

# System port definition

## Overview

This interface document defined content is the API interface data required for the standard version of the mobile APP function to be used (including android and ios).

## Module interface detail definition

### System setting part

url:<http://vipapi.18gps.net/>

Macid: [Device IMEI number](#).

### Login

Interface address: url/GetDateServices.asmx/loginSystem

Apply method: HTTP GET

Apply E.g:

url/GetDateServices.asmx/loginSystem?LoginName=test&LoginPassword=123456&LoginType= ENTERPRISE&language=cn&ISMD5=0&timeZone=8&apply=APP&loginUrl=http://vip.18gps.net/

Request parameter:

Name	Type	Must to fill in	Instruction
LoginName	String	Yes	Login account
LoginPassword	String	Yes	Login password
LoginType	String	Yes	Login user type [Company user:ENTERPRISE;Use device number to login:USER]
language	String	Yes	Language type [cn,Chinese;en,English]. According to the language of the mobile phone system to judge , must fill in
ISMD5	Int	Optional	1
timeZone	Int	Yes	Time zone
apply	String	Yes	APP
loginUrl	String	Optional	<a href="http://vip.18gps.net/">http://vip.18gps.net/</a> (URL)
PushID	String	Yes	V4. 18gps. Login name

Return successfully: [JSON]

```
{"success":"true","mds":"verify","LoginType":"ENTERPRISE","grade":"8","id":"555","msg":""}
```

Failure return: [JSON]

```
{"success":"false"}
```

### Device list

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

Apply E.g:

url/GetDateServices.asmx/GetDate?method=getDeviceList&mds=

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call Function

Return [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe"(Means get the device list successfully),"rows":[{"objectid":"00","platenumber","fullName","macid","blockdate"(Means expire date),"uptime"(Means data update time), "offline", "speed", "defenceStatus"}]}
```

Individual return field descriptions:

Name	Type	Instruction
objectid	Int	Unique ID of one row data
offline	Int	0: off line; 1: static; 2: moving
defenceStatus	Int	0: Undefence; 1:Defence

### Get vehicle information

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

Apply E.g:

url/GetDateServices.asmx/GetDate?method=getCurrentDefault&mds=&macid=

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call Function
macid	String	Yes	Device system unique code

Return [JSON]

```
{ "success": "true", "errorCode": "200", "errorDescribe": "(Means take the last time success vehicle or default to view the vehicle successfully)", "rows": [{"platenumber", "speed", "defenceStatus", "fullName", "macid", "objectid", "oilPrice"}]}
```

Individual return field descriptions:

Name	Type	Instruction
objectid	Int	Unique ID of one row data
defenceStatus	Int	0: not enabled; 1: enabled

## Historical track

Interface address: url/GetDataServices.aspx/GetData

Apply method: HTTP GET

Apply E.g:

url/GetDataServices.aspx/GetData?method=getHistoryMByMUtc&option=cn&mids=qwesfs&school\_id=0&custid=0&userID=0&mapType=BAIDU&pwd=123456&from=11122215&to=112223444

Request parameter:

Name	Type	Must to fill in	Instruction
method	String	Yes	Call Function
option	String	Yes	Language type [cn,Chinese;en,English]
mids	String	Yes	Token user identity
school	Int	Yes	Login user ID
custid	Int	Yes	Current user ID
userID	Int	Yes	Device ID
mapType	String	Yes	Map type[,BAIDU;GOOGLE]
pwd	String	Yes	Device password
from	String	Yes	History trajectory start time (in milliseconds)
to	String	Yes	History trajectory end time (in milliseconds)

Return [STRING]:

"Longitude, latitude, time (milliseconds), speed, direction;" ( separated by comma)

## Get the online vehicle latest location.

Interface address: url/GetDataServices.aspx/GetData

Apply method: HTTP GET

E.g:

url/GetDataServices.aspx/GetData?method=getOnlineGpsInfoByIDUtc&school\_id=0&custid=00&userIDs=00&mapType=BAIDU&option=cn&t=1408960100602&mids=desfs

Request parameter:

Name	Type	Must to fill in	Instruction
method	String	Yes	Call Function
school	Int	Yes	Login user ID
custid	Int	Yes	Current user ID
userIDs	Int	Yes	Device ID
mapType	String	Yes	Map type[,BAIDU;GOOGLE]
option	String	Yes	Language type [cn,Chinese;en,English]
t	String	Yes	Timestamp
mids	String	Yes	Token user identity

Return [JSON]:

```
{ "key": {"sys_time": "0", "jingdu": "1", "weidu": "2", "ljingdu": "3", "lweidu": "4", "datetime": "5", "heart_time": "6", "su": "7", "status": "8", "hangxiang": "9", "user_id": "10", "server_time": "11", "statenumber": "12"}, "records": [{"system time", "longitude", "latitude", "longitude(LBS)", "latitude(LBS)", "Update record time", "Heart beat time", "speed", "status 【 As below status 】", "Direction", "Device ID", "server time", "291.09,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00"}]}
```

Note:The data in the key is explanatory, corresponding to each value of each array in the records array, such as status corresponding to "status 【 As below status 】", generally used: records[0][ status]

status(instruction):

Data: abcdef	Six consecutive characters
a : acc status	0:off, 1: on
b : Defence status	0: un-defence, 1: defence
c : oil status	0: off, 1: on
d : charging status	
e : Door status	0: off, 1: on
f : position status	1:positioned, 0:not positioned

--Number 1:accState,

--Number 2: defenceStatus ,

--Number 3: oilState ,

--Number 4: charging status,

--Number 5: doorState

--Number 6: position status

Statenumber ( instruction):

Data: "291.09,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00"	Sixteen consecutive comma separated characters
--	---

--Number 1: mileage(mil),  
 --Number 2: oil ,  
 --Number 3: weight ,  
 --Number 4: temperature 1 (tempc),  
 --Number 5: standby battery capacity (betteryV),  
 --Number 6: Mains power voltage (powerV),  
 --Number 7: Number of GPS satellites  
  
 (gpscount),  
 --Number 8: gsm signal strength (gsmlevel) ,  
 --Number 9: clockwise (ClockwiseState) ,  
 --Number 10: vehicle status(VehicleState),  
 --Number 11: lock number (lockcnt) ,  
 --Number 12: temperature 2 (temp1),  
 --Number 13: temperature 3 (temp2) ,  
 --Number 14: temperature 4 (temp3)  
 --Number 15: height (height)  
 --Number 16: position signal type (SignalType)

## Get the vehicle latest location.

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get the login port)&method=getUserAndGpsInfoByIDsUtcNew

Request example:

http://myautogps.com/GetDateServices.asmx/GetDate?timetick=1437106122537&method=getUserAndGpsInfoByIDsUtcNew&school\_id=00&custid=00&mapType=BAIDU&option=cn&user\_id=88888&mds=44897c9325f61a1bb&time=402

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function

Return successfully [JSON]:

```

{
  "data": [
    {
      "key": {
        "sys_time": "0",
        "user_name": "1",
        "jingdu": "2",
        "weidu": "3",
        "ljingdu": "4",
        "lweidu": "5",
        "datetime": "6",
        "heart_time": "7",
        "su": "8",
        "status": "9",
        "hangxiang": "10",
        "sim_id": "11",
        "user_id": "12",
        "sale_type": "13",
        "iconType": "14",
        "server_time": "15",
        "product_type": "16",
        "expire_date": "17",
        "group_id": "18",
        "statenumber": "19"
      },
      "records": [
        [
          "1437104751000",
          "00F2FD0071",
          "113.39563833636",
          "22.9804579500247",
          "0",
          "0",
          "1437104779230",
          "1437104779230",
          "0.00",
          "000001",
          "0",
          "F2FD0071",
          "2371",
          "",
          "01",
          "1437118571878",
          "XYZYL",
          "1468713600000",
          "",
          "0,0,0,0,0,0,99.00,0,0,0,0,0,0,0"
        ]
      ]
    }
  ]
}

```

```

    ],
    "groups": []
  }
],
"success": "true",
"errorCode": "200",
"errorDescribe": ""
}

```

Failure return [JSON]:

```
{"success":"false","errorCode":"500","errorDescribe":"description","data":[]}
```

## User login (Note: After this interface is called, the device interface is activated in succession according to the returned content)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/azxuserRegister?reg\_phone=18124622462&reg\_pass=123456&reg\_macid=123451

Request parameter:

Name	Type	Must to fill in	Instruction
reg_phone	String	Yes	Device SIM card number
reg_pass	String	Yes	Password
macid	String	Yes	Device system unique number

Return successfully [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe":"egistered successfully","returnID":"ReturnID"}
```

Failure return [JSON]:

```
{"success":"false","errorCode":"400","errorDescribe":"registered failed"}
```

## Active device

Interface address: url/GetDateServices.asmx/macActivatings

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/macActivatings?reg\_num&password&macid&login\_id

Request parameter:

Name	Type	Must to fill in	Instruction
reg_num	String	Yes	Vehicle plate number
password	String	Yes	Password
macid	String	Yes	Device system unique number
login_id	String	Yes	The feedback ID after login

Return successfully [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe":"registered successfully"}
```

Failure return [JSON]:

```
{"success":"false","errorCode":"400","errorDescribe":"registered failed"}
```

## Send command 2015-04-13 to modify

Interface address: url/GetDateServices.asmx/setUserStatus

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/setUserStatus?method=sendCommand&macid =12313&cmd=SAFEON&pwd=123&mds=dwesfs&param=

Request parameter:

Name	Type	Must to fill in	Instruction
method	String	Yes	Call function
macid	String	Yes	Device system unique number
mds	String	Yes	Token user identity
cmd	String	Yes	Command code sent
pwd	String	Yes	Optional parameter, whether the command code requires a password
param	String	Yes	Optional parameter, corresponding to the instruction code

Cmd instruction:

SAFEON	Position defence
SAFEOFF	Position un-defence
OPENTRUNK	Open the trunk
OPENRELAY	Ignition on, device password required
CLOSERELAY	Turn off, need device password
RESET	Restart device, device password required
OPENENGINE	Start the engine
CLOSEENGINE	Stop the engine
SETALARM	Equipment vibration alarm level, requires parameter [A, B, C] A: Alarm mode. 0, no alarm; 1, call; 2, SMS; 3, call + SMS, default 1 B: Alarm level. Level1-Level 5, default to be level 3. C: Alarm cooling time, no alarm is repeated during cooling time. Unit: minutes. Default 5 minutes. (Alarm time interval )
LOCKDOOR	Lock vehicle door
OPENDOOR	Open vehicle door
LOCKWINDOW	Lock vehicle window
OPENWINDOW	Open vehicle window
VOICELIGHTSEEKCAR	Sound and light search vehicle

AIRCONDITIONSWITCH	Air conditioning switch, requires parameter [A, B] A: Boot time. (default 12 minutes, range 1-20 minutes) B: Temperature limit. (default 26 ° C, range 16-30 ° C)
CHARGESET	Charge management, need parameter [A, B] A: Charging speed. 0, slow; 1, fast B: Charging time (default 12h, range 1-24h).
SPEEDLIMIT	Speed limit, requires parameter [A, B] A: Speed limit (default 90km/h, range 0km/h -120km/h)

Return successfully: [JSON]

```
{ "success": "true", "cmdNo": "return to command ID", "type": "0", "cmdtype": "command type", "macid": "device ID", "mactype": "protocol code", "user_id": "user ID", "platenumber": "vehicle plate number", "err": "0", "info": "Return a successful description" }
```

Failure return: [JSON]

```
{ "success": "false", "message": "View the corresponding code below", "err": "1", "info": "Failed to send" }
```

Message instruction:

PERMISSIONS	Demo users cannot send instructions
SEND_FAIL	Command failed to send
PWD_ERROR	Wrong password
USER_LEAVE	While device offline, cannot send command

## Get vehicle information 2015-02-10

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?method=loadUserV3&macid=&mids=

Request parameter:

Name	Type	Must to fill in	Instruction
method	String	Yes	Call function
macid	String	Yes	Device system unique number
mids	String	Yes	Token user identity

Return [JSON]:

```
{ "success": "true", "errorCode": "error code", "data": [
{ "platenumber": "vehicle plate number", "adminTelr": "admin mobile number", "macid": "device ID", "fullName": "device name", "linktel": "contact mobile number", "adminTel1": "SOS number 1", "adminTel2": "SOS number 2", "pass": "password", "sim": "SIM card number" } ] }
```

## Get the alarm list

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?method=getAlarmList&macid=设备号&classify=报警类型&mids=(get login port)&currentPage=1 (累加)

Request parameter:

Name	Type	Must to fill in	Instruction
method	String	Yes	Call function
macid	String	Yes	Device system unique number
mids	String	Yes	Token user identity

Return [JSON]:

```
{ "success": "true", "errorCode": "200", "errorDescribe": "取车辆报警成功", "rows": [
{ "classify": "alarm type", "fullName": "device name", "speed": "speed", "lat": "latitude", "lon": "longitude", "ptime": "alarm time", "id": "alarmid" } ] }
```

Which alarm types are required to use this interface, attached the list .

## Get vehicle status data

Interface address: url/GetDateServices.asmx/GetDate

Apply method: http get

E.g: url/GetDateServices.asmx/GetDate?mids=(get login port)&method=getCarHardWare&macid=device name

Request parameter:

Name	Type	Must to fill in	Instruction
mids	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Return [JSON]:

```
{ "success": "true", "errorCode": "200", "errorDescribe": "get vehicle status data successfully", "obdList": [ { "obdcode": "healthy score", "obdcodestr": "healthy score", "obddescription": "100" } ], "rows": [ { "macid": "device ID", "P1": "0", "P2": "699", "P3": "85", "P4": "0", "P5": "0", "P6": "48", "P7": "4", "P8": "0", "P9": "", "P10": "37", "P11": "0", "P12": "7.5", "P13": "1.05", "P14": "0", "P15": "0", "P16": "0", "P17": "0", "P18": "14.7", "P47": "0", "speed": "8", "fraction": "100", "StateTime": "2014/9/16 7:43:13", "CurCodeTime": "2014/9/16 15:43:13", "HisCodeTime": "" } ] }
```

Refer to the attachment here: ( OBD Uniform agreement.doc )

Individual return field description:

Name	Instruction
obddescription	healthy score
fraction	healthy score
StateTime	Vehicle condition update time
CurCodeTime	Fault code update time
HisCodeTime	Historical fault code update time

## Get vehicle status data (Home page)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: http get

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getCarHardWares&macid=device name

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Return [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe":"get vehicle status data successfully","rows":{"macid":"00000365869053521933","iTime":"Insurance expiration time","aTime":"Annual review expiration time","fraction":"healthy score","carWashNumber":"No need to use","WaxingNumber":"No need to use"}}
```

## Set GPS status

Interface address: url/GetDateServices.asmx/GetDate

Apply method: http get

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=gpsOnOff&macid=device name& state=(0:online,1: stealth)

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
state	Int	Yes	0: set to be online,1:stealth

Return [JSON]:

```
{"success":"true|false","errorCode":"200","errorDescribe":"description"}
```

## Get GPS status of defence and un-defence

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getUserStatus&macid=device name

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Return [JSON]:

```
{"success":"true|false","errorCode":"200","errorDescribe":"description","rows":{"defenceStatus":"defence and un-defence","gpsOnOff":"gps status"}}
```

Return string description

Name	Type	Instruction
defenceStatus	Int	Defence status[0:un-defence; 1:defence]
gpsOnOff	Int	Gps status [0:online; 1:stealth]

## Get the itinerary (the trip within one day) 2015-01-21 add V3(20150130)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getStrokeV3&macid=device name&beginTime&endTime

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
beginTime	String	Yes	Start time(timestamp, number of milliseconds), start time and end time must be within the day
endTime	String	Yes	End time (timestamp, number of milliseconds), start time and end time must be within the day

Return [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe":"get vehicle itinerary successfully","rows":{"totalfuelUsed":"total fuel capacity","distance","fuelHKM":"100 kilometers fuel consumption","totalSpeedoverSeconds":"over speed times","maxspeed":"maximum speed","brakeTimes","emergencyBrakeTimes","speedupTimes","emergencySpeedupTimes","speed","maxTempc":"maximum temperature","btime":"begin time","etime":"end time","strokeTime","fraction":"healthy score","jsType":"(0:rapidly 1: Standard 2: Steady 3: Slowly)","driveTime(second)"},"avgFuel":"average 100 kilometers fuel consumption","totalfuel":"total fuel consumption","totalmil":"total milage"}}
```

## Get APP configuration [device offline time, static time etc.]2015-01-21 add V3

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getAPPconfgV3

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function

Return [JSON]:

```
{"success": "true", "outTime", "One-click detection, offline time (minutes)", "CurCodeURL", "Fault repair entranceURL", "OFF_LINE_TIME": "offline (minutes)", "STATIC_SECONDS": "static (minutes)"} 
```

## Set fuel/oil price

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=setOilPrice&macid=device name&oilPrice=fuel/oil price

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
oilPrice	double	Yes	Price

Return [JSON]:

```
{"success": "true|false"}
```

## Add fueling information

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=AddFuelInfo&Fuel&Price

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
Fuel	double	Yes	Oil capacity
Price	double	Yes	Price

Successfully Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "描述", "data": []}
```

Failed Return [JSON]:

```
{"success": "false", "errorCode": "500", "errorDescribe": "描述", "data": []}
```

## Get fuel added information

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=GetFuelList&beginTime&endTime

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
beginTime	String	Yes	Start time(timestamp, number of milliseconds),
endTime	String	Yes	End time(timestamp, number of milliseconds),

Successfully Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "description", "data": [{"Id": "Row data unique ID", " UserId", " Fuel", " Price", " AddTime"}, {"Id": "Row data unique ID", " UserId", " Fuel", " Price", " AddTime"}]}
```

Failed Return [JSON]:

```
{"success": "false", "errorCode": "500", "errorDescribe": "description", "data": []}
```

## Delete fuel added information

## Set geo-fence range

Interface address: url/GetDateServices.asmx/GetDate?method=SetFenceRange

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?method=SetFenceRange&macid=device ID &defenceRad=Fence radius&mds=

Request parameter:

Name	Type	Must to fill in	Instruction
method	String	Yes	Call function
macid	String	Yes	Device ID
defenceRad	double	Yes	Fence radius
mids	String	Yes	Token user identity

Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "modified successfully", "data": []}
```

## Get geo-fence range

Interface address: url/GetDateServices.asmx/GetDate?method=GetFenceRange

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?method=GetFenceRange&macid=device ID&mids=

Request parameter:

Name	Type	Must to fill in	Instruction
macid	String	Yes	Device ID
method	String	Yes	Call function
mids	String	Yes	Token user identity

Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "修改成功", "data": [{"defenceRad": "200"}]}
```

## Get Service provider

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mids=(get login port)&method=serviceProviders&macid=device name&mapType=BAIDU&trade=服务商&lat=纬度&lng=经度

Request parameter:

Name	Type	Must to fill in	Instruction
mids	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
mapType	String	Yes	Map type(BAIDU; GOOGLE)
trade	String	Yes	Service provider description, such as: car wash shop, beauty shop, etc., service provider is car wash shop
lat	Double	Yes	longitude
lng	Double	Yes	latitude

Return:

```
{"success": "true", "info": "search successful", "err": "0", "data": [{"id": "5752", "deptname": "service provider name", "tel": "service provider telephone", "contact", "lat": "latitude", "trade": "service provider", "addr": "address", "email": "", "totalMacCount": "", "lng": "longitude", "distance": "distance(Kilometer)"}]}
```

## Get fault code

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mids=(get login port)&method=getCurCode&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mids	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Return [JSON]:

```
{"success": "true", "info": "search successful", "err": "0", "data": [{"id": "14532", "obdcode": "4886", "obdcodestr": "fault code", "obddescription": "fault description"}]}
```

## Get fault code 2015-01-23 add V3(20150130)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mids=(get login port)&method=getCurCodeV3&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mids	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Return [JSON]:

```
{"success": "true", "info": "search successful", "err": "0", "fraction": "healthy score", "data": [{"id": "14532", "obdcode": "4886", "obdcodestr": "fault code", "obddescription": "fault description", "sysId": "system ID"}], "obdSys": [{"sysId": "system ID", "obdTitle": "system title", "obdCount": "Number of faults"}], ...}
```



## Take the vehicle latitude and longitude

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getCarLocation&macid=device ID&mapType=(BAIDU|GOOGLE)

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
mapType	String	Yes	Map type(BAIDU; GOOGLE)

Return [JSON]:

```
{"success":"true|false","lat":"latitude","lon":"longitude","visited":"SOS status","ptime":"SOS time"}
```

## Get 4s information

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getInfoList&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Successfully Return [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe":"get vehicle information successfully
```

```
","rows":[{"nextMantanceMil":"Mileage before maintenance expired ","insurance":"Insurance expire time","annual":"Annual review expiration time","tireReplace":"Time to change tires","brakeReplace":"Time to change brake wire time","oilReplace":"Time to change oil ","stel":"SOS Contact"}]}
```

Failed Return [JSON]:

```
{"success":"false","errorCode":"500","errorDescribe":"get wrong vehicle information"}
```

## Get SMS inquiry 20150325

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=GetGpsUser4SInfo&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Successfully Return [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe":"get 4s shop information successfully"," data":[{"nextMantanceMil":"Mileage before maintenance expired 【display-10000: Not set】","nextMantanceDate":"maintenance expire time【display-10000: Not set】","insurance":"Insurance expire time 【display-10000: Not set】","annual":"Annual review expiration time【display-10000: Not set】","tireReplace":" change tires","brakeReplace":" change brake wire ","oilReplace":"change oil","stel":"SOS Contact"}]}
```

Failed Return [JSON]:

```
{"success":"false","errorCode":"500","errorDescribe":"get 4s shop information successfully"," data":[]}
```

## Update unit password

Interface address: url/ GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=&method=modifyEnterprisePwd&reg\_pass=&reg\_olbpass=

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
reg_pass	String	Yes	New password
reg_olbpass	String	Yes	Old password

Successfully Return [JSON]:

```
{"success":"true","errorCode":"200","errorDescribe":"modified successfully"}
```

Failed Return [JSON]:

```
{"success":"false","errorCode":"500","errorDescribe":"description"}
```

## Reservation

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=reservation&macid =device ID &linktel =contact mobile phone number&linkman=contact&serviceType=reservation type(["General Maintenance", "Maintenance Car", "Decorative Beauty", "Car Wash Cleaning", "Other Items"]);&appointmentTime=reservation time

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
linktel	String	Yes	Contact mobile phone number
linkman	String	Yes	Contact
serviceType	Int	Yes	Reservation type 0: General Maintenance, 1:Maintenance Car, 2: Decorative Beauty, 3: Car Wash Cleaning, 4: Other Items
appointmentTime	DateTime	Yes	Reservation time

Successfully Return:

```
{"success":"true"}
```

Failed Return:

```
{"success":"false"}
```

## Reservation list

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=reservationList&macid =device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Successfully Return [JSON]:

```
{"success":"true","rows":[{"id":"","linktel":"contact mobile phone number","linkman":"contact","reserveTime":"reservation time","reserveType":"reservation type","macid":"device ID","depiction":"reply content","reserveStatus":"reservation status"}]}
```

Individual return field description:

Name	Type	Instruction
reserveTime	DateTime	reservation time
reserveType	Int	Reservation type 0: General Maintenance, 1:Maintenance Car, 2: Decorative Beauty, 3: Car Wash Cleaning, 4: Other Items
reserveStatus	Int	Reservation status [0: unprocessed; 1: processed;]

Failed Return [JSON]:

```
{"success":"false"}
```

## SOS calling number

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getHelp&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Successfully Return [JSON]:

```
{"factoryPhone":"Repair call","FirePhone":"","safetyPhone":"insurance phone number","ctel":"SOS contact","sorttel":"service provider phone number","linktel":"contact phone number"}
```

## Submit service provider

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=submitRescue&macid =device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Successfully Return [JSON]:

```
{"success": "true"}
```

Failed Return [JSON]:

```
{"success": "false"}
```

## Dispatching vehicles

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getDispatchLocation&macid =device ID&mapType=(BAIDU GOOGLE)

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
mapType	String	Yes	Map type(BAIDU; GOOGLE)

Successfully Return [JSON]:

```
{"success": "true", "dispatchList": [{"dispatchLat": "latitude", "dispatchLon": "longitude"}]}
```

## Processing alarm ID

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=setAlarmID&macid=device ID

&id=alarm ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
id	Int	Yes	Unique identifier of the alarm message ID

Successfully Return [JSON]:

```
{"success": "true|false"}
```

## Handling alarm types

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=setAlarmClassify&macid =device ID&classify=alarm type

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
classify	Int	Yes	Alarm type

Successfully Return [JSON]:

```
{"success": "true|false"}
```

Remarks: Alarm type classifyYesInt type

## Number of alarms

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getAlarmCount&macid=device ID &classify=Type of alarm to be counted (Comma separated)

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
classify	String	Yes	Alarm type

Successfully Return [JSON]:

```
{"success": "true", "row": {"classify10": "733", "classify5": "0", "classify8": "0", "classify6": "0", "classify21": "0", "classify26": "0", "classify9": "0"}}
```

Remarks: Alarm type classifyYesInt type

## Set standard information(20150130)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=setInfo&macid=device ID&aTime=Annual review date&iTime=Insurance date&platenumber=vehicle plate number&engineNumber=engine number&framenumber=vehicle frame number&brand=brand ID&model=model ID&displacement=Displacement&era=Age

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
aTime	DateTime	Yes	Annual review date
iTime	DateTime	Yes	Insurance date
platenumber	String	Yes	vehicle plate number
engineNumber	String	Yes	engine number
framenumber	String	Yes	vehicle frame number
brand	Int	Yes	Brand ID
model	Int	Yes	Model ID
displacement	String	optional	Displacement
era	String	optional	Age

Successfully Return [JSON]:

```
{"success": "true|false"}
```

## Take standard information(20150130)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getInfo&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Success Return [JSON]:

```
{"success": "true", "platenumber", "annualAuditDate", "insuranceDate", "fullName", "username": "mobile phone number(user name)", "engineNumber", "framenumber": "frame number", "brandname": "brand name", "brandmodel", "displacement", "era": "Age", "brandID": "brand ID", "modelID": "model ID" }
```

## Payment callback

Interface address: url/notity\_url.aspx

Apply method: HTTP GET

E.g:

url/notity\_url.aspx?mds=(get login port)&method=getInfo&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Success Return [JSON]:

```
{"success": "true", "platenumber", "annualAuditDate", "insuranceDate", "fullName", "username", "engineNumber" }
```

## Take the brand (20150130)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getBrand

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function

Success Return:

```
{"success": "true", "errorCode": "200", "errorDescribe": "get vehicle information Successfully", "data": [{"bid", "brandname"}]}
```

## Take model (20150130)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=getModels&brand=brand ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
brand	Int	Yes	Brand ID

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "get vehicle information Successfully", "data": [{"id": "model ID", "brandmodel" }]}
```

## Modify the license plate number (20150324)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=modifyUserPlatenumber&macid=device ID &platenumber

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
platenumber	String	Yes	Plate number

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify SIM card number(20150324)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=modifyUserSim& macid=device ID &sim=sim card number

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
sim	String	Yes	Sim card number

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify contact phone number(20150324)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=modifyUserTel&macid=device ID&tel=contact person mobile phone number

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
tel	String	Yes	contact person mobile phone number

Success Return [JSON]:

```
{ "success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": [] }
```

## Modify SOS call number 1 (20150324)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=modifyUserAdminTel1&macid=device ID&adminTel1=SOS number 1

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
adminTel1	String	Yes	SOS number 1

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify SOS call number 2 (20150324)

Interface address: url/GetDataServices.asmx/GetData

Apply method: HTTP GET

E.g:

url/GetDataServices.asmx/GetData?mds=(get login port)&method=modifyUserAdminTel2&macid=device ID &adminTel2=SOS number 2

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device system unique number
adminTel2	String	Yes	SOS number 2

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify device password (20150324)

Interface address: url/GetDataServices.asmx/GetData

Apply method: HTTP GET

E.g:

url/GetDataServices.asmx/GetData?mds=&method=modifyUserPwd&macid=&pwd=&oldPwd=

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity (get login port)
method	String	Yes	Call function
macid	String	Yes	Device system unique number
pwd	String	Yes	New password
oldPwd	String	Yes	Old password

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify the annual review expiration time(20150324)

Interface address: url/GetDataServices.asmx/GetData

Apply method: HTTP GET

E.g:

url/GetDataServices.asmx/GetData?mds=(get login port)&method=modifyUser4sAnnualAuditDate&macid=device ID&AnnualAuditDate=time **【Retain the year month date, timestamp format】**

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity (get login port)
method	String	Yes	Call function
macid	String	Yes	Device system unique number
AnnualAuditDate	String	Yes	<b>【Retain the year month date, timestamp format, Number of milliseconds】</b>

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify the number of maintenance kilometers(20150325 modify)

Interface address: url/GetDataServices.asmx/GetData

Apply method: HTTP GET

E.g:

url/GetDataServices.asmx/GetData?mds=(get login port)&method=modifyUser4sSetMil& macid=device ID&nextMantanceMil=maintenance kilometers&mil=Current kilometers

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity (get login port)
method	String	Yes	Call function
macid	String	Yes	Device system unique number
nextMantanceMil	double	Yes	maintenance kilometers
mil	double	Yes	Current kilometers

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify the next maintenance time (20150324)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=modifyUser4snextMantanceDate&macid=device ID&nextMantanceDate= **【 Retain the year month date, timestamp format 】**

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity ( get login port )
method	String	Yes	Call function
macid	String	Yes	Device system unique number
nextMantanceDate	String	Yes	<b>【 Retain the year month date, timestamp format, Number of milliseconds 】</b>

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## Modify insurance expiration time(20150324)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=modifyUser4sInsuranceDate&macid=device ID&InsuranceDate= **【 Retain the year month date, timestamp format 】**

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity ( get login port )
method	String	Yes	Call function
macid	String	Yes	Device system unique number
InsuranceDate	String	Yes	<b>【 Retain the year month date, timestamp format, Number of milliseconds 】</b>

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": []}
```

## View modified 4S shop information **【 maintenance, insurance 】** (20150327)

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=GetGpsUser4SSetInfo&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity ( get login port )
method	String	Yes	Call function
macid	String	Yes	Device system unique number

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "Successfully modified", "data": [{"nextMantanceMil": "mileage before Maintenance expired", "nextMantanceDate": " Maintenance expiration time", "insurance": "time before Insurance expiration", "annual": "Annual review expiration time", "mil": "Current mileage" }]}
```

## Get 4s shop service expiration time or mileage

Interface address: url/GetDateServices.asmx/GetDate

Apply method: HTTP GET

E.g:

url/GetDateServices.asmx/GetDate?mds=(get login port)&method=GetGpsUser4SInfoV3&macid=device ID

Request parameter:

Name	Type	Must to fill in	Instruction
mds	String	Yes	Token user identity
method	String	Yes	Call function
macid	String	Yes	Device ID

Success Return [JSON]:

```
{"success": "true", "errorCode": "200", "errorDescribe": "description", "data": [{"
```

```
  "nextMantanceMil": "mileage before Maintenance expired",  
  "nextMantanceDate": " Maintenance expiration time",  
  "insurance": "time before Insurance expiration",  
  "annual": "Annual review expiration time",  
  "airConditionerTime": "Maintenance air conditioning expiration time",  
  "driveTime": "Driver's license expiration time",  
  "tireReplace": "Distance change tire mileage",  
  "oilReplace": "Distance to change oil mileage",  
  "mil": "Current mileage",
```

```
"stel": "emergency contact",
```

```
}
```

```
]
```

```
[Note: Date class default -1000; mileage class default 0]
```

```
Failed Return [JSON]:
```

```
{"success":"false","errorCode":"500","errorDescribe":"description","data":[]}
```