陪护床 NB 蓝牙锁通信协议

修订

修订版本	作者	修订内容	修改时间
V1.0	吴志益	整理协议	2017.3.10
V1.1	吴志益	添加查找设备指令	2017.8.19
V1.2	吴志益	添加查询锁状态	2018.3.20
V1.3	吴志益	添加关锁返回说明	2018.5.29
V1.4	陈诚	添加 NB 测试指令	2018.11.16
V1.5	陈诚	添加设置 http 域名	2018.11.16
V1.6	陈诚	添加查询域名，查询 IMEI，查询 ICCID 指令，NB 模块正常上报	2019.05.09

审核

审核部门	签字	日期
		年 月 日
		年 月 日
		年 月 日

批准

批准人	签字	日期
		年 月 日

1.广播数据

广播数据内容格式为：

PRO[2]	MAC[6]	NC[1]	POWER	LOCK_STATE

字段	说明
PRO[2]	用来表示遵循壹家软件蓝牙锁协议的设备，以过滤其他设备,如 0x0102,0x1111
MAC[6]	锁的物理地址
NC[1]	预留一位
POWER	电量百分比，0-100
LOCK_STATE	锁的开关状态，0：开，1：关

注意：广播信息至少会包含 PRO[2]MAC[6],POWER 和 LOCK_STATE 不同的设备可能不会包含。

2.交互命令总体描述

命令格式：

CMD1	CMD2	LEN	DATA1	…	DATAN	**TOKEN[4]**

命令总长固定为 16 字节，不足 16 字节的用随机数补足。

字段	说明
CMD1	命令字节 1
CMD2	命令字节 2
LEN	命令数据长度，即 DATA1 到 DATAN 的总字节数
DATA1 …DATAN	命令数据
TOKEN[4]	四个字节的命令令牌

命令响应格式：

CMD1	CMD2	LEN	RESULT

字段	说明
CMD1	命令字节 1
CMD2	命令字节 2

LEN	命令数据长度
RESULT	命令操作结果：0x00 命令正确，0x01 命令错误

注意：对每条命令，接收方都要返回响应，以确保命令正确送达

3.各交互命令详解

3.1 检测电量

检测电量

0x02	0x01	0x01	0x01

检测电量响应

响应内容	说明
0x02 0x02 0x01 POWER(0~100)	检测电量成功
0x02 0x02 0x01 0xff	检测电量失败

3.2 开锁

开锁命令

0x05	0x01	0x06	PSW[6]

PSW[6]开锁密码，初始密码是'0''0''0''0''0''0'，即 0x30 0x30 0x30 0x30 0x30 0x30

开锁响应

响应内容	说明
0x05 0x02 0x01 0x00	开锁成功
0x05 0x02 0x01 0x01	开锁失败

3.3 改密码

分两个包发送，第一次发送旧密码的数据包，第二次发送新密码的数据包

改密码第一条命令

0x05	0x03	0x06	OLDPSW[6]

字段	说明
OLDPSW[6]	旧密码

改密码第一条命令响应

响应内容	说明
0xCB 0x050x03 0x010x00	旧密码成功
0xCB 0x050x03 0x01 0x01	旧密码失败

改密码第二条命令

0x05	0x04	0x06	NEWPSW [6]

字段	说明
NEWPSW [6]	新密码

改密码第二条响应

响应内容	说明
0x050x05 0x010x00	改密码成功
0x050x05 0x01 0x01	改密码失败

3.4 关锁

关锁命令

0x05	0x0C	0x01	0x01

关锁响应

当设备带有关锁检测的时候，返回如下 0x050301,关锁结果看下条返回的 0x050801

通知内容	说明
0x05 0x03 0x01 0x00	关锁成功
0x05 0x03 0x01 0x01	关锁失败

当设备不支持下发关锁指令，但带有关锁检测的会主动上报 0x0508 的内容

通知内容	说明
0x05 0x08 0x01 0x00	关锁成功
0x05 0x08 0x01 0x01	关锁失败

当设备不带有关锁检测的时候，返回如下 0x050d 的内容

通知内容	说明
0x05 0x0d 0x01 0x00	关锁成功
0x05 0x0d 0x01 0x01	关锁失败

3.5 查询锁开关状态

查询锁开关状态命令

0x05	0x0E	0x01	0x01

查询锁开关状态响应

响应内容	说明
0x05 0x0F 0x02 STA LATC	STA:感应磁体开关，01表示吸住磁铁，00表示拿开磁铁
	LATC：锁状态，01表示锁舌伸出，00表示锁舌压入

3.6 获取令牌

获取令牌命令

0x06	0x01	0x01	0x01

获取令牌响应

0x06	0x02	0x07	TOKEN[4]	VER[2]	DEVTYPE

字段		说明		
TOKEN		4个字节的令牌		
VER	VER[0]	固件版本号	主版本号	
	VER[1]		次版本号	
DEVTYPE		设备类型	01	蓝牙单车版
			02	2G、GPS通信版
			03	蓝牙挂锁
			04	蓝牙柜锁

3.7 修改密钥

分两个包发送，第一次发送密钥的前八位数据包，第二次发送密钥的后八位数据包

修改密钥第一条命令

0x07	0x01	0x08	KEY1[8]

修改密钥第一条响应

响应内容	说明
0xCB 0x07 0x01 0x010x00	成功
0xCB 0x07 0x01 0x01 0x01	失败

修改密钥第二条命令

0x07	0x02	0x08	KEY2[8]

字段	说明
KEY2[8]	密钥的后八位

修改密钥第二条响应

响应内容	说明
0x07 0x03 0x010x00	成功
0x07 0x03 0x01 0x01	失败

3.8 固件升级

固件升级命令

0x03	0x01	0x01	0x01

注：当APP收到响应200ms后，设备会断开连接重启，改变广播名，进入固件升级模式，升级完成后恢复到正常广播状态

固件升级响应

响应内容	说明
0x03 0x02 0x010x00	成功
0x03 0x02 0x01 0x01	失败

3.9 查找设备

查找设备命令

0x03	0x03	0x01	0x01

查找设备响应

响应内容	说明
0x03 0x04 0x010x00	成功
0x03 0x04 0x01 0x01	失败

响应内容(中间添加操作返回)	说明

0xE4 0x08 0x01 0x00	成功
0xE4 0x08 0x01 0x01	失败

3.10 NB 测试

开启 NB 测试命令

0x0E	0x01	0x01	0x01

开启 NB 测试响应

响应内容	说明
0x0E 0x02 0x01 0x00	成功
0x0E 0x02 0x01 0x01	失败

3.11 NB 上报 sim 卡检测

NB 上报 RSSI 信号值

上报内容	说明
0x0E 0x03 0x01 0x00	NB未检测到sim卡插入
0x0E 0x03 0x01 0x01	NB检测到sim卡

3.12 NB 上报 RSSI 信号值

上报内容	说明
0x0E 0x04 0x01 0xXX	0xXX表示rssi信号质量，范围0 -- 99,正常在15以上才能联网，99为unknown

3.13 NB 上报是否注册(预留)

上报内容	说明
0x0E 0x05 0x01 0x00	NB未注册
0x0E 0x05 0x01 0x01	NB已注册

3.14 NB 上报模组是否正常

上报内容	说明
0x0E 0x06 0x01 0x00	NB模组异常
0x0E 0x06 0x01 0x01	NB模组正常

3.15 查询 http 域名

查询 http 域名

0x0E	0x07	0x01	Flag

Flag从1开始累加，当flag=1时，表示开始查询域名，等到设备响应，响应的数据包中包括总分片数量，解析出总分片数之后加1查询后面的分片。

查询 http 域名响应

0x0E	0x08	Data_len	Flag	Data

Data_len：占1字节，表示flag + data的长度
Flag：占1字节，高4位表示分片的总数，低4位表示分片的序号。同一查询http域名指令中高4位相同（即分片总数相同），低4位递增加1，从1开始。最后一片数据高4位和低4位相同。
Data：占Data_len - 1字节。

3.16 查询 IMEI

查询 IMEI 命令

0x0E	0x09	0x01	Flag

Flag从1开始累加，当flag=1时，表示开始查询域名，等到设备响应，响应的数据包中包括总分片数量，解析出总分片数之后加1查询后面的分片。

查询 IMEI 响应

0x0E	0x0A	Data_len	Flag	Data

Data_len：占1字节，表示flag + data的长度
Flag：占1字节，高4位表示分片的总数，低4位表示分片的序号。同一查询http域名指令中高4位相同（即分片总数相同），低4位递增加1，从1开始。最后一片数据高4位和低4位相同。
Data：占Data_len - 1字节。

3.17 查询 ICCID

查询 ICCID 命令

0x0E	0x0B	0x01	Flag

Flag从1开始累加，当flag=1时，表示开始查询域名，等到设备响应，响应的数据包中包括总分片数量，解析出总分片数之后加1查询后面的分片。

查询 ICCID 响应

0x0E	0x0C	Data_len	Flag	Data

Data_len：占1字节，表示flag + data的长度
Flag：占1字节，高4位表示分片的总数，低4位表示分片的序号。同一查询http域名指令中高4位相同（即分片总数相同），低4位递增加1，从1开始。最后一片数据高4位和低4位相同。
Data：占Data_len - 1字节。

3.18 查询 IMSI

查询 IMSI 命令

0x0E	0x0D	0x01	Flag

Flag从1开始累加，当flag=1时，表示开始查询域名，等到设备响应，响应的数据包中包括总分片数量，解析出总分片数之后加1查询后面的分片。

查询 IMSI 响应

0x0E	0x0E	Data_len	Flag	Data

Data_len：占1字节，表示flag + data的长度

Flag：占1字节，高4位表示分片的总数，低4位表示分片的序号。同一查询http域名指令中高4位相同（即分片总数相同），低4位递增加1，从1开始。最后一片数据高4位和低4位相同。
Data：占Data_len - 1字节。

3.19 设置 http 域名

设置 http 域名

0x0F	0x01	Data_len	Flag	Data

Data_len：占1字节，表示flag + data的长度
Flag：占1字节，高4位表示分片的总数，低4位表示分片的序号。同一设置http域名指令中高4位相同（即分片总数相同），低4位递增加1，从1开始。最后一片数据高4位和低4位相同。
Data：占Data_len - 1字节。

设置 http 域名响应

每收到一片数据设备都将响应

响应内容	说明
0x0F 0x02 0x02 0xXX 0x00	设备收到第0xXX分片处理成功
0x0F 0x02 0x02 0xXX 0x01	设备收到第0xXX分片处理失败

4.UUID 定义

Server UUID:

"0000fee7-0000-1000-8000-00805f9b34fb"

ReadData UUID:

"000036f6-0000-1000-8000-00805f9b34fb"

WriteDataUUID:

"000036f5-0000-1000-8000-00805f9b34fb"

CLIENT_CHARACTERISTIC_CONFIG UUID:

"00002902-0000-1000-8000-00805f9b34fb"

OAD_SERVICE_UUID:

"f000ffc0-0451-4000-b000-000000000000"

CC_SERVICE_UUID:

"f000ccc0-0451-4000-b000-000000000000"

5.通信协议

为了方便叙述，iOS 和Android 应用程序统称为主机，蓝牙锁简称为锁。主机和锁之间通过基本的通信帧进行通信，通信帧固定为16 个字节，除有效的指令和数据之外，剩余部分可以填充任意数据。发送方需要先把通信帧加密后再发送，接收方收到数据需要解密还原通信帧。加密算法约定为AES-128，它是蓝牙BLE 通信时最常选用的加密方式。以下为初始的密钥：

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3A	60	43	2A	5C	01	21	1F	29	1E	0F	4E	0C	13	28	25

初始密码为：’0’’0’’0’’0’’0’’0’（十六进制都为：0x30）

参考以下AES-128 数据加密的JAVA 实现：

```
public static byte[] Encrypt(byte[] sSrc, byte[] sKey){
    try{
        SecretKeySpec skeySpec = new SecretKeySpec(sKey, "AES");
        Cipher cipher = Cipher.getInstance("AES/ECB/NoPadding");
        cipher.init(Cipher.ENCRYPT_MODE, skeySpec);
        byte[] encrypted = cipher.doFinal(sSrc);
        return encrypted;
    }catch(Exception ex){
        return null;
    }
}
```

参考以下AES-128 数据解密的JAVA 实现：

```
public static byte[] Decrypt(byte[] sSrc, byte[] sKey){
    try{
        SecretKeySpec skeySpec = new SecretKeySpec(sKey, "AES");
        Cipher cipher = Cipher.getInstance("AES/ECB/NoPadding");
```

```
        cipher.init(Cipher.DECRYPT_MODE, skeySpec);
        byte[] dncrypted = cipher.doFinal(sSrc);
        return dncrypted;
    }catch(Exception ex){
        return null;
    }
}
```

6.交互流程

6.1 获取 Token

(1).获取令牌指令和参数

0x06 0x01 0x01 0x01

(2).随机数补全16字节

0x06 0x01 0x01 0x01 0x2C 0x2C 0x62 0x58 0x26 0x67 0x42 0x66 0x01 0x33 0x31

0x41

(3).AES128加密

(4).0xCA 0xFB 0x4A 0xC9 0x88 0x96 0x6A 0x67 0x63 0x39 0xFE 0x49 0x07 0x05 0xB1 0xF9

(5).发送命令

(6).收到获取令牌响应

0xBA 0x13 0x9E 0xF9 0xC0 0xE4 0x80 0xA5 0xAB 0xD4 0xE2 0xAB 0xD3 0xDB 0x91

0x47

(7).AES128 解密

设备返回 4 字节令牌

0x06 0x02 0x08 0x08 0x66 0x84 0x23 0x02 0x01 0x00 0x01 0x00 0x35 0x27 0x4A

0xDE

注：需要获取令牌才能进行其他操作。

6.2 开锁

(1).开锁指令和参数

0x05 0x01 0x060x30 0x30 0x30 0x30 0x30 0x300x08 0x66 0x84 0x23

(2).随机数补全16字节

0x05 0x01 0x060x30 0x30 0x30 0x30 0x30 0x300x08 0x66 0x84 0x23 0x5E 0x26 0x36

(3).AES128加密

0x0F 0x36 0x17 0x05 0x03 0x17 0x80 0xDD 0x00 0x7A 0x4B 0x42 0x33 0x60 0x17

0x4B

(4).发送命令

(5).设备收到响应

0x1F 0xBE 0x9E 0x1C 0x27 0x45 0x4E 0x9D 0xEB 0x4D 0xA8 0x79 0xF0 0xB3 0xE5

0x09

(6).AES128 解密

0x05 0x02 0x01 0x000x00 0xA0 0xC7 0xCA 0x7F 0x47 0x49 0x7D 0x4B 0x74 0x6B

0x81

7.NB 模块与服务器通讯协议

7.1 格式说明

锁是电池供电，不用的时候将处于休眠模式，此时NB-IoT模块处于PSM休眠状态，接收和发送射频均处于关闭状态，以达到省电的目的。因此，NB-IoT模块只能是主动发送请求给服务器，而服务器不能主动推送消息给NB-IoT模块。

NB-IoT模块与服务器通讯采用http通用协议格式，方便开发和维护。

NB-IoT模块采用http post方式进行请求，请求格式为：http地址为“域名+路径”，http header中Content-Type为空或者"application/x-www-form-urlencoded"，Body内容为“key=value”字符串形式，body中如果存在多个key，使用“&”符号隔开。

注：服务器响应数据不得超过1024个字节。

7.2 通讯命令格式

7.2.1 同步时间及配置参数

锁上电初始化时将会和服务器进行一次时间同步，另外还会定时向服务器进行同步时间，只有同步时间了之后才能扩展一些涉及到时间问题的功能，例如定时向服务器上报锁的相关状态信息。

设备在每天的早上9点和下午3点定时上报状态信息以及向服务器进行时间及配置参数的同步。

- **请求格式**

http 地址	域名+/api/General/timesync
Content-Type	空
Body	空

- **服务器响应**

Content-Length	Body 内容长度
Body	时间戳数值,状态定时上报时间 1(0-23),状态定时上报时间 2(0-23),低电量阈值（1-100）

注：定时时间 1 和定时时间 2 数值不能相同，若无定时时间 1 或者 2，不填值即可，但要使用“,”隔开，例如响应数据可以是“1542007917,,18,20”，“1542007917,6,,10”“1542007917,,,30”，“1542007917,,,”等等。

后面如果要扩展去设置什么参数，都可以将参数以“,”进行隔开区设置，软件上就可以前后兼容。

- **举例说明**

NB-IoT 模块发送请求：

http 地址	http://share.one-more.cn/api/General/timesync
Content-Type	空
Body	空

服务器响应：

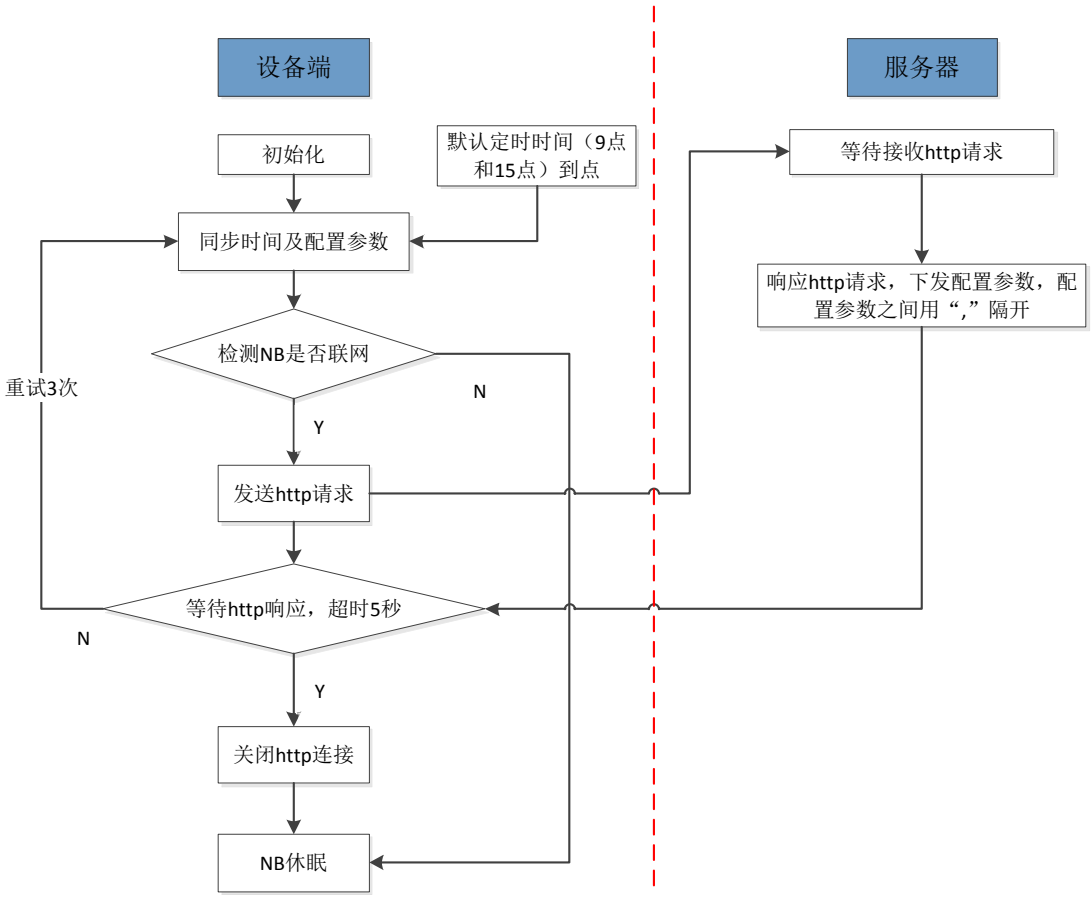
Content-Length	18
Body	1542007917,9,15,20

Content-Length: 10

Connection: close

1542007917,9,15,20

- **流程图**



7.2.2上报相关状态信息

锁在开锁和关锁的时候会立刻上报状态信息，另外在每天的早上 9 点和下午 3 点（可更改）定时上报状态信息。

状态信息包括设备 MAC 地址，NB-IoT 模块中 SIM 卡的 IMEI 号，锁的开关状态（0：锁舌收缩磁铁远离（开锁成功），1：锁舌伸出磁铁靠近（关锁成功），2：锁舌收缩磁铁靠近（开锁中），3：锁舌伸出磁铁远离（异常状态），100：关锁失败）和电量百分比。（可扩展任何状态信息，只需硬件支持即可，例如防撬，经纬度定位信息等等）

- **请求格式**

http 地址	域名+/api/Device/rptstat
Content-Type	application/x-www-form-urlencoded
Body	mac=%s&imei=%s&lockstat=%s&power=%s

- **服务器响应**

Content-Length	Body 内容长度
Body	错误码（0 表示成功，大于 0 表示错误，详见附录错误码）

- **举例说明**

NB-IoT 模块发送请求：

http 地址	http://share.one-more.cn/api/Device/rptstat
Content-Type	application/x-www-form-urlencoded
Body	mac=CC8D3B0541AD&imei=868334031521754&lockstat=0&power=80

服务器响应：

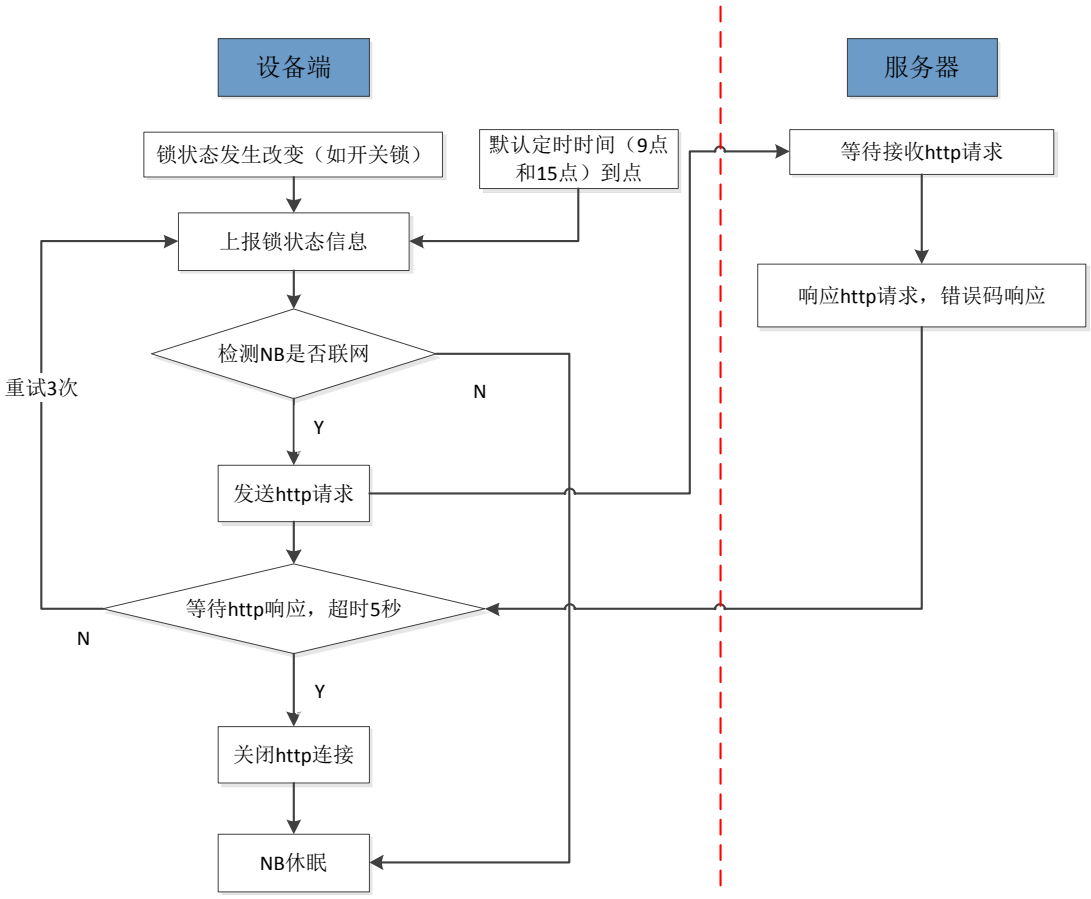
Content-Length	1
Body	0

Content-Length: 1

Connection: close

0

- **流程图**



7.2.3请求是否能够开锁

用户使用手机 APP 登录以后，利用手机扫描锁上的二维码，APP 会上报用户 ID 等以及锁的 MAC 到服务器上，服务器即可将用户 ID 和锁关联起来。服务器返回数据给 APP，此时用户就可以使用蓝牙开锁了。

当用户使用蓝牙无法开锁或者不想使用蓝牙进行开锁的时候，可以使用锁体上的按键进行一键开锁。

服务器根据用户 ID 和时间来判断此用户是否有权限来开锁，设备端将会通过错误码来判断此次是否能够开锁。

- **请求格式**

http 地址	域名+/api/Bedorder/queryauth
Content-Type	application/x-www-form-urlencoded
Body	mac=%s

- **服务器响应**

Content-Length	Body 内容长度
Body	错误码（0 表示成功，大于 0 表示错误，详见附录错误码）

- **举例说明**

NB-IoT 模块发送请求：

http 地址	http://share.one-more.cn/api/Bedorder/queryauth
Content-Type	application/x-www-form-urlencoded
Body	mac=CC8D3B0541AD

服务器响应：

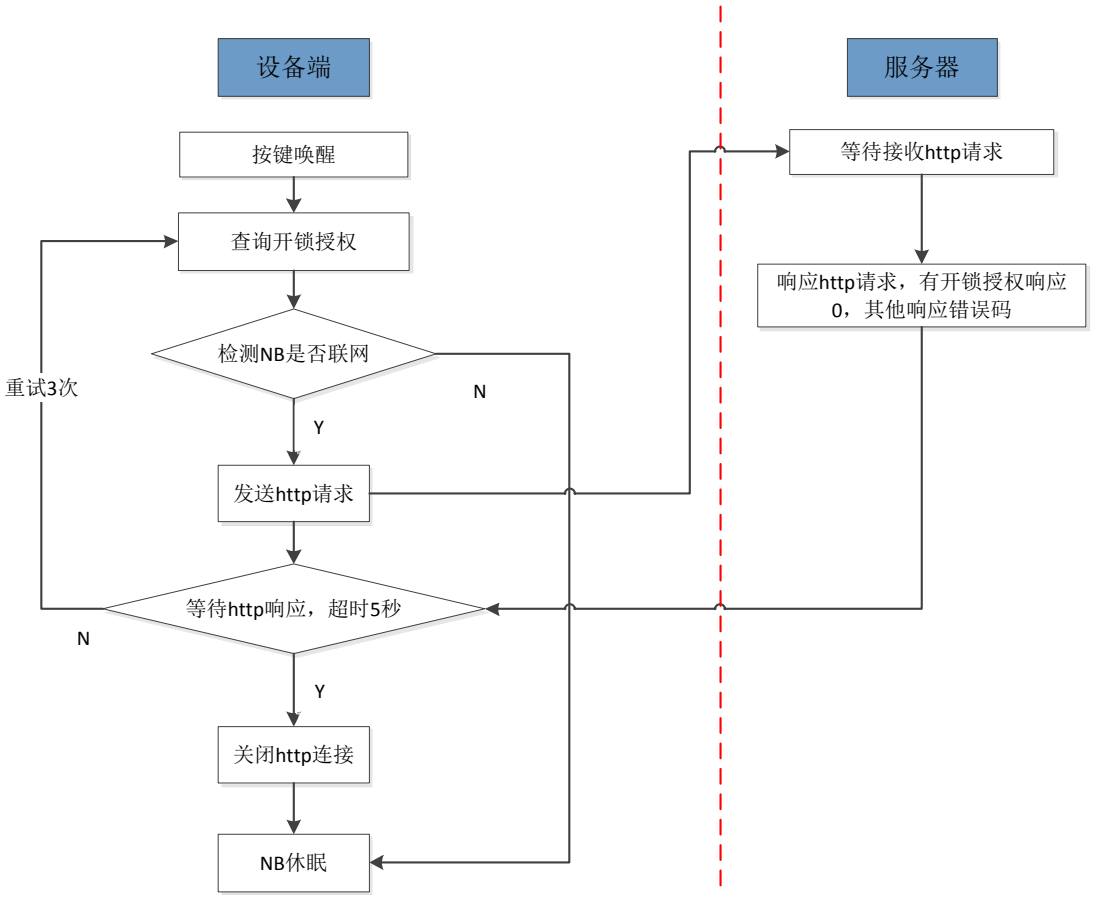
Content-Length	1
Body	0

Content-Length: 1

Connection: close

0

- **流程图**



附录

错误码

错误码数值	表示含义
0	成功
1	请求成功但无授权
97	参数错误
98	服务器异常