ALIBABA CLOUD

阿里云

IoT设备身份认证

我是芯片/模组厂商

文档版本：20201216

阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。

2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。

4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。

5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。

6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击**设置**> **网络**> **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
Courier字体	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.申请ID²烧录许可证

本文介绍申请ID²烧录授权许可证的流程。

请您注册阿里云账号并完成个人实名认证后，再进行下面的操作。建议您使用阿里云账号申请入驻。

请单击下面的链接了解详细内容：

- 阿里云账号注册流程
- 个人实名认证

步骤一：申请开通产线烧录管理或芯片入驻管理

1. 登录物联网设备认证。
2. 在左侧导航栏选择**扩展服务 > 我的服务**。
3. 单击**芯片入驻管理**或**产线烧录管理**后，填写信息。

   > **说明** 阿里接口人请填写与您联系的阿里云IoT小二姓名，如果没有可以不填。

4. 单击**确定**，提交申请并等待审核。
5. 当您通过审核后，即可看到**芯片入驻管理**或**产线烧录管理**页面。

步骤二：下载产线烧录SDK

1. 在左侧导航栏选择**扩展服务 > 芯片入驻管理 > 工具列表**。
2. 在**工具列表**页面下载配套相应产线的SDK。

   > **说明** 目前SDK支持SE（安全芯片）、TEE、 Secure MCU、软件沙盒等载体类型，包含Java和C两种编程语言的SDK。

3. 下载产线SDK并解压。

   ID² SDK Release Package包括以下文件。

文件	说明
sdk-lib	ID² SDK的库文件。
sdk-sample	示例代码。
genkeypairs.jar	用于生成产线公私钥对的工具。

步骤三：生成产线公私钥

在安装有Java环境的PC机上运行公私钥对生成程序，并获取keypairs.txt文件中的公私钥对。

```
java -jar genkeypairs.jar > keypairs.txt
```

公私钥对信息如下图所示。

```
Generate key pair:
pubKey:
MIGfMA0GCSqGSIb[redacted].UzbGHCB
privKey:
MIICdQIBADANBgk[redacted]sF3zwwc
```

> **注意** 请务必妥善保存密钥，产线烧录环节会使用pubKey拉取ID²。

步骤四：获取ID²烧录许可证

1. 在左侧导航栏选择**扩展服务 > 产线烧录管理**。
2. 在**产线烧录管理**页面，单击**产线管理**页签。
3. 单击**新增产线**。
4. 在**新增产线**对话框，填写产线信息后，单击**确定**提交。

参数	说明
产品型号	实际的产品或设备的型号。
芯片型号	芯片的完整型号。
载体类型	SE、TEE、KM等。
操作系统类型	按照设备实际运行的操作系统信息来填写。
操作系统版本	按照设备实际运行的操作系统信息来填写。

5. 在左侧导航栏选择**扩展服务 > 产线烧录管理**。
6. 单击**新增许可证**，打开**新增许可证**弹窗，填写信息后，单击**确定**提交。

参数	说明
产线	从已经创建的产品中选择。
公钥	厂商公私钥对中的公钥。 公私钥对生成方法，请参见步骤三：生成产线公私钥。
密钥类型	根据实际产品密钥类型选择。
申请配额	填写要烧录的ID²数量，单次烧录不超过100万个。

2.自主验证

本文介绍如何使用设备端适配验证工具完成ID²全链路的调试验证。

前提条件

- 已在设备端上集成安全SDK。获取安全SDK，请访问ID² Client SDK。
- ID²数据已烧录到设备上。若还未完成烧录，请参考烧录ID²到芯片。
- 已获取ID²认证授权。购买ID²认证授权，请访问ID²认证授权-预付费。

步骤一：使用设备端适配验证工具验证ID²

完成ID²设备端适配后，可以通过ID²设备端适配验证工具验证ID²的设备认证和解密功能。设备端适配验证工具适用于不同的载体（如TEE，SE，MCU），流程如下图。

1. 获取设备端调试工具。
2. 编译生成自主验证的固件（id2_app）。
3. 将自主验证的固件烧录到设备。

   调试结果如下图。

步骤二：在服务端验证调试结果

1. 登录物联网设备身份认证。
2. 选择**扩展服务 > 设备端验**。
3. 在**设备端适配验证**页面单击**验证调试结果**，将设备端生成的调试结果粘贴在**验证调试结果**对话框。
4. 单击**验证**，开始设备端适配验证。
5. 验证完成后，在**完成验证**对话框查看验证结果。

步骤三：在设备端解密

1. 获取服务端生成的密文。

   请在步骤二：在服务端验证调试结果描述的**完成验证**对话框中复制密文。

2. 在设备端导入密文。

```c
/*
 * Copyright (C) 2017-2019 Alibaba Group Holding Limited
 */

#include "id2_test.h"

/* Hex String, getting from id2 console */
#define ID2_CIPHER_DATA     "[illegible]"

int main(int argc, char *argv[])
{
    int ret;
    uint32_t cipher_len = 0;
    char *cipher_data = ID2_CIPHER_DATA;

    if (argc >= 1) {
        if (!strcmp(argv[1], "--get_data")) {
            cipher_data = argv[2];
        }
    }

    ret = id2_client_unit_test();
    if (ret < 0) {
        ID2_DBG_LOG("id2 client unit test fail!\n");
        return -1;
    }

    cipher_len = strlen(cipher_data);

    ret = id2_client_generate_authcode();
    if (ret < 0) {
        ID2_DBG_LOG("id2 client generate authcode fail!\n");
        return -1;
    }

    cipher_len = strlen(cipher_data);
    if (cipher_len > ID2_ID_LEN * 2) {
        ret = id2_client_decrypt_data(cipher_data, cipher_len);
        if (ret < 0) {
            ID2_DBG_LOG("id2 client decrypt data fail!\n");
            return -1;
        }
    }

    return 0;
}
```

3. 在设备端进行解密验证。
   i. 重新编译生成固件（id2_app）。
   ii. 烧录固件到设备。
   iii. 运行固件，进行解密验证。

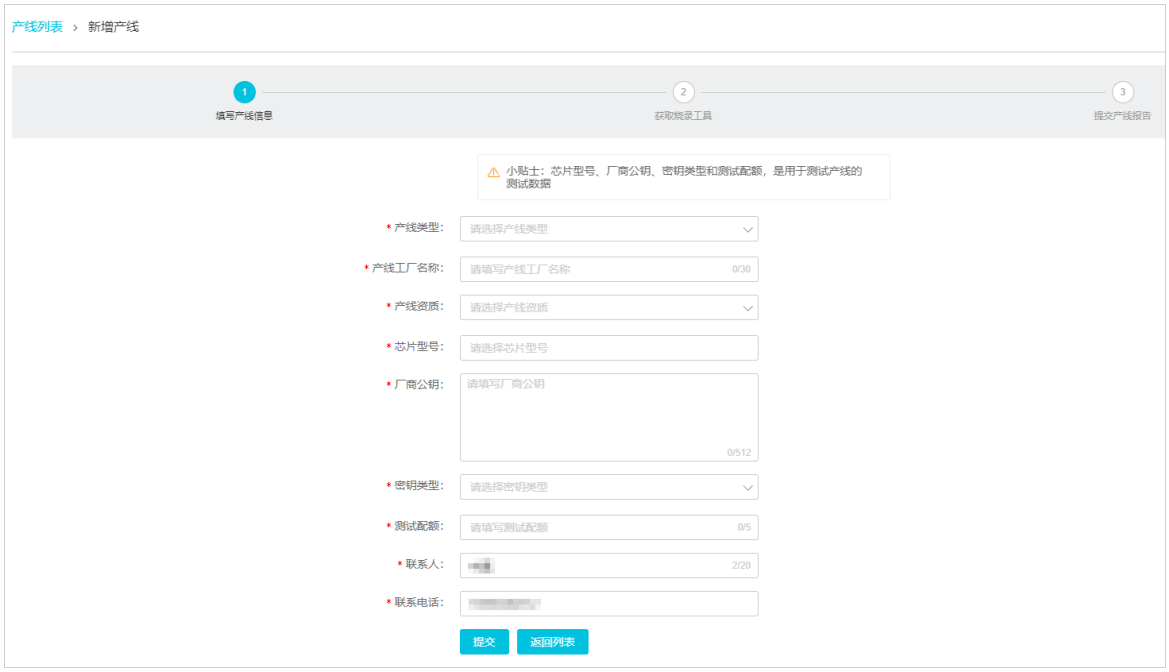
```
                        "args": {
                        },
                        "result":       "000F[illegible]9100"
                }, {
                        "api":  "id2_client_get_challenge_auth_code",
                        "args": {
                                "challenge":    "55B8[illegible]33DC",
                                "extra":        "abcd1234"
                        },
                        "result":       "2~2~C4B[illegible]Ec02//C4i/ciSU6rEupdmiisx"
                }, {
                        "api":  "id2_client_get_timestamp_auth_code",
                        "args": {
                                "timestamp":    "1512022279204",
                                "extra":        "abcd1234"
                        },
                        "result":       "3~2~C4B[illegible]rpS+a//C4i/ciSU6rEupdmiisx"
                }]
}

<LS_LOG> id2_client_cleanup 579: [id2_client_cleanup enter.]
<LS_LOG> id2_client_generate_authcode 186: =====>ID2 Client Generate AuthCode End.

<LS_LOG> id2_client_decrypt_data 202: ====> ID2 Client Test Decrypt Start.
<LS_LOG> id2_client_init 556: [id2_client_init enter.]
<LS_LOG> ID2 Client Version: 0x00020000
<LS_LOG> ID2 Client Build Time: Aug 27 2019 10:18:30
<LS_LOG> ----------------------------------------
<LS_LOG> CONFIG_ID2_DEBUG is defined!
<LS_LOG> CONFIG_ID2_OTP is defined!
<LS_LOG> CONFIG_ID2_KEY_TYPE: ID2_KEY_TYPE_AES
<LS_LOG> ----------------------------------------
<LS_LOG> id2_client_get_id 606: [id2_client_get_id enter.]
<LS_LOG> id2_client_get_id 649: ID2: 000FFFFFEAEDFDF4137D9100
<LS_LOG> id2_client_decrypt 700: [id2_client_decrypt enter.]
<LS_LOG> id2_log_hex_dump 99: id2 cipher input: [length = 0x0020]
<LS_LOG> id2_log_hex_dump 109: 85 5C 18 1E C6 21 94 E5   44 1F C0 09 EC AA 16 BE
<LS_LOG> id2_log_hex_dump 109: BC AE 76 E0 36 86 AA EA   B7 67 37 FD 95 D2 4E 05
<LS_LOG> id2_log_hex_dump 99: id2 cipher output: [length = 0x0020]
<LS_LOG> id2_log_hex_dump 109: 30 30 30 46 46 46 46 46   45 41 45 44 46 44 46 34
<LS_LOG> id2_log_hex_dump 109: 31 33 37 44 39 31 30 30   08 08 08 08 08 08 08 08
<LS_LOG> id2_sym_cipher_154: id2_cipher_pkcs5_unpadding.
<LS_LOG> id2_client_decrypt_data 262: ================>ID2 Client Test Decrypt Pass.
```

3.产线审核

本文介绍产线审核流程。

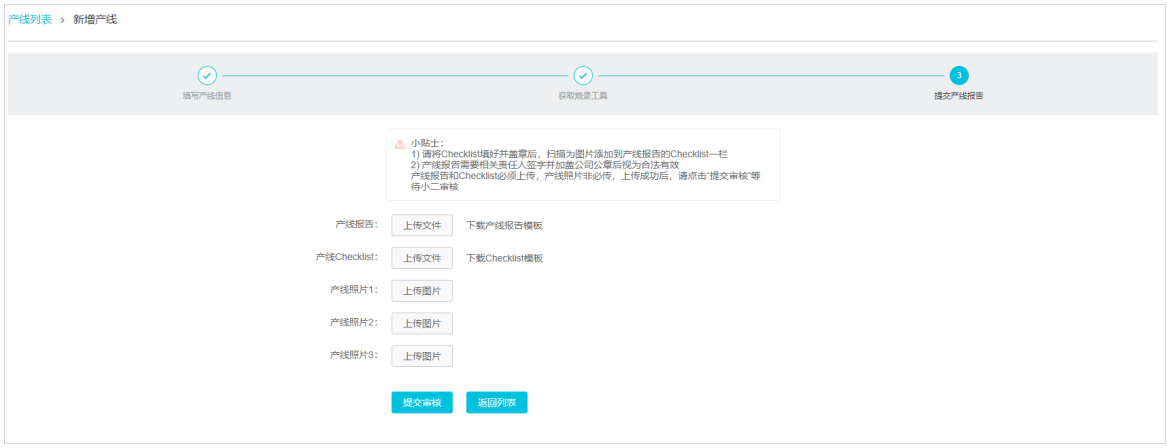
1. 登录安全芯片接入平台，在左侧导航栏选择**芯片接入管理 > 产线列表**，然后单击**新增产线**。

2. 在**填写产线信息**页面中填写产线的实际信息，并选择一款芯片，填写芯片的相关测试数据，单击**提交**。



3. 在**获取烧录工具**页面，下载产线SDK，进行产线烧录测试。测试完成后，单击**下一步**。如果您已经完成了烧录的测试，请跳过此步骤。

4. 在**提交产线报告**页面，上传产线报告、产线Checklist和产线照片（产线照片非必传）。上传之后，请通知您的阿里云IoT接口人，并等待审核。



> **说明**
>
> - 请将Checklist填好并盖章后，扫描为图片添加到产线报告的Checklist一栏。
> - 产线报告需要相关责任人签字并加盖公司公章后视为合法有效。
> - 若您已上传产线审核相关信息但未提交审核，可稍后在**产线列表**页面，找到待审核的产线，单击其操作列的**提交审核**发起产线信息审核。

5. 审核通过后，**产线列表**页面中，该产线对应的**审核状态**列显示**审核通过**。产线即开启了ID²安全芯片的烧录资质。