**接口API**

|  |  |
| --- | --- |
| 文档编写： | 当前版本： |
|  |  |
|  |  |
|  |  |

目录

[1.主动安全车辆信息上传 1](#_Toc20492595)

[2.主动安全报警信息上传 3](#_Toc20492596)

[3.主动安全报警附件上传 8](#_Toc20492597)

[4.实时位置信息 9](#_Toc20492598)

# 1.主动安全车辆信息上传

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口地址 | http://bdapi.qhd12328.com/acceptSaftyCarInfo.jsn | | | | |
| **请求参数** | **序号** | **参数** | **数据类型** | **是否必填** | **参数说明** |
| **1** | **compId** | **字符串** | 是 | 运营商应用编号 |
| **2** | **sign** | **字符串** | 是 | 参数签名 |
| **3** | **dataList** | **字符串** | 是 | 车辆信息字符串-使用json格式的字符串表示，一次可以传递多个车辆信息。  [{"carPlateNum":"冀C12345","carPlateColor":"2","carFrameNum":"123456","installTime":"20190829191919","modelCode":"DX5","manufacturerName":"锐明","manufacturerId":"123456","modelId":"123456","sim":"20190829153232","dsmId":"123456","adasId":"123456","channelEnable":"1,2,3,4","channelNum":10," protocol":"*808*",}]  数据格式说明:  ==== carPlateNum: 车牌号  ==== carPlateColor:车牌颜色（1-蓝色,2-黄色,3-黑色,4-白色,9-黄绿相间）  ==== carFrameNum:车架号  ====installTime:安装时间（格式yyyymmddhhmmss）  ====modelCode: 设备型号  ==== manufacturerName: 制造商名称  ==== manufacturerId: 制造商id  ==== modelId:设备ID  ====sim: sim卡号  ==== dsmId: dsm设备ID  ==== adasId: adas设备ID  ==== channelEnable:可用视频通道  ==== channelNum:通道总数  ==== protocol:协议类型 808/1078/n9m |
| **返回json** | 成功返回: {"success":true}  失败返回：{"success":false,"errMsg":"失败原因"} | | | | |
| **响应参数** | **序号** | **参数** | **参数说明** | | |
| **1** | success | 请求成功或者失败 | | |
| **2** | errMsg | 失败原因 | | |
|  |  |  | | |
|  |  |  | | |

# 2.主动安全报警信息上传

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口地址 | http://bdapi.qhd12328.com/acceptSaftyWarning.jsn | | | | |
| **请求参数** | **序号** | **参数** | **数据类型** | **是否必填** | **参数说明** |
| **1** | **compId** | **字符串** | 是 | 运营商应用编号 |
| **2** | **sign** | **字符串** | 是 | 参数签名 |
| **3** | **dataList** | **字符串** | 是 | 报警信息字符串-使用json格式的字符串表示，一次可以传递多个报警信息。  [{"bId":"123456","deviceno":"64S0112","sim":"40281816810","lon":"118.804231","lat":"40.124660","speed":"50","direction":"12","starttime":"20200130191919","endtime":"20200130191919","carPlateNum":"冀C12345","carPlateColor":"2","driverName":"张三","driverCard":"123456","warningType":"1","warningInfo":"123456","alarmTimeSecond":"20","detailInfo":"123456","alarmState":"开始","handleType":"电话提醒","handleTime":"20200130191919","handlePeople":"测试"}]  数据格式说明:  ==== bId: 必填,唯一,用于关联数据 报警主键  ==== deviceno:设备编号  ==== sim:SIM卡号  ====lon:经度  ====lat: 纬度  ==== speed: 速度  ==== direction: 方向  ==== starttime:报警开始时间（格式yyyymmddhhmmss）  ====endtime: 报警结束时间（格式yyyymmddhhmmss）  ==== carPlateNum: 车牌号  ==== carPlateColor: 车牌颜色（1-蓝色,2-黄色,3-黑色,4-白色,9-黄绿相间）  ==== diverName: 驾驶员姓名  ==== driverCard: 驾驶员身份证号  ==== warningType: 报警类型（见下面2.2）  ==== warningInfo: 报警详情  ==== detailInfo: 处理结果  ==== alarmState: 报警状态（开始，结束，处理）  ==== handleType:处理方式  ==== handleTime: 处理时间（格式yyyymmddhhmmss）  ==== handlePeople: 处理人  ==== roadLevel: 道路等级(见下面2.3)  ==== roadSpeedLimit: 道路限速值  ==== alarmTimeSecond:报警时长（秒） |
| **返回json** | 成功返回: {"success":true}  失败返回：{"success":false,"errMsg":"失败原因"} | | | | |
| **响应参数** | **序号** | **参数** | **参数说明** | | |
| **1** | success | 请求成功或者失败 | | |
| **2** | errMsg | 失败原因 | | |
|  |  |  | | |
|  |  |  | | |

**2.2报警类型**

1--------------紧急报警');

2--------------超速报警');

3--------------疲劳驾驶');

4--------------危险预警');

5--------------GNSS模块发送故障');

6--------------GNSS天线未接或被剪断');

7--------------GNSS天线短路');

8--------------终端主电源欠压');

9--------------终端主电源掉电');

10--------------终端LCD或显示器故障');

11--------------TTS模块故障');

12--------------摄像头故障');

13--------------道路运输证IC卡模块故障');

14--------------超速预警');

15--------------疲劳驾驶预警');

16--------------抛洒报警');

17--------------不在指定区域卸料报警');

18--------------非指定区域行驶报警');

19--------------当天累计驾驶超时');

20--------------超时停车');

21--------------进出区域');

22--------------进出线路');

23--------------路段行驶时间不足/过长');

24--------------路线偏离报警');

25--------------车辆VSS故障');

26--------------车辆油量异常');

27--------------车辆被盗');

28--------------车辆非法点火');

29--------------车辆非法移位');

30--------------碰撞报警');

31--------------侧翻报警');

32--------------非法开门报警');

33--------------平台禁行时段运行报警');

34--------------平台区域报警');

35--------------平台线路偏移报警');

36--------------平台分段限速报警');

37--------------平台线路关键点报警');

38--------------司机打电话检测报警');

39--------------平台超速报警');

40--------------平台夜间行驶(2-5点行驶)报警');

41--------------平台疲劳驾驶报警');

42--------------平台夜间限速报警');

43--------------三级路面报警');

44--------------车辆非法运营报警');

45--------------告警区域报警');

46--------------平台控制区域报警');

47--------------平台核心区域报警');

49--------------平台超速报警');

101--------------视频信号丢失报警');

102--------------主存储器故障报警');

103--------------灾备存储单元故障报警');

104--------------其他视频设备故障报警');

105--------------客车超载报警');

107--------------特殊报警录像达到存储阈值报警');

130--------------图像移动侦测报警');

131--------------急弯/S弯报警');

132--------------激烈颠簸');

133--------------川标疲劳驾驶');

134--------------打电话');

135--------------抽烟');

136--------------未注视前方');

137--------------静默求助');

138--------------漏油报警');

139--------------车道偏离');

140--------------急刹车报警');

141--------------加油提示');

145--------------控制区域报警');

146--------------核心区域报警');

151--------------前车碰撞预警');

152--------------车道偏离预警');

153--------------行人避撞预警');

154--------------车距检测预警');

155--------------超速预警');

156--------------无驾驶员报警');

157--------------驾驶员分心报警');

161--------------SIM卡月流量报警');

162--------------SIM卡日流量报警');

163--------------高温报警');

164--------------低温报警');

165--------------湿度报警');

171--------------GPS未定位报警');

180--------------前向碰撞预警');

181--------------车辆偏离预警');

182--------------车距过近预警');

183--------------限速标志识别');

184--------------行人碰撞报警');

185--------------频繁变道预警');

186--------------疲劳驾驶报警');

187--------------接打电话报警');

188--------------抽烟报警');

189--------------注意力分散报警');

190--------------异常报警');

191--------------轮胎异常报警');

195--------------驾驶员变更');

196--------------道路标识超限');

197--------------道路标志识别');

198--------------主动抓拍事件');

199--------------障碍物报警');

200--------------实线变道');

301--------------视频信号丢失报警');

302--------------视频信号遮挡报警');

303--------------存储单元故障报警');

304--------------其他视频设备故障报警');

306--------------异常驾驶行为报警');

307--------------特殊报警录像达到存储阈值报警');

330--------------低头');

331--------------玩手机');

332--------------未系安全带报警');

333--------------双手离开方向盘报警');

340--------------后方接近报警');

341--------------左侧后方接近报警');

342--------------右侧后方接近报警');

350--------------车道超载报警');

351--------------站外上客报警');

352--------------驾驶员不匹配');

260--------------未系安全带报警');

261--------------无派车单报警');

262--------------路口未减速报警');

305--------------客车超员报警');

345--------------盲区报警');

353--------------频率异常报警');

334--------------单手离开方向盘报警');

365--------------乘客干扰驾驶员报警');

370--------------未礼让行人报警');

371--------------路口超速报警');

**2.3道路等级**

41000 高速公路,42000 国道,43000 主要大街、城市快速路,51000 省道,  
44000 主要道路,45000 次要道路,52000 乡公路,53000 县乡村内部道路  
54000 县乡村内部道路,47000 普通道路,49 非导航道路

# 3.主动安全报警附件上传

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口地址 | http://bdapi.qhd12328.com/acceptSaftyWarningFiles.jsn | | | | |
| **请求参数** | **序号** | **参数** | **数据类型** | **是否必填** | **参数说明** |
| **1** | **compId** | **字符串** | 是 | 运营商应用编号 |
| **2** | **sign** | **字符串** | 是 | 参数签名 |
| **3** | **dataList** | **字符串** | 是 | 附件信息字符串-使用json格式的字符串表示，一次可以传递多个附件信息。  [{"bId":"123456","carPlateNum":"冀C12345","carFrameNum":"12345","fileType":"picture","fileName":"00\_05\_6402\_0\_9001df7bec914c5cba20ebd581513ba8.jpg","fileUrl":"http://39.106.199.153:6006/vehicleAlarmData/20190823/9001df7bec914c5cba20ebd581513ba8/00\_05\_6402\_0\_9001df7bec914c5cba20ebd581513ba8.jpg"}]  数据格式说明:  ==== bId: 必填,唯一,用于关联数据  ==== carPlateNum: 车牌号  ==== carFrameNum:车架号  ====fileType:文件类型（照片:picture 视频:video）  ====fileName: 文件名  ==== fileUrl: 文件的地址路径 |
| **返回json** | 成功返回: {"success":true}  失败返回：{"success":false,"errMsg":"失败原因"} | | | | |
| **响应参数** | **序号** | **参数** | **参数说明** | | |
| **1** | success | 请求成功或者失败 | | |
| **2** | errMsg | 失败原因 | | |
|  |  |  | | |
|  |  |  | | |

# 4.实时位置信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口地址 | http://bdapi.qhd12328.com/acceptCarGps.jsn | | | | |
| **请求参数** | **序号** | **参数** | **数据类型** | **是否必填** | **参数说明** |
| **1** | **compId** | **字符串** | 是 | 运营商应用编号 |
| **2** | **sign** | **字符串** | 是 | 参数签名 |
| **3** | **dataList** | **字符串** | 是 | 附件信息字符串-使用json格式的字符串表示，一次可以传递多个附件信息。  [{"altitude":1216,"vehicle\_color":"黄色","vehicle\_no":"冀C16569","vehicle\_type":"危险品运输车","driver\_card":"","lon":111.578743,"deviceno":"78S0164","speed":48,"driver\_name":"","datetime":"20190927152416","sim":"40281815601","alarm":0,"state":1,"lat":37.29578,"mileage":306.2,"direction":348}]数据格式说明:  ==== altitude: 海拔高度  ==== vehicle\_color: 车牌颜色  ==== vehicle\_no: 车牌号  ==== vehicle\_type: 车辆类型  ====driver\_card: 驾驶员身份证  ==== lon: 经度  ==== deviceno: 终端设备编号  ==== speed: 速度  ==== driver\_name: 驾驶员姓名  ==== datetime: Gps时间 （yyyymmddhhmmss）  ==== sim: 终端sim卡号  ==== alarm: 报警状态(1:报警，0：正常)  ==== state: 车辆状态(3:未定位  , 2:故障, 1:运营, 0:停运)  ==== lat: 纬度  ==== mileage: 行车里程  ==== direction: 方向 |
| **返回json** | 成功返回: {"success":true}  失败返回：{"success":false,"errMsg":"失败原因"} | | | | |
| **响应参数** | **序号** | **参数** | **参数说明** | | |
| **1** | success | 请求成功或者失败 | | |
| **2** | errMsg | 失败原因 | | |
|  |  |  | | |
|  |  |  | | |

**参数签名:**

第一步，设所有发送的数据为集合M，将集合M内非空参数值的参数按照参数名ASCII码从小到大排序（字典序），使用URL键值对的格式（即key1=value1&key2=value2…）拼接成字符串stringA。

特别注意以下重要规则：

1. ◆ 参数名ASCII码从小到大排序（字典序）；
2. ◆ 如果参数的值为空不参与签名；
3. ◆ 参数名区分大小写；
4. ◆ 验证调用返回或微信主动通知签名时，传送的sign参数不参与签名，将生成的签名与该sign值作校验。

第二步，在stringA最后拼接上key得到stringSignTemp字符串，并对stringSignTemp进行MD5运算，再将得到的字符串所有字符转换为大写，得到sign值signValue。

**例子代码:**

String compId = "000001";  
  
//秘钥key  
String key = "48af4b28-552e-a508-c9cf-c1571b0fb16d";  
  
SortedMap<String,Object> parameters = new TreeMap<String,Object>();  
parameters.put("compId ", compId);  
parameters.put("cardRecs ", cardRecsJson);  
//签名  
String sign = createSign("UTF-8", parameters,key);  
//使用post的方式将compId,cardRecs等业务 参数和sign签名，提交到接口

**createSign方法**

*/\*\*characterEncoding  
 \** ***@param characterEncoding*** *编码格式  
 \** ***@param parameters*** *请求参数  
 \** ***@param*** signKey*签名秘钥  
 \** ***@return*** *\*/*public static String createSign(String characterEncoding,  
 SortedMap<String,Object> parameters,String signKey){  
 StringBuffer sb = new StringBuffer();  
 Set es = parameters.entrySet();  
 Iterator it = es.iterator();  
 while(it.hasNext()) {  
 Map.Entry entry = (Map.Entry)it.next();  
 String k = (String)entry.getKey();  
 Object v = entry.getValue();  
 if(null != v && !"".equals(v)   
 && !"sign".equals(k) && !"key".equals(k)) {  
 sb.append(k + "=" + v + "&");  
 }  
 }  
 sb.append("key=" + signKey);  
 String sign = MD5Util.MD5Encode(sb.toString(), characterEncoding).toUpperCase();  
 return sign;  
}

注意：compId由大数据平台统一分配,signKey由双方共同约定