Link ID^2 安全模组

Doc. No. AN-xxxxx-xx-xx

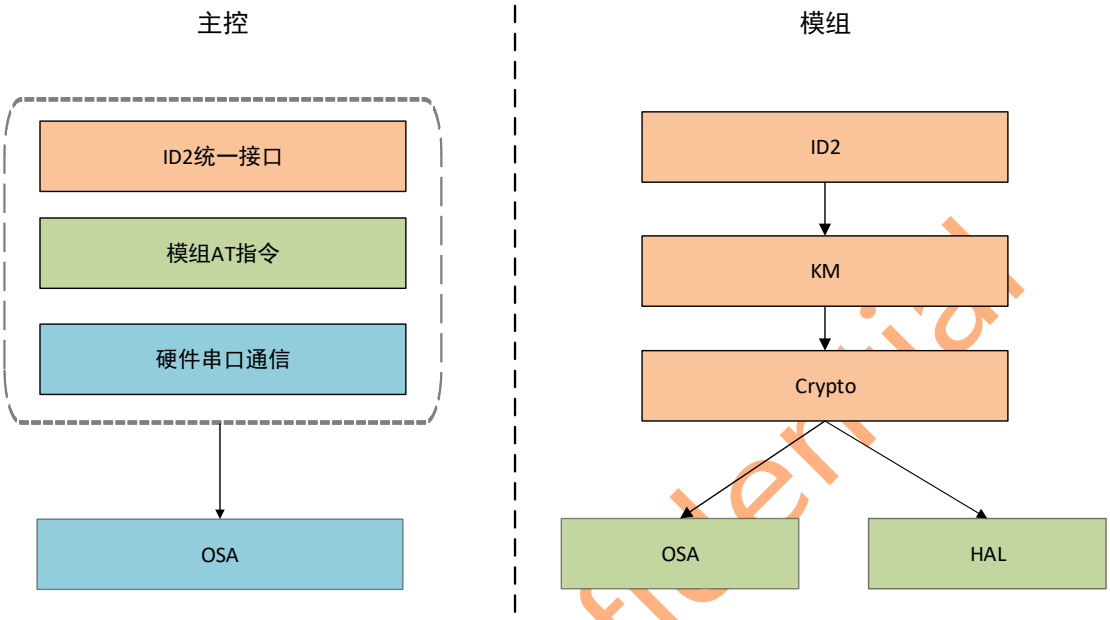
Rev 1.0

Release Date: 2019-09-16

Link ID² 安全模组

一．简介：

Link ID² 安全模组，是指在安全模组中预制 ID² ID 和密钥，并提供基于 ID² 的认证和解密功能；主控通过外接安全模组，可以直接使用 ID² 的能力。



二．模组 AT 指令：

模组 AT 指令封装和解析，提供主控和模组间进行交互的 ID² 安全报文的处理。根据 ID² 提供的能力不同，需适配如下接口：

1. int mdu_id2_init(void)
   - 功能：模组 ID² 初始化。
   - 参数：N/A
   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

2. int mdu_id2_cleanup(void)
   - 功能：模组 ID² 资源释放。
   - 参数：N/A
   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

3. int mdu_id2_get_version(uint32_t* version)
   - 功能：获取模组 ID² SDK 的版本号。
   - 参数：

名称	输入/输出	描述
version	输出	32 位整型数据

   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

4. int mdu_id2_get_id(uint8_t* id, uint32_t len)
   - 功能：获取模组 ID² 的 ID 字符串。
   - 参数：

名称	输入/输出	描述
id	输出	ID² ID 字符串
len	输入	内存的长度，大于等于 ID2_ID_LEN

   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

5. int mdu_id2_get_challenge_auth_code(uint32_t type, const char* random,
   const uint8_t* extra, uint32_t extra_len,
   uint8_t* auth_code, uint32_t* auth_code_len)
   - 功能：获取模组 ID² 不同模式下的认证码。
   - 参数：

名称	输入/输出	描述
type	输入	生成设备端 ID² 认证码的模式， 0 – challenge; 1- timestamp
random	输入	ID² 服务端下发的随机数字串
extra	输入	extra 字串，可选
extra_len	输入	extra 字串的长度
auth_code	输出	生成的设备端认证码字符串
auth_code_len	输入+输出	输入-内存长度；输出-认证码实际长度

   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

6. int mdu_id2_decrypt(const uint8_t* in,
   uint32_t in_len, uint8_t* out, uint32_t* out_len)
   - 功能：模组 ID² 解密
   - 参数：

名称	输入/输出	描述
in	输入	16 进制密文
in_len	输入	密文的长度，最大 4096 字节
out	输出	解密后的 16 进制明文
out_len	输入+输出	输入-明文内存长度；输出-明文实际长度

   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

7. int mdu_id2_get_device_challenge(uint8_t* random, uint32_t* random_len)
   - 功能：获取设备端的随机数字符串，最长 16 字节。
   - 参数：

名称	输入/输出	描述
random	输出	获取的随机数字符串

random_len	输入+输出	输入-内存长度；输出-随机数实际长度

- 返回值：
  IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

8. int mdu_id2_verify_server(const uint8_t* auth_code, uint32_t auth_code_len,
   const uint8_t* device_random, uint32_t device_random_len,
   const uint8_t* server_extra, uint32_t server_extra_len)
   - 功能：验证服务端的认证码
   - 参数：

名称	输入/输出	描述
auth_code	输入	服务端的认证码字符串
auth_code_len	输入	服务端认证码的长度
device_random	输入	设备端随机数字符串
device_random_len	输入	设备端随机数字符串的长度
server_extra	输入	server extra 字符串，可选
server_extra_len	输入	server extra 字符串长度

   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

9. int id2_client_get_secret(const char* seed, uint8_t* secret, uint32_t* secret_len)
   - 功能：获取基于 ID² 生成的设备密钥。
   - 参数：

名称	输入/输出	描述
seed	输入	生成密钥的因子，字符串，以’\0’结尾
secret	输出	生成的设备密钥，可显字符串
secret_len	输入+输出	输入-内存的长度，需大于等于 64 字节； 输出-生成的设备密钥字符串的长度

   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

三．硬件串口通信：

串口驱动集成，为主控和模组之间提供基本的数据交互通道。

1. int mdu_open_session(void **handle)
   - 功能：与模组建立会话连接。
   - 参数：

名称	输入/输出	描述
handle	输出	会话句柄

   - 返回值：
     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

2. int mdu_transmit_command(void *handle,

const uint8_t *req_buf, uint32_t req_len, uint8_t *rsp_buf, uint32_t *rsp_len)

- 功能：发送 AT 指令到模组，并等待模组的响应。
- 参数：

名称	输入/输出	描述
handle	输入	会话句柄
req_buf	输入	模组 AT 指令请求
req_len	输入	模组 AT 指令请求的长度
rsp_buf	输出	模组 AT 指令响应
rsp_len	输入+输出	输入–内存的长度；输出–实际响应的长度

- 返回值：

  IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

3. int mdu_close_session(void *handle)
   - 功能：关闭会话连接。
   - 参数：

名称	输入/输出	描述
handle	输入	会话句柄

   - 返回值：

     IROT_SUCCESS - 成功；其他 – 失败，参见错误码。

四．错误码：

IROT_SUCCESS	0	The operation was successful
IROT_ERROR_GENERIC	-1	The generic error
IROT_ERROR_BAD_PARAMETERS	-2	Input parameters are invalid
IROT_ERROR_SHORT_BUFFER	-3	The supplied buffer is too short for output
IROT_ERROR_EXCESS_DATA	-4	Too much data for the requested operation
IROT_ERROR_OUT_OF_MEMORY	-5	Out of memory
IROT_ERROR_COMMUNICATION	-7	Communication error
IROT_ERROR_NOT_SUPPORTED	-8	The request operation is not supported
IROT_ERROR_NOT_IMPLEMENTED	-9	The request operation is not implemented
IROT_ERROR_TIMEOUT	-10	Communication timeout
IROT_ERROR_ITEM_NOT_FOUND	-11	The item is not exist