping may be utilized:

* **Network Mapping** – This is done when multiple users access the mapped drives over a network. The parameter and web servers must be shared before these are mapped as O: and P: drives, for example. When sharing these servers, define the Permission settings to ensure that proper access rights are given to the users.

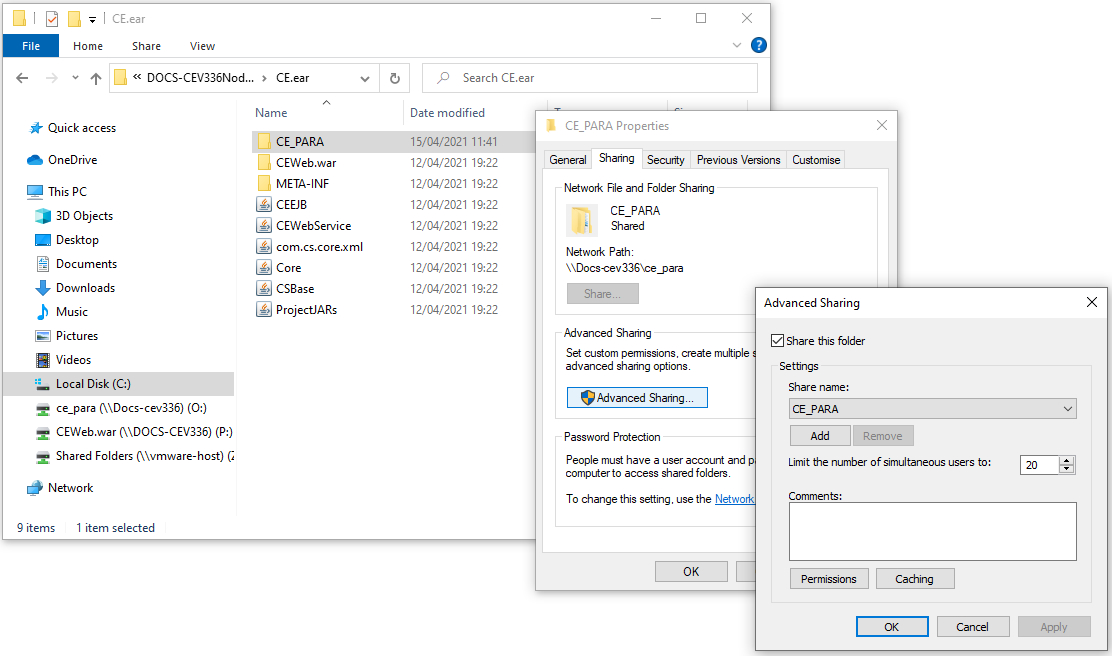


Figure 9. Permission Setting

* **Local Mapping** –This is done when parameter and web servers are not accessed over a network. To create this kind of mapping, the following codes may be executed through a batch file.

|  |
| --- |
| (Start of code)  subst O: /d  subst P: /d  subst O: C:"\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\installedApps\DOCS-CEV330Node01Cell\CE.ear\CE\_PARA"  subst P: C:"\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\installedApps\ DOCS-CEV330Node01Cell\CE.ear\CEWeb.war"  (End of code) |

Client-Side Mapping

To map the shared CE parameters and CEWeb.war folders to the system parameter-defined drives. (e.g. O:\ and P:\) of the client workstation, the following procedure must be performed.

|  |
| --- |
| To Configure The CE Utility on The Client-side |

|  |  |  |
| --- | --- | --- |
| 1. To identify the system parameter-defined drives to be mapped with the CE Parameters and CEWeb.war folders, log on the CE Utility as an administrator user. |  |  |
|  |  |  |
| 2. Select the System Parameter function of the Parameter Manage group. |  |  |
|  |  |  |
| 3. The parameters GEN\_WEB\_ROOTPATH and GEN\_XML\_ROOTPATH display the parameter values of the system-parameter drives to be mapped. |  |  |
|  |  |  |
| 4. Locate the shared CE Parameters and CEWeb.war folders from the network. These folders are to be mapped to the client workstation. |  | 14 |
|  |  |  |
| 5. To map the required folders to the client workstation, select the Map Network Drive option from the Windows Explorer menu bar. |  |  |
|  |  |  |
| 6. In the Map Network Drive window, map the CE Parameters and to the GEN\_XML\_ROOTPATH drive that is indicated in the Systems Parameter function. (E.g. O:\)  Use the browse button to specify the folder to be mapped and click on the OK button once done. |  |  |
|  |  |  |
| 7. In the Map Network drive window, click on the Finish button to finalize the mapping. |  |  |
|  |  |  |
| 8. Map the CEWeb.war folder to the GEN\_WEB\_ROOTPATH drive that is indicated in the Systems Parameter function. (E.g. P:\)  Click on the Finish button when done. |  |  |
|  |  |  |
| 9. The mapped folders are displayed in the drive index of the Windows Explorer. |  |  |

CE Log Files

Logs aid the user to narrow down the scope and trace the source of any errors that occur in the system. For this purpose, CE provides the following log types and functionalities:

* DAOLOG
* TIMELOG
* CHARFILTERLOG
* RPTLOG
* CELOG
* STPLOG
* CE Utility Log

noteNOTE: For detailed information on CE logs, refer to the *CE Log Settings* documentation.

SQL Parser

The SQL Parser is a tool that interprets SQL statements and is most commonly used to understand log files. This tool does not require installation and can be directly used once run. JAVA\_HOME environment variable is required to be defined in the machine.

To use the SQL Parser tool:

|  |
| --- |
| Do the following … |

|  |  |  |
| --- | --- | --- |
| 1. Run the SQLParser batch file from the SQL Parser home directory. |  |  |
|  |  |  |
| 2. The SQL Parser window is displayed.  In the upper section of the window, specify the SQL statement that requires interpretation. Click on the Parse button when done. |  |  |
|  |  |  |
| 3. The lower section of the window shows the parsed data. |  |  |

Microsoft® Script Editor

The Microsoft® Script Editor has a built-in JavaScript debugger that allows JavaScript debugging of web-based applications. Thus, this tool may be used to easily troubleshoot errors that are encountered in the CE system.

|  |
| --- |
| Setting up the Microsoft® Script Editor |

|  |  |  |
| --- | --- | --- |
| 1. Run the Microsoft® Script Debugger executable file. |  |  |
|  |  |  |
| 2. To proceed with the installation, click on the Run button in the dialog box that is displayed. |  |  |
|  |  |  |
| 3. A confirmation message is displayed for the installation of Microsoft® Script Debugger. Click on the Yes button. |  |  |
|  |  |  |
| 4. Accept the License Agreement terms by clicking on the Yes button. |  |  |
|  |  |  |
| 5. In the Script Debugger Setup window that is displayed, specify the directory from which the script debugger is to be installed by using the Browse button. Click on the OK button when done. |  |  |
|  |  |  |
| 6. The installation process is initialized. |  |  |
|  |  |  |
| 7. A message is displayed confirming the successful installation.  **NOTE:**  Restart the computer or server once the installation is completed. |  |  |
|  |  |  |
| 8. The Internet Explorer settings must first be configured before the Microsoft® Script Debugger can be utilized.  To do this, run the Internet Explorer program. In the View Menu, check if the Script Debugger command is displayed. |  |  |
|  |  |  |
| 9. In the Tools Menu, select Internet Options. |  |  |
|  |  |  |
| 10. In the Advanced tab of the Internet Options window that is displayed, make sure that the Disable Script Debugging (Internet Explorer) option under the Browsing section is not marked.  Click on the OK button. |  |  |

|  |
| --- |
| Using the Microsoft® Script Editor in CE |

|  |  |  |
| --- | --- | --- |
| 1. When using Customer Enterprise at runtime, an error message is displayed when an error in the script is encountered. Click on the Yes button. |  |  |
|  |  |  |
| 2. The Microsoft Script Debugger program is then displayed and the user can start the debugging process. |  |  |

Ultra Edit Text Editor

UltraEdit™ is a Microsoft Windows editor for data and programming. This tool is particularly useful for handling large data files and column-oriented data entry operations, and aids users to easily view and edit codes and variables.

In Customer Enterprise, the business requirements are coded in JavaScript and saved as JS Script file (JS file). After successfully installing a text editor, this can be used to open and easily edit the codes in a JS file.

|  |
| --- |
| To Install the Ultra Edit Text Editor |

|  |  |  |
| --- | --- | --- |
| 1. Run the UltraEdit™ installer. |  |  |
|  |  |  |
| 2. A dialog box is displayed to verify if the system is to proceed with the installation. Click on the Run button. |  |  |
|  |  |  |
| 3. The Installation process is initialized. |  |  |
|  |  |  |
| 4. To proceed with the installation of the tool, click on the Next button. |  |  |
|  |  |  |
| 5. Accept the license agreement terms by clicking on the ‘I accept the agreement’ option.  Click on the Next button when done. |  |  |
|  |  |  |
| 6. Select the Installation option and click on the Next button. |  |  |
|  |  |  |
| 7. Select the Setup Type and click on the Next button. |  |  |
|  |  |  |
| 8. Once the settings have been defined, click on the Install button to begin the installation. |  |  |
|  |  |  |
| 9. The installation process is started. |  |  |
|  |  |  |
| 10. A message is displayed confirming a successful installation. |  |  |

|  |
| --- |
| To Utilize the Ultra Edit Text Editor |

|  |  |  |
| --- | --- | --- |
| 1. In Ultra Edit, colors are used to highlight document objects, elements, functions, and comments. In this way, mistyped keywords, keywords accidentally used as an identifier, or a string that has not been terminated, can easily be identified. |  |  |
|  |  |  |
| 2. In the UltraEdit™ screen, line numbers are displayed on the left margin. Through this, every line is easily located and referred to. |  |  |
|  |  |  |
| 3. Viewing multiple files at the same time is more efficient than starting a new editor to view another file. In this tool, multiple files can be viewed at the same time. |  |  |
|  |  |  |
| 4. Auto-indentation saves time for every line written. It also improves the consistency of the code. |  |  |
|  |  |  |
| 5. When the cursor is positioned on either of the curly bracket or parenthesis, the corresponding pair is highlighted. This saves a lot of time in debugging syntax errors. |  |  |

Macromedia® Dreamweaver®

Macromedia® Dreamweaver® from Adobe is the industry-leading, license-required web development tool which allows users to efficiently design, develop and maintain standard-based websites and applications. It provides a powerful combination of visual layout tools, application development features, and code editing support. Once the Macromedia® Dreamweaver® is installed, it can then be used to modify JS,JSP or CSS files.

|  |
| --- |
| To Install Dreamweaver |

|  |  |  |
| --- | --- | --- |
| 1. Run the Macromedia® Dreamweaver® installer. |  |  |
|  |  |  |
| 2. A dialog box is displayed to verify if the system is to proceed with the installation. Click on the Run button. |  |  |
|  |  |  |
| 3. The installation process is initialized. |  |  |
|  |  |  |
| 4. To proceed with the installation, click on the Next button. |  |  |
|  |  |  |
| 5. Accept the license terms agreement and click on the Next button. |  |  |
|  |  |  |
| 6. The path to which Dreamweaver is to be installed is displayed. Use the Change button to select a different path.  Click on the Next button when done. |  |  |
|  |  |  |
| 7. Select the file type from which Dreamweaver is to be set as the default editor.  Click on the Next button. |  |  |
|  |  |  |
| 8. Click on the Install button to begin the installation. |  |  |
|  |  |  |
| 9. A message is displayed confirming a successful installation. Click on the Finish button. |  |  |

|  |
| --- |
| To Use Dreamweaver |

|  |  |  |
| --- | --- | --- |
| 1. Access the JS, JSP, or CSS file to be modified by using Dreamweaver. |  |  |
|  |  |  |
| 2. In the Dreamweaver screen, code modification is simplified by the use of line numbers and font colors. |  |  |

Veracode Scanning

To ensure the overall security quality of the Customer Enterprise (CE) application, the system is thoroughly validated by Veracode, an application security company that provides cloud-based service for securing web applications such as CE. Both static and dynamic automated scans are performed to identify security issues. Any issue detected is addressed and closed until the system achieves a score of 100 on both the static and dynamic analyses, which marks the completion of the Veracode scanning.

The CE New Utility

The Customer Enterprise New Utility, or CE New Utility , is the web-based parameter-setting tool of the Customer Enterprise (CE) system. It is used to define the parameters and rules that are required to carry out business operations and system tasks in CE. These parameters and rules are defined according to specific business requirements.

Run on Apache Tomcat, CE NU serves as an independent and standalone environment that is not linked to any networks. Multiple CE NU environments can be set up for application development: each environment may be utilized for a specific function, area, or phase in the project. The parameters defined on multiple environments may be committed to, or merged with, the main CE server through a third-party version control software.

NOTE:

i. For details on the directory structures of the CE New Utility and CE Utility, refer to the following discussion in Chapter 3: [CE Utility and CE New Utility Directory Structures](#CE_Directory_Structures_CH3)

ii. Refer to the CE New Utility documentation set for more details on using the CE New Utility.

Glossary

Glossary

Glossary

a

|  |  |
| --- | --- |
| ***Administrator*** | The type of CE Utility user with rights to create Administrator and Operator users and define transaction function parameters such as GAPI rules, SWIFT settings, and accounting rules. |
|  |  |
| ***Application Server*** | Server where the CE system is actually deployed and where CE processes the business logic and parameter operations. |
|  |  |
| ***Attribute*** | Pertains to a Business component. Its specific settings regulate the behavior of functions in processing transactions. Unlike the main program (Control component), which provides the general purpose of the function, attributes define the actual actions to be performed by the function. A function may use several attributes; this group of one or more attributes is called an Attribute rule. |
|  |  |
| ***Authorization Rule*** | A group of settings that determine when a transaction is considered authorized. |

B

|  |  |
| --- | --- |
| ***Bank-Country Group*** | The combination or grouping of bank groups and countries that is created to facilitate the sharing of parameters and authorization rules among business units that utilize similar rules and parameters for business transaction processing. |

C

|  |  |  |
| --- | --- | --- |
| ***Calculation*** | The set of calculations and methods that determine the behavior of the system. | |
|  |  | |
| ***Catalog*** | A set of criteria that is used for filtering the records to be retrieved into, and subsequently processed in, a transaction function. | |
|  |  | |
| ***CE New Utility*** | The web-based parameter setting cool of the CE system. Refer to the CE New Utility set of manuals. | |
|  |  | |
| ***CE Utility*** | Short for Customer Enterprise Utility Workbench. This is the main tool for building parameters in CE. | |
|  |  | |
| ***CEWeb.war*** | | The folder that contains the business parameter files which are in the original format as JS or JSP. |
|  |  | |
| ***CE\_PARA*** | The folder that contains the business parameter files which are in an XML file format. | |
|  |  | |
| ***Client*** | The CE system’s user browser interface. | |
|  |  | |
| ***Customer Enterprise (CE)*** | The Eximbills business-to-business (B2B) solution that provides bank customers with a convenient and secure single window for processing and inquiring on all their trade finance, open account, and payments transactions. | |
|  |  | |
| ***Customer Enterprise Utility Workbench*** | See CE Utility. | |

D

|  |  |
| --- | --- |
| ***Database*** | An application used for the management, storage, and retrieval of data. |
|  |  |
| ***Data Source*** | A source of digitized data (e.g., database, connection to a database). |
|  |  |

E

|  |  |
| --- | --- |
| ***Eximbills*** | The flagship product of China Systems, which provides support for advanced e-commerce, open account, and trade-related services. |
|  |  |
| ***Eximbills Enterprise (EE)*** | The Eximbills integrated system that runs on Java EE. It automates and audits the complete cycle of Trade Finance, Open Account, and Payments transactions, in real time and in accordance with SWIFT, UCP, and ISO20022 standards. |

F

|  |  |
| --- | --- |
| ***FAP*** | Stands for Function Assignment Permission, the functions to which an entity – a company or user – has access rights to. |
|  |  |
| ***Field*** | The smallest unit that can hold data. |
|  |  |
| ***Function*** | A unified set of elements, operations, and configurations that produce a target setting, process, and/or output. This typically refers to a CE Utility function or a transaction function. |
|  |  |
| ***Function Group*** | A grouping of CE browser-side functions that is organized based on the common usage and purpose of the functions. A function group is set up in the CE Utility. |

G

|  |  |
| --- | --- |
| ***GEN\_WEB\_ROOTPATH*** | The system parameter that is used to define the path of the WEB parameter files. It is a Utility Workbench type of system parameter, which is defined through the System Parameter function in the CE Utility. |
|  |  |
| ***GEN\_XML\_ROOTPATH*** | The system parameter that is used to define the path of the XML parameter files. It is a Utility Workbench type of system parameter, which is defined through the System Parameter function in the CE Utility. |

H

|  |  |
| --- | --- |
| ***HTML*** | Stands for Hyper Text Markup Language; the main markup language for web pages, HTML elements are the basic building-blocks of web pages. It consists of tags enclosed in angle brackets within the web page content, and has start tags (opening tags) and end tags (closing tags). In between these tags are text, tags, comments and other types of text-based content. |

I

|  |  |
| --- | --- |
| ***IBM WebSphere Application Server*** | The application server service by IBM WebSphere that facilitates the operation and maintenance of the application processes of systems such as CE. |
|  |  |
| ***Inbox*** | A facility for accessing and processing transactions in three ways: by product; by transaction status; and by product and transaction status. |

J

|  |  |
| --- | --- |
| ***Java Standard Edition Development Kit*** | Also simply referred to as JDK. It is a type of software required in developing and testing programs that are built using Java language and are run using the Java platform. This must first be installed before the CE server, workstation, and CE Utility are set up. |
|  |  |
| ***JavaScript*** | A scripting language that can interact with HTML source codes and has dynamic content. |
|  |  |
| ***JSP*** | Stands for JavaServer Pages. This technology enables the creation of dynamic web content and development of web-based applications that are server- and platform- independent. Transaction screens, which are the actual web pages displayed when running a function in the CE browser, are saved as JSP files. |

K

|  |  |
| --- | --- |
| ***Key Fields*** | The fields that, when combined together, make each of the records of a table unique. The key fields that are automatically added to the table must be retained to ensure proper runtime processing. |

M

|  |  |
| --- | --- |
| ***Module*** | A group of functions that perform interrelated processes and operate under a general principle or objective (e.g., a system module, which is essential to system processes; a business or transaction module, which pertains to a bank service or product). |
|  |  |
| ***Multi-Language*** | A functionality of CE that allows users to translate the elements in transaction screens – as well as messages in the transaction screens – into their preferred language. |

O

|  |  |
| --- | --- |
| ***Operator*** | The type of CE Utility user with parameter and transaction module configuration rights. |
|  |  |
| ***Oracle*** | A database server product that is compatible with the CE system. |

P

|  |  |
| --- | --- |
| ***Parameter*** | Any user-controlled configuration that defines a factor or logic within a set of interrelated operations; performs a specific action in a group of processes; or produces a categorical result or setting. |
| Protocol Manager Function The CE Utility function that is used to configure, manage, and maintain the communication protocols used by CE for connecting with other systems. |  |
| ***Parameter Drive*** | The location of the CE parameter files, one of the two main folders of an CE environment. Its path is defined in the GEN\_XML\_ROOTPATH system parameter. |
| Protocol Manager Function The CE Utility function that is used to configure, manage, and maintain the communication protocols used by CE for connecting with other systems. |  |
| ***Parameter Files*** | See XML. |

R

|  |  |
| --- | --- |
| ***Reformat*** | A function that is used to commit and update all maintenance operations on the table structures and CE data types, to the relevant database tables. |

S

|  |  |
| --- | --- |
| ***Schema*** | The structure and layout of objects within the database. In Oracle, a schema is associated with a specific database user and is comprised of database objects such as tables and views. |
|  |  |
| ***Screen*** | A function’s transaction screen. This is the actual webpage displayed when running a function in the CE browser. |
| ***Security Module*** | A group of CE security and system maintenance functions. With these functions, the required security and system settings are initially configured to enable a CE user to run any security or business function from the CE system. |
|  |  |
| ***SQL*** | Stands for Structured Query Language. This is a standard interactive and programming language for retrieving and updating data in the database. |
|  |  |
| ***SQL Statements*** | The building blocks of SQL software and programming. An SQL statement definition leads to specific output and always begins with a command (e.g., Create, Select, Insert). |
|  |  |
| ***Super Administrator*** | The type of CE Utility user with rights to administrative tasks such as configuring business units; defining data sources; and creating Super Administrator and Administrator user profiles. |
|  |  |
| ***Super Officer*** | A user that can perform all the security operations offered in CE. |

W

|  |  |
| --- | --- |
| ***Web Server*** | Server that receives data from the client (i.e., JSP and JS information), converts it to XML format, passes it to the application server for processing, and passes the processed info from the application server to the client. |

X

|  |  |
| --- | --- |
| ***XML*** | Stands for Extensible Markup Language. This is the format used by CE for the communication between the client (browser) and the server. |
|  |  |
| ***XML Generator Function*** | The CE Utility function that is used to generate the corresponding XML files for a specific parameter setting. |