PATAC 5 字节 DLL 应用需求

PATAC 5 Bytes DLL Application Requirements

Version：V1.0.0

Date：2016-1-22

Author：Sun Wang

Contents

1. 特殊声明/Special Statement:

本文描述的 PATAC 5 字节 DLL 皆为“SGM SERVICE PROGRAMMING SYSTEM (SPS) NON-DISCLOSURE AGREEMENT”和“GA –Security Access_ 0x27_Service-Unlocking_Supplier_Requirements_v1.4.4” 中 定 义 的 “sa015bcr.dll”。

In this document, the PATAC 5 Bytes DLL is the “sa015bcr.dll” which be defined in the “SGM SERVICE PROGRAMMING SYSTEM (SPS) NON-DISCLOSURE AGREEMENT” and “GA –Security Access_ 0x27_ Service- Unlocking _Supplier_Requirements_v1.4.4”.

2. GMLAN 协议的特殊需求/Special Requirement For GMLAN：

使用 PATAC DLL 的 GMLAN 协议模块，在开发刷新使用的 Utility 文件时，必须做如下应用场景的区分：

- ME 工具执行的 Op-Code$27 的 AC2 必须为:$F0；
- AE 工具货执行的 Op-Code$27 的 AC2 必须为:$20；

上述应用场景的区分，由 Op-Code$56 实现。

During the ECU utility file development, those GMLAN ECUs which use PATAC DLL shall identify the following use cases:

- The AC2 of the Op-Code$27 which implemented by the ME tool: $F0;
- The AC2 of the Op-Code$27 which implemented by the AE tool: $20;

The different use case shall be identified by the Op-Code$56.

3. PATAC DLL 接口定义/PATAC DLL Interfaces Definition：

PatacDLL 接口定义为/PatacDLL interfaces definitions：

int PatacSha(char* szPara2, char* szPassword, char* szPara3, char *szHash);

4. 关键参数定义/Key Parameters Definition：

PATAC DLL 需要调用 4 个 char*类型的关键参数。

PATAC DLL shall use 4 key parameters. The format of each key parameter shall char*.

int ir = PatacSha(a,b,c,d);

- szPara2 为 Seed 值，长度为 5 字节的 16 进制数，5 字节的最后一个字节为 Index 值/ The szPara2 is the seed. The length of seed shall be 5 bytes. The format of the seed shall be Hex. The last bytes of the seed shall be index value.
- szPassword 为 password 密文，长度为 62 字节的 16 进制数/ The szPassword is password. The length of the password is 62 bytes. The format of the password shall be Hex.
- szPara3 为算法号(AC0)，长度为 1 字节的 16 进制数/The szPara3 is the algorithm number(AC0). The length of the szPara3 is 1 byte. The format of the szPara3 shall be Hex.
- szHash 为计算结果返回的缓冲区地址信息，用于存储 5 字节 16 进制的输出 Key 值/ The szHash is the address info which used to store the output of PATAC DLL (key value). The length of the key is 5 bytes. The format of the key shall be Hex.

5. Appendix. DLL 调用示例/The DLL Calling Example：

```
// dll_test_explicit.cpp : Defines the entry point for the console application.
//
#include "stdafx.h"
#include <windows.h>
#include <iostream>

typedef int (*PatacFunc)(char *, char *, char *, char *);

int _tmain(int argc, _TCHAR* argv[])
{
    PatacFunc _PatacFunc;
    TCHAR const    *path = _T("patacdll.dll");
    HINSTANCE hInstLibrary = LoadLibrary(path);
    char *a = "0101010104";    // 10个16进制字符
    char *b = "04a78ea8913c64161ec743e1185cc83a5c7741e29bf54
e92dedaf26923cf4adb11fd691265d11fada9ca4a2fbba635c7652ce0f922a09b040801364206
c8";
    char *c = "04";                    // 2个16进制字符
    char *d = new char[11];     //  定义输出缓冲区
    d[10] = 0;    // the end of the string.

    if (hInstLibrary)
    {
        _PatacFunc = (PatacFunc)GetProcAddress(hInstLibrary, "PatacSha");
     std::cout<< "load ok." << std::endl;
        if (_PatacFunc)
        {
            std::cout << "test return:" << _PatacFunc(a,b,c,d) << std::endl;
        std::cout << "sha:"<< d <<std::endl;
        }
        FreeLibrary(hInstLibrary);
    }
    else
    {
        std::cout << "DLL Failed To Load!" << std::endl;
    }
    std::cin.get();
    return 0;
}
```