|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 删除人员信息 | REQ\_DELETE\_PERSON\_INFO | 0x0203 | 删除人员信息 | 双向 |
| 响应删除人员信息 | ACK\_DELETE\_PERSON\_INFO | 0x4203 | 回应删除人员信息请求 | 双向 |

### UUID结构UuidT

struct UuidT

{

UUID uuid; // uuid，例如可能为人员id或特征id等

int token; // 操作对应的token

};

此结构为删除人员、特征或照片时，传送的数据结构。

数据定长20Bytes

|  |  |  |
| --- | --- | --- |
| uuid | token |  |
| 16Bytes(UUID) | 4Bytes |  |

注：之前版本的说明有误，认为删除人员、特征或照片时只发送UUID数组。实际是发送UuidT结构组成的数组。

相关请求：[REQ\_DELETE\_PERSON\_INFO](#_删除人员信息请求)，[REQ\_DELETE\_PERSON\_IRIS](#_删除虹膜特征请求)，[REQ\_DELETE\_PERSON\_PHOTO](#_删除人员照片请求)

#### 删除人员信息请求

请求方向：服务器端↔客户端，任意一方均可发起此请求。

此请求用于向接收方删除已有的人员信息。

请求名称：REQ\_DELETE\_PERSON\_INFO

请求帧：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 同步头 | 请求码 | 请求子码 | 实际数据长度 | 传送的数据 | 校验 |
| 0x5353  (2 Bytes) | (2 Bytes) | 0xaaaa  (2 Bytes) | 传送的数据长度  (4 Bytes) | 人员UuidT数组  (4Bytes+20Bytes\*数组长度) | (2 Bytes) |

请求帧字节长度不是固定值，以传送的实际数据长度为准。

**实际传送的数据结构**为：

typedef list<UuidT> lstPersonId;

|  |  |  |  |
| --- | --- | --- | --- |
| len | UuidT | UuidT | …… |
| 4Bytes | 20Bytes | 20Bytes | …… |

其中，len表示后面的人员UuidT数组长度，即有多少个人员的UuidT。

其中人员UuidT的数据结构参见[UuidT](#_UUID结构UuidT)。

#### 响应删除人员信息

请求方向：服务器端↔客户端，任意一方均可响应。

此请求用于回应删除人员信息的请求，可能成功或失败。

请求名称：ACK\_DELETE\_PERSON\_INFO

请求帧格式参见“一般回应格式”。

### 一般回应格式

大部分请求的回应帧格式如没有特殊说明，则均按照如下形式。

请求帧：

如果错误码=0，表示对应的请求成功，格式为

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 同步头 | 请求码 | 请求子码 | 实际数据长度 | 错误码 | 传送的数据 | 校验 |
| 0x5353  (2 Bytes) | (2 Bytes) | 0xaaaa  (2 Bytes) | 4  (4 Bytes) | 0  (4Bytes) | （无数据） | (2 Bytes) |

请求帧字节长度固定为：16Bytes。

如果错误码<0，表示对应的请求失败，并给出错误描述，格式为

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 同步头 | 请求码 | 请求子码 | 实际数据长度 | 错误码 | 传送的数据 | 校验 |
| 0x5353  (2 Bytes) | (2 Bytes) | 0xaaaa  (2 Bytes) | 264  (4 Bytes) | (4Bytes) | ErrorString (260Bytes) | (2 Bytes) |

请求帧字节长度固定为：276Bytes。