



Centers for Medicare & Medicaid Services
CMS eXpedited Life Cycle (XLC)

Electronic Submission of Medical Documentation (esMD)

R3.0 RC Client (.NET) Implementation Guide

Version 1.3

6/13/2014

Document Number: R_3_0_RC_Client_dot_Net_Imp_Guide_V_1_3docx

Contract Number: HHSM-500-2007-00024I

Table of Contents

1. Introduction	1
2. Overview	2
3. System Requirements	3
3.1 Processor	3
3.2 Disk Space	3
3.3 Memory	3
3.4 Permissions	3
3.5 Network	3
3.6 Microsoft .NET Framework	3
4. TIBCO MFT File Transfers	4
5. Installation	5
5.1 Out-of-the-box	5
5.1.1 Keystore	5
5.1.2 Configuring the RC Client	5
5.1.3 Running the RC Client	7
5.2 Custom RC Client	8
6. Operation	10
7. XML Messages	11
7.1 Inbound	11
7.1.1 Payload Files	11
7.1.2 Metadata File	11
7.1.3 Pickup HIH Status Response	12
7.1.4 PMD PA Review Result HIH Status Response	13
7.1.5 PMD PA Review Result Validation Error Response	13
7.1.6 Pickup Virus Error Response	14
7.1.7 PMD PA Review Result Virus Scan Error Response	14
7.2 Outbound	15
7.2.1 Pickup Notification	15
7.2.2 Error Pickup Notification	16
7.2.3 PMD PA Review Result	17
8. RC Client Components	18
8.1 SFTP Client	18
8.2 Compression Utility	19
8.3 Encryption Utility	19
8.4 XML Processor	19
8.5 Scheduler	19
8.6 Housekeeping Manager	19

9. RC Client Workflow.....	20
9.1 Start RC Client	20
9.1.1 Login and Encrypt	20
9.2 Outbound Process	20
9.2.1 Outbound Start	20
9.2.2 Get Outbound Documents	20
9.2.3 Connect	20
9.2.4 Push	22
9.3 Inbound Processes	22
9.3.1 Inbound Start	22
9.3.2 Housekeeping.....	22
9.3.3 Extraction.....	22
9.3.4 Checksum Verification	22
9.4 Acknowledgements	22
9.4.1 Pickup Notification	22
9.4.2 Error Pickup Notification	23
9.5 Connect.....	23
9.6 Get Notifications.....	23
9.7 Process Document.....	23
9.8 Pull Document.....	23
10. Release 3.0 Changes in the API.....	24
11. .NET Client API.....	27
11.1 Security	27
11.2 .NET API Documentation	28
11.2.1 Login.....	28
11.2.2 Inbound.....	28
11.2.3 Outbound.....	30
11.2.4 Utilities – PMDPA Result	30
11.2.5 Utilities - Encryption	31
11.2.6 Utilities - Handshake.....	32
11.3 Logs	32
11.4 Utilities.....	32
12. Error Codes	33
13. Contacts	34
Appendix A: New User Registration	35
Appendix B: Registered User Login Instructions.....	41
Appendix C: XML Message Details.....	44
Acronyms.....	45
Glossary.....	47

Record of Changes	49
Approvals.....	50

List of Figures

Figure 1: RC Client Inbound and Outbound Process	2
Figure 2: RC Client Popup	8
Figure 3: RC Client Components	18
Figure 4: RC Client Workflow	21
Figure 5: Encryption and Decryption Process	27
Figure 6: New User Registration	35
Figure 7: New User Registration Menu	36
Figure 8: Terms and Conditions	36
Figure 9: New Registration Form	37
Figure 10: Email Verification	38
Figure 11: New Registration - Contact Information	38
Figure 12: Authentication Questions	39
Figure 13: Review Registration Details	39
Figure 14: Registration Acknowledgement.....	40
Figure 15: Registered User Login Window.....	41
Figure 16: Terms and Conditions	41
Figure 17: My Profile	42
Figure 18: Modify Account Profile	43

List of Tables

Table 1: Inbound Files.....	4
Table 2: Outbound Files	4

Table 3: Sample RC Client Configuration File.....	6
Table 4: E_123456-metadata.xml	11
Table 5: N_123456_Pickup_HIH_Status_Response.xml	12
Table 6: N_123456_PMDPA_Review_Result_HIH_Status_Response.xml	13
Table 7: R_123456_PMDPA_Review_Result_Validation_Error_Response.xml	13
Table 8: R_123456_Pickup_Virus_Scan_Error_Response.xml	14
Table 9: R_123456_PMDPA_Review_Result_Virus_Scan_Error_Response.xml	15
Table 10: P_186303_Pickup_Request.xml	16
Table 11: R_186303_Pickup_Error_Request.xml	16
Table 12: E_186303_PMDPA_Review_Result_Request.xml.....	17
Table 13: Client Method Comparison	24
Table 14: ESMD.RcClient.Login.LoginProcess Methods	28
Table 15: ESMD.RcClient.Inbound.Inbound Methods	28
Table 16: ESMD.RcClient.Outbound.Outbound Methods	30
Table 17: Manual Submission of PMDPAResult	31
Table 18: EMSD.RcClient.Encryption.EncryptionUtil Methods.....	31
Table 19: Remote Troubleshooting	32
Table 20: Error Codes	33
Table 21: Support Points of Contact.....	34
Table 22: XML Message Details	44
Table 23: Acronyms	45
Table 24: Glossary	47
Table 25: Record of Changes	49

1. Introduction

The Electronic Submission of Medical Documentation (esMD) system provides a mechanism for exchanging medical documentation (and responses) between the Medicare Provider community and the Medicare Review Contractor (RC) community. The purpose is to enable the electronic transmission of information between Health Information Handlers (HIHs) who represent Providers and the Medicare RCs, replacing paper documents where possible.

The RC Client is a utility that enables RCs to communicate with esMD by exchanging files via TIBCO® Managed File Transfer (MFT) server. An example of a file is the Power Mobility Device (PMD) Prior Authorization (PA) review results response.

Please note: esMD identifies submissions and requests are sent from HIHs to RCs as inbound files, transactions, and responses are sent from RCs to HIHs as outbound files.

The RC Client provides the following functionality:

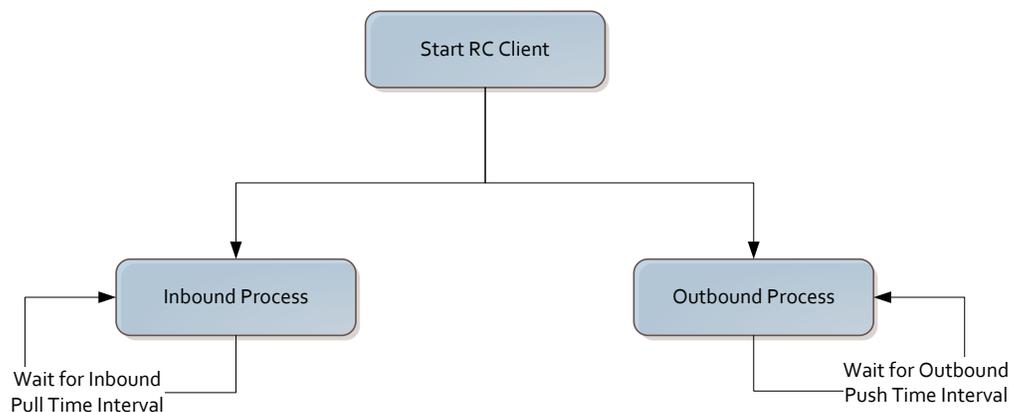
- Pull
 - Inbound documents (submitted by HIHs) from the TIBCO MFT server
 - HIH acknowledgement that they received PMD PA review results responses
 - Metadata validations for the outbound process
- Push
 - Outbound PMD PA review results responses to esMD
 - Error messages generated due to file decompression and checksum verification
 - Acknowledgement messages for receipt of documents and Authorization requests
- Site-Specific Configuration settings
 - Push frequency/Pull frequency
 - Folder locations for both Inbound and Outbound files

2. Overview

The esMD RC .NET Client is a standalone .NET Windows desktop application that runs outside the Centers for Medicare and Medicaid Services (CMS) network on the RC's machine, computer, or server. The purpose of the RC .NET Client is to connect to the TIBCO MFT server at the Baltimore Data Center (BDC) and push and pull files. The RC .NET Client uses Individuals Authorized Access to the CMS Computer Services (IACS) login credentials to authenticate with the TIBCO MFT server. The RC Client users (at the RC site) provide their login credentials when they start the RC Client on their machines. See Appendix A: New User Registration for details on how to request an IACS Identification (ID).

Users enter their login credentials only once at the program startup. When the RC Client starts, it initiates and then continuously runs two parallel threads as shown in Figure 1: RC Client Inbound and Outbound Process. When a user starts the RC Client, it will run continuously; it pulls and pushes files automatically without continual user intervention, based on the frequencies set by the RC.

Figure 1: RC Client Inbound and Outbound Process



In the inbound process when the RC Client connects to the BDC TIBCO MFT server, the RC Client immediately executes a pull cycle. The documents are pulled into the RC's inbound user directory for the authenticated user, and then the RC Client disconnects and waits for the next cycle, as determined by the Inbound Pull Time Interval setting.

In the outbound process when the RC Client connects to the BDC TIBCO MFT server, the RC Client executes a push cycle, pushes documents from the RC's outbound user directory to the TIBCO MFT server, and then disconnects and waits for the next cycle as determined by the Outbound Push Time Interval setting.

The inbound pull frequency is independent of the outbound push frequency. After each successful push or pull process, the RC Client thread disconnects from the TIBCO MFT server. To ensure continuous operation of the RC Client, it must preserve each user's IACS login credentials during the program execution.

Caution: Running multiple instances of the .NET RC Client for the same jurisdiction could result in duplicate file downloads.

3. System Requirements

3.1 Processor

The RC Client requires a Pentium 2 266-Megahertz (MHz) processor or greater.

3.2 Disk Space

The disk requirement for the RC .NET Client is 50 Megabytes (MB) for the RC Client itself. The documents that the RC Client pulls from the TIBCO MFT server may require additional disk space.

3.3 Memory

The RC .NET Client requires a minimum of 128 MB of free memory.

3.4 Permissions

The RC Client must have read, write, and execute permissions on all the directories under the installation home.

3.5 Network

The RC Client requires internet connectivity that supports more than 32-Kilobits Per Second (Kbps) transfer speeds.

3.6 Microsoft .NET Framework

The RC .NET Client requires Microsoft .NET Framework 4.5 to run properly.

4. TIBCO MFT File Transfers

The RC Client uses the TIBCO MFT server to interact with the esMD system. It uses the Secure File Transfer Protocol (SFTP) to connect to the TIBCO MFT server and uses the `ls/get/put` commands to interact with the files. There are four (4) types of inbound files that the RC Client pulls from the TIBCO MFT server, described in Table 1: Inbound Files.

Note “ES0001” is a sample mailbox number that the TIBCO MFT server uses to identify the RC, and “0977890” is a sample transaction ID also shown in Table 2: Outbound Files. The final two qualifiers in the file name that are prefixed with D and T are the Date and Timestamp, respectively. The VAL files will have a ‘T’ prefix and the Production files will have a ‘P’ prefix.

Only 1,022 files will be visible in the TIBCO MFT Server at one time, associated with the MFT Mailbox Routing number. As each file is pulled, the TIBCO MFT Server will bring new files from the mainframe and place at the bottom of the queue.

Table 1: Inbound Files

Type	Example File Name	Delivery Type Description
Inbound	T.ES0001.E0977890.D140116.T1033445	The E in prefix to the 0977890 transaction ID indicates an esMD payload
Inbound	T.ES0001.A0977890.D140116.T1033445	A indicates an esMD acknowledgment
Inbound	T.ES0001.R0977890.D140116.T1033445	R indicates an esMD error
Inbound	T.ES0001.N0977890.D140116.T1033445	N indicates a notification file

Table 2: Outbound Files

Type	Example File Name	Delivery Type Description
Outbound	T#EFT.ON.ESMD.E0977890.D140116.T1033445	E indicates an esMD payload
Outbound	T#EFT.ON.ESMD.R0977890.D140116.T1033445	R indicates an esMD error
Outbound	T#EFT.ON.ESMD.P0977890.D140116.T1033445	P indicates an esMD pickup notification

5. Installation

You can install the RC Client in two ways:

- Out of the box
- Custom RC Client (.NET)

5.1 Out-of-the-box

The RC .NET Client Application Programming Interface (API) comes packaged with a sample client. To run this sample client out-of-the-box, the RCs need to follow the procedures in the following sections.

5.1.1 Keystore

Important: The RC .NET Client uses Asymmetric encryption to store the IACS user credentials securely. For this encryption to work, the RC needs to use the machine-level Rivest, Shamir & Adleman (RSA) key container provided by Microsoft Windows.

5.1.1.1 Microsoft Windows Machine-Level RSA Key Container

Microsoft Windows provides machine-level RSA key containers to all users who can log in to a computer by default. RSA key containers are useful as you can use them to encrypt or decrypt protected configuration sections while logged in with an administrator account. You can use a machine-level RSA key container to protect information for a single application, all the applications on a server, or a group of applications on a server that runs under the same user identity. Although machine-level RSA key containers are available to all users, they can be secured with New Technology File System (NTFS) Access Control Lists (ACLs) so that only required users can access them. You can use the `aspnet_regiis.exe` tool to create, export, import, or delete an RSA key container:

1. Type the command below at a command console to create a new RSA key container.

```
cd C:\Windows\Microsoft.NET\Framework64\<v4 . xxxxxxx>  
aspnet_regiis -pc <yourKeyName>
```

2. Replace `<v4 . xxxxxxx>` with the actual .NET framework version on your machine, and the `<yourKeyName>` with a name for your key so that you can retrieve it later.

5.1.1.2 Key Handling

The RC .NET Client delegates the key handling to the Windows Operating System Environment.

5.1.2 Configuring the RC Client

Once the keystore is created, the RC Client is ready to be configured to use the keystore.

1. Update the keystore information in the configuration file. (required)

Important: The XML configuration file (i.e., config/esmd-rc-client-config.xml) is used by the RC Client to retrieve important configuration parameters necessary for its operation.

2. Use the comments for each configuration parameter shown in Table 3: Sample RC Client Configuration File as a guide in entering your data.

Table 3: Sample RC Client Configuration File

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:ESMDCConfig xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc/config"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config esmd-config.xsd"
">

  <!--The TIBCO MFT Server Configuration-->
  <ESMDSFTPSTPServer>
    <!--TIBCO MFT Sever host name or IP -->
    <host>eftp2.cms.hhs.gov</host>
    <!--The TIBCO SFTP PORT-->
    <port>11022</port>
    <!--Update: Use T for VAL, P for PROD-->
    <environmentId>T</environmentId>
    <!--The EFT File Name Prefix-->
    <eftFilePrefix>#EFT</eftFilePrefix>
  </ESMDSFTPSTPServer>
  <!--The Keycontainer Settings-->
  <KeyStoreInfo>
    <!--N/A for .net-->
    <keyStoreLocation></keyStoreLocation>
    <!--N/A for .net-->
    <encKeyInfo></encKeyInfo>
    <!--N/A for .net-->
    <encKeyInfoExt></encKeyInfoExt>
    <!-- Update: The Key container Name-->
    <certAlias>yourKeyName</certAlias>
  </KeyStoreInfo>
  <!--The Inbound Process Configuration-->
  <InboundConfig>
    <!--Enable the Inbound Process? true/false-->
    <enabled>true</enabled>
    <!--The Pull Frequency for the Inbound Process in minutes; the
default is 240 minutes i.e. 4 hours-->
    <checkFrequency>240</checkFrequency>
    <!-- Update: The RC Client installation/home directory-->
    <rcHomeDirectory>c:\RCClient</rcHomeDirectory>
    <!-- Update: The target directory to extract the downloaded inbound
files before routing-->
    <targetDirectory>c:\RCClient\data\download</targetDirectory>
    <!-- Update: The input directory where the inbound payloads and the
metadata will be routed after the extraction-->
    <inputDirectory>c:\RCClient\data\input</inputDirectory>
    <!-- Update: The temp directory where the files are pulled from
TIBCO-->
    <tempDirectory>c:\RCClient\data\temp</tempDirectory>
    <!-- Update: The Error directory for routing the inbound error
notifications from esMD/HIH-->
```

```

    <errorDirectory>c:\RCClient\data\error\</errorDirectory>
    <!-- Update: The configuration directory for RC Client-->
    <configDirectory>c:\RCClient\data\conf\</configDirectory>
    <!-- Update: The notifications directory for routing the inbound
notifications from esMD/HIH-->

<notificationsDirectory>c:\RCClient\data\notification\</notificationsDirector
y>
    <!-- Update: The Remote Inbound Directory path on the TIBCO Server-->
    <remoteInboundDir>/ES####</remoteInboundDir>
    <!-- Update: The routing id for the inbound files used to pick the
inbound files to pull-->
    <inboundRoutingId>ES####</inboundRoutingId>
  </InboundConfig>
  <!--The Outbound Process Configuration-->
  <OutboundConfig>
    <!-- Update: Enable the Outbound Process? true/false-->
    <enabled>true</enabled>
    <!--The push frequency for the Outbound process in minutes default is
15 minutes-->
    <pushFrequency>15</pushFrequency>
    <!-- Update: The temp directory to use for the outbound process for
creating the PMPDA/Notification files-->
    <tempDirectory>c:\RCClient\data\temp\</tempDirectory>
    <!-- Update: The local outbound directory to push the outbound files
from-->
    <outputDirectory>c:\RCClient\data\output\</outputDirectory>
    <!-- Update: The Remote Outbound directory to push files-->
    <remoteOutboundDir>/esMD_UPLOAD</remoteOutboundDir>
    <!-- Update: The Remote Outbound Routing ID to push files onto esMD
servers via TIBCO-->
    <outboundRoutingId>ESMD</outboundRoutingId>
    <!--The Outbound File name prefix-->
    <outboundFilePrefix>ON</outboundFilePrefix>
  </OutboundConfig>
  <!--The PMD PA Files Configuration-->
  <PMDPAConfig>
    <!--The Content Type Code for the PMDPA Files i.e. 8-->
    <contentTypeCode>8</contentTypeCode>
    <!--The Delivery Type for the PMDPA Files i.e.E-->
    <deliveryType>E</deliveryType>
  </PMDPAConfig>
</ns2:ESMDConfig>

```

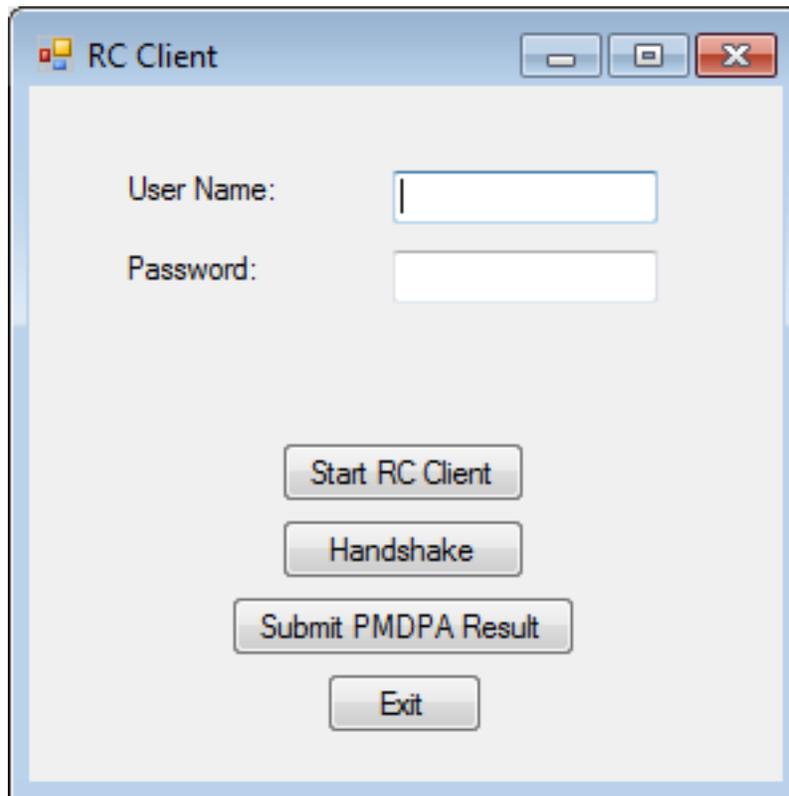
5.1.3 Running the RC Client

Before you, as the RC, run the sample RC Client, you must double-check all the configuration parameters in the Extensible Markup Language (XML) configuration file, especially the ones with the "Update" prefix in the comments of the sample XML configuration file as shown in Table 3: Sample RC Client Configuration File.

1. To run the sample RC Client, run the "RcClientUI.exe" utility provided in the distribution package to bring up the RC Client popup (see Figure 2: RC Client Popup).
2. Start the RC Client by clicking the "Start RC Client" button or clicking on the "Handshake" button to test the connectivity.
3. To manually submit PMD PA Result, click on the "Submit PMDPA Result" button.

4. To stop the RC Client, click the “Exit” button.

Figure 2: RC Client Popup



5.2 Custom RC Client

The RC .NET Client provides an API so the RC can extend the RC Client to fit your environmental needs. The API enables you perform the following functions:

- Log in to the TIBCO MFT server
- Get Notifications from the TIBCO MFT server using the SFTP protocol. (Refer to Section 11.2.2 Inbound).
- Decrypt/encrypt and store the login credentials using a secure RSA algorithm (Refer to Section 11.2.5 Utilities - Encryption)
- Pull medical documentation from the TIBCO MFT server. (Refer to Section 11.2.2 Inbound).
- Extract the downloaded packages. (Refer to Section 11.2.2 Inbound).
- Check the payloads using checksums in the metadata. (Refer to Section 11.2.2 Inbound).
- Push the outbound files from the “output” directory (Refer to Section 11.2.3 Outbound)
- Create custom files (for example, custom PMD PA files. Refer to Section 11.2.4 Utilities – PMDPA Result).

Note: The procedures for customizing the RC Client API are beyond the scope of this document; they are in the Help file packaged in the RC Client product.

6. Operation

The RC Client runs in a cyclical manner sleeping for a specified time interval between the operating cycles. The sleep intervals are configured in the “checkFrequency” parameter for the Inbound Process and the “pushFrequency” parameter for the Outbound process. The RC is advised to use the default of 240 minutes (4 hours) for the Inbound process and 15 minutes for the Outbound process.

The RC Client operation is interrupted in two events:

1. The IACS Passwords expires (IACS passwords expire every 60 days – contact the CMS help desk to reset).
2. Virus Scan error notification is received from esMD.

In the first scenario, when the IACS Password expires; the RC Client suspends its operation and is terminated. The RC must restart the RC Client and the user must provide the right credentials to login to the TIBCO MFT Server. The IACS notifies the user notified 15 days prior to the password expiring.

In the second scenario, when a Virus Scan error notification has been received from esMD, all the processes of the RC Client are suspended and the client is terminated. In addition, the RC Client is locked and cannot pull/push files even if the client is restarted. The RC is advised to contact the esMD Team (refer to Section 13 Contacts for more details) to unlock the RC Client.

7. XML Messages

This section describes the various XML messages transferred during the inbound and outbound processes.

Note: See Appendix C: XML Message Details for details on the name of the XML Message Files transferred between the RC Client and esMD.

7.1 Inbound

The RC Client transfers the following messages during the inbound process:

- Payload Files
- Metadata File
- Pickup HIH Status Response
- PMD PA Review Result HIH Status Response
- PMD PA Review Result Validation Error Response
- Pickup Virus Scan Error Response
- PMD PA Review Result Virus Scan Error Response

7.1.1 Payload Files

The RC Client will receive Portable Document Format (PDF) files as payloads in the inbound documents with delivery type 'E'. An example payload file name is: E_185457-esmdQSSIVG0407141396893280928-0.pdf

7.1.2 Metadata File

The metadata file accompanies the payload files in the inbound documents with delivery type 'E'. The metadata file contains information about the payloads like the Object Identifiers (OIDs), Transaction ID (TID), Submission Metadata and Optional Metadata. The Content Type Code will change for each line of business. Details described in Table 4: E_123456-metadata.xml below.

Note: The metadata file will remain the same for all lines of business including Additional Documentation Requests (ADRs), PMD PA Requests, Appeals, and Advance Determination of Medicare Coverage (ADMC).

Table 4: E_123456-metadata.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:RetrieveMedicalDocumentationResponse returnCode="0"
serviceSuccessful="true" xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc">
  <statusDescription>The RetrieveMedicalDocumentationRequest processed
successfully.</statusDescription>
  <NumberOfDocuments>1</NumberOfDocuments>
  <ESMDPackage>
    <ESMDTransaction TransactionId="185456" DeliveryType="E"/>
    <SendingOID>urn:oid:123.456.657.132</SendingOID>
```

```

    <TargetOID>urn:oid:2.16.840.1.113883.13.34.110.1.999.1</TargetOID>
    <CompleteSubmission>true</CompleteSubmission>
    <SubmissionMetadata xsi:type="ns4:ADR"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ns4="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <CreationTime>2014-04-07T13:54:45.936-04:00</CreationTime>
    <SubmissionTime>2014-04-07T13:54:45.936-04:00</SubmissionTime>
    <EFTSubmissionTime>2014-04-07T13:54:45.937-
04:00</EFTSubmissionTime>
    <ContentTypeCode>1</ContentTypeCode>
    <NPI>1234567890</NPI>
    <ClaimId>TestLD ClaimID 8071302</ClaimId>
    <CaseId>LoadTest Case ID 123</CaseId>
    </SubmissionMetadata>
    <Documentation DocumentUniqueIdentifier="E_185456-
esmdQSSIVG0407141396893280928-0">
    <OptionalMetadata>
    <FieldName>esMDDocumentCreationTime</FieldName>
    <FieldValue>1396893285937</FieldValue>
    </OptionalMetadata>
    <OptionalMetadata>
    <FieldName>Description</FieldName>
    <FieldValue>From esMD</FieldValue>
    </OptionalMetadata>
    <OptionalMetadata>
    <FieldName>Checksum</FieldName>
<FieldValue>73d1ba48402985bac6ddab12f47c179dddbbe4c6</FieldValue>
    </OptionalMetadata>
    </Documentation>
  </ESMDPackage>
</ns2:RetrieveMedicalDocumentationResponse>

```

7.1.3 Pickup HIH Status Response

When the RC Client sends a pickup notification to the esMD system, the esMD application processes this notification and esMD sends the response to the HIH. The esMD application then generates the Pickup Status Response and sends it to the RC indicating the response was sent to the HIH, as detailed in Table 5: N_123456_Pickup_HIH_Status_Response.xml. Please refer to the code in Appendix C: N_TID_Pickup_HIH_Status_Response.xml.

Note: The HIH Pickup Status Response will remain the same for all lines of business including ADRs, Appeals, PMD PA Responses, Recover Audit Contractor (RAC) Discussion Requests, and ADMC Requests.

Table 5: N_123456_Pickup_HIH_Status_Response.xml

```

<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<ns2:RCPickupNotificationResponse
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc/config">
  <ESMDTransactionId>186303</ESMDTransactionId>
  <ErrorInfo xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>

```

```

<status>Success</status>
<statusDesc>SENT PICKUP STATUS TO HIH</statusDesc>
</ns2:RCPickupNotificationResponse>

```

7.1.4 PMD PA Review Result HIH Status Response

When the RC Client sends a PMD PA Review Result to esMD, the esMD application processes the file and sends the PMD PA Review Result to the HIH. The esMD application submits the PMD PA Review Result HIH Status Response, detailed in Table 6:

N_123456_PMDPA_Review_Result_HIH_Status_Response.xml , and sends it to the RC, indicating the result was sent to the HIH. Please refer to the code located in Appendix C: N_TID_PMDPA_Review_Result_HIH_Status_Response.xml.

Table 6: N_123456_PMDPA_Review_Result_HIH_Status_Response.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<esmd:SubmitPMDPADeterminationResponse
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc
../../../../config/esmd-rc.xsd "
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:esmdl="http://esmd.ois.cms.hhs.gov/v1/rc/transaction"
xmlns:esmd="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:cmsbt="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
  <statusDescription>PMD PA Review results - Successfully delivered
to HIH</statusDescription>
  <ESMDTransaction DeliveryType="N" TransactionId="186303"/>
</esmd:SubmitPMDPADeterminationResponse>

```

7.1.5 PMD PA Review Result Validation Error Response

When the RC Client sends a PMD PA Review Result to esMD, the esMD application processes and sends the PMD PA Review Result to the HIH. If there is an error in processing the PMD PA Review Result submitted by the RC, the esMD application generates the PMD PA Results Response Error, detailed in Table 7:

R_123456_PMDPA_Review_Result_Validation_Error_Response.xml , and sends it to the RC. The RC will then resubmit the PMD PA Results Result. Please refer to the code located in Appendix C: R_TID_PMDPA_Review_Result_Validation_Error_Response.xml.

Table 7: R_123456_PMDPA_Review_Result_Validation_Error_Response.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<esmd:SubmitPMDPADeterminationResponse
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config/esmd-
rc.xsd "
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:esmdl="http://esmd.ois.cms.hhs.gov/v1/rc/transaction"
xmlns:esmd="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:cmsbt="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
  <statusDescription>statusDescription</statusDescription>
  <ESMDTransaction DeliveryType="R" TransactionId="186303"/>

```

```

<ValidationFailure>
  <FailureCode>541</FailureCode>
  <FailureReason>ESMD validation error: Transaction ID is
invalid</FailureReason>
</ValidationFailure>
<ValidationFailure>
  <FailureCode>556</FailureCode>
  <FailureReason>ESMD validation error: Decision Indicator must
be A, N, or R</FailureReason>
</ValidationFailure>
</esmd:SubmitPMDPADeterminationResponse>

```

7.1.6 Pickup Virus Error Response

When the RC Client sends a Pickup Notification to esMD, the esMD application sends it to the Virus Scan Gateway for virus scan. If there are any viruses detected in the pickup notification, the esMD application sends the message detailed in Table 8:

R_123456_Pickup_Virus_Scan_Error_Response.xml to the RC. The RC Client will then pull this Virus Scan Error, stop the inbound and outbound processes and locks down the RC Client to prevent RC Client from interacting with esMD. In this situation, the RC Client does not enable recovery, and the RC will contact esMD Help Desk. Please refer to the code located in Appendix C: R_TID_Pickup_Virus_Scan_Error_Response.xml.

Table 8: R_123456_Pickup_Virus_Scan_Error_Response.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<tns:RCPickupNotificationResponse
xmlns:tns="http://esmd.ois.cms.hhs.gov/v1/rc/config"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config esmd-
config.xsd">
  <ESMDTransactionId>123456</ESMDTransactionId>
  <ErrorInfo>
    <ErrorCode>560</ErrorCode>
    <ErrorName>VirusFound</ErrorName>
    <ErrorDescription>ESMD validation error: Submission is infected with
virus</ErrorDescription>
  </ErrorInfo>
  <Status>FAILED</Status>
  <StatusDesc>Outbound Response File contains virus and so the response is
rejected.</StatusDesc>
</tns:RCPickupNotificationResponse>

```

7.1.7 PMD PA Review Result Virus Scan Error Response

When the RC Client sends a PMD PA Review Result to esMD, the esMD application sends it to the Virus Scan Gateway for virus scan. If there are any viruses detected in the PMD PA Review Result, the esMD application sends the message detailed in Table 9:

R_123456_PMDPA_Review_Result_Virus_Scan_Error_Response.xml to the RC. The RC Client will then pull this PMD PA Review Result Virus Scan Error, stop the inbound and

outbound processes and locks down the RC Client to prevent RC Client from interacting with esMD. In this situation, the RC Client does not enable recovery, and the RC will contact esMD Help Desk. Please refer to the code located in Appendix C: R_TID_Pickup_Virus_Scan_Error_Response.xml.

Table 9: R_123456_PMDPA_Review_Result_Virus_Scan_Error_Response.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<esmd:SubmitADMCDeterminationResponse
xmlns:cmsbt="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt"
xmlns:esmd="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:esmdl="http://esmd.ois.cms.hhs.gov/v1/rc/transaction"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc ../config/esmd-rc.xsd
">
  <statusDescription>Outbound Response File contains virus and so the
response is rejected.</statusDescription>
  <ESMDTransaction TransactionId="123456" DeliveryType="R"/>
  <ValidationFailure>
    <FailureCode>560</FailureCode>
    <FailureReason>ESMD validation error: Submission is infected with
virus</FailureReason>
  </ValidationFailure>
</esmd:SubmitADMCDeterminationResponse>
```

7.2 Outbound

The RC Client transfers the following messages during the outbound process:

- Pickup Notification
- Error Pickup Notification
- PMD PA Review Result

7.2.1 Pickup Notification

The RC Client generates pickup notifications for all inbound files pulled from the TIBCO MFT server and processed successfully, as detailed in Table 10: P_186303_Pickup_Request.xml. Please refer to the code located in Appendix C: P_TID_Pickup_Request.xml.

Table 10: P_186303_Pickup_Request.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:RCPickupNotification
  xmlns:tns="http://esmd.ois.cms.hhs.gov/v1/rc/config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config/esmd-
  config.xsd ">
  <ESMDTransactionId>186303</ESMDTransactionId>
  <PickupTime>2014-04-09T09:00:00</PickupTime>
  <SubmissionTime>2014-04-09T09:00:00</SubmissionTime>
  <ErrorInfo xsi:nil="true"/>
</tns:RCPickupNotification>
```

7.2.2 Error Pickup Notification

The RC Client generates pickup error notifications for all inbound files pulled from TIBCO MFT and processed unsuccessfully, detailed in Table 11: R_186303_Pickup_Error_Request.xml. The processing errors are generated in two scenarios:

- Checksum verification failed (i.e., the file received by the RC client does not match the file sent by esMD).
- Extraction was unsuccessful (i.e., the RC client could not successfully unzip the file received from the TIBCO MFT server).
- Refer to the code located in Appendix C: R_TID_Pickup_Error_Request.xml.

Table 11: R_186303_Pickup_Error_Request.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:RCPickupNotification
  xmlns:tns="http://esmd.ois.cms.hhs.gov/v1/rc/config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config/esmd-
  config.xsd ">
  <ESMDTransactionId>186303</ESMDTransactionId>
  <PickupTime>2001-12-31T12:00:00</PickupTime>
  <SubmissionTime xsi:nil="true"/>
  <ErrorInfo>
    <ErrorCode>0</ErrorCode>
    <ErrorName>ErrorName</ErrorName>
    <ErrorDescription>ErrorDescription</ErrorDescription>
  </ErrorInfo>
</tns:RCPickupNotification>
```

7.2.3 PMD PA Review Result

The PMD PA Determination Result is the XML message from the RC to the HIH, detailed in Table 12: E_186303_PMDPA_Review_Result_Request.xml. Please refer to the code located in Appendix C: E_TID_PMDPA_Review_Result_Request.xml.

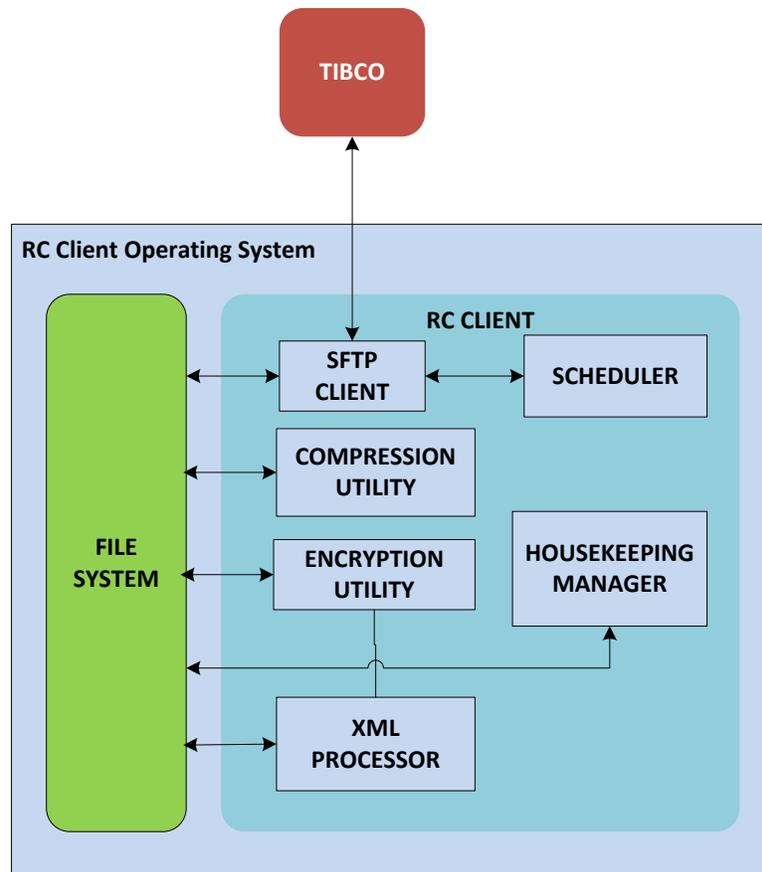
Table 12: E_186303_PMDPA_Review_Result_Request.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<esmd:SubmitPMDPADeterminationRequest
  xmlns:cmsbt="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt"
  xmlns:esmd="http://esmd.ois.cms.hhs.gov/v1/rc"
  xmlns:esmdl="http://esmd.ois.cms.hhs.gov/v1/rc/transaction"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config/esmd-rc.xsd ">
  <ESMDTransaction TransactionId="186303" DeliveryType="E"/>
  <PMDPAResponse>
    <CreationTime>2014-04-09T10:00:00</CreationTime>
    <SubmissionTime>2014-04-09T10:00:00</SubmissionTime>
    <EFTSubmissionTime>2014-04-09T09:00:00</EFTSubmissionTime>
    <ContentTypeCode>8</ContentTypeCode>
    <NPI>1234567890</NPI>
    <DecisionIndicator>A</DecisionIndicator>
  </PMDPAResponse>
  <UniqueTrackingNumber>UniqueTrackingNumber</UniqueTrackingNumber>
  <ReasonCodeRecord>
    <ReasonCode>123</ReasonCode>
    <ReasonCodeDescription>test123</ReasonCodeDescription>
  </ReasonCodeRecord>
</esmd:SubmitPMDPADeterminationRequest>
```

8. RC Client Components

Figure 3: RC Client Components shows a diagram of the internal components of RC Client application. The following sections describe each component in detail.

Figure 3: RC Client Components



8.1 SFTP Client

The SFTP Client is an internal component of the RC Client. It provides the following functionality:

- Connect to the TIBCO MFT server using IACS ID
- List the available documents on the TIBCO MFT server
- Pull the documents to the RC Client
- Push the outbound documents from RC Client to the TIBCO MFT server

8.2 Compression Utility

The Compression utility allows the RC Client to extract the payload, metadata file, and messages from the compressed file downloaded from the TIBCO MFT server. The RC Client uses the zip file format.

The same utility is used to create compressed file logs for extraction.

8.3 Encryption Utility

The Encryption utility encrypts the login credentials that will be stored in memory for the duration of the RC Client program execution. The Encryption utility is described in detail in Section 11.1 Security.

8.4 XML Processor

The XML Processor supports creating XML messages to send to esMD, as well as loading the configuration files for the RC Client.

8.5 Scheduler

After the RC Client starts, the polling cycle begins. The poll is a redundant cycle; you can configure the interval (for example, 1 hour or 4 hours) through the RC Client property file. The Schedule component controls the RC Client threads and ensures the RC Client runs in regular intervals determined by the “checkFrequency” parameter in the XML Configuration File.

8.6 Housekeeping Manager

The Housekeeping Manager allows the RC Client to recover from any abnormal terminations with the exception of a Virus lockdown. In this situation, the RC Client does not enable recovery, and the RC must contact the esMD Help Desk.

9. RC Client Workflow

The workflow associated with Figure 3: RC Client Components is broken down in Figure 4: RC Client Workflow, followed by a detailed description of the workflow.

9.1 Start RC Client

The RC Client starts on the RC machine or server. It loads the XML Configuration File.

9.1.1 Login and Encrypt

The RC Client prompts the user for the following details:

1. IACS User ID
2. IACS Password

After successful login, TIBCO login credentials are encrypted in memory and used when needed to log in to the TIBCO MFT server. The RC Client initiates two threads, one for the inbound process and one for the outbound process in sections 9.2 Outbound Process and 9.3 Inbound Processes, respectively. These processes are described in the next section.

9.2 Outbound Process

9.2.1 Outbound Start

The RC Client loads configuration parameters for the outbound process from the configuration file (XML). The configuration parameters are as follows:

- Directories used by the RC Client to create the outbound files (`outputDirectory`)
- The remote outbound directory to push the files to (`remoteOutboundDir`)
- Push frequency (`pushFrequency`)
- The outbound file name prefix for the TIBCO MFT server (`outboundFilePrefix`)
- SFTP server details for the chosen environment (`ESMDSFTPServer`)

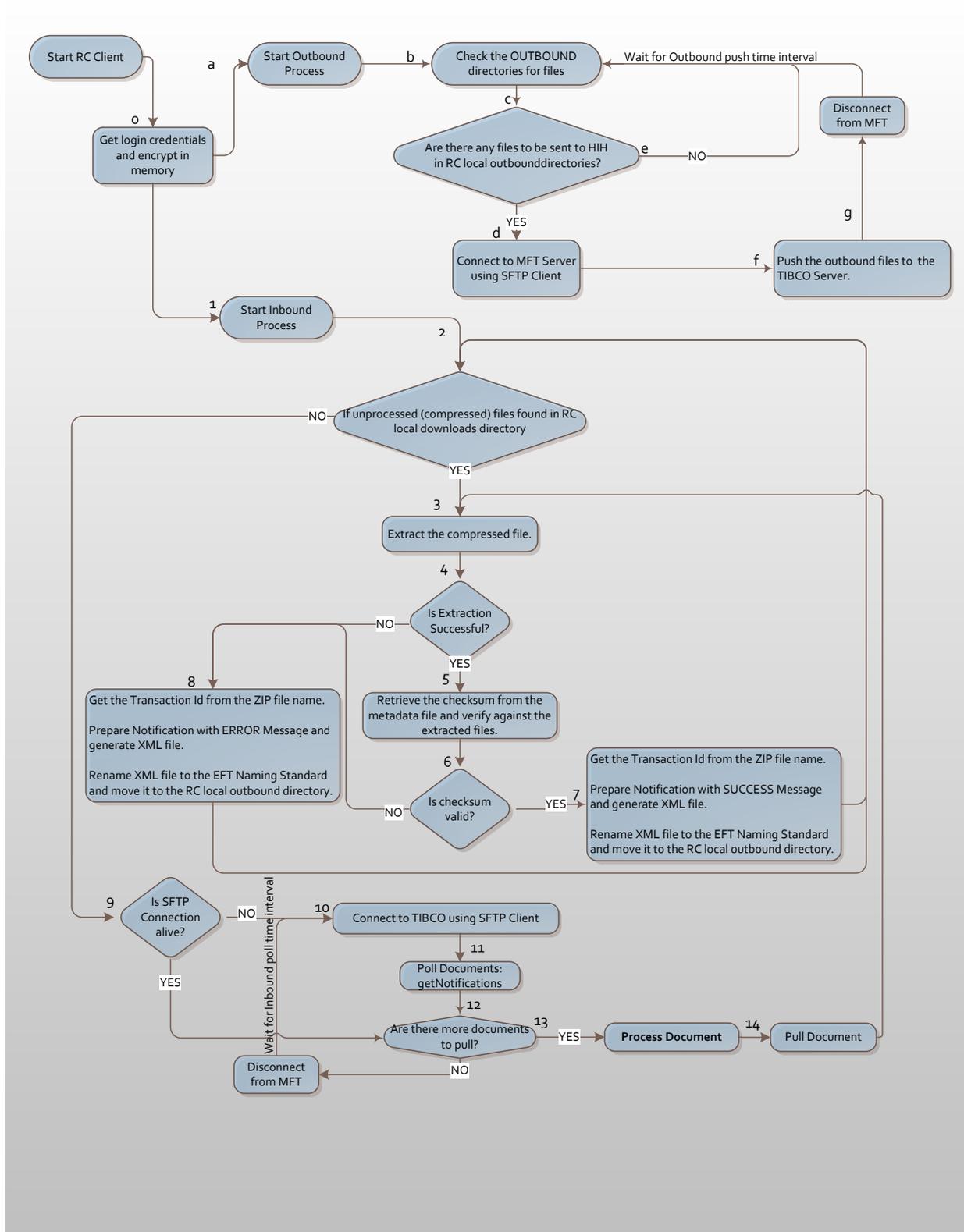
9.2.2 Get Outbound Documents

The RC Client checks the output directory for any files to be sent to the HIH. If any such files exist, the process continues to Step d (Connect); otherwise, the outbound process thread sleeps for the time interval determined by the pushFrequency parameter in the XML Configuration File.

9.2.3 Connect

The RC Client connects to the TIBCO MFT server using IACS login credentials. The Encryption utility decrypts the credentials in memory and logs in to the TIBCO MFT server. If the user password is expired, the connection fails, prompting the user to provide the login information again.

Figure 4: RC Client Workflow



9.2.4 Push

The RC Client pushes outbound files to the TIBCO MFT server. After that, the outbound process thread sleeps. The sleep time interval is determined by the outbound push frequency configuration parameter in the XML Configuration File.

9.3 Inbound Processes

9.3.1 Inbound Start

- The RC Client loads Configuration parameters from the configuration file (XML). The Configuration parameters are for the following inbound processes:
- Pull frequency
- SFTP server details for the chosen environment

9.3.2 Housekeeping

The Housekeeping Manager is responsible for the cleanup and recovery from any abnormal terminations. If the extraction process was interrupted during extraction in the previous run, then there will be compressed files in the local "temp" directory.

9.3.3 Extraction

The Housekeeping Manager extracts compressed files found in the local "temp" directory for the RC Client before it pulls any new documents from the TIBCO MFT server. It will extract the oldest files first. If the extraction is successful, RC Client proceeds to "checksum verification"; otherwise, RC Client creates an error pickup notification.

9.3.4 Checksum Verification

After the extraction is complete, the RC Client uses the XML Processor to parse the metadata file from the zip package. This metadata file contains the checksums for all payloads in the package. The RC Client verifies the checksum for each file in the package against the checksum in the metadata file. If the checksum is valid for all files, the RC Client will create a pickup notification; otherwise, the RC Client will create an error pickup notification.

9.4 Acknowledgements

9.4.1 Pickup Notification

If the RC Client successfully extracts and verifies compressed files, the RC Client sends a success notification through esMD to inform the HIH that the document has been received and successfully processed. To generate this SUCCESS notification, the RC Client should:

- Get the transaction ID from the compressed file name
- Prepare the notification with a SUCCESS message and generate an XML notification file
- Rename the XML notification file to the EFT naming standard and move it to the outbound directory. Refer to Table 2: Outbound Files in Section 4 TIBCO MFT File Transfers for more information.

9.4.2 Error Pickup Notification

If the RC Client encounters an error indicating failure while either extracting the compressed file or verifying the checksum for the contents of the package, the RC Client sends an error notification through esMD, asking the HIH to resubmit the package. In order to generate this error notification, the RC Client must:

- Obtain the TID from the compressed file name.
- Prepare the notification with an ERROR message.
- Generate an XML notification file.
- Rename the XML notification file to the EFT naming standard and move it to the outbound directory. This file will be handled by the outbound process.

9.5 Connect

After the Housekeeping Manager completes preprocessing, the RC Client checks for an active connection to the TIBCO MFT server. If a connection is active, the RC Client uses this connection. If the connection is inactive, the RC Client uses the Encryption utility to decrypt the login credentials from memory and connects to the TIBCO MFT server.

9.6 Get Notifications

The RC Client uses the SFTP Client to get a list of the available inbound documents for the RC on the TIBCO MFT server.

9.7 Process Document

If any documents are available for the RC Client to pull from the TIBCO MFT server, the RC Client will go through the list to pull each document.

9.8 Pull Document

The RC Client uses the SFTP Client to pull each inbound document from the TIBCO MFT server. The RC Client then extracts the contents of the zip file and continues processing.

10. Release 3.0 Changes in the API

Table 13: Client Method Comparison compares similar methods in the Enterprise Content Management (ECM) Client and the Release 3.0 RC Client.

Table 13: Client Method Comparison

ECM Client (Inbound)	RC Client (Inbound)
<p>process()</p> <ul style="list-style-type: none"> • Calls getNotifications() to get list of available downloads from ECM. • Calls processMedicalDocumentation() for each available document from ECM. • Calls acknowledge() each document processed with status 100. • Sleeps for the checkFrequency time interval before next pull. 	<p>process()</p> <ul style="list-style-type: none"> • Collects the IACS login credentials provided in the login prompt or by decrypting the encrypted login details in memory. • Calls getNotifications() with the login credentials to get list of available downloads from TIBCO. • Calls processMedicalDocumentation() for each available document from TIBCO. • Calls acknowledge() each document processed with a success/error XML response message. • Sleeps for the checkFrequency time interval before next pull.
<p>getNotifications()</p> <ul style="list-style-type: none"> • Creates a "RetrieveNotificationsRequest" with the contentTypeCode and deliveryType. • Retrieves the list of ESMDTransactions available. 	<p>getNotifications()</p> <ul style="list-style-type: none"> • Connects to TIBCO MFT server with IACS Login and Password. • Retrieves the list of files available for download for that environment.

ECM Client (Inbound)	RC Client (Inbound)
<p>processMedicalDocumentation()</p> <ul style="list-style-type: none"> • Creates a “RetrieveMedicalDocumentationRequest” from the ESMDTransaction. • Retrieves the Medical Document for that transaction in “RetrieveMedicalDocumentationResponse”. • Calls saveDocuments() method to save the metadata from the response and the attachments to the target directory from the XML configuration file. 	<p>processMedicalDocumentation()</p> <ul style="list-style-type: none"> • Pulls the zip file from the TIBCO MFT server using the pullDocument() method based on the name passed to it. • Extracts the zip file into the “download” directory using the extractDocument() method. • If extraction fails, calls the acknowledge method with an error event and exits. • After successful extraction, verifies the extracted payloads against the checksum in the metadata file using the checkPayloads() method. • If checksum fails, calls the acknowledge method with an error event. • If checksum passes, calls the acknowledge method with a success event.
<p>acknowledge()</p> <ul style="list-style-type: none"> • Creates and submits the “SubmitExternalEventRequest” for the ESMDTransaction passed with eventCode 100. • Retrieves the “SubmitExternalEventResponse” and logs it. 	<p>acknowledge()</p> <ul style="list-style-type: none"> • Creates the error/success message and puts it in the output directory. • Logs the event.
<p>promptForInfo()</p> <ul style="list-style-type: none"> • Gathers the following information <ul style="list-style-type: none"> ○ Delivery Type ○ Transaction Id ○ Content Type Code ○ NPI ○ ADMC Decision Indicator ○ Unique Tracking Number (enter for none) ○ Add a reason code Y/N ○ Reason Code ○ Reason Code Description (enter for none) • Builds the ADMCResult and the ESMDTransaction needed for submission. 	<p>promptForInfo()</p> <ul style="list-style-type: none"> • Gathers the following information <ul style="list-style-type: none"> ○ Delivery Type ○ Transaction Id ○ Content Type Code ○ NPI ○ PMDPA Decision Indicator ○ Unique Tracking Number (enter for none) ○ Add a reason code Y/N ○ Reason Code ○ Reason Code Description (enter for none) • Builds the PMDPAResult object.

ECM Client (Inbound)	RC Client (Inbound)
<p>submitADMCRestult()</p> <ul style="list-style-type: none">• Creates and submits the "SubmitADMCDeterminationRequest".• Retrieves the "SubmitADMCDeterminationResponse" and logs it.	<p>submitPMDPARestult()</p> <ul style="list-style-type: none">• Takes the PMDPAResult object and creates the XML response file in the input directory.• Compresses and pushes the XML file to TIBCO MFT server.

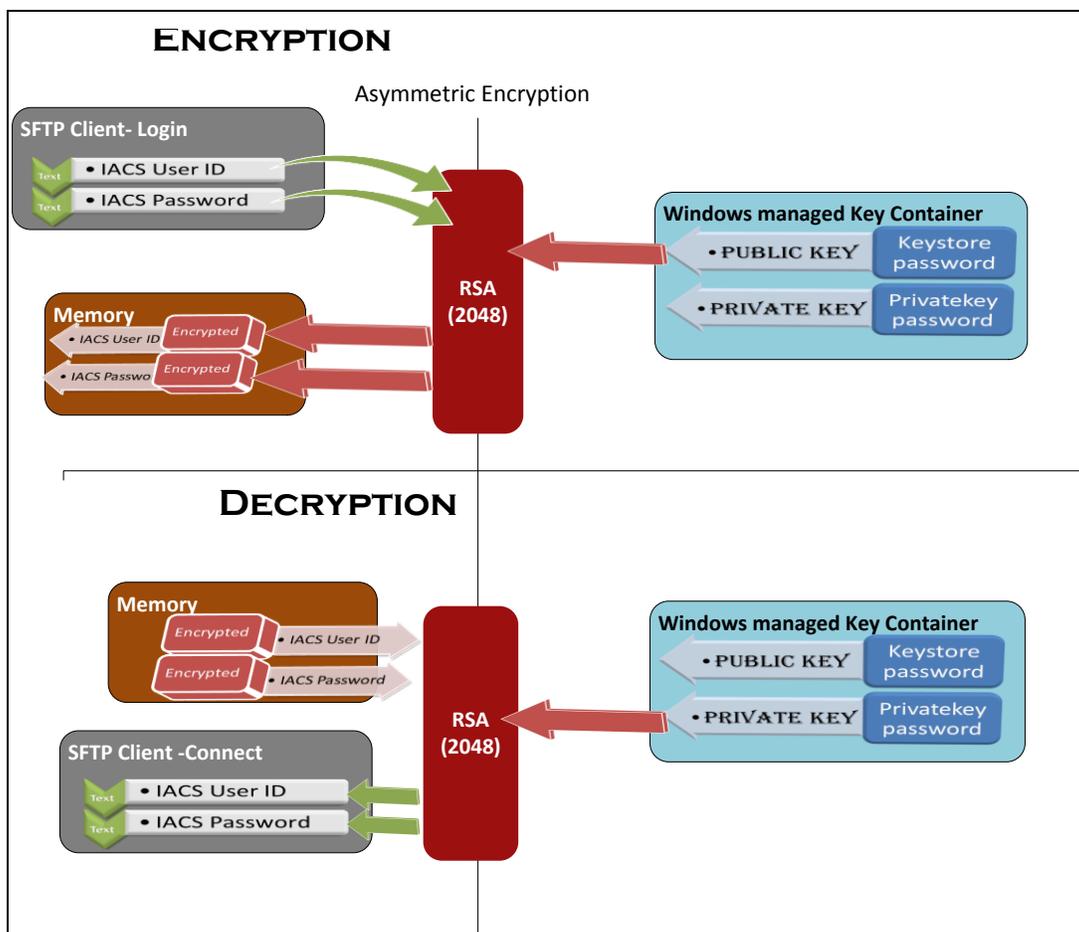
11. .NET Client API

11.1 Security

Note: The .NET Client release from April 28, 2014 does not include the encryption of login credentials as that feature is still under development. This section depicts the planned design and is subject to change. This guide will be updated as required when the security implementation is completed.

When the RC Client starts, the user credentials are provided because they are stored in encrypted form in memory. Figure 5: Encryption and Decryption Process shows the processes used to safeguard the IACS user credentials from exposure.

Figure 5: Encryption and Decryption Process



The RC .NET Client uses (RSA) asymmetric encryption algorithms to secure the login credentials.

11.2 .NET API Documentation

This section discusses API methods that can be called for a custom solution to interface with the TIBCO MFT server. If you, as the RC, choose to use the RC .NET client out-of-the-box, skip this section.

11.2.1 Login

Table 14: ESMD.RcClient.Login.LoginProcess Methods describes the RC Client Login Process.

Table 14: ESMD.RcClient.Login.LoginProcess Methods

#	Method	Description
1.	public bool Authenticate()	User login procedure. The username and password properties are encrypted and set only when this method returns TRUE. Returns: TRUE if the user logs in successively.

11.2.2 Inbound

Table 15: ESMD.RcClient.Inbound.Inbound Methods details the RC Client Inbound Process.

Table 15: ESMD.RcClient.Inbound.Inbound Methods

#	Method	Description
1.	public SortedList<long, string> GetNotifications(string remoteDownloadDirectoryPath , string filePattern)	This method connects to the TIBCO MFT server and checks for any available notifications. Parameters: <ol style="list-style-type: none"> 1. remoteDownloadDirectoryPath – The remote directory path to download from as a String. 2. filePattern – The File Name Pattern to look for as a String. Returns: A list for file names sorted by last modified time, oldest first.
2.	public string PullDocument(string remoteDocName, string localDocName)	This method is used to pull the document (i.e., zip file) from the TIBCO MFT server using the remoteDocName and saves it locally in the "temp" directory as the localDocName. Parameters: <ol style="list-style-type: none"> 1. remoteDocName – The remote file to pull as a String 2. localDocName – The local file name to save as a String. Returns: Error message if any. A null return value means downloading succeeded.

#	Method	Description
3.	public string ExtractDocument(string zipFileName, string targetDirectory)	<p>Extracts the zip file that was downloaded from the TIBCO MFT server.</p> <p>Parameters:</p> <ol style="list-style-type: none"> 1. zipFileName - The local zip file to extract 2. targetDirectory - The target directory to place the extracted contents. <p>Returns: The directory name - the location where the extracted file(s) stored in the local file system.</p>
4.	public bool ProcessMedicalDocumentatio n(string remoteDocumentName)	<p>This is the "housekeeping" method. It does the following:</p> <ol style="list-style-type: none"> 1. Pulls the zip file from the TIBCO MFT server using the PullDocument() method based on the name passed to it to the "temp" directory. 2. Extracts the zip file into the "download" directory using the ExtractDocument() method. 3. If extraction fails, calls the Acknowledge() method with an error event. 4. After successful extraction, verifies the extracted payloads against the checksum in the metadata file using the CheckPayloads() method. 5. If checksum fails, calls the Acknowledge() method with an error event. 6. If checksum passes, calls the Acknowledge() method with a success event. <p>Parameters:</p> <ol style="list-style-type: none"> 1. remoteDocumentName - The remote document name to pull and process. <p>Returns: The Boolean status of the processing for that document.</p>
5.	public string Acknowledge(RCPickupNotific ation rcPickupNotification)	<p>Generates the pickup notification for a downloaded document. If the ErrorInfo object is populated, it generates an error pickup notification. If the ErrorInfo object is null, it generates a pickup notification.</p> <p>Parameters:</p> <ol style="list-style-type: none"> 1. rcPickupNotification - The RCPickupNotification object. <p>Returns: The compressed file name (in TIBCO naming conventions) created in the output directory as a String.</p>

#	Method	Description
6.	public bool CheckPayloads(string localExtractedDirectory, ESMDDocument[] esmdDocuments)	Checks the payload files against the metadata from the package. Parameters: 1. localExtractedDirectory – The directory in which the payloads were extracted to as a File. 2. esmdDocuments – The payloads metadata captured in ESMDDocument objects. Returns: The status of the checksum verification.

11.2.3 Outbound

Table 16: ESMD.RcClient.Outbound.Outbound Methods details the esMD RC Client Outbound Process.

Table 16: ESMD.RcClient.Outbound.Outbound Methods

#	Methods	Description
1.	public SortedList<long, string> GetOutboundDocuments(string outboundDir, string filePattern)	This method is used to retrieve the list of outbound documents in the "output" directory to be pushed. Parameters: 1. outboundDir – The local "output" directory to push files from as a String. 2. filePattern – The file name pattern to push as a String. Returns: A list of file names (without a directory path).
2.	public string PushDocument(string localDocName, string remoteDirectory)	This method used to push a local compressed document from the "output" directory to the TIBCO MFTserver. Parameters: 1. localDocName _ – The name of the file to push as a String. 2. remoteDirectory –The remote directory name to push to as a String. Returns: an error message if any. A null return value means uploading succeeded.

11.2.4 Utilities – PMDPA Result

Table 17: Manual Submission of PMDPAResult details the Methods to submit the PMD PA Result.

Table 17: Manual Submission of PMDPAResult

#	Methods	Description
1.	public SubmitPMDPADeterminationRequest PromptForInfo()	This method is used to prompt the user for information needed to populate the SubmitPMDPADeterminationRequest object. It reads the input from the command line. Returns: The SubmitPMDPADeterminationRequest object populated with the data provided by the user.
2.	public string CreateCompressedTIBCOFileForPMPDPARequest(SubmitPMDPADeterminationRequest submitPMDPADeterminationRequest)	This method is used to create the XML file and compress it into a TIBCO file. Parameters: 1. submitADMCDeterminationRequest – The SubmitPMDPADeterminationRequest object to use. Returns: The compressed outbound file name ready to be pushed by the outbound process.

11.2.5 Utilities - Encryption

Note: The .NET Client release from April 28, 2014 does not include the encryption of login credentials as that feature is still under development. This section depicts the planned design and is subject to change. This guide will be updated as required when the security implementation is completed.

Refer to Table 18: EMSD.RcClient.Encryption.EncryptionUtil Methods for details on the EMSD.RcClient.Encryption.EncryptionUtil methods.

Table 18: EMSD.RcClient.Encryption.EncryptionUtil Methods

#	Methods	Description
1.	public string EncryptCredential(string credential)	This method is used to encrypt the IACS login credentials using an RSA Public Key from the key container. Parameters: 1. credential – user's login name or password to encrypt as a String. Returns: The encrypted credential.

#	Methods	Description
2.	public string DecryptCredential(string credential)	This method is used to decrypt the IACS login credentials using an RSA Private Key from the key container. Parameters: 1. credential – user's encrypted login name or password. Returns: The decrypted credential.

11.2.6 Utilities - Handshake

Refer to Table 19: Remote Troubleshooting for details on the ExecuteHandshake method.

Table 19: Remote Troubleshooting

Methods	Description
public bool ExecuteHandshake()	This sample method invokes a call to the TIBCO MFT server to pass login information to assist in remote troubleshooting. Returns: TRUE if handshake succeeded.

11.3 Logs

The RC .NET Client Sample application is a Windows desktop application. All log messages are written to the RcClient.log file.

11.4 Utilities

RC Client provides a popup window for the user to invoke all the features it performs:

1. Start the RC Client
2. Handshake
3. Manually submit the PMDPA Response Messages

12. Error Codes

Table 20: Error Codes lists the types of error codes and their descriptions. These codes are used to populate the ErrorInfo object inside the error pickup notification XML (e.g., R_TID_Pickup_Error_Request.xml).

Table 20: Error Codes

Error Type	Error Code	Description
UNZIP ERROR	534	ESMD_534 - RC Client processing error (Unzip failure). Please resubmit.
CHECKSUM ERROR	535	ESMD_535 - RC Client processing error (Checksum issue). Please resubmit.
METADATA ERROR	536	ESMD_536 - RC Client processing error (Metadata issue). Please resubmit.

13. Contacts

Table 21: Support Points of Contact list for esMD.

Table 21: Support Points of Contact

Contact	Phone	Email	Hours of Operation
CMS esMD Help Desk	(443) 832-1856	CMSesMDHelpdesk@qssinc.com	Regular Business hours 9 AM to 5 PM ET

Appendix A: New User Registration

The following section provides instructions for creating a new IACS user account.

1. To register in IACS, access the CMS website, copy and paste the following link in your browser, or press Ctrl and click the link:
<https://idm.cms.hhs.gov/idm/user/welcome.jsp>

Figure 6: New User Registration shows the Request Access window that opens.

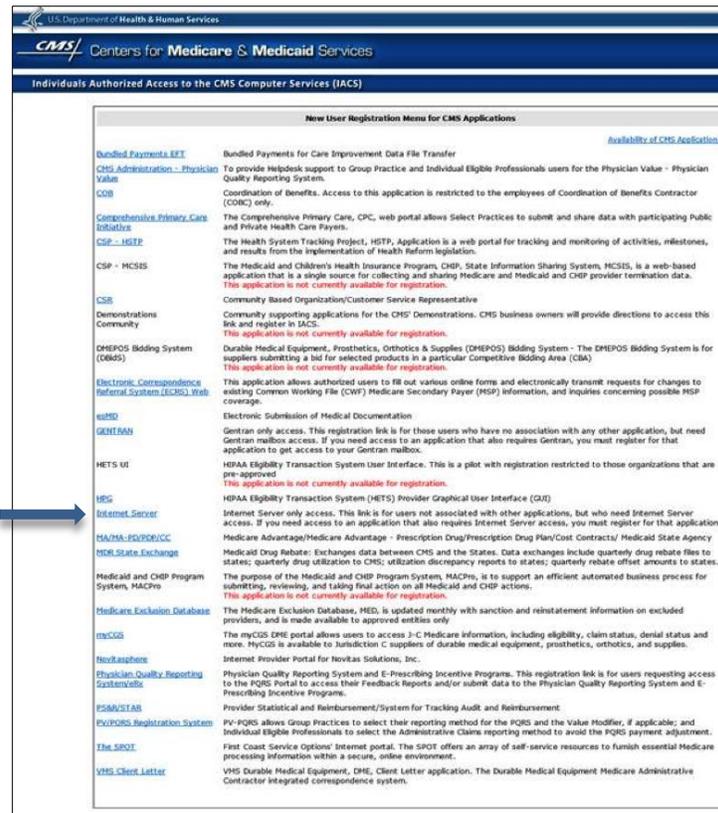
Figure 6: New User Registration



2. Click the **New User Registration** hyperlink indicated at the arrow in Figure 6: New User Registration.

Figure 7: New User Registration Menu shows the next window that opens.

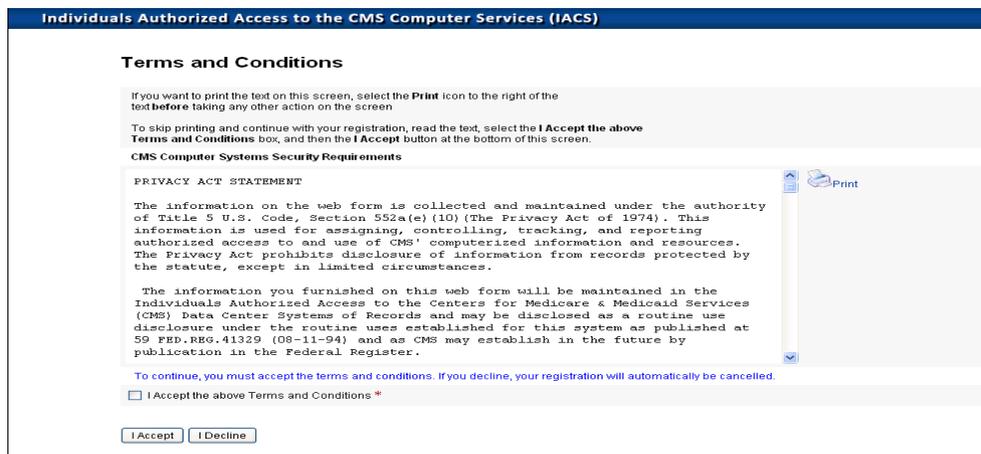
Figure 7: New User Registration Menu



3. Click the **Internet Server** hyperlink indicated at the arrow in Figure 7: New User Registration Menu.

Figure 8: Terms and Conditions shows the Privacy Act Statement and the Rules of Behavior, which are the required terms and conditions for accessing CMS computer systems.

Figure 8: Terms and Conditions



Type the code accurately (without spaces or characters) in the Verification Code field shown in Figure 10: Email Verification. The verification code will not work properly if cut or copied and pasted into the screen.

Note: You have up to 30 minutes to access your email, obtain the verification code, and enter it on this page.

Figure 10: Email Verification

If you do not receive the verification E-mail, click the **Re-send verification code** link to the right of the Verification Code field.

You can request a resend up to three times or contact the Help Desk. If you realize you may have entered an incorrect e-mail address, you must start the entire process again. After three unsuccessful email attempts, you must start the entire process again.

7. After you successfully enter the verification code, click **Next**.

The Contact Information page is highlighted, as shown in Figure 11: New Registration - Contact Information and now contains pre-populated fields.

Figure 11: New Registration - Contact Information

8. As on the previous pages, fill in the required fields indicated by an asterisk (*) and click **Next**.

Once the data is validated, the system displays the **Authentication Questions** page, shown in Figure 12: Authentication Questions.

Figure 12: Authentication Questions

Individuals Authorized Access to the CMS Computer Services (IACS)

Authentication Questions

Please answer at least 2 of the following questions, and then select "Next" to proceed with registration.

Question	Answer
What is your grandmother's maiden name?	esmd
What was the model of your first car?	esmd
What is the middle name of your oldest cousin?	esmd
What was the name of your first pet?	
What was your childhood phone number?	
What was the first name of your first boyfriend?	
What was the first name of your first girlfriend?	
What is the name of your first elementary school?	
What was your childhood street name?	
What was the name of your first employer?	
What was your grandfather's profession?	
What was the name of your first college roommate?	
Where was your wedding reception held?	

9. Answer a minimum of two Authentication Questions to complete your registration and then click **Next**.

These answers will be used to validate your identity if you need to recover your user ID or password using IACS' **Forgot your User ID?** or **Forgot your Password?** self-service features.

The system displays the **Review Registration Details** page, shown in Figure 13: Review Registration Details.

Figure 13: Review Registration Details

Individuals Authorized Access to the CMS Computer Services (IACS)

Review Registration Details

The following is the information you entered on the New User Registration Form. Please review the information below to verify correctness.

- To modify any of the information, click **Edit**.
- If the information is correct and you wish to proceed, click **Submit**.

First Name:	John Doe	MI:	N	Last Name:	Admin
Title:	Mr.	Suffix:	Jr.	Professional Credentials:	BA
Social Security Number:	*****9999				
Date of Birth:	10/19/1972				
E-mail:	pthompson-2012_02-4570@dm.com				
Office Telephone:	456-456-4522 X452				
Company Name:	ABCD Inc.			Company Telephone:	451-452-4529 X453
Address 1:	101 Main Street			Address 2:	Suite 102
City:	Baltimore	State/Territory:	MD	Zip Code:	45274-4535
User Type:	esMD				
Role:	esMD Admin				

Authentication Questions

Question	Answer
What is your grandmother's maiden name?	esmd
What was the model of your first car?	esmd
What is the middle name of your oldest cousin?	esmd

10. Review your information.

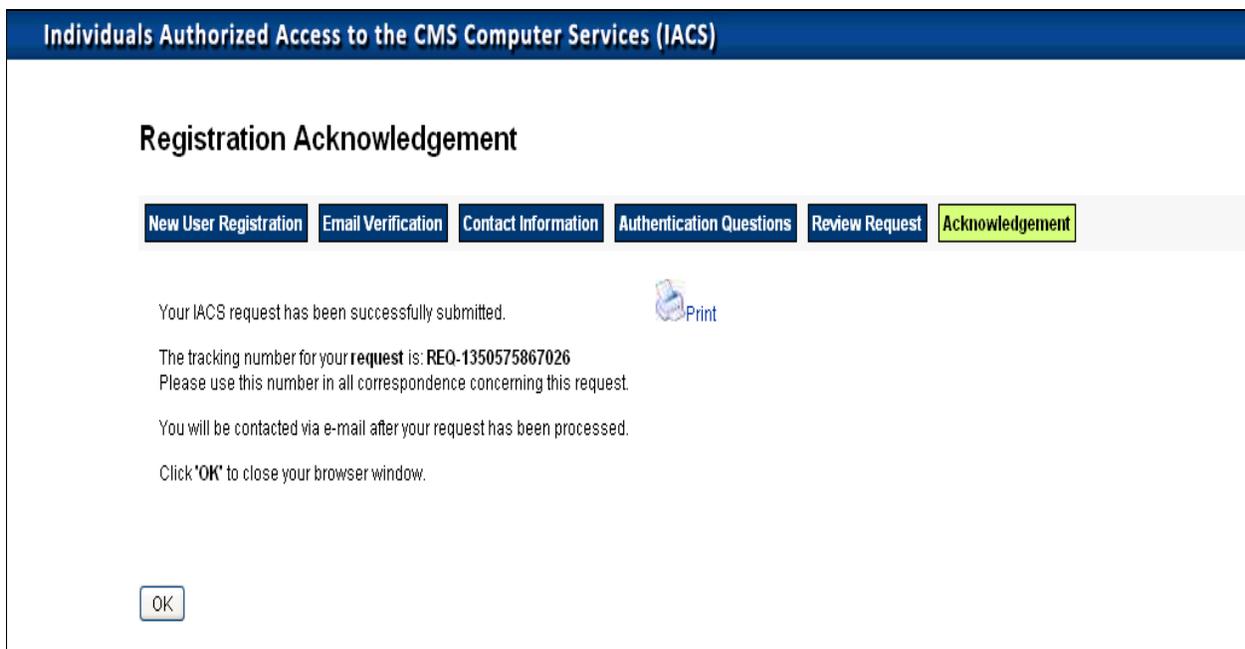
Note: If you need to change anything, click the **Edit** button to make the change. The New User Registration pages will be displayed to make any changes. However, you cannot change the E-mail and Confirm E-mail fields using the **Edit** button.

11. Click the **Submit** button when you are satisfied that your registration information is correct.

The Acknowledgement is displayed, shown Figure 14: Registration Acknowledgement, indicating your registration request was successfully submitted and a request tracking number was assigned. You should record and use this number when you have questions about the status of your request. You will be contacted by e-mail within 48 hours.

Note: You can print your registration information by clicking the **Print** icon.

Figure 14: Registration Acknowledgement



12. Click **OK** (this is required to complete your registration).

Appendix B: Registered User Login Instructions

This appendix provides instructions for existing IACS account users.

1. To login into IACS, access the CMS website.
2. Copy and paste the link in your browser, or press Ctrl and click the link:
https://idm.cms.hhs.gov/idm/user/welcome.jsp

Figure 15: Registered User Login Window shows the Request Access window that opens.

Figure 15: Registered User Login Window

Please select one of these options:

Request Access

[New User Registration](#) - Select this link to register in IACS

[Registered User Login](#) - Select this link to login to use IACS

[Forgot Your User ID?](#) - Select this link to retrieve your IACS User ID

Warning Warning Warning UNAUTHORIZED ACCESS TO THIS COMPUTER SYSTEM IS PROHIBITED BY LAW REFERENCE TITLE 18 U.S.C. SECTION 1030 This is a Centers for Medicare & Medicaid Services (CMS) computer system. This computer system is provided for the processing of Official U.S. Government information. All data contained on this system is owned by CMS and, for the purpose of protecting the rights and property of CMS, be monitored, intercepted, recorded, read, copied, or captured in any manner and disclosed in any manner by authorized personnel. If you are not authorized access to this system you must immediately exit. Federal, State, and/or local laws may provide criminal penalties for any person illegally accessing or using a Government owned or operated computer system. THERE IS NO RIGHT OF PRIVACY IN THIS SYSTEM If you are authorized to access this system, you are subject to the requirements of the Office of Information Services' Security Handbook. The Handbook incorporates Article 35 of the 2004 CMS/AFGE Master Labor Agreement (June 9, 2004), governing bargaining unit employees. System security personnel may give to law enforcement officials any potential evidence of crime found on this system. USE OF THIS SYSTEM BY ANY USER, AUTHORIZED OR UNAUTHORIZED, CONSTITUTES CONSENT TO THIS MONITORING, INTERCEPTION, RECORDING, READING, COPYING, OR CAPTURING AND DISCLOSURE.

3. Click the **Registered User Login** hyperlink indicated at the arrow.

Figure 16: Terms and Conditions shows the Privacy Act Statement and the Rules of Behavior, which are the required terms and conditions for accessing CMS computer systems.

Figure 16: Terms and Conditions

Individuals Authorized Access to the CMS Computer Services (IACS)

Terms and Conditions

If you want to print the text on this screen, select the **Print** icon to the right of the text before taking any other action on the screen

To skip printing and continue with your registration, read the text, select the **I Accept the above Terms and Conditions** box, and then the **I Accept** button at the bottom of this screen.

CMS Computer Systems Security Requirements

PRIVACY ACT STATEMENT

The information on the web form is collected and maintained under the authority of Title 5 U.S. Code, Section 552a(e) (10) (The Privacy Act of 1974). This information is used for assigning, controlling, tracking, and reporting authorized access to and use of CMS' computerized information and resources. The Privacy Act prohibits disclosure of information from records protected by the statute, except in limited circumstances.

The information you furnished on this web form will be maintained in the Individuals Authorized Access to the Centers for Medicare & Medicaid Services (CMS) Data Center Systems of Records and may be disclosed as a routine use disclosure under the routine uses established for this system as published at 59 FED. REG. 41329 (08-11-94) and as CMS may establish in the future by publication in the Federal Register.

To continue, you must accept the terms and conditions. If you decline, your registration will automatically be cancelled.

I Accept the above Terms and Conditions *

4. After you read the content and agree, click the **I Accept** button.

Figure 17: My Profile is the next window that opens.

Figure 17: My Profile



5. Select and click Modify Account Profile as indicated by the red arrow in Figure 17: My Profile.

Figure 18: Modify Account Profile is the next window that opens and the fields will be populated with the user's information.

Note: The fields in the example are empty for security purposes.

Figure 18: Modify Account Profile

U.S. Department of Health & Human Services
CMS Centers for Medicare & Medicaid Services
 Individuals Authorized Access to the CMS Computer Services (IACS)

Modify Account Profile

Modify Account Profile | Email Verification | Review Request | Acknowledgement

User Information

User ID:

Title: First Name: Last Name: Suffix:

Middle Initial: Professional Credentials:

Date of Birth: Valid Date of Birth format is mm/dd/yyyy

E-mail: Valid E-mail address format is user@internetprovider.domain. List of allowed domains: com, gov, net, org, us, mil, biz, edu, vi, pr, md, coop

Professional Contact Information

Office Telephone: Ext.: Valid Telephone Number Format is XXX-XXX-XXXX

Company Name: Company Telephone: Ext.:

Country:

Address 1: Address 2:

City: State/Territory: Zip Code:

Access Request

Select Action:

Select Application: Availability of CMS Applications

Justification for Action:

6. Verify user information is correct within the populated fields.
7. Under Access Request, in the Select Action field, select and click Add Application.
8. The same screen above will refresh.
9. Select and click Select Application.
10. Click the Internet Server_application.
11. Enter the appropriate information in the Justification for Action field.

Appendix C: XML Message Details

Table 22: XML Message Details provides details of the XML message.

Note: TID stands for the Transaction ID.

Table 22: XML Message Details

Business	#	Folder	File Name	Process
PMDPA	1	P_TID	P_TID_Pickup_Request.xml	Outbound
	2	R_TID	R_TID_Pickup_Virus_Scan_Error_Response.xml	Inbound
	3	N_TID	N_TID_Pickup_HIH_Status_Response.xml	Inbound
	4	R_TID	R_TID_Pickup_Error_Request.xml	Outbound
	5	E_TID	E_TID_PMDPA_Review_Result_Request.xml	Outbound
	6	R_TID	R_TID_PMDPA_Review_Result_Virus_Scan_Error_Response.xml	Inbound
	7	R_TID	R_TID_PMDPA_Review_Result_Validation_Error_Response.xml	Inbound
	8	N_TID	N_TID_PMDPA_Review_Result_HIH_Status_Response.xml	Inbound
ADR	9	P_TID	P_TID_Pickup_Request.xml	Outbound
	10	R_TID	R_TID_Pickup_Virus_Scan_Error_Response.xml	Inbound
	11	N_TID	N_TID_Pickup_HIH_Status_Response.xml	Inbound
	12	R_TID	R_TID_Pickup_Error_Request.xml	Outbound
ADMC	13	P_TID	P_TID_Pickup_Request.xml	Outbound
	14	R_TID	R_TID_Pickup_Virus_Scan_Error_Response.xml	Inbound
	15	E_TID	N_TID_Pickup_HIH_Status_Response.xml	Inbound
	16	R_TID	R_TID_Pickup_Error_Request.xml	Outbound
APPEAL	17	P_TID	P_TID_Pickup_Request.xml	Outbound
	18	R_TID	R_TID_Pickup_Virus_Scan_Error_Response.xml	Inbound
	19	N_TID	N_TID_Pickup_HIH_Status_Response.xml	Inbound
	20	R_TID	R_TID_Pickup_Error_Request.xml	Outbound
RAC Request	21	P_TID	P_TID_Pickup_Request.xml	Outbound
	22	R_TID	R_TID_Pickup_Virus_Scan_Error_Response.xml	Inbound
	23	R_TID	R_TID_Pickup_HIH_Status_Response.xml	Inbound
	24	R_TID	R_TID_Pickup_Error_Request.xml	Outbound

Acronyms

Table 23: Acronyms

Acronym	Literal Translation
ACL	Access Control List
ADMC	Advance Determination of Medicare Coverage
ADR	Additional Documentation Request
API	Application Programming Interface
BDC	CMS Baltimore Data Center
CERT	Comprehensive Error Rate Testing
CMS	Centers for Medicare & Medicaid Services
DMAC	DMC MAC
DME	Durable Medical Equipment
ECM	Enterprise Content Management
esMD	Electronic Submission of Medical Documentation
HHS	Department of Health and Human Services
HIH	Health Information Handler
HIPAA	Health Insurance Portability & Accountability Act of 1996
IACS	Individuals Authorized Access to CMS Computer Services
ID	Identifier or Identification
Kbps	Kilobits Per Second
MAC	Medicare Administrative Contractor
MB	Megabytes
MFT	Managed File Transfer (in TIBCO product)
MHZ	Megahertz
NPI	National Provider Identifier
NTFS	New Technology File System
OID	Object Identifier
PA	Prior Authorization
PDF	Portable Document Format
PERM	Payment Error Rate Measurement

Acronym	Literal Translation
PMD	Power Mobility Devices
POC	Point of Contact
RAC	Recover Audit Contractor
RC	Review Contractor
RSA	Rivest, Shamir & Adleman (public key encryption technology)
SFTP	Secured File Transfer Process
SSN	Social Security Number
TID	Temporary ID
XML	Extensible Markup Language
W3C	World Wide Web Consortium

Glossary

Table 24: Glossary

Term	Definition
Application Programming Interface	In computer programming, an Application Programming Interface (API) specifies how some software components should interact with each other and can be used to ease the work of programming graphical user interface components.
Centers for Medicare & Medicaid Services	The Centers for Medicare & Medicaid Services (CMS) is the Department of Health and Human Services (HHS) agency responsible for Medicare and parts of Medicaid.
Electronic Submission of Medical Documentation	Electronic Submission of Medical Documentation (esMD) is a mechanism for submitting medical documentation via an internet gateway connecting Providers to the CMS. In its second phase, esMD will enable Medicare Review Contractors (RCs) to electronically send claim related Additional Document Request (ADR) letters to Providers when their claims are selected for review.
Extensible Markup Language	Extensible Markup Language (XML) is a set of rules for encoding documents electronically. It is defined in the XML 1.0 Specification produced by the World Wide Web Consortium (W3C) and several other related specifications; all are fee-free open standards.
Health Information Handler	A Health Information Handler (HIH) is any company that handles information on behalf of a provider or Durable Medical Equipment (DME) supplier. Examples include Release of Information vendors, Health Information Exchanges, Regional Health Information Organizations, Electronic Health Record vendors, and Claim Clearinghouses.
National Provider Identifier	The National Provider Identifier (NPI) is a unique identification number for use in standard health care transactions. The NPI is issued to health care providers and covered entities that transmit standard Health Insurance Portability & Accountability Act of 1996 (HIPAA) electronic transactions (e.g., electronic claims and claim status inquiries).
Object Identifier	Object Identifier (OID) is an identifier used to name an object. In computer security, OIDs serve to name almost every object type in X.509 certificates, such as components of Distinguished Names.

Term	Definition
PMD PA	Power Mobility Devices Prior Authorization (PMD PA) is a covered item of Durable Medical Equipment (DME) that is in a class of wheelchairs that includes power wheelchairs.
Review Contractor	A Review Contractor (RC) is an entity designated as a recipient of requested medical documentation. Examples are Recover Audit Contractors (RACs), Medicare Administrative Contractors (MACs), Durable Medical Equipment (DME) Medicare Administrative Coordinators (DMACs), and Payment Error Rate Measurement (PERM) or Comprehensive Error Rate Testing (CERT) contractors.
Social Security Number	A Social Security Number (SSN) is a unique identification number assigned to individuals by the SSA.

Record of Changes

Table 25: Record of Changes

Version Number	Date	Author/Owner	Description of Change
0.9	4/21/2013	Yong Luo	Initial draft
1.0	4/28/2014	John Dondapati	Final draft
1.0	4/30/2014	Melony Stehlik	Updated IACS Registered User Instructions Updated Table of Contents Updated List of Figures.
1.0	4/30/2014	John Dondapati	Updated the XML Messages Removed the Property utilities Updated the references.
1.1	5/14/2014	Melony Stehlik	Updated Error Codes
1.2	6/10/2014	Melony Stehlik	Updated Section 4. Updated Table 3: RC Client Sample Configuration file, removed _Download. Deleted Section 5.3, SFTP Utility. Cross referenced Sample Messages in Section 7 to appropriate code in Appendix C; Table 1: XML Message Details. Updated Section 9 RC Client Workflow diagram. Updated Approvals page.

Approvals

The undersigned acknowledge that they have reviewed the User Manual and agree with the information presented within this document. Changes to this User Manual will be coordinated with, and approved by, the undersigned, or their designated representatives.

Signature: _____ Date: _____

Print Name: Braeyon Terry-Connor

Title: Contracting Officer's Representative

Role: CMS Approving Authority