How to do debugging.

---- an introduction to how to use this program and debug it.

Nov 21 2008

Introduction

At the first of all, I will describe the structure of program. Then, I will give the definitions and functions to each one of all sub-folders. Main purpose of this program is designed to communicated with union-pay mainframe server. Then, it would receive the response messages from union-pay,and parsed the received message into proper format,which is used as understandable for programmers or other staffs.

Architecture of System

See the analysis of system requirements. Or other technical documents.

Structure of System

The system contains about these folders which is defined as beneath.

```
Roots-----|
          |  COMMUN
          |  DOC
          |  ENCRYPT
          |  LOG
          |  MAIN
          |  MAKE
          |  POS
          |  PUBLIC
          |  README
          |  SQL
          |  SYSTEM
          |  TEST (NOT BE USED)
```

Abilities

Before we describe the system in details, some abilities you should acquire, or you should know that.

1. The knowledge of LINUX system
2. The knowledge of C-STD LIB programming .
3. The knowledge of how to use GDB
4. The knowledge of how to use MAKE
5. Know how to use VI OR VIM OR EMACS as your source edit tools.

Functions of Sub-Folders

INCLUDE

In INCLUDE folder, it contains the head files, which is used by source code, and the declarations of variables and definitions of variables,or some public global variables and global function which is used by all procedures in other units, named by dot h.

SOURCE

In SOURCE folder, it contains the source file. All implementations of procedures were implemented in source file, named by dot c.

System Folders description

1. COMMUN

   It contains tow sub-folders, One is INCLUDE folder, the other one is SOURCE. It is used to communicate with external systems. Such as, third party companies. In these tow folders, we can provider interfaces to external systems.

2. DOC

   It contains all documents need by programmer or other company staffs .

3. ENCRYPT

   Same as COMMUN folder, it also contains tow sub-folders, INCLUDE, SOURCE.

   When we sent the message to union-pay, all messages should be identified MAC. The value of MAC is generated by encryption machine. We use the hardware encryption to protect our messages from illegal accesses. All procedures, involving in encryption, are implemented in SOURCE folder. All declarations&definitions of procedures , variables could be founded in INCLUDE folder.

4. LOG

It contains the system log files. To ensure system trace-abilities, by providing log is a good idea. System generate three types log-file, The 1st One, raw message log, named by dot raw. The 2nd, SQL statements logs, named by dot SQL. The 3rd, Error logs, named by dot err. Before or after sending, the sent message should be written into raw log-files in raw format. The similar rules were applied in SQL log-files and ERROR log-files.

5. MAIN

   It contains the main procedure. The 1st procedure, with which system started.

6. MAKE

It contains the rules of compiling the source code. How to compile the source code and make the executable file? Just type the “make” command in shell environment. If you had some new modules were added to system, you should modify the Makefile to keep up with the latest changes.

7. POS

Same as we described above, it also contains tow folders, the 1st one is INCLUDE, the 2nd is SOURCE. In include folder, it contains all definitions and declarations of procedures, and variables. In source folder, all procedures involved in communication with POS mainframe were implemented in this folder. Such as, Sending the request messages, receiving the respond messages, parsing the received messages, answering the request message from upper level system.

8. PUBLIC

It contains two sub-folders, INCLUDE, SOURCE. The global functions and global variables were defined and declared in include folder and implemented in SOURCE folder.

9. README

All description files were stored in this folder.

10. SQL

It contains interfaces with DB server. In INCLUDE folder, it stored declarations and definitions of interfaces. In SOURCE folder, it stored the implementations of interfaces. All functions or procedures, which communicated with DB could be found in this folder.

11. SYSTEM

It contains the procedures,which is used to deal with system user's command.

12. TEST (obsoleted)

13. CONFIG

It contains the configuration file needed by system. Before you start up the system, You should copy this configuration file into same folder where the executable file was stored.

Configuration file description

Before the system was executed or in executing phase, System needed some fixed values. Such as, remote machine IP, remote machine listening port, Time-Out value of TCP/IP, and field lengths of POS transaction protocols, etc. We write these fixed value into a file, called configuration file, named by “sysconfig.cfg”. If you want to comment out a line, you can put a character-- # in front of this line.

The detailed explanation as following. (main items in configuration file)

LISTEN_IP --- The system will listen the client request on LISTEN_IP.

LISTEN_PORT--- The system will start listening on LISTEN_PORT.

REMOTE_IP --- The POS mainframe Listening IP address .

REMOTE_PORT --- The POS mainframe Listening Port.

TIMEOUT_VAL --- The timeout value of TCP connections. (Unit: second)

ENCYPT_IP --- The IP address of Encryption machine.

ENCYPT_PORT -- The Port Of Encryption machine.

MAC_VALUE --- The working key of MAC of Encryption machine.

PIN_VALUE --- The working key of PIN of Encryption machine.

FIELDS_VAL --- The lengths of each filed of POS transactions protocol.

MSGSWITCH --- The flag of whether to show message or not.

LOGWRITE --- The flag of whether to write the log-files or not.

The preparation before debug

1. The working key of MAC of encryption machine. If we do not do encryption in test phase, you can leave it alone.
2. The working key of PIN of encryption machine. Same as above.
3. The POS Mainframe serve's IP and Port.
4. Terminal ID.
5. Sender Institution ID .
6. Merchant Type and ID.

etc .

How to count the BITMAP value

The most important field of POS transactions is BITMAP. It indicates which filed is on, which field is off. The Bitmap has 128 bits. Each bit represents a protocol field. If 14th field is available in a transaction protocol, for an example, the 14th bit of bitmap is 1, otherwise, 0.

As following shown.

```
80       20       00       00       00       41       40       10
10000000 00100000 00000000 00000000 00000000 01000001 01000000 00010000
00       78       d7       80       80       00       00       00
00000000 01111000 11010111 10000000 10000000 00000000 00000000 00000000
```

```
80        20       00        00        00       41       00       10
10000000  00100000 00000000  00000000  00000000 01000001 00000000 00010000
04        00       00        00        00       00       00       00
00000100  00000000 00000000  00000000  00000000 00000000 00000000 00000000
```

Others

If you have any questions, you can mail it to me or call me !

My email { mixtrue at hotmail dot com }

My Mobile: 15158132949