到闸业务

注：本协议严格按照互联互通协议文档开发

Data 参数内容：

```
"Data" : {
  " parkId": "",
  " orderNo": "string",
  " plateNo": "string",
  " startTime": "string",
  " endTime": "string"
}
```

字段	类型	含义	字段限制说明
parkId	String	停车场站 id	必填 0-100 字符
orderNo	String	订单号	必填 0-100 字符
plateNo	String	车牌号	必填 0-100 字符
startTime	String	充电开始时间	必填 0-100 字符
endTime	String	充电结束时间	必填 0-100 字符

返回 Data 参数内容

```
Data:{

    "code": "sring" （0:成功、1：失败）
    "message": "sring" （成功或失败原因）
}
```

加密方式

数据传输与安全

本部分确立了电动汽车交换电服务信息交换的数据传输和安全防护的一般原则，包含充换电服务信息交换的数据传输体系、平台认证要求、密钥的管理和使用要求。

数据传输体系

数据传输接口的基本要求

电动汽车充换电服务信息交换应根据国家信息安全等级保护相关要求。

运营商须提供严格的系统安全保密机制，保障信息交换接口安全、稳定、可靠地运行，包括信息的存取控制、应用系统操作的安全等。基本要求：

1) 采用身份认证、访问控制、数据加密、数字签名等安全措施；

2) 采用安全可靠并且普遍使用的加密算法；

3) 密钥的存贮和交易信息的加密 / 解密需要在安全的环境中；

4) 遵循数据安全保密的国家和行业标准；

5) 定期更换密钥；

6) 具备对报文做来源正确性鉴别的机制（HMAC）。

密钥体系

每个运营商与省平台交互前需要分配平台标识（OperatorID）、平台密钥（OperatorSecret）、消息密钥（DataSecret）、消息密钥初始化向量（DataSecretIV）和签名密钥（SigSecret）。

1)平台标识（OperatorID）: 固定9位，运营商的组织机构代码，作为运营商的唯一标示。

2) 平台密钥（OperatorSecret）:可采用32H、48H 和64H，由 0-F 字符组成，为申请认证使用。

3)消息密钥（DataSecret）:用于对所有接口中 Data 信息进行加密。

4)消息密钥初始化向量（DataSecretIV）：固定16位，用户 AES 加密过程的混合加密。

5)签名密钥（SigSecret）： 可采用32H、48H 和64H，由 0-F 字符组成，为签名的加密密钥。

平台认证方式及规则

概述

电动汽车充换电服务信息交换应具备平台认证服务提供平台之间的鉴权认证功能。平台之间在信息交换前，需完成平台认证，获得平台交换能力。

平台认证模式

平台认证采用中心交换认证模式，中心交换认证模式由市级平台提供鉴权认证服务，运营商与中心认证服务方确定运营商标识（OperatorID）、运营商密钥（OperatorSecret）、消息密钥（DataSecret）、消息密钥初始化向量（DataSecretIV）和签名密钥（SigSecret），具体认证方式由各运营商和认证服务方共同确定。

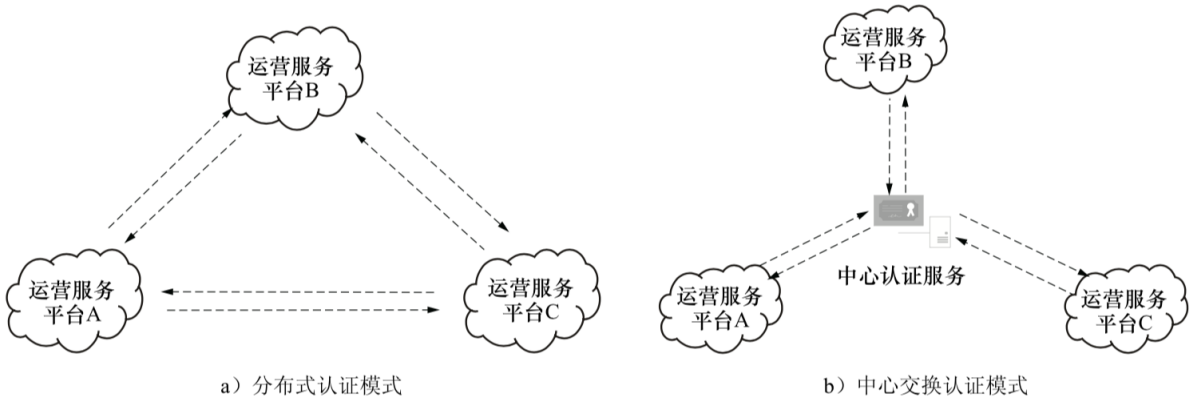

图 1　认证模式

平台认证方法

平台认证宜采取身份认证和访问控制相结合的方式进行。

身份认证可采取用户名/口令认证、密钥认证或数字证书认证等方式进行；访问控制可

采取 IP 访问控制、时间访问控制等多种手段结合。

用户身份认证成功后授予 Token，每次向服务端请求资源的时候需要带着服务端签发的 Token，服务端验证 Token 成功后，才返回请求的数据。Token 的有效期由服务方确定，最长不应超过7天，Token 丢失或失效后需要再次发起认证服务。

图 2　平台认证方式

数据传输方式及规则

数据传输接口规则

所有数据传输接口均采用 HTTP(S)接口，每个接口的 URL 均采用如下格式定义：

http(s)://[域名]/hlht/v[版本号]/[接口名称]

1)域名：各接入平台所属域名。

2) 版本号：代表接口版本号，不同的版本地址对应相应版本代码。系统升级期间，新旧版本可同时存在，待所有接入方都切换到新接口，旧接口即可下线。从而达到平滑升级的目的。

3)接口名称：所请求/调用接口的名称，具体接口名称见接口定义。

为保证各接口的功能明确清晰，每个 URL 只允许对应一种功能。其中测试例分类:

接口调用方式

所有接口均使用 HTTP(S)/POST 方式传输参数,传输过程中应包含消息头和消息主体两部分。

消息头规范

消息头一般需包含内容类型和授权信息（Authorization）。

内容类型（Content-Type）字段用于标识请求中的消息主体的编码方式，本标准中所规范的信息交换内容均采用 JSON 的方式，参数信息采用 utf-8编码，因此需要配置消息头中的 Content-Type 为 application/json;charset=utf-8。

授权信息（Authorization ）字段用于证明客户端有权查看某个资源，本标准中所规范的授权信息采用凭证（Token）的方式，因此需要在配置消息头中的 Authorization 为 Bearer Tonken。

请求参数规则

一般由运营商标识（OperatorID）、参数内容（Data）JSON 串、时间戳（TimeStamp）、自增序列（Seq）和数字签名（Sig）组成。

表1　消息主体内容表

参数名	说明	举例
OperatorID	运营商标识	

Data	各接口具体参数信息	"Data": {
		"parkId": "",
		"orderNo": "string",
		"plateNo": "string",
		"startTime": "string",
		"endTime": "string"
		}

TimeStamp	时间戳	接口请求时时间戳信息，格式为 yyyyMMddHHmmss
Seq	自增序列	4 位自增序列取自时间戳，同一秒内按序列自增长，新秒重计。如 0001
Sig	参数签名	

返回参数规则

数据传输接口的返回参数一般由返回值（Ret）、返回信息（Msg）、参数内容（Data）和数字签名（Sig）组成。

1)Ret:必填字段，返回编码参考下表。

2)Msg:必填字段，有错误表示具体错误信息，无错误返回成功信息。

3)Data:参数内容,采用 utf-8编码，JSON 格式。

表2　返回参数编码表

Ret 值	说明
-1	系统繁忙，此时请求方稍后重试
0	请求成功
4001	签名错误
4002	Token 错误
4003	POST 参数不合法,缺少必须的示例： OperatorID,sig,TimeStamp,Data，Seq 五个参数
4004	请求的业务参数不合法，各接口定义自己的必须参数
500	系统错误

批量数据传输

数据传输接口中的 Data 字段可为数组型的 JSON 格式，数据发送方可通过该字段实现批量数据的传输。

密钥的使用及管理

各运营商系统间在消息传递时，需要保障传输和接收数据的安全和完整。

基本安全要求

运营商必须满足数据安全传输控制方面的要求。

运营商必须提供严格的系统安全保密机制，保障信息交换接口安全、稳定、可靠地运行，包括信息的存取控制、应用系统操作的安全等。

密钥的安全要求

密码算法用于密钥的产生、分发、HMAC 以及加密等安全功能，相关的算法模块在其生命周期内不能被修改、导出至安全环境外部。

指定功能的密钥仅能做指定功能使用，不能被其他任何功能使用。

密钥的产生

数据密钥应具备随机产生特性，密钥产生后要检查密钥的有效性，弱密钥和半弱密钥需被剔除。

运营商加入信息交换时，必须申请独立的密钥文件，密钥可由运营商协商产生。

密钥的分发

密钥的分发应该由安全方式进行，可通过线下分发、联机报文或数字信封的方式加密传输。

密钥的存储

密钥保存在数据库中以密文方式存储。

密钥注入、密钥管理和密钥档案的保管应由专人负责。使用密钥和销毁密钥要在监督下进行并应有使用、销毁记录。

密钥的销毁

当新密钥产生后，生命期结束的旧密钥必须从数据库和内存中清除，防止被替换使用；同时所有可能重新构造此密钥的信息也必须清除。新密钥成功启用和旧密钥自动销毁的记录将被更新。

数据的加密处理

数据加密规则

消息发送方需要对 Data 字段中涉及交易及隐私等数据利用消息密钥（DataSecret）进行加密，加密算法宜使用 AES 128位加密，加密模式采用 CBC，填充模式采用 PKCS5Padding 方式。

消息接收方收到消息之后，根据消息密钥（DataSecret）对消息体中的 Data 数据进行

解密，校验参数合法性等后续业务处理。

数据加/解密方法

数据传输的加密使用对称加密算法 AES 加密，AES 算法的密钥长度、分组长度和轮数的关系如表3所示。

表3　Key-Block-Round 关系

密钥长度 (Nk words)	分组长度 (Nb words)	轮数 (Nr)
4	4	10
6	4	12
8	4	14

对于 AES 加密和解密变换，AES 算法使用的轮函数由4个不同的以字节为基本单位的变换复合而成，该过程由四个不同的阶段组成：

1)S 盒变换，用一个 S 盒完成分组中的按字节代替；

2)行移位变换，一个简单的置换；

3)列混淆变换，一个利用在域 GF(28)上的算术性的代替；

4)轮密钥加变换，一个利用当前分组和扩展密钥的一个部分进行按位异或。

AES 对数据的加密过程是通过把输入的明文和密钥由轮函数经 Nr 轮迭代来实现的，结尾轮与前 Nr-1轮不同。前 Nr-1轮依次进行 S 盒变换、行移位变换、列混淆变换和轮密钥加变换；结尾轮与前 Nr-1轮相比去掉了列混淆变换。

而解密过程与加密过程相反，通过把输入的密文和密钥由轮函数经 Nr 轮迭代来实现的，结尾轮与前 Nr-1轮不同。前 Nr-1轮依次进行逆行移位变换、逆 S 盒变换、轮密钥加变换和逆列混淆变换；结尾轮与前 Nr-1轮相比去掉了逆列混淆变换。

AES 算法的加密解密过程如图5所示。

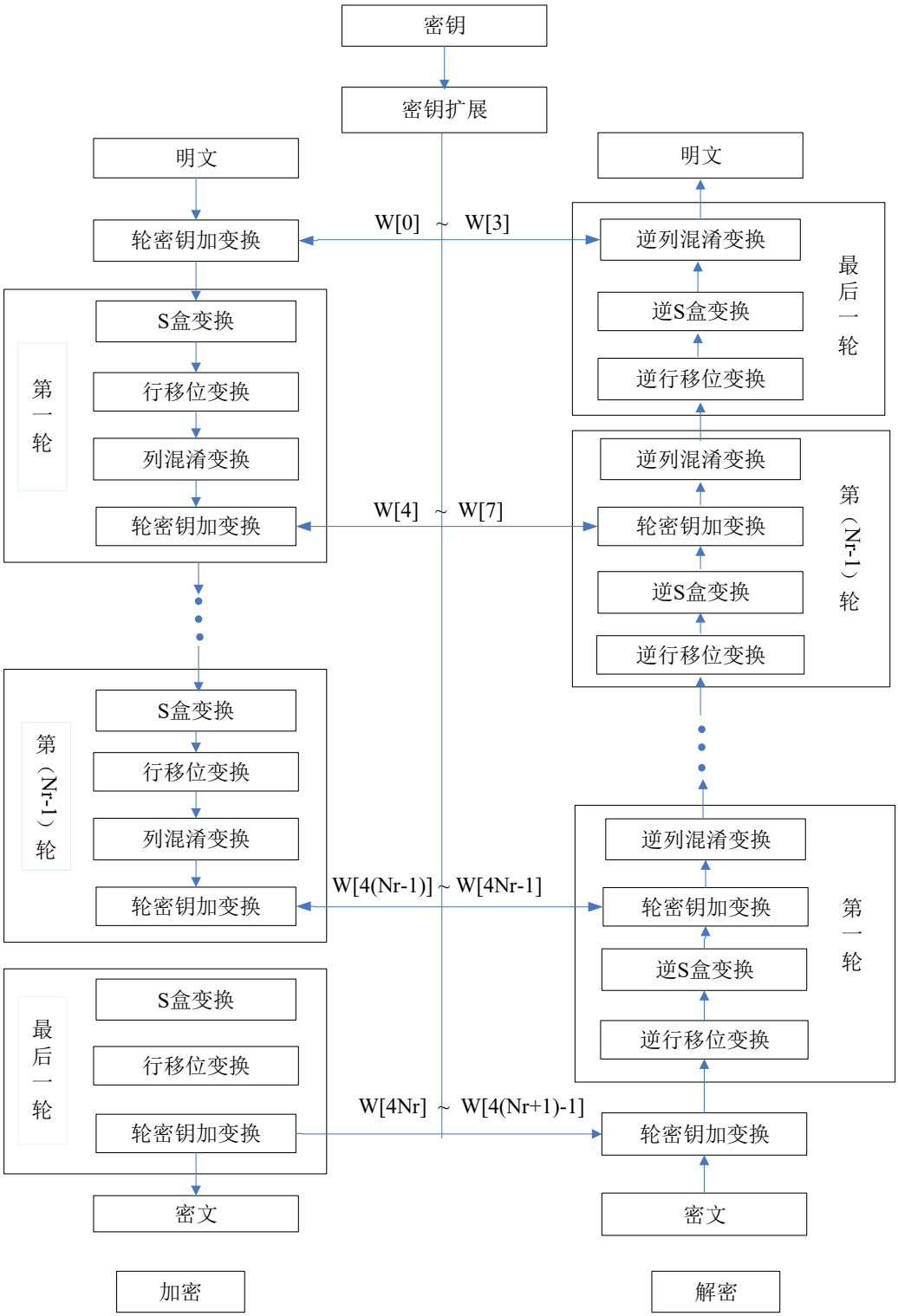

图1　AES 加/解密过程图

数据加/解密示例（密钥、向量不作为固定值）

密钥：1234567890abcdef

初始向量：1234567890abcdef

明文信息：

示例：{"total":1,"stationStatusInfo":{"operationID":"123456789","stationID":"111111111111111","connectorStatusInfos":{"connectorID":1,"equipmentID":"10000000000000000000001","status":4,"currentA":0,"currentB":0,"currentC":0,"voltageA":0,"voltageB":0,"voltageC":0,"soc":10,}}}

秘文：

示例：il7B0BSEjFdzpyKzfOFpvg/Se1CP802RItKYFPfSLRxJ3jf0bVl9hvYOEktPAYW2nd7S8MBcyHYyacHKbISq5iTmDzG+ivnR+SZJv3USNTYVMz9rCQVSxd0cLlqsJauko79NnwQJbzDTyLooYoIwz75qBOH2/xOMirpeEqRJrF/EQjWekJmGk9RtboXePu2rka+Xm51syBPhiXJAq0GfbfaFu9tNqs/e2Vjja/ltE1M0lqvxfXQ6da6HrThsm5id4ClZFIi0acRfrsPLRixS/IQYtksxghvJwbqOsbIsITail9Ayy4tKcogeEZiOO+4Ed264NSKmk7l3wKwJLAFjCFogBx8GE3OBz4pqcAn/ydA=

参数签名规范

参数签名要求

参数签名采用 HMAC-MD5算法，采用 MD5 作为散列函数，通过签名密钥（SigSecret）对整个消息主体进行加密，然后采用 Md5信息摘要的方式形成新的密文，参数签名要求大写。

请求参数的签名顺序按照消息体顺序拼接后执行，拼接顺序为运营商标识（OperatorID）、参数内容（Data）、时间戳（TimeStamp）、自增序列（Seq）。

返回参数的签名顺序按照消息体顺序拼接后执行，拼接顺序为返回值（Ret）、返回信息（Msg）、参数内容（Data）。

参数签名方法

（1）HMAC-MD5算法

HMAC（K，M）=H（K⊕opad｜H（K⊕ipad｜M））

其中:K 是密钥（OperatorSecret），长度可为64字节，若小于该长度，在密钥后面用“0”补齐。

M 是消息内容;

H 是散列函数;

opad 和 Ipad 分别是由若干个0x5c 和0x36组成的字符串;

⊕表示异或运算;

｜表示连接操作。

（2）HMAC-MD5流程

1) 在签名密钥（SigSecret）后面添加0来创建一个长为64字节的字符串(str);

2) 将上一步生成的字符串(str)与 ipad(0x36)做异或运算，形成结果字符串(istr);

3)将消息内容 data 附加到第二步的结果字符串(istr)的末尾;

4)做 md5运算于第三步生成的数据流(istr);

5) 将第一步生成的字符串(str)与 opad(0x5c)做异或运算，形成结果字符串(ostr);

6)再将第四步的结果(istr)附加到第五步的结果字符串(ostr)的末尾;

7)做 md5运算于第六步生成的数据流(ostr)，输出最终结果(out)。

参数签名示例

签名密钥：1234567890abcdef

运营商标识（OperatorID）：123456789

参数信息（Data）：

il7BOBSEjFdzpyKzfOFpvg/Se1CP8O2RItKYFPfSLRxJ3jfObVl9hvYOEktPAYW2nd7S8MBcyHYyacHKb
ISq5iTmDzG+ivnR+SZJv3USNTYVMz9rCQVSxdOcLlqsJauko79NnwQJbzDTyLooYoIwz75qBOH2/xOMir
peEqRJrF/EQjWekJmGk9RtboXePu2rka+Xm51syBPhiXJAqOGfbfaFu9tNqs/e2Vjja/ltE1MOlqvxfXQ
6da6HrThsm5id4ClZFIiOacRfrsPLRixS/IQYtksxghvJwbqOsbIsITail9Ayy4tKcogeEZiOO+4Ed264
NSKmk7l3wKwJLAFjCFogBx8GE3OBz4pqcAn/ydA=

时间戳（TimeStamp）：20160729142400

自增序列（Seq）：0001

签名（Sig）：745166E8C43C84D37FFEC0F529C4136F

分布式认证的认证接口规范

此接口用于平台之间认证 Token 的申请，Token 作为全局唯一凭证，调用各接口时均需要使用。此接口也应实现加解密规范要求和验签要求。

接口定义

接口名称：query_token
接口使用方法：由服务端实现此接口，需求端调用。

输入参数

参数名称	定义	参数类型	描述
运营商标识	OperatorID	字符串	运营商组织机构代码
运营商密钥	OperatorSecret	字符串	运营商分配的唯一识别密钥

返回值

参数名称	定义	参数类型	描述
运营商标识	OperatorID	字符串	运营商组织机构代码
成功状态	SuccStat	整型	0:成功; 1:失败
获取的凭证	AccessToken	字符串	全局唯一凭证
凭证有效期	TokenAvailableTime	整型	凭证有效期，单位秒
失败原因	FailReason	整型	0:无; 1:无此运营商; 2:密钥错误; 3～99:自定义