EC2X-QuecOpen Important Data Backup and Update Solution

LTE Module Series

Rev.EC2XQuecOpen_Important_Data_Backup_and_Updare_Solution

Date: 2018-03-07

Status: Preliminary

www.quectel.com

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
7th Floor, Hongye Building, No.1801 Hongmei Road, Xuhui District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://quectel.com/support/sales.htm

For technical support, or to report documentation errors, please visit:
http://quectel.com/support/technical.htm
Or email to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2017-09-08	Ramos ZHANG	Initial
1.1	2018-03-07	Ramos ZHANG	Updated reference solution

Contents

1 Introduction

Principle: The customer's app (or important configuration parameters) is stored in both the system partition and the usr_data partition. Check whether the apps of the two partitions exist when every time start up, if the app does not exist, copy it from another partition. If app exist both in two partition, check version number, if the version is different, copy the latest app to another partition. Because these two partitions will not be damaged at the same time, it can guarantee that the app will not be lost. Besides, even if the app has been upgraded and the file system restored, the app can be guaranteed to be the latest version.

Some important configuration parameters also can take same solution to maintain.

2 Partition Related With Customer

Partition related with customer refer to the partitions that can be adjusted for customer application and configuration parameter data storage. All other unrelated partition cannot be used for any modification.

Table 1: Partition related with customer

Partition	Default Size	Available Size	Partition Format	Purpose
usr_data	124.25M	88.75M	UBI file system	Store read and write user data, application and parameter configuration.
sys_back	58M	\	Image	System UBI image backup, can be resized according to the actual situation
system	88.75M	About 19M	UBI file system	rootfs volume, readable-writable partition, can store customer app and parameter configuration.

NOTES

1. If integrating quectel fota upgrade function is required, usr_data must reserve at least 60M space. Sys_back is used to back up the system partition, If system partition holds a large amount data, the sys_back partition should be adjusted accordingly, the minimum size is the customer system ubi size+6M, and in the later use, system cannot have big data storage increase.
2. The usr_data partition is loaded in the /etc/init.d/find_partitions.sh script by default, and if usr_data partition mount fails, it will be reformatted. Therefore, there is risk that customer' app or data stored in this partition may be lost . Please refer to the code in SDK.

3 How to implement backup and restore of customer APP

3.1 Reference Program Flow

After booting and running the /etc/init.d/find_partitions.sh script (because all the partitions of the module are loaded after this script is run by default), run an automatic backup restore detection script .

app_auto_backup_restore.sh (Only for reference)

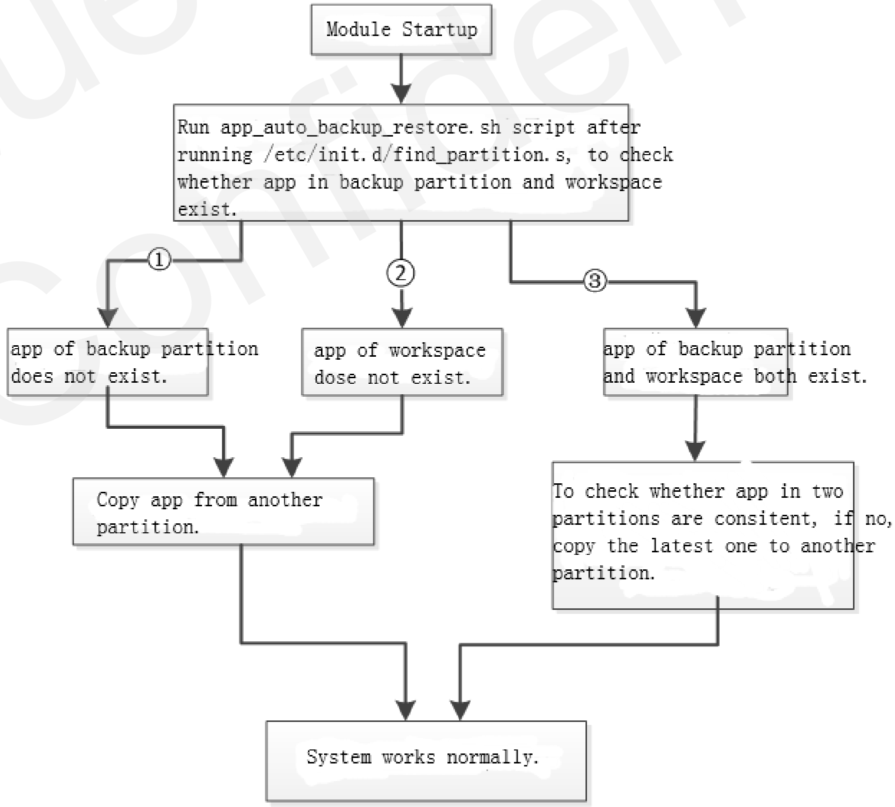
Module Startup
Run app_auto_backup_restore.sh script after running /etc/init.d/find_partition.s, to check whether app in backup partition and workspace exist.
①
②
③
app of backup partition does not exist.
app of workspace dose not exist.
app of backup partition and workspace both exist.
Copy app from another partition.
To check whether app in two partitions are consitent, if no, copy the latest one to another partition.
System works normally.

Figure 1: Script Processing Flow Chart

NOTES

1. Due to the cache mechanism of Linux file operations, if the power is suddenly turned off during backup, the backup app may be incomplete, but the code is not "detected", and cause next restore cannot run. Solution: Make sure to copy the file first, it is better to judge whether the copy operation is completed, then do the sync action, and finally copy the tag file of version.
2. The app version tag file should be saved separately from the apps directory (which stores all apps and even important parameters), then do the sync action and finally copy the tag file of app version. Please refer to the provided script. If the customer's important parameters and applications are stored in other directories, please refer to the script to add the corresponding two directories to achieve the same purpose.
3. The two folders backed up with each other should be placed in different partitions because if the kernel detects the file system corruption, will format the entire partition.

3.2 Important Data Backup and Update Scenarios

This chapter still take script as example, meanwhile, analyze the scenes and characteristics that may appear in actual operation. The important parameters and the application are located in the /usrapp/apprun/apps directory, backup directory is /usrdata/appbackup/apps, and below are four scenes.

A. The module run for the first time after downloading, the /usrdata/appbackup/apps backup partition is empty, the /usrdata/appbackup/app_ver.txt tag also does not exist. This script will copy contents of the /usrapp/apprun/apps folder to the backup partition, then copy the tag file and complete the backup.
B. If the partition of usr_data is damaged, the usr_data partition will be reformatted when /etc/init.d/find_partitions.sh is loaded, at this time, the backup data and backup tag are lost. This script will copy contents under the /usrapp/apprun/apps folder, and version number in /usrapp/apprun/app_ver.txt to the usr_data backup partition.
C. After the customers' app successfully upgraded, the file version number of the /usrapp/apprun/app_ver.txt is increased. Even if the backup area is not updated, the script will automatically copy the file under usrapp/apprun/apps to the /usrdata/appbackup/apps directory after detecting the change of version number at the next startup, and then copy the app version file to complete synchronization.
D. The customers' app has been upgraded and the backup partition has been updated. However, in the later use, the system partition maybe damaged and restored. At this time, the contents of the /usrapp/apprun/apps folder are restored to the factory state. This script will judge that the version of the backup parttion is higher, and the content under /usrapp/apprun/apps folder in backup partition will be synced to the /usrapp/apprun/apps folder and the version number will be also synced.

3.3 Test Method

(1) Delete apps and version tag file under usr_data, then reboot (or rerun the script), check whether the content under system partition is synced to usr_data partition.

(2) Delete apps and version tag file under system partition, then reboot (or rerun the script), check whether the content under usr_data partitions synced to system partition.

(3) Add files to the system partition apps and increase the version tag file number by one, then reboot (or rerun the script), check whether the content under system partition is synced to usr_data partition.

(4) Add files to the usr_data partition apps and increase the version tag file number by one, then reboot (or rerun the script), check whether the content under usr partition is synced to system partition.

(5) After normal operation, erase the system partition by fastboot method. After rebooting the system, check whether the app exists in the system partition and it is consistent with the one in usr_data partition.

(6) After normal operation, erase the usr_data partition by fastboot method. After rebooting the system, check whether the app exists in the usr_data partition and it is consistent with the one in system partition.

4 Reference Script

```
#!/bin/sh
# Copyright (c) 2014, The Linux Foundation. All rights reserved.
#
# example app_auto_backup_restore.sh
#
# if app.bin    stored in system partition (/usrapp/app.bin, /usrapp/app_Ver)
# app backup partition is usr_data in (/usrdata/appbackup/app.bin, /usrdata/appbackup/app_Ver)
#
#     /usrapp/apprun/apps
#     /usrapp/apprun/app_ver.txt
#     /usrdata/appbackup/apps
#     /usrdata/appbackup/app_ver.txt
```

```
Ver1=0
Ver2=0
app1="/usrapp/apprun/apps"
app1_Ver="/usrapp/apprun/app_ver.txt"
#just support 1,2,3,4,5,6,7 ......
app2="/usrdata/appbackup/apps"
app2_Ver="/usrdata/appbackup/app_ver.txt"
# you'd better make sure the usr_data partition is mount ok in here
if [ ! -f $app1_Ver ];then
     Ver1=-1
else
     Ver1=`cat $app1_Ver`
fi
if [ ! -f $app2_Ver ];then
     Ver2=-1
else
     Ver2=`cat $app2_Ver`
fi
```

```
if [ "$Ver1" -eq "$Ver2" ];then
     echo "app verion same , exit !!!"
     exit 0
fi
```

```
echo -n " app version update now!!!!"
if [ "$Ver1" -gt "$Ver2" ];then
        echo -n " app update to usr_data partition /usrdata/appbackup/"
        rm -rf $app2
        rm -rf $app2_Ver
        mkdir -p $app2
        cp -rf $app1    $app2
        sync
        cp -rf $app1_Ver    $app2_Ver
        sync
fi
if [ "$Ver1" -lt "$Ver2" ];then
        echo -n " app    update to system partition    /usrapp/apprun/"
        rm -rf $app1
        rm -rf $app1_Ver
        mkdir -p $app1
        cp -r $app2 $app1
        sync
        cp -rf $app2_Ver $app1_Ver
        sync
fi
```