|  |
| --- |
| **Design Document- Process Bulk Update file via back-end** |

|  |  |
| --- | --- |
| **Process Bulk update File via back-end** | **<<Version No:>>** |
| **Version No:v00.02** | **<<Approved By>>** |
| **2136 –Project Eagle** | **<<Department:>>** |
| **Department: New Product and Innovations** |  |

**Document Control**This document contains proprietary information of Mpower Softcomm Pvt. Ltd. No part of this document may be reproduced, stored, copied, or transmitted in any form or by means electronic, mechanical, photocopying or otherwise, without the explicit written Permission of Mpower Softcomm Pvt. Ltd.

© 2015 All rights reserved. MPower Softcomm Pvt. Ltd.

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author/Reviewer** |
| 12/04/2016 | v00.01 | Document Creation | Suma B.Y |
| 15/04/2016 | V00.02 | Note added in this session | Suma B.Y |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Content**

[1.1 Document Purpose 3](#_Toc448481372)

[1.2 Scope of Document 3](#_Toc448481373)

[1.3 Assumptions 3](#_Toc448481374)

[1.4 Intended Audience 3](#_Toc448481375)

[1.5 Acronyms and Abbreviation 3](#_Toc448481376)

[1.6 References 4](#_Toc448481377)

[2. Design Overview 5](#_Toc448481378)

[3. Feature Design 6](#_Toc448481379)

[3.1 File Upload mechanism 6](#_Toc448481380)

[3.2 Security 7](#_Toc448481381)

[Encryption/Decryption for voucher 7](#_Toc448481382)

[3.3 Scheduler mechanism for File pulling. 8](#_Toc448481383)

[3.4 Bulk File Integrity Check 8](#_Toc448481384)

[3.4.1 File not being processed in past 9](#_Toc448481385)

[3.4.2 Header and Trailer 10](#_Toc448481386)

[3.5 Reports 10](#_Toc448481387)

[3.6 Logging 10](#_Toc448481388)

[3.7 Concurrency 11](#_Toc448481389)

[3.8 Error Codes 11](#_Toc448481390)

[4. Feature Flow 11](#_Toc448481391)

[5. Traceability Matrix 13](#_Toc448481392)

[6. Open Issues 13](#_Toc448481393)

Introduction

## Document Purpose

This document presents the high level design of “Process Bulk Update file via back-end”.

## Scope of Document

This document provides the complete design of the “Process Bulk Update file via back-end”, process that shall be used as a reference by the development, testing and implementation teams. The encrypted Bulk file shall be provided to OTA by the provider on request from the client (e.g. Edcon, OTA etc.).

## Assumptions

1. The Bulk Update file shall be supplied by the Edcon.

2. The Bulk Update file can have multiple denominations.

3. The Edcon file has been taken as reference while writing this document.

## Intended Audience

The document shall serve its benefits to the following groups:

1. Project management team
2. Designers and Architects.
3. Developers.
4. Testers.
5. Implementation Team.
6. QA

## Acronyms and Abbreviation

Refer 2136\_Terms\_And\_Conditions\_v00.03

## References

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Name** | **Version** | **Date** | **Company/Organization** |
| 2136\_M04H\_Process Bulk Update File via back-end\_v01.00 20160105 | 01.00 | 20160105 | RMCS |
| OTA\_Architecture\_v0.11 | 0.11 | 20150630 | MPower |

# Design Overview

This process shall file provided by Bulk update provider on the of request client (e.g.OTA).

The Bulk update provider shall send encrypted Bulk update file either by email or by SFTP to RMCS SFTP. IT support of RMCS shall move the encrypted file from RMCS SFTP site to OTA SFTP site. OTA shall be having the scheduler, which will ping the OTA SFTP site to check the new bulk file arrival. As soon as a new bulk stock file is available, OTA shall copy the file from OTA SFTP site to OTA via Admin portal.

OTA shall then decrypt the encrypted bulk file and check the file integrity. OTA shall also check the integrity of the file. Finally OTA shall send the file uploaded load report to IT support.

Note : We should create the following folder structure where the pe\_sftp server is deployed:

OTA\_Share

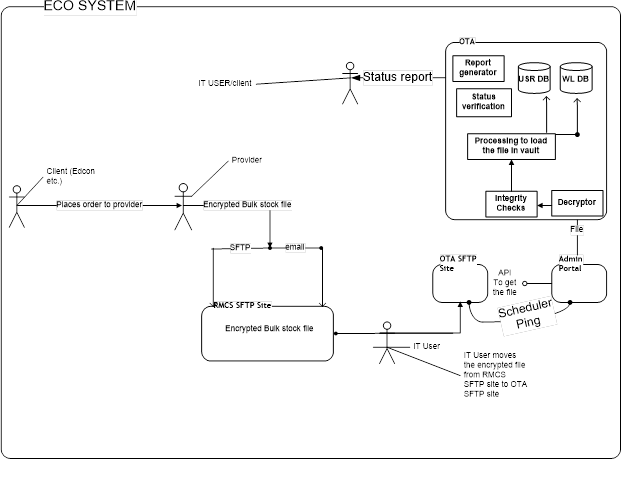
-> BulkFile

* BulkFilePending
* BulkFileError
* BulkFileSuccess

BulkFilePending: The uploaded file will be stored in this folder.

BulkFileError: The file containing error records is stored in this folder temporarily until the file is sent to IT user.

BulkFileSuccess: After processing all the records the file is stored in this folder.



# Feature Design

## File Upload mechanism

* **New File Layout🡪**

Sample file name: Customer request PIN reset 06102014\_Bulk\_Update.txt

Refer: TPC Prepaid Services Bulk Stock Interface

* **File Layout🡪**

File contain three sections.

1. Header- One per file.

2. Detail-pipe limited.

3. Trailer- not applicable.

* **Service Rules🡪**

1. File Provider should drop the file at some particular time, which shall be known to IT support of RMCS.

2. The file shall be transferred via SFTP or mailed from Bulk file provider.

3. OTA SFTP site shall be having a folder structure for each client or chain (e.g. Edcon etc.), this main folder shall be having subfolders per supplier who will be providing the stock for that particular chain.

4. IT support of RMCS shall move the file to an appropriate folder on the OTA SFTP site.

* **Business Rules 🡪**

Refer 2136\_Business Rules Inventory\_v0.06.xlsx

**OTA File Upload Mechanism🡪**

* Admin shall be having a scheduler which will keep on pinging the OTA SFTP site for new file at every interval of N minutes. N is configurable parameters with default value as 30 minutes.
* If there is/are new file/s available on OTA SFTP site, OTA SFTP site shall call the load API to upload the file in admin file system.
* Admin portal to call the file upload API to copy the file in OTA file system and move the original encrypted file into OTA archive.
* OTA shall load the Bulk file into database.

## Security

1. The Bulk update file provider shall send the encrypted Bulk file. The decryption shall be done with the public key of supplier.

2. The OTA shall exchange the public/private key with each supplier.

### Encryption/Decryption for voucher

[12:25:36 PM] Suma Y: AES (Advanced encryption standard)  The AES algorithm is symmetric i.e. single key shall be used for encrypting/ decrypting the data. Encryption/Decryption for encrypted Bulk file.

OTA shall exchange the private/public key with each of the MNO/supplier. Upon receiving the encrypted bulk file, OTA shall determine the source of the file and decrypt the bulk file using the private key of the source. If the decryption of the file fails, the audit log will be updated with decryption of file failed with reason “ERR\_FILE\_DECRYPTION\_FAILED”.

The private key of the source shall be shared with the OTA. OTA shall

[12:25:58 PM] Suma Y: keep the record of the key.

To decrypt the encrypted bulk update file, OTA shall be using 3DES algorithm.

## Scheduler mechanism for File pulling.

A scheduler is set with the ping details of each folder that contains the encrypted Bulk file. The scheduler time shall be configured through Admin portal. The admin shall have the flexibility to change the ping location, ping time etc. The scheduler picks the file and moves it to the OTA file system via the admin portal.

**Scheduler**



## Bulk File Integrity Check

OTA shall perform the below validation checks on the decrypted bulk file to ensure that

The received file has not been processed before. If the processed has already been processed, then the file is invalid and update the file validation status with reason “ERR\_INVALID\_FILE\_1”

The file has a valid header. If the file does not have either of the header, then the file is invalid and update the file validation status with reason “ERR\_INVALID\_FILE\_2”

The format of the file is correct. If the file does not comply with a valid file format, the files is invalid and update the file validation status with reason “ERR\_INVALID\_FILE\_3”.

OTA shall maintain the record of the folder structure from where a particular file is being copied. (e.g. file name+OTA SFTP folder structure).

### File not being processed in past

The decrypted file name shall be of the type “Customer request PIN reset 06102014\_Bulk\_Update.txt”. OTA shall also be keeping the record of the OTA SFTP folder from where the file is being copied.

If the response from the Database in NULL, then the file is good to go for the next processing.



### Header and Trailer

Create the structures which contain the header and trailer attributes of the stock files for all suppliers. File processor shall determine the supplier and match the header and trailer attributes of the file with pre-defined structures for that supplier.

**Refer🡪 Customer request PIN reset 06102014\_Bulk\_Update.txt**

## Reports

The Bulk update load report shall be send to recipient based on pattern matched mechanism. The process of sending email shall be automated. If there is/are any error/s during the file’s structural integrity then OTA shall mention all the details in the load report.

Report shall be send from the IFA-OUT-EMAIL. All the details of the reports .e.g. recipient’s email ID,

File’s structural integrity failure.

## Logging



File processor shall be updating the audit log for all encryption/decryption, validation and file upload status. This logging shall help to find the root cause in case of any failure or error.

## Concurrency

The File upload system does not support concurrent file upload. Concurrent file upload can compromise the file integrity in database and can end up system throwing multiple errors.

## Error Codes

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Error Code Mnemonics** | **Description** |
| 1 | ERR\_FILE\_DECRYPTION\_FAILED | Decryption of the bulk file failed. |
| 2 | ERR\_INVALID\_FILE\_1 | File has already been processed in past |
| 3 | ERR\_INVALID\_FILE\_2 | File does not have either header or trailer |
| 4 | ERR\_INVALID\_FILE\_3 | File does not comply with a valid file format. |

# Feature Flow

**File Upload Feature Flow**





# Traceability Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement Spec** | **Interface Spec** | **Design Spec** | **Development** | **Testing** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Open Issues

Not Applicable.