
产线烧录 SDK

(VC 版)

Rev 2.1

Release Date: Oct 23, 2017

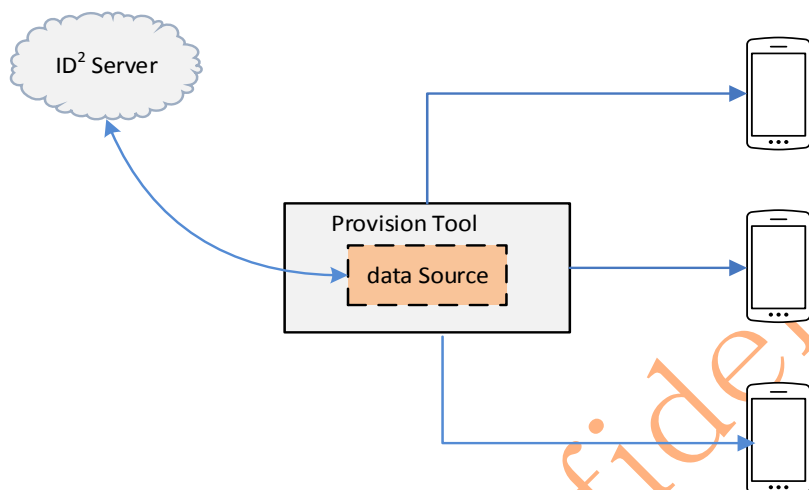
Alibaba Confidential

产线烧录 SDK

一. 简介：

ID²产线烧录 SDK 是由阿里开发和维护，提供给合作厂商，用于集成到厂商烧录工具，完成 ID²在产线的获取和烧录。

ID² Provisioning



请联系合作项目接口同学签署相关 NDA 以获取 SDK。

ID² SDK Release Package：

1. inc: sdk 的库文件。
2. lib: dll 库、许可证文件和日志配置文件。
3. sample: 示例代码
4. genkeypairs.jar: 用于生成厂商产线 rsa1024 公私钥对。
5. ReleaseNotes.txt: 发布说明。

二. 许可证申请：

合作厂商需先申请与项目对应的 ID² 许可证，才能开始 ID² 烧录的测试和生产。

1. 在安装有 java 环境的 PC 机上运行 `java -jar genkeypairs.jar > keypairs.txt` 将生成一对 rsa1024 的公私钥对，如图所示（请妥善保管此文件）：

```
1 Generate key pair:
2 pubKey:
3 MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCilYqbBd88MHOqFiUzbGHCB:
4 privKey:
5 MIICdQIBADANBgkqhkiG9w0BAQEFAASCAl8wggJbAgEAAoGBAKKVipsF3zwwc
```

其中：

pubkey: 申请 ID² 许可证的时候，需要提交到 ID² 服务端。

产线烧录 SDK

privKey: 产线私钥，用于解密 ID²服务端下发的 ID² 密钥。

2. 合作厂商提供以下的信息，发送给阿里负责此项目合作的同事。当前厂商线上申请 ID² 平台尚在搭建中，后续客户将可以直接进行线上申请。

序号	项目	内容
1	芯片名称	
2	公钥	
3	密钥类型 (RSA-1024、AES-128、AES-256)	
4	厂商名称	
5	产品名称	
6	用途 (生成、测试)	
7	配额	

注：

后台根据提交信息生成唯一的许可证序列号，其中包括厂商唯一编码和产品唯一编码。

三．接口说明：

1. ds_stat_t ds_init(ds_cfg_t *cfg)

功能：初始化烧录 SDK，只需启动时做一次

参数：指定烧录工具支持同时烧录的设备数和许可证文件路径

返回：成功 - DS_STATUS_OK；失败 - 参考错误码

2. ds_stat_t ds_create_dev(ds_dev_t *dev)

功能：创建烧录数据的句柄

参数：数据的句柄

返回：成功 - DS_STATUS_OK；失败 - 参考错误码

3. ds_stat_t ds_get_rec_num(ds_dev_t dev, uint32_t *num)

功能：获取 record 的数量

参数：dev - 数据的句柄； num - record 数量

返回：成功 - DS_STATUS_OK；失败 - 参考错误码

4. ds_stat_t ds_get_rec(ds_dev_t dev, uint32_t rec_idx, void *rec, uint32_t *size)

功能：通过 dev 句柄获取知道 index 对应的 record 数据和 record 的大小

参数：当 rec == NULL 时，size 返回指定 index 的 record 的大小

当 rec != NULL 时，rec 返回指定 index 的 record 数据

返回：成功 - DS_STATUS_OK；失败 - 参考错误码

5. ds_stat_t ds_set_dev_prov_stat(ds_dev_t dev, ds_dev_prov_stat_t prov_stat)

功能：烧录工具通过 dev 句柄设置 ID²的烧录状态

产线烧录 SDK

参数：烧录状态

返回：成功 - DS_STATUS_OK；失败 - 参考错误码

6. void ds_destroy_dev(ds_dev_t dev)

功能：注销烧录数据

7. void ds_cleanup(void)

功能：对应 ds_init, 用于 ID²烧录退出时的清理操作

四. 使用说明：

ID² SDK 提供示例代码，可以基于此代码进行调试和测试。

1. 修改示例代码，设置烧录端口数和许可证文件路径：

```
cfg.port_num = threadCnt;  
strcpy(cfg.lic_path, "licenseConfig.ini");
```

2. 配置许可证文件 (licenseConfig.ini)：

```
[config]  
#select server type: 0 means for local test, 1 for factory model  
serverType = 1  
  
#select id^2 synchronize policy: 0 means fast but some waste of id^2, 1 means a little slow but no waste.  
id2SyncPolicy = 1  
  
#The license from id^2 server  
license = "aa7249375d352dd0c0c01a4123456789"  
  
#factory prikey, put by factory Line  
factPrikey = "Base64 Encode RSA1024 Private Key"
```

其中：

serverType: 固定值 1 - factory mode

id2SyncPolicy: 获取 ID² 数据的策略：

0：对应异步（速度较快，但会有一定的浪费）

1：对应同步（按需获取数据，不会产生浪费，使用于网络良好的场景）

license: 申请的许可证序列号

factPriKey: 产线私钥，用于解密服务端下发的 ID² 密钥，对应的公钥用于许可证申请。

3. 配置日志文件 (log.conf)：

```
path=./  
level=ALL
```

其中：

path: 日志输出路径

level: 日志输出的级别：

■ NONE_O: 不输出日志

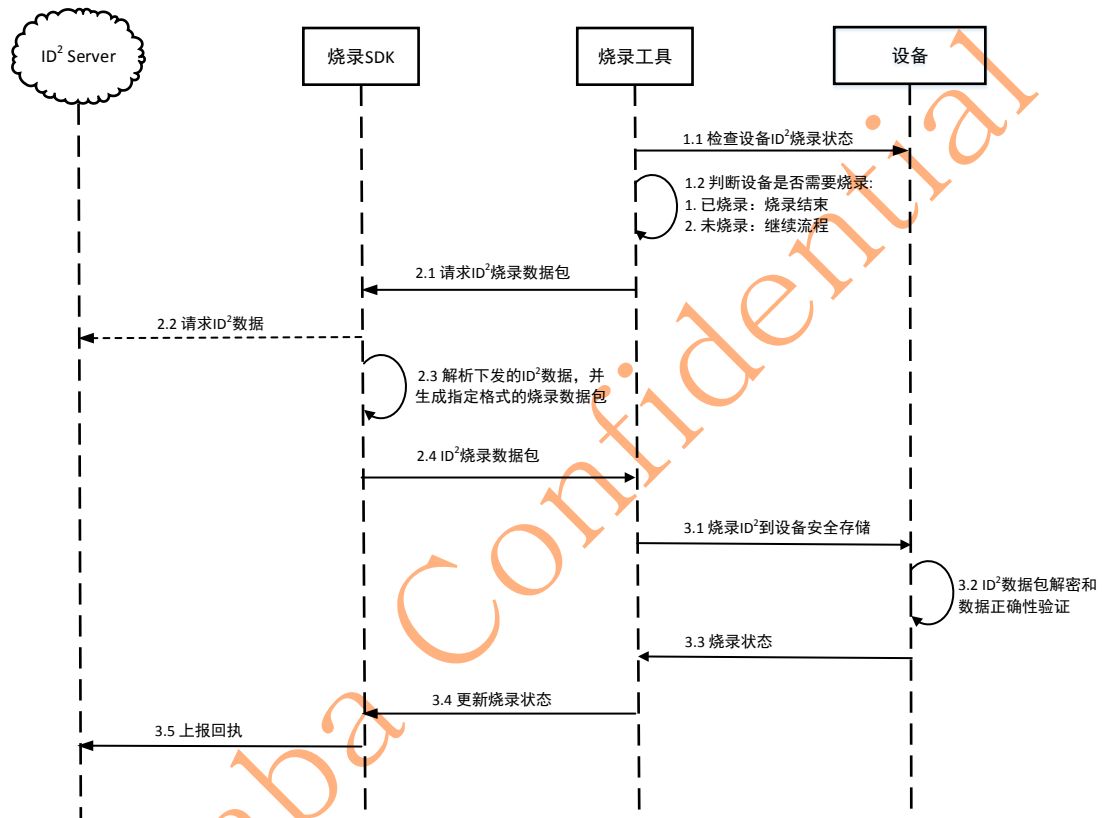
■ INFO_O: 输出 INFO 类型的日志

产线烧录 SDK

- DEBUG_O: 输出 DEBUG 类型的日志
- ERROR_O: 输出 ERROR 类型的日志
- ALL : 输出所有类型的日志

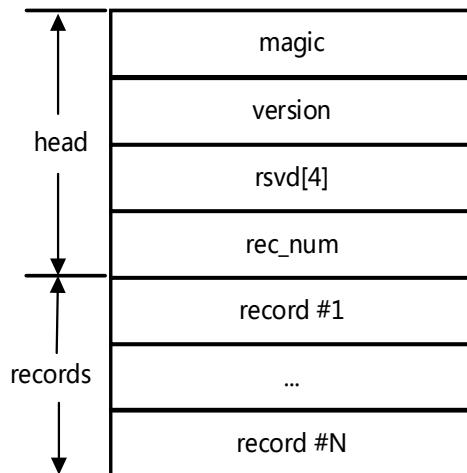
日志配置文件 (log.conf) 需与 dll 库放置在相同目录, 生成的日志以日期命名, 如 2017-10-23.log。

五. 烧录流程:



ID² 烧录数据包格式:

产线烧录 SDK



其中：

magic: 0x764F7250

version: 0x00030000

rsvd[4]: 0x00000000 for each rsvd field

rec_num: record 数量, 调用 ds_get_rec_num 获得

records: record 数据, 调用 ds_get_rec 获得, 按顺序组装生成

六 . Q&A:

1. 如果产线没有网络, 如何进行 ID² 的烧录?

针对此种情况, ID² 产线烧录可以分为两步:

1) 生成烧录数据包:

开发 VC 工具集成 ID² 烧录 SDK, 在连网的环境, 生成 ID² 数据包 (保存在不同文件中), 同时产生 ID² 烧录回执 (set_dev_prov_stat(DEV_PROV_STAT_SUCCESS))。

2) 数据包烧录到设备:

将 ID² 烧录数据包导入产线数据库, 烧录工具从数据库中获得烧录数据, 并烧到设备。

2. 烧录工具不是 VC 开发的, 如何进行烧录工具和烧录 SDK 的集成?

可以采用问题 1 的解决方案, 将 ID² 烧录分为两个独立的流程。