

LocalSense 客户端通信协议

v0.9

版本说明

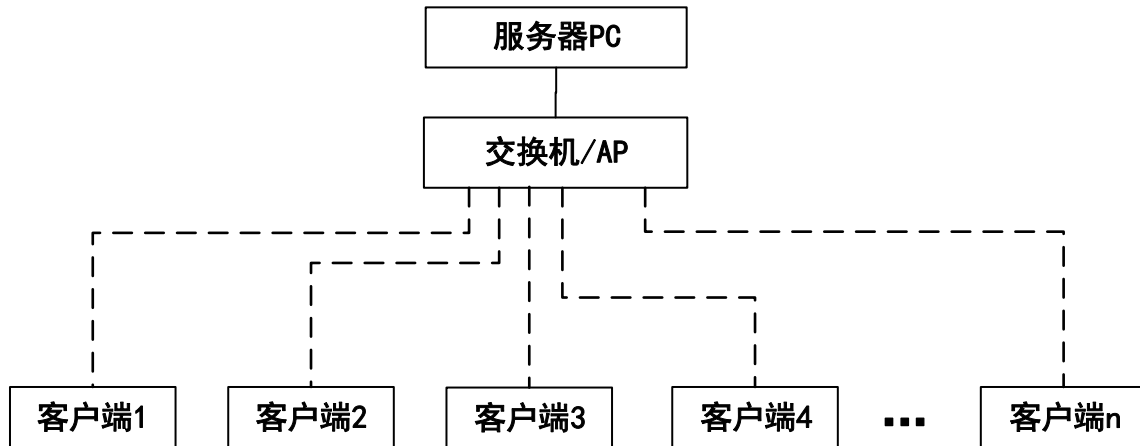
[illegible]

目录

- 1. 通信流程..... 4
 - 1.1 系统框图..... 4
 - 1.2 服务器与客户端之间的通信..... 4
- 2. 实时数据..... 5
 - 2.1 标签实时信息..... 5
 - 2.2 报警信息..... 6
 - 2.3 电量信息..... 7
 - 2.4 扩展数据信息（客户定制）..... 7
 - 2.5 扩展报警信息..... 8

1. 通信流程

1.1 系统框图



采用 C/S 结构，一个服务器同时和多个客户端交互。客户端的形态可以是 PC 软件，移动端软件。

1.2 服务器与客户端之间的通信

- 1) 实时数据：位置信息、状态变化等数据，采用 websocket 协议交互：在服务器上启动相关服务，根据 websocket 协议对客户端进行数据推送。

2. 实时数据

websocket服务端的服务名称：localsense_distribute_server，采用非安全模式，端口号：9001。应用数据格式如下。采用网络字节序。

子协议为：**localSensePush-protocol**的实时数据

2.1 标签实时信息

CRC16 校验区					
帧头	帧类型	标签个数(N)	标签信息	CRC16 校验	帧尾
2B	1B	1B	N*21B	2B	2B

标签信息（21B）									
标签 ID	X 坐标	Y 坐标	Z 坐标	定位指示/楼层指示 (已废弃，该位预留)	电量	休眠标志/充电标志	时间戳	地点编号	楼层编号
2B	4B	4B	2B	1B	1B	1B	4B	1B	1B

- 1、帧头 2 字节，固定值：0xCC5F
- 2、帧类型 1 字节，固定值：0x01
- 3、标签个数(N) 1 字节，该帧中标签信息的总数目 N
- 4、标签信息 每个标签 21 字节

1) 标签 ID，2 字节，标签的编号。

2) X 坐标、Y 坐标各 4 字节，Z 坐标 2 字节，标签在地图中的坐标，单位 cm。

3) 定位指示/楼层指示，(已废弃)

定位指示（高 4 位）:标签解算结果，0:存在性；1:1D 定位结果；2:2D 定位结果；3:3D 定位结果，-1：未指示。

楼层指示（低 4 位）:1~15 楼层索引

4) 电量百分比 0： 0%， 1： 20%， 2： 40%， 3： 60%， 4： 80%， 5： 100%

5) 休眠标志/充电标志：

休眠标志(高 4 位)： 0 没休眠 1 休眠

充电标志（低 4 位）： 0 没充电 1 充电

6) 时间戳，从 0 时 0 分 0 秒经历的毫秒数

- 7) 地点编号，1 字节，标签所在的地点编号，例如定位结果在 4 栋，地点编号为 4，最大支持 256 个地点
- 8) 楼层编号，1 字节，标签所在的楼层编号，最大支持 256 层

- 5、CRC16 校验 不包含帧头帧尾
- 6、帧尾 2 字节，固定值：0xAABB

示例：

帧头	类型	一个 标签	标签 ID(17)	X 坐标 (1002cm)	Y 坐标 (5100cm)	Z 坐标 (1500cm)	2D 定 位/3 楼	电量 (20%)	不休 眠、 充电	时间 (1:00)
0xCC5F	0x01	0x01	0x0011	0x000003EA	0x000013EC	0x05DC	0x23	0x01	0x01	0x0036ee80
保留	CRC16 校验	帧尾								
0x0000		0xAABB								

2.2 报警信息

CRC16 校验区							
帧头	帧类型	报警类型	关联 ID	报警时 间	报警 信息	CRC16 校验	帧尾
2B	1B	1B	2B	8B	120B	2B	2B

- 1、帧头 2 字节，固定值：0xCC5F
- 2、帧类型 1 字节，固定值：0x03
- 3、报警类型 1 字节，0x01 电子围栏报警 0x02 表示 SOS 报警 0x03 表示剪断报警 0x04 表示消失报警 0x05 电子围
栏报警消除, 0x06 剪断报警消除 0x07 表示低电量报警
- 4、关联 ID 关联的标签 ID,指示谁触发了报警
- 5、报警时间 自 1970-01-01 经过的毫秒数
- 6、报警信息 警情描述，GB2312 编码
- 7、CRC16 校验 不包含帧头帧尾
- 8、帧尾 2 字节，固定值：0xAABB

示例：

帧头	类型	电子围栏报警	关联的标签为17	报警时间 (2015-05-14 11:26:39.604)	报警信息	CRC16 校验	帧尾
0xCC5F	0x03	0x01	0x0011	1431573999604	进入区域报警，电子围栏名称：禁区 1		0xAABB

2.3 电量信息

CRC16 校验区					
帧头	帧类型	标签数量	电量信息	CRC16 校验	帧尾
2B	1B	1B	N*4B	2B	2B

标签 ID	电量状态	充电状态
2B	1B	1B

- 1、帧头 2 字节，固定值：0xcc5f
- 2、帧类型 1 字节，固定值：0x05
- 3、标签数量：1 字节
- 4、标签 ID 2 字节 标签的编号
- 5、电量状态 1 字节: 0: 0%，1: 20%，2: 40%，3: 60%，4: 80%，5: 100%
- 6、充电状态 1 字 :0 没充电 1 充电
- 7、CRC16 校验 不包含帧头帧尾
- 8、帧尾 2 字节，固定值：0xAABB

2.4 扩展数据信息（客户定制）

CRC16 校验区							
帧头	帧类型	标签 ID	有效数据长度(N)	扩展数据信息	时间戳	CRC16 校验	帧尾

2B	1B	2B	2B	NB	8B	2B	2B
----	----	----	----	----	----	----	----

- 1、帧头 2 字节，固定值：0xCC5f
- 2、帧类型 1 字节，固定值：0x08
- 3、标签 ID 2 字节
- 4、有效数据长度(N) 2 字节，指扩展数据信息长度
- 5、扩展数据信息 N 字节 指有效数据内容
- 6、时间戳 8 字节 自 1970-01-01 经过的毫秒数
- 7、CRC16 校验 不包含帧头帧尾
- 8、帧尾 2 字节，固定值：0xAABB

2.5 扩展报警信息

该报警信息电子围栏告警更详细。

CRC16 校验区							
帧头	帧类型	报警类型	关联 ID	报警时间	附加信息	CRC16 校验	帧尾
2B	1B	1B	2B	8B	120	2B	2B

附加信息（仅围栏有效）2015-10-09					
围栏 ID	当前位置		边信息	顶点信息	
围栏标识	X 位置	Y 位置	边名称	顶点名称	离顶点距离
8B	4B	4B	30B	30B	4B

- 1、帧头 2 字节，固定值：0xCC5F
- 2、帧类型 1 字节，固定值：0x09

- 3、**报警类型** 1 字节，0x01 电子围栏报警 0x02 表示 SOS 报警 0x03 表示剪断报警 0x04 表示消失报警, 0x07 表示低电量报警 (非电子围栏报警附加信息不可用)
- 4、**关联 ID** 关联的标签 ID,指示谁触发了报警
- 5、**报警时间** 自 1970-01-01 经过的毫秒数
- 6、**附加信息** 120 字节，仅当报警类型为 0x01 时，前 32 字节有效，其他内容预留，对其他报警类型数据内容预留。当报警类型为 0x01 时，格式如上所示：8 字节围栏 ID 标识，8 字节当前标签位置（**X, Y 坐标值，各占 4 字节，单位：cm**），30 字节闯入围栏边界的边名称（GBK 编码），30 字节闯入围栏边界的边顶点名称（GBK 编码），4 字节当前位置距边顶点的距离。其他预留。
- 7、**CRC16 校验** 不包含帧头帧尾
- 8、**帧尾** 2 字节，固定值：0xAABB

附录:

1. CRC16 校验 (C 语言代码):

```

/*****
// crc check function
/*****

static const unsigned char aucCRCHi[] =

{

    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40

};

```

```

static const unsigned char aucCRCLo[] =
{
    0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7,
    0x05, 0xC5, 0xC4, 0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E,
    0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09, 0x08, 0xC8, 0xD8, 0x18, 0xD9,
    0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD, 0x1D, 0x1C, 0xDC,
    0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
    0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32,
    0x36, 0xF6, 0xF7, 0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D,
    0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A, 0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38,
    0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE, 0x2E, 0x2F, 0xEF,
    0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
    0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1,
    0x63, 0xA3, 0xA2, 0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4,
    0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F, 0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB,
    0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8, 0xB9, 0x79, 0xBB, 0x7B, 0x7A, 0xBA,
    0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
    0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,
    0x50, 0x90, 0x91, 0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97,
    0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C, 0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E,
    0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98, 0x88, 0x48, 0x49, 0x89,
    0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
    0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83,
    0x41, 0x81, 0x80, 0x40
};

```

```

unsigned short crc16Check(unsigned char *frame, unsigned char len )
{
    unsigned char ucCRCHi = 0xFF;
    unsigned char ucCRCLo = 0xFF;
    unsigned short iIndex;

    while( len-- )
    {
        iIndex = ucCRCLo ^ ( *frame );

```

```
ucCRCLo = ( unsigned char )( ucCRChi ^ aucCRChi[iIndex] );
```

```
ucCRChi = aucCRCLo[iIndex];
```

```
frame++;
```

```
}
```

```
return (unsigned short)( (unsigned short)ucCRChi << 8 | ucCRCLo );
```

```
}
```