PalmSecure™

U4PS-LCL3-01ENZ3(2)

FUJITSU Biometric Authentication

PalmSecure™ SDK V02

Sample Source for C Language Manual

Professional Edition

(V01)

FUJITSU

◆ Revision History

Revision	Issued Date	Revised Page	Modification Details
Rev. 1	Oct 2020	Entire document	Newly created
Rev. 1.1	Jul 2021	Page 3	• Added Makefile for Linux(armhf) and Linux(arm64). in “1,2 List of Contents”,
		Page 7	• Modified the following descriptions in “2.2 How to Build C Sample Source”. - Added information for Linux(armhf)/Linux(arm64). - Updated the tools for development in each OS.
Rev. 1.2	Nov 2022	Page 7 to 8	• Modified the following descriptions in “2.2 How to Build C Sample Source”. - Modified the build options for Linux(x64). - Added tested settings to Note 4).

◆ Introduction

Thank you for purchasing PalmSecure™ SDK V02 (hereinafter called "this product").
This document describes how to use the Sample source for C language aiming at readers with the following knowledge.

- Basic knowledge of Windows and Linux
- Basic knowledge of C language
- Basic knowledge of the Authentication library
- Basic knowledge of Arm processor

November 2022 : Rev. 1.2

Caution for This Manual

You are required to use this product within the specification described in this document.

Regarding to High Safety Required Usage

This Product is designed, developed and manufactured as contemplated for general use, including without limitation, general office use, personal use, household use, and ordinary industrial use, but is not designed, developed and manufactured as contemplated for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could lead directly to death, personal injury, severe physical damage or other loss (hereinafter "High Safety Required Use"), including without limitation, nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system. You shall not use this Product without securing the sufficient safety required for the High Safety Required Use. If you wish to use this Product for High Safety Required Use, please consult with the sales representatives in charge before such use.

Cautions for Exporting This Product

When exporting or providing this product and this document to a foreign country, check relevant laws such as "Foreign Exchange and Foreign Trade Control Law", and regulations such as U.S. export control law, and follow the necessary procedures.

Warnings

- Reprinting or reproducing this document in part or in whole without permission is forbidden
- Items described in this document are subject to change without prior notice.

PalmSecure is a trademark of Fujitsu Ltd.

Microsoft, Windows, and Visual Studio are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Linux® is a registered trademark or trademark of Linus Torvalds in the United States and/or other countries.

Arm is a trademark or registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

Other company names and product names described in this document are trademarks or registered trademarks of each company.

◆ Abbreviations and Common Terms

Abbreviations and common terms used in this document are as follows:

Abbreviation/ Common Term	Description
This product	Abbreviation for "PalmSecure™ SDK V02".
Sensor	Common term for “PalmSecure Sensor V2” and “PalmSecure-F Pro”.
Authentication library	Abbreviation for “Authentication library V34 Professional Edition”.
C Sample source	Abbreviation for “Sample source for C language V01 Professional Edition”.
Visual Studio 2019	Abbreviation for “Microsoft® Visual Studio® 2019”.

◆ Notations

The following symbols are used in this document.

Symbol	Description
!Caution	Describes things that you have to look out for. You must read it.
★Tip	Provides reference information. Read it as necessary.
>See>	Indicates an item to be referred.

◆ Table of Contents

Chapter1 Introduction of C Sample Source

1.1 Overview

The C Sample source shows the basic usage of the Windows version and Linux version of Authentication library.
If you are developing the Palm vein authentication system application in C/C++, you can check how to call the basic functions of the Authentication library by referring to the C Sample source.
You can also build the C Sample source, run the created CUI application, and experience the basic features of palm vein authentication on a stand-alone configuration.

1.2 List of Contents

The C Sample source provides the following files.

Folder		Stored file	Description
1st Hierarchy	2nd Hierarchy		
PalmSecure Sample Source	source	BioAPI_sample_C_Verify.c	Sample source of functions required for enrollment and verification.
		BioAPI_sample_C_Identify.c	Sample source of functions required for enrollment and identification.
		Makefile	Makefile to create the program for Linux(armhf)/Linux(arm64).

1.3 Hardware and Software Requirements

The following describes the required hardware and software to use the C Sample source.

<table>
<tr><th colspan="2">Hardware and Software Requirements</th><th>Description</th></tr>
<tr><td colspan="2">Hardware Requirements</td><td rowspan="3">Follow the requirements for the Windows version andor Linux version of Authentication library V34 Professional Edition referring to the “Authentication Library Reference Guide”.</td></tr>
<tr><td rowspan="3">Software Requirements</td><td>OS (Note)</td></tr>
<tr><td>Sensor driver</td></tr>
<tr><td>Authentication library</td><td>Windows version and Linux version of Authentication library V34 Professional Edition
(Download the latest version from SDK V02L03 in the SDK V02 Support Website.)</td></tr>
</table>

Note) For information on OSes that the C sample source is tested on, refer to the “System Development Guide”.

Chapter2 How to Use C Sample Source

2.1 Before Using C Sample Source

Prepare the environment to use the C Sample source as follows.

(1) Install the Sensor driver.

>See> For information on how to install the Sensor driver, refer to the “Sensor Driver Installation Guide”.

(2) Install the Authentication library and set the operational environment setting file “PvAPI.INI”.

>See> For information on how to install the Authentication library, and information on the operational environment setting file, refer to the “Authentication Library Reference Guide”.

(3) Confirm firmware version level of the Sensor unit and update the firmware if necessary.

>See> For information on how to confirm the firmware version, refer to the “Sensor Maintenance Tool Operation Guide”.

>See> For information on the latest version of the firmware, refer to the “System Development Guide”.

(4) Set your application key.

Define your “application key” printed in the “License Agreement” to the identifier APPLICATION_KEY (see below) in the "BioAPI_sample_C_Verify.c" and "BioAPI_sample_C_Identify.c" as a string constant.

```
#define  APPLICATION_KEY          "your application key"
```

2.2 How to Build C Sample Source

Locate the C Sample source files, header files and library files of the Authentication library appropriately and then build them.
Note that the build options are different depending on the OS.
The following shows the build options for each OS. (Note 1) (Note 2) (Note 3)

OS	Build options
Windows (x86)	cl BioAPI_sample_C_Verify.c /D WIN32 /I .¥include /link .¥lib¥PvFw.lib
	cl BioAPI_sample_C_Identify.c /D WIN32 /I .¥include /link .¥lib¥PvFw.lib
Windows (x64)	cl BioAPI_sample_C_Verify.c /D WIN32 /I .¥include /link .¥lib¥F3BC4BIO.lib
	cl BioAPI_sample_C_Identify.c /D WIN32 /I .¥include /link .¥lib¥F3BC4BIO.lib
Linux (x64)	gcc BioAPI_sample_C_Verify.c -DUNIX -I ./Inc -L ./LM -lf3bc4bio -lf3bc4com -lf3bc4bsp -lf3bc4mat -lf3bc4cap -Wl,-rpath,./ -o ./BioAPI_sample_C_Verify
	gcc BioAPI_sample_C_Identify.c -DUNIX -I ./Inc -L ./LM -lf3bc4bio -lf3bc4com -lf3bc4bsp -lf3bc4mat -lf3bc4cap -Wl,-rpath,./ -o ./BioAPI_sample_C_Identify
Linux (armhf) (arm64)	Modify and execute the Makefile provided in C Sample source, (Note 4)

Note 1) Prepare a build tool set suitable to your environment.
The C Sample source is confirmed to be built by the following compiler.
Windows (x86) : Microsoft® C/C++ Optimizing Compiler Version 19.28.29914 for x86 (Build Tools for Visual Studio 2019)
Windows (x64) : Microsoft® C/C++ Optimizing Compiler Version 19.28.29914 for x64 (Build Tools for Visual Studio 2019)
Linux(x64) : gcc 4.8.5 (Included in CentOS 7.9)
Linux(armhf) : arm-linux-gnueabihf-gcc (Linaro GCC 6.3-2017.02) 6.3.1 20170109
Linux(arm64) : aarch64-linux-gnu-gcc (Linaro GCC 6.3-2017.02) 6.3.1 20170109

Note 2) For include folder, specify the folder that stores the Authentication library header files.

Note 3) For library folder, specify the folder that stores library file of the Authentication library ("*.lib" for Windows, "libf3bc4bio.so" for Linux).

Note 4) Set your toolchain to the "CROSS" and "CFLAGS" in the Makefile.

Note that the Sample source is confirmed to be built with the following settings after setting the location of the compiler described in (Note 1) to the environment variable.

Linux(armhf) :

- "CROSS" : arm-linux-gnueabihf-
- "CFLAGS" : no setting (set according to your environment)

Linux(arm64) :

- "CROSS" : aarch64-linux-gnu-
- "CFLAGS" : not setting (set according to your environment)

```
CROSS := "set your toolchain"
  .
  .
  .
CFLAGS = -c -g -Wall -DUNIX ¥
         -I ./ -I ./Inc ¥
         -I "set your toolchain"
  .
  .
  .
```

Also, change the include folder and library folder if necessary.

2.3 Running the Built Program

Confirm that the Sensor is connected, and then run the built program by double-clicking it or command-line input.

The C Sample source is programmed to do the following process in sequence.

- **BioAPI_sample_C_Verify**

  (1) Calling "BioAPI_Enroll" 1 time. (Note)

  Note) If enrollment data file "BIRData.dat" exists in the current folder, this process is skipped.

  (2) Calling "BioAPI_Verify" 1 time.

  (3) Calling "BioAPI_Capture" 1 time.

  (4) Calling "BioAPI_VerifyMatch" 1 time.

!Caution **Output files of the program**

BioAPI_sample_C_Verify stores enrollment data "BIRData.dat", capture data "BIRCapData.dat", and silhouette image data "silhouette.bmp" in the current folder.
Delete these files yourself as necessary.

- **BioAPI_sample_C_Identify**

  (1) Calling "BioAPI_Enroll" 2 times.

  (2) Calling "BioAPI_Identify" 1 time.

  (3) Calling "BioAPI_Capture" 1 time.

  (4) Calling "BioAPI_IdentifyMatch" 1 time.

★Tip **Output data of the program**

BioAPI_sample_C_Identify stores enrollment data and capture data on the memory.
The data is automatically deleted when you close the program.

Note that error information is output when an error occurred in the Authentication library.

>See> For error information, refer to the "Authentication Library Reference Guide".


PalmSecure