Security interface documentation

Document management

Version history

date	version	modify the record	author

Content

Content **3**

1 Introduction **5**

2 Key Management Module **5**

2.1 MK/SK Method 5

2.1.1 mfsdk_save_plaintext_key *5*

2.1.1.1 Interface Prototype: 5

2.1.1.2 Function Description: 5

2.1.1.3 Interface Description: 5

2.1.2 mfsdk_save_encrypted_key *6*

2.1.2.1 Interface Prototype: 6

2.1.2.2 Function Description: 6

2.1.2.3 Interface Description: 6

2.1.3 mfsdk_3des_run *7*

2.1.3.1 Interface Prototype: 7

2.1.3.2 Function Description: 7

2.1.3.3 Interface Description: 7

2.1 DUKPT Method 8

2.1.1 dukpt_init *8*

2.1.1.1 Interface Prototype: 8

2.1.1.2 Function Description: 8

2.1.1.3 Interface Description: 8

2.1.2 dukpt_load_init_key 8

2.1.2.1 Interface Prototype: 8

2.1.2.2 Function Description: 9

2.1.2.3 Interface Description: 9

2.1.3 dukpt_get_key 9

2.1.3.1 Interface Prototype: 9

2.1.3.2 Function Description: 9

2.1.3.3 Interface Description: 9

1 Introduction

This document will provide a comprehensive description of the security interface to help application developers better perform secondary development.

2 Key Management Module

2.1 MK/SK Method

2.1.1 mfsdk_save_plaintext_key

2.1.1.1 Interface Prototype:

int mfsdk_save_plaintext_key(int type, int gid, unsigned char * key, unsigned char *kvc);

2.1.1.2 Function Description:

Save the key in plaintext

2.1.1.3 Interface Description:

	Parameter name	Effective value	Description
Input	type		MFSDK_KT_MAINKEY to MFSDK_KT_TRANSKEY
Input	gid		Key Index 0-9
Input	key		16-byte key plaintext

Input	kvc		4 bytes key check
Output	No		
return value			0 successfully

2.1.2 **mfsdk_save_encrypted_key**

2.1.2.1 Interface Prototype:

int mfsdk_save_encrypted_key(int type, int gid, unsigned char * key, unsigned char *kvc);

2.1.2.2 Function Description:

Save the key ciphertext

2.1.2.3 Interface Description:

	Parameter name	Effective value	Description
Input	type		MFSDK_KT_MAINKEY to MFSDK_KT_MAGDEC
Input	gid		Key Index 0-9
Input	key		16-byte key ciphertext
Input	kvc		4 bytes key check
Output	No		

return value			0 successfully

2.1.3 **mfsdk_3des_run**

2.1.3.1 Interface Prototype:

int mfsdk_3des_run(int type, int gid, int mode, unsigned char *ind, int size, unsigned char *outd);

2.1.3.2 Function Description:

Encryption and decryption operation

2.1.3.3 Interface Description:

	Parameter name	Effective value	Description
Input	type		MFSDK_KT_MAINKEY to MFSDK_KT_TRANSKEY
Input	gid		Key Index 0-9
Input	mode		MFSDK_ENCRYPT or MFSDK_DECRYPT
Input	ind		Input data
Input	size		Data size
Output	outd		Output Data
return value			0 successfully

2.1 DUKPT Method

2.1.1 **dukpt_init**

2.1.1.1 Interface Prototype:

void dukpt_init();

2.1.1.2 Function Description:

Dukpt module initial

2.1.1.3 Interface Description:

	Parameter name	Effective value	Description
Input	no		
Output	No		
return value			

2.1.2 **dukpt_load_init_key**

2.1.2.1 Interface Prototype:

int dukpt_load_init_key(unsigned char gid, unsigned char* init_ksn, unsigned char* init_key);

2.1.2.2 Function Description:

Dukpt module initial

2.1.2.3 Interface Description:

	Parameter name	Effective value	Description
Input	gid		Key Index fix 0
Input	init_ksn		Initial Key Serial Number
Input	init_key		16-byte Initial key plaintext
Output	No		
return value			0 successfully

2.1.3 **dukpt_get_key**

2.1.3.1 Interface Prototype:

int dukpt_load_init_key(unsigned char gid, unsigned char* init_ksn, unsigned char* init_key);

2.1.3.2 Function Description:

Dukpt module initial

2.1.3.3 Interface Description:

	Parameter name	Effective value	Description

Input	gid		Key Index fix 0
Input	key		16-byte key plaintext
Input	ksn		Key Serial Number
Output	No		
return value			0 successfully