|  |  |
| --- | --- |
|  | **Alcatel** |
| **SmartRelease API Guide** | |
| **Domain: Alcatel**  **Rubric: Technical document** | |
| **CONTENTS:**  **This file describes the data based APIs which can be used for 3rd party to develop the PC/PHONE/TABLET based application to have the basic control to Alcatel mobile WIFI-router device.** | |

# 1. Catalog

[1. Catalog 2](#_Toc470684599)

[2. Change History 6](#_Toc470684600)

[3. Descriptions 13](#_Toc470684601)

[4. General Instructions 14](#_Toc470684602)

[4.1 HTTP Protocol 14](#_Toc470684603)

[4.1.1 HTTP Request 14](#_Toc470684604)

[4.1.2 HTTP Response 14](#_Toc470684605)

[4.2 Design principals 14](#_Toc470684606)

[4.2.1 Request Messaging Patterns 14](#_Toc470684607)

[4.2.2 Response Messaging Patterns 15](#_Toc470684608)

[4.2.3 Full Example 17](#_Toc470684609)

[5. Basic Part API 18](#_Toc470684610)

[5.1 User Management Requests 18](#_Toc470684611)

[5.1.1 Login 18](#_Toc470684612)

[5.1.2 Logout 19](#_Toc470684613)

[5.1.3 GetLoginState 20](#_Toc470684614)

[5.1.4 ChangePassword 21](#_Toc470684615)

[5.1.5 HeartBeat 22](#_Toc470684616)

[5.1.6 ForceLogin 23](#_Toc470684617)

[5.1.7 SetPasswordChangeFlag 24](#_Toc470684618)

[5.1.8 GetPasswordChangeFlag 25](#_Toc470684619)

[5.2 SIM Management Requests 26](#_Toc470684620)

[5.2.1 GetSimStatus 26](#_Toc470684621)

[5.2.2 UnlockPin 28](#_Toc470684622)

[5.2.3 UnlockPuk 29](#_Toc470684623)

[5.2.4 ChangePINCode 30](#_Toc470684624)

[5.2.5 ChangePinState 31](#_Toc470684625)

[5.2.6 GetAutoValidatePinState 32](#_Toc470684626)

[5.2.7 SetAutoValidatePinState 32](#_Toc470684627)

[5.2.8 UnlockSimlock 34](#_Toc470684628)

[5.2.9 ActiveSimlock 35](#_Toc470684629)

[5.2.10 EraseSimlock 36](#_Toc470684630)

[5.3 Connection 36](#_Toc470684631)

[5.3.1 GetConnectionState 36](#_Toc470684632)

[5.3.2 Connect 38](#_Toc470684633)

[5.3.3 DisConnect 39](#_Toc470684634)

[5.3.4 GetConnectionSettings 39](#_Toc470684635)

[5.3.5 SetConnectionSettings 41](#_Toc470684636)

[5.4 Network Management Requests 42](#_Toc470684637)

[5.4.1 GetNetworkInfo 42](#_Toc470684638)

[5.4.2 SearchNetwork 45](#_Toc470684639)

[5.4.3 SearchNetworkResult 46](#_Toc470684640)

[5.4.4 RegisterNetwork 47](#_Toc470684641)

[5.4.5 GetNetworkRegisterState 48](#_Toc470684642)

[5.4.6 GetNetworkSettings 49](#_Toc470684643)

[5.4.7 SetNetworkSettings 51](#_Toc470684644)

[5.4.8 GetNetworkModeListType————Firmware端实现，UI未调用接口 53](#_Toc470684645)

[5.4.9 GetDomRoamGuardForced————Firmware端实现，UI未调用接口 54](#_Toc470684646)

[5.5 WLAN 54](#_Toc470684647)

[5.5.1 GetWlanState 54](#_Toc470684648)

[5.5.2 SetWlanOff 55](#_Toc470684649)

[5.5.3 SetWlanOn 56](#_Toc470684650)

[5.5.4 GetWlanSettings 57](#_Toc470684651)

[5.5.5 SetWlanSettings 59](#_Toc470684652)

[5.5.6 SetWPSPin 61](#_Toc470684653)

[5.5.7 SetWPSPbc 62](#_Toc470684654)

[5.5.8 GetWlanSupportMode 63](#_Toc470684655)

[5.6 SMS 64](#_Toc470684656)

[5.6.1 getSmsInitState 64](#_Toc470684657)

[5.6.2 GetSMSContactList 65](#_Toc470684658)

[5.6.3 GetSMSContentList 67](#_Toc470684659)

[5.6.4 GetSMSStorageState 69](#_Toc470684660)

[5.6.5 DeleteSMS 70](#_Toc470684661)

[5.6.6 SendSMS 71](#_Toc470684662)

[5.6.7 GetSendSMSResult 72](#_Toc470684663)

[5.6.8 SaveSMS 73](#_Toc470684664)

[5.6.9 GetSMSSettings 74](#_Toc470684665)

[5.6.10 SetSMSSettings 75](#_Toc470684666)

[5.6.11 GetSingleSMS 76](#_Toc470684667)

[5.7 Statistics 77](#_Toc470684668)

[5.7.1 GetUsageRecord 77](#_Toc470684669)

[5.7.2 SetUsageRecordClear 78](#_Toc470684670)

[5.7.3 GetUsageSettings 79](#_Toc470684671)

[5.7.4 SetUsageSettings 81](#_Toc470684672)

[5.7.5 GetFDTimer 82](#_Toc470684673)

[5.7.6 SetFDTimer 83](#_Toc470684674)

[5.8 USSD 84](#_Toc470684675)

[5.8.6 SendUSSD 84](#_Toc470684676)

[5.8.7 GetUSSDSendResult 85](#_Toc470684677)

[5.8.8 SetUSSDEnd 86](#_Toc470684678)

[5.9 Update 87](#_Toc470684679)

[5.9.6 GetDeviceNewVersion 87](#_Toc470684680)

[5.9.7 SetFOTAStartDownload 88](#_Toc470684681)

[5.9.8 GetDeviceUpgradeState 89](#_Toc470684682)

[5.9.9 SetDeviceUpdateStop 90](#_Toc470684683)

[5.9.10 SetCheckNewVersion 90](#_Toc470684684)

[5.9.11 SetDeviceStartUpdate 91](#_Toc470684685)

[5.10 Firewall 92](#_Toc470684686)

[5.10.6 GetMacFilterSettings 92](#_Toc470684687)

[5.10.7 SetMacFilterSettings 93](#_Toc470684688)

[5.10.8 getPortFwding 94](#_Toc470684689)

[5.10.9 addPortFwding 95](#_Toc470684690)

[5.10.10 deletePortFwding 97](#_Toc470684691)

[5.10.11 editPortFwding 98](#_Toc470684692)

[5.10.12 SetPortFwding————暂不支持 99](#_Toc470684693)

[5.10.13 getDMZInfo 100](#_Toc470684694)

[5.10.14 setDMZInfo 101](#_Toc470684695)

[5.10.15 getIPFilterList 102](#_Toc470684696)

[5.10.16 addIPFilter 104](#_Toc470684697)

[5.10.17 deleteIPFilter 105](#_Toc470684698)

[5.10.18 editIPFilter 106](#_Toc470684699)

[5.10.19 getFirewallSwitch 107](#_Toc470684700)

[5.10.20 setFirewallSwitch 108](#_Toc470684701)

[5.10.21 SetIPFilter————暂不支持 109](#_Toc470684702)

[5.10.22 GetVPNPassthrough————暂不支持 110](#_Toc470684703)

[5.10.23 SetVPNPassthrough————暂不支持 111](#_Toc470684704)

[5.10.24 getUrlFilterSettings 112](#_Toc470684705)

[5.10.25 SetUrlFilterSettings 113](#_Toc470684706)

[5.10.26 GetWanAccess 114](#_Toc470684707)

[5.10.27 SetWanAccess 115](#_Toc470684708)

[5.11 LAN 116](#_Toc470684709)

[5.11.6 GetLanSettings 116](#_Toc470684710)

[5.11.7 SetLanSettings 118](#_Toc470684711)

[5.12 ConnectionDevices 119](#_Toc470684712)

[5.12.6 GetConnectedDeviceList 119](#_Toc470684713)

[5.12.7 GetBlockDeviceList 120](#_Toc470684714)

[5.12.8 SetConnectedDeviceBlock 122](#_Toc470684715)

[5.12.9 SetDeviceUnblock 123](#_Toc470684716)

[5.12.10 SetDeviceName 124](#_Toc470684717)

[5.13 System Information Requests 125](#_Toc470684718)

[5.13.6 GetSystemInfo 125](#_Toc470684719)

[6.7.7 SetLanguage 126](#_Toc470684720)

[7.7.7 GetCurrentLanguage 127](#_Toc470684721)

[8.7.7 GetSystemStatus 128](#_Toc470684722)

[9.7.7 SetDeviceReboot 131](#_Toc470684723)

[9.7.8 SetDeviceReset 132](#_Toc470684724)

[9.7.9 SetDeviceBackup 132](#_Toc470684725)

[5.13.8 SetDeviceRestore————已取消使用 133](#_Toc470684726)

[5.13.9 SetDevicePowerOff 134](#_Toc470684727)

[5.13.10 SetAppBackup——————非UI接口 135](#_Toc470684728)

[5.13.11 SetAppRestoreBackup——————非UI接口 135](#_Toc470684729)

[5.13.12 GetWizardStatus——————非UI接口 136](#_Toc470684730)

[5.13.13 SetWizardStatus——————非UI接口 137](#_Toc470684731)

[5.13.14 GetRedirectSwitch——————非UI接口 138](#_Toc470684732)

[5.13.15 SetRedirectSwitch——————非UI接口 139](#_Toc470684733)

[5.13.16 GetUpnpSettings 140](#_Toc470684734)

[5.13.17 SetUpnpSettings 141](#_Toc470684735)

[5.13.18 GetDebugInfo——————未实现接口 142](#_Toc470684736)

[5.13.19 uploadBackupSettings 144](#_Toc470684737)

[5.13.20 GetHfaActivationSatus——————未实现接口 145](#_Toc470684738)

[5.13.21 CancelHfaActivation——————未实现接口 146](#_Toc470684739)

[5.13.22 getNewPageFlagSemaphore 147](#_Toc470684740)

[5.13.23 setNewPageFlagSemaphore 148](#_Toc470684741)

[5.13.24 SendPingURL——————Firmware端未实现接口 149](#_Toc470684742)

[5.13.25 SetMyNumber——————Firmware端未实现接口 149](#_Toc470684743)

[5.14 Sharing 150](#_Toc470684744)

[5.14.1 GetDLNASettings——————Firmware端未实现接口 150](#_Toc470684745)

[5.14.2 SetDLNASettings——————Firmware端未实现接口 151](#_Toc470684746)

[5.14.3 GetSambaSettings 152](#_Toc470684747)

[5.14.4 SetSambaSettings 153](#_Toc470684748)

[5.14.5 GetFtpSettings 154](#_Toc470684749)

[5.14.6 SetFtpSettings 155](#_Toc470684750)

[5.14.7 GetSDCardSpace 155](#_Toc470684751)

[5.14.8 GetSDFileList 156](#_Toc470684752)

[5.14.9 GetSDcardStatus 158](#_Toc470684753)

[5.14.10 GetUsbcardSettings 159](#_Toc470684754)

[5.14.11 SetUsbcardSettings 160](#_Toc470684755)

[5.15 Profile 160](#_Toc470684756)

[5.15.1 GetProfileList 160](#_Toc470684757)

[5.15.2 AddNewProfile 162](#_Toc470684758)

[5.15.3 EditProfile 164](#_Toc470684759)

[5.15.4 DeleteProfile 165](#_Toc470684760)

[5.15.5 SetDefaultProfile 166](#_Toc470684761)

[5.15.6 getCurrentProfile 167](#_Toc470684762)

[5.16 Power Management 169](#_Toc470684763)

[5.16.1 GetBatteryState 169](#_Toc470684764)

[5.16.2 GetPowerSavingMode 170](#_Toc470684765)

[5.16.3 SetPowerSavingMode 171](#_Toc470684766)

[5.17 WIFI Extender——————未实现接口 172](#_Toc470684767)

[5.17.3 GetWIFIExtenderSettings 172](#_Toc470684768)

[5.17.4 SetWIFIExtenderSettings 173](#_Toc470684769)

[5.17.5 GetWIFIExtenderCurrentStatus 174](#_Toc470684770)

[5.17.6 GetHotspotList 175](#_Toc470684771)

[5.17.7 EditHotspotInfo 176](#_Toc470684772)

[5.17.8 DeleteHotspot 178](#_Toc470684773)

[5.17.9 AddHotspot 179](#_Toc470684774)

[5.17.10 SearchHotspot 180](#_Toc470684775)

[5.17.11 GetSearchHotspotList 180](#_Toc470684776)

[5.17.12 ConnectHotspot 182](#_Toc470684777)

[5.17.13 DisconnectHotspot 183](#_Toc470684778)

[5.18 OMA——————未实现接口 184](#_Toc470684779)

[5.18.1 SetPRLUpdate 184](#_Toc470684780)

[5.18.2 GetPRLUpdateProcess 184](#_Toc470684781)

[5.18.3 SetPRLUpdateCancel 185](#_Toc470684782)

[5.18.4 SetProfileUpdate 186](#_Toc470684783)

[5.18.5 GetProfileUpdateProcess 187](#_Toc470684784)

[5.18.6 SetProfileUpdateCancel 188](#_Toc470684785)

[5.18.7 SetFactoryReset 189](#_Toc470684786)

[5.18.8 SetDeviceRenew 189](#_Toc470684787)

[5.18.9 GetServerIndicator 190](#_Toc470684788)

[5.18.10 SetServerIndicator 191](#_Toc470684789)

[5.18.11 SetOMAOEMServer 192](#_Toc470684790)

[5.18.12 GetOMALinkInfo 194](#_Toc470684791)

[5.18.13 GetOMABrandAlpha 195](#_Toc470684792)

[5.18.14 GetSimLockStatus 196](#_Toc470684793)

[5.18.15 GetOMACarrierID 197](#_Toc470684794)

[5.19 TR069 198](#_Toc470684795)

[5.19.1 GetClientConfiguration——————Firmware端未实现接口 198](#_Toc470684796)

[5.19.2 SetClientConfiguration——————Firmware端未实现接口 199](#_Toc470684797)

[5.20 FeatureGuide 201](#_Toc470684798)

[5.20.1 Get feature list 201](#_Toc470684799)

[17 6. Appendix 205](#_Toc470684800)

[6.1 Error Definition 205](#_Toc470684801)

[6.1.1 关于各模块Error Code的定义 205](#_Toc470684802)

[6.1.2 JSON RPC保留的Error code 205](#_Toc470684803)

[6.1.3 Common Error Tables 205](#_Toc470684804)

# 2. Change History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Person** | **Descriptions** |
| **2014-08-04** | V1.0 | Hou Ailing | Create document |
| **2014-08-06** | V1.1 | Kuang Li | Modify6.1.1 关于各模块Error Cod的定义  Remove 6.1.2 关于通用Error Code的定义  Modify 4.2.2.1 Error object  将API的Error Code不满6位的前面补0 |
| **2014-08-22** | V1.2 | Hou Ailing | Completed 5.1-----5.15 |
| **2014-09-11** | V1.3 | Kuang Li | 给每个error code加上双引号.  将SMS模块里的参数“Id”改成"ContactId".  Remove 5.6.3.1GetSMSContentList request里的参数"PhoneNumber".  Change 5.6.3 GetSMSContentList里参数“SMSType”的枚举值以便跟以前的项目保持一致. |
| **2014-09-22** | V1.4 | Kuang Li | 5.6.2 GetSMSContactList, add parameter"TotalPageCount"  5.6.3 GetSMSConentList, add parameter"TotalPageCount" |
| **2014-09-23** | V1.5 | Hu Qiao | 5.11.2 SetLanSettings, change data type of parameter “IPv4IPAddress”, “SubnetMask” and “DHCPServerStatus”, remove “” of “DHCPLeaseTime” value. |
| **2014-09-25** | V1.5 | Xueqin.jin | 5.5.5 修改为"SetWlanSettings",  5.6.11 添加接口GetSingleSMS |
| **2014-09-25** | V1.5 | Fengtian.wang | 5.3.4将GetConnectionType合并到GetConnectionSettings中  5.3.5将SetConnectionType合并到 SetConnectionSettings |
| **2014-09-26** | V1.6 | Ailing.hou | Modify connection device:  5.12.1 GetBlockDeviceList  5.12.2 SetConnectedDeviceBlock  5.12.3 SetDeviceUnlock  5.12.4 SetDeviceName  Modify 5.14.3 GetSDCardSettings, add parameter  Add 5.14.5 GetSDFileList |
| **2014-10-01** | V1.6 | Jinbo.huang | 5.14.3 modify params TotalSpace/UsedSpace type(string) and unit(GB)  5.14.5 add params SIZE unit(KB) |
| **2014-10-08** | V1.7 | Changqin.li | 1. modify 5.15.1 GetprofileList 0:predefine profile   1:User Createprofile  2. modify 5.15.2 AddNewProfile "AuthType":0  3. Add 5.2.9 ActiveSimlock  4. Add 5.2.10 EarseSimlock  5. modify 5.9.3 GetDeviceUpgradeState Status: 2 is complete |
| **2014-10-17** | V1.7 | Kuang Li | 1. Modify 5.6.2 GetSMSContactList "SMSType" Description |
| **2014-10-17** | V1.7 | Long Zengyong | 1、Modify5.4.1 GetNetworkInfo 中 5.4.1.2 Response参数  2、Modify 5.4.6 GetNetworkSetting中5.4.6.2 Response参数 |
| **2014-10-21** | V1.7 | Huqiao | 1. Add5.14.1 GetDLANSettings中参数”DlanName” 2. Add5.14.2 SetDLANSettings中参数”DlanName” |
| **2014-10-22** | V1.7 | Kuang Li | 1. Add 5.16 Power Management |
| **2014-10-22** | V1.8 | Jinbo.huang | 1. change DLNA api (5.14) 2. add FTP/SAMBA api (5.14) 3. change sdshare api (5.14) |
| **2014-10-22** | V1.9 | Fengzhou.zhao | 1. Modify 5.13.4.   Change “barrery” of GetSystemStatus to “chg\_state” and “bat\_leve”. |
| **2014-10-22** | V1.9 | Kuang Li | 1. Modify 5.5.4, add 5G paramers 2. Modify 5.5.5, add 5G paramers 3. Add 5.5.8, GetWlanSupportMode |
| **2014-10-28** | V2.0 | Huqiao | 1. Modify 5.5.6, modify wps pin error code 2. Modify 5.5.7, modify wps pbc error code 3. Modify 5.5.6 paramers WpsPin |
| **2014-10-28** | V2.1 | Changqin li | 1. Modify 5.2.8 paramers SIMLockState |
| **2014-10-28** | V2.2 | Kuang Li | 1. Modify 5.17.1 Get feature list |
| **2014-10-31** | V2.2 | Kuang Li | 1. Modify 5.5.1 GetWlanState param WlanState 2. Modify 5.4.1 and 5.13.4 param SignalStrength |
| **2014-10-31** | V2.3 | Jinbo.huang | 1. Add 5.13.9 sdcard status api 2. Add 5.14.9 set device power off api |
| **2014-11-3** | V2.4 | Kuang Li | 1. 5.17.1 Get feature list, change “Power Management” to “PowerManagement” 2. 5.16.1 GetBatteryState, change “ChargeState” to “chg\_state” |
| **2014-11-5** | V2.5 | Zengyong.long | 1. 5.14.3 GetSystemStatus, change ”Roaming”: 0:Roaming, 1:no |
| **2014-11-5** | V2.6 | Changqin li | 1. Add 5.9.5 SetCheckNewVersion |
| **2014-11-6** | V2.7 | Huqiao | 1. Modify MacFilter Api |
| **2014-11-6** | V2.8 | hai.huang | 1. Modify for Statistics. |
| **2014-11-7** | V2.9 | Kuang li | 1. Modify 5.5.4 and 5.5.5 Channel value |
| **2014-11-7** | V2.10 | Kuang li | 1. Add "NetworkName" to 5.13.4 GetSystemStatus |
| **2014-11-10** | V2.11 | Changqin li | 1. Add 5.13.10 SetAppBackup 2. Add 5.13.10 SetAppRestoreBackup |
| **2014-11-10** | V2.12 | Kuang li | 1. Modify 5.6.6 and 5.6.8 "SMSTime" format 2. Modify 5.7.2 "clear\_time" type |
| **2014-11-11** | V2.13 | Fengzhou.zhao | 1. Add “bat\_cap” to 5.13.4 GetSystemStatus |
| **2014-11-11** | V2.14 | Huqiao | 1. Add” TotalConnNum” to 5.13.4 GetSystemStatus |
| **2014-11-12** | V2.15 | hai.huang | 1. Modify the word of “GetUsageRecord” in “Statistics” |
| **2014-11-13** | V2.16 | Zengyong.long | 1. Modify Network Type |
| **2014-11-14** | V2.17 | Changqin li | add 5.13.11 response error code RestoreErr |
| **2014-11-14** | V2.18 | Zengyong.long | 1. Add “NO SERVER” to Network Type |
| **2014-11-20** | V2.19 | Jinbo.huang | 1. Add 5.14.10“GetUsbcardStatus” 2. Add 5.14.11“SetUsbcardStatus” |
| **2014-11-26** | V2.20 | Yanping.an | 1. Add “DeviceName” to GetFeatureList |
| **2014-11-26** | V2.21 | Kuang Li | 1. Add new error “060802 Fail with store space full” for 5.6.8 SaveSMS |
| **2014-11-27** | V2.22 | Huqiao | 1. Modify 5.10.1 and 5.10.2 Macfilter Parameters |
| **2014-11-27** | V2.23 | Kuang Li | 1. Modify 5.16.2 and 5.16.3, change “WiFiMode” as optional parameter. |
| **2014-12-04** | V2.24 | Kuang Li | 1. Modify 5.17.1 GetFeatureList, Set “Sharing” group as Optional. Not required if not support SD card. |
| **2014-12-16** | V2.25 | Kuang Li | 1. Modify 5.6.4 GetSMSStorageState, add LeftCount and UnreadReport |
| **2015-01-07** | V2.26 | Ailing.hou | Add WIFI extender function |
| **2015-01-10** | V2.26 | Ailing.hou | Modify featureGuide. |
| **2015-01-28** | V2.27 | Ailing.hou | Add 5.13.12GetWizardStatus,5.13.13 SetWizardStatus  Add 5.10.3 GetPortFwding, 5.10.4 AddPortFwding, 5.10.5 DeletePortFwding, 5.10.6 EditPortFwding |
| **2015-01-28** | V2.27 | Ailing.hou | Modify featureGuide. |
| **2015-04-03** | V2.28 | Xiaoqing.liu | Modify explain 5.7.1 |
| **2015-05-19** | V2.29 | Xiaoqing.liu | Add param “current\_time” for getUseHistory |
| **2015-06-05** | V2.30 | Ailing.hou | Add 5.10.7 GetDMZSettings; 5.10.8 SetDMZSettings |
| **2015-06-19** | V2.31 | Fangsi.Pi | Add 5.1.6 “ForceLogin” |
| **2015-06-19** | V2.32 | Yunyang.Zhang | Add 5.7.5 “GetDevicePushInfo” , 5.7.6 “SetDevicePushInfo” , 5.7.7 “DeleteDevicePushInfo” . |
| **2015-06-29** | V2.33 | Yunyang.Zhang | Modify 5.7.5, 5.7.6 “LimMsg” to “MonMsg” |
| **2015-06-30** | V2.34 | Yunyang.Zhang | Add 5.7.5, 5.7.6 “LimMsg” |
| **2015-07-01** | V2.35 | Yunyang.Zhang | Add “AppKey” , “AppMasterSecret” for 5.7.5 “GetDevicePushInfo” , 5.7.6 “SetDevicePushInfo” |
| **2015-07-06** | V2.36 | Yunyang.Zhang | Add error code “-32604” “Access is denied” |
| **2015-07-28** | V2.36 | Jinhui.lu | GetNetworkInfo Add LTE+ (4G+) |
| **2015-07-29** | V2.37 | Xiaoqing.liu | This function first add for W800 Yota:  1).5.10.3 GetPortFwding add new parameter: fwding\_status;  2). add 5.10.7 SetPortFwding  3) 5.4.1 GetNetworkInfo add new parameters: mcc,mnc,SINR,RSRP,RSRQ,RSSI,eNBID,CGI,CenterFreq,TxPWR;  4) 5.3.1 GetConnectionState add new parameter:I sIdle,SessionId,MaxDIRate, MaxUIRate;  5) GetSystemInfo add new parameter:UICCID,HNBN,CSGT,IMEISV,RFVersion,AsicVersion;  6) Add 5.3.6 GetRoamConnPromptSettings, 5.3.7 SetRoamConnPromptSettings;  7) Add 5.3.8 GetRoamingPopShowFlag, 5.3.9 SetRoamingPopShowFlag; |
| **2015-07-30** | V2.38 | Jinbo.huang | Change fota update to download api |
| **2015-08-01** | V2.39 | 1)--- 7) Ailing.hou  8)—11) Xueqin.jin  12)—14) Jinhui.lu | Modify this modification for Y856 Sprint:   1. 5.4.6GetNetworkSettings,5.4.7SetNetworkSettings addparameters: “CellularData”, ”DomesticRoam”, ”DomesticRoamCarrier”, ”DomesticRoamGuard”; 2. 5.10.1 GetMacFilterSettings,5.10.2 SetMacFilterSettings change format and add parameters for Y856 Sprint; 3. 5.7.3 GetUsageSettings, 5.7.4 SetUsageSettings add new parameters: UsadDataWarn; 4. 5.15.1 GetProfileList, 5.15.2 Add NewProfile, 5.15.3 EditProfile add new parameter:PhpType;   Add 5.15.6 GetDataProfile;  Add 5.15.7 EditDataProfile;  Add 5.15.8 SetMSL;   1. 5.9.1 GetDeviceNewVersion add parameter: Date;   Add 5.9.6 SetUpdateMode;  Add 5.9.7 GetManualUpdateProcess  Add 5.9.8 GetLastUpdateInfo  Add 5.9.9 GetUpdateHistory   1. Add 5.18 OMA 2. 5.13.1 GetSystemInfo add parameters: “msisdn”,“IMSI”, “Model”,”FMVersion”,”FWBuildDate”,”AppVersion”,“BootloaderVersion”, “Manager”, “Manufacturer”, “AMSSVersion”, ”Chipset”;   Add 5.13.14 GetRedirectSwitch;  Add 5.13.15 SetRedirectSwitch;  Add 5.13.16 GetUpnpSettings;  Add 5.13.17 GetUpnpSettings;  Add 5.13.18 GetDebugInfo   1. Add 5.3.10 GetConnectHistoryList   Add 5.3.11 SetConnectionHistoryClear   1. 5.6.11 GetSingleSMS add parameters: ”SMSId”, ”PhoneNumber”, ”SMSType”, ”SMSTime”;   Add 5.6.12 GetSMSList;   1. Add 5.10.10 GetIPFilterList   Add 5.10.11 AddIPFilter  Add 5.10.12 DeleteIPFilter  Add 5.10.13 EditIPFilter  Add 5.10.14 GetFirewallSwitch  Add 5.10.15 SetFirewallSwitch   1. 5.11.1 GetLanSettings, 5.11.2 SetLanSettings add parameters: “VPNPassthrough”,”DNSMode”,”DNSAddress1”,”DNSAddress2” 2. 5.1.3 GetLoginState add parameters: “LoginRemainingTimes”, “LockedRemainingTime” 3. 5.5.4 GetWlanSettings, 5.5.5 SetWlanSettings add parameters; 4. 5.12.1 GetConnectedDeviceList add parameter: “ConectedGuestList”;   5.12.5 SetDeviceName add parameter: “ConnectedType”; |
| **2015-08-04** | V2.40 | Xueqin.jin | for Y856 Sprint:  5.10.14 GetFirewallSwitch Add parameters port\_forward\_status 5.10.15 SetFirewallSwitch Add parameters port\_forward\_status |
| **2015-08-06** | V2.41 | Xueqin.jin | for Y856 Sprint:  Add 5.10.16 SetIPFilter |
| **2015-08-06** | V2.42 | Jinhui.lu | GetSystemStatus add parameters “LoginState” |
| **2015-08-11** | V2.43 | Ailing.hou | For Y856 Sprint:  Update 5.18 OMA order；add new 5.18.3 SetPRLUpdateCancel; |
| **2015-08-14** | V2.44 | Ailing.hou | 5.18.5 Get ProfileUpdateProcess---profile\_update\_status add new value:3 |
| **2015-08-17** | V2.45 | Xueqin.jin | For Y856 Sprint:  5.10.3 GetPortFwding add parameter portfwd\_name  5.10.7 SetPortFwding add parameter portfwd\_name  5.4.1 GetNetworkInfo and 5.13.4 GetSystemStatus  Add WCDMA、CDMA、GSM to NetworkType  5.5.4 GetWlanSettings and 5.5.5 SetWlanSettings add parameter  PrivacySeparator,Guest\_PrivacySeparator |
| **2015-08-18** | V2.46 | Ailing.hou | For Y856 Spring:  Add 5.18.9 GetServerIndicator;  5.18.10 SetServerIndicator;  5.18.11 SetOMAOEMServer |
| **2015-08-19** | V2.47 | 1)maotang.yang  2)Xueqin.jin | 1. 5.7.3 GetUsageSettings, 5.7.4 SetUsageSettings add new parameters: UnitWarn;   BillingDay初始返回一个默认值0。   1. 5.7.1 GetUsageRecord add parameters: NextCycleDate, RemainingDays |
| **2015-08-24** | V2.48 | Xueqin.jin | 1)add API 5.9.10 DownloadManualUpdatefile  2)modify 5.9.7 params IsReadyForUpdate description, 1:updating  3) add API 5.13.19 uploadBackupSettings |
| **2015-08-27** | V2.49 | Qiuping.xu | Update  5.4.1 GetNetworkInfo Network Type  5.4.6 GetNetworkSetting NetworkMode and NetworkBand  5.4.7 SetNetworkSetting NetworkMode and NetworkBand |
| **2015-08-28** | V2.50 | Ailing.hou | Update for Y856 Sprint  5.4.1 GetNetworkInfo add parameter: lte\_state, PLMN\_name, Band, DL\_channel, UL\_channel, RSRQ,  5.18.8 SetMSL 修改为CheckMSL |
| **2015-08-31** | V2.51 | Ailing.hou | Update for Y856 Sprint  Add 5.3.12 GetMobileIPErrorState; 5.3.13 SetMobileIPErrorState;  Add 5.7.8 GetUsageSetFlag;  Add 5.18.12 GetOMALinkInfo;  Modify 5.5.4 GetWlanSettings, 5.5.5 SetWlanSettings, add parameter:  TcpNatTimer,UdpNatTimer,AnyConnectingRejection,Guest\_AnyConnectingRejection; Delete NatTimer; |
| **2015-08-31** | V2.52 | Qiuping.xu | Add for Y856UB Sprint  5.4.8 GetNetworkModeListType |
| **2015-09-06** | V2.53 | Qiuping.xu | Modify 5.4.6 GetNetworkSettings,5.4.7 SetNetworkSettings: Delete DomesticRoamCarrier; Modify 5.4.8 GetNetworkModeListType response para name |
| **2015-09-07** | V2.54 | Jinhui.lu | GetDebugInfo add param ‘LteLastError’ |
| **2015-09-08** | V2.55 | Xingmei.Luo | Add for Y856UB Sprint  5.1.1 Login add error code 010104-- Guest AP has no ui access right. |
| **2015-09-09** | V2.56 | Qiuping.xu | Update for Y856UB Sprint  Delete CellularData para on 5.4.6 and 5.4.7  Add 5.4.9 GetDomRoamGuardForced |
| **2015-09-10** | V2.57 | Jinhui.lu | Add API :GetHfaActivationStatus |
| **2015-09-11** | V2.58 | Jinhui.lu | Uptate API ‘5.13.20 GetHfaActivationStatus’;  Add API ‘5.13.21CancelHfaActivation’ |
| **2015-09-21** | V2.59 | Jinhui.lu | Uptate API ‘5.13.20 GetHfaActivationStatus’,param ‘HfaActivationStatus’add value (6、7、8、9、10) |
| **2015-10-08** | V2.60 | Xueqin.jin | Modify API 5.3.8 GetRoamingPopShowFlag 、5.3.9 SetRoamingPopShowFlag param show\_flag description ，5.3.12 GetMobileIPErrorState、5.3.13 SetMobileIPErrorState param webUI\_show\_flag description ：  0:Web UI need show message（需要显示）  1: Web UI needn't show message（不需要显示） |
| **2015-10-12** | V2.61 | Jinhui.lu | Add API 5.18.13 GetOMABrandAlpha |
| **2015-10-13** | V2.62 | Jinhui.lu | Update API 5.18.13 GetOMABrandAlpha |
| **2015-10-19** | V2.63 | Maotang.yang | Add API 5.1.7 setPasswordChangeFlag 5.1.8 getPasswordChangeFlag |
| **2015-10-22** | V2.64 | Jinhui.lu | Update API 5.13.20 GetHfaActivationStatus param HfaActivationStatus add value:11:canceling;12: completed  Update API 5.18.2 GetPRLUpdateProcess and API 5.18.5 GetProfileUpdateProcess |
| **2015-10-23** | V2.65 | Qingli.wang | Add API 5.18.14 GetSimLockStatus |
| **2015-10-26** | V2.66 | Qiuping.xu | Add 5.4.10 GetLteForceConf For Y856UB Sprint |
| **2015-10-28** | V2.67 | Maotang.yang | Add 5.3.1 GetConnectionState param ClearCode(val 29 or 33) |
| **2015-11-6** | V2.68 | Xueqin.jin | Modify for Y856UB Sprint according to the real firmware return data  1.Move LteLastError from API 5.13.18 GetDebugInfo to API 5.4.1 GetNetworkInfo  2.Modify PdpIPType Mapping value To:1:IPV4; 2:IPV63:IPV4V6 in API 5.15.1 GetProfileList; 5.15.2 AddNew Profile ;5.15.3 EditProfile |
| **2015-12-10** | V2.69 | Qiuping.xu | Add DomesticRoamPref para on 5.4.6 and 5.4.7 For Y856UB Sprint |
| **2015-12-24** | V2.70 | Xueqin.jin | Add params "private\_port\_end"&"global\_port\_end"for API : 5.10.3 GetPortFwding  5.10.4 AddPortFwding  5.10.5 DeletePortFwding  5.10.6 EditPortFwding |
| **2016-01-14** | V2.71 | Xueqin.jin | Add API 5.10.19 GetUrlFilterSettings  5.10.20 SetUrlFilterSettings |
| **2016-4-12** | V2.72 | Xueqin.jin | Add API 5.13.22 getNewPageFlagSemaphore  5.13.23 setNewPageFlagSemaphore |
| **2016-11-24** | V2.73 | Xueqin.jin | 修改API name 为getSmsInitState |
| **2016-11-29** | V2.74 | Xueqin.jin | 1.修改API getCurrentProfile标题显示错误问题1.  2. 添加wan侧访问开关接口：  5.10.21 GetWanAccess  5.10.22 SetWanAccess  3.添加update API 5.9.6 SetDeviceStartUpdate  添加接口SetFOTAStartDownload描述 |
| **2016-11-30** | V2.75 | Xueqin.jin | 根据实际接口修改SetPasswordChangeFlag和GetPasswordChangeFlag返回参数及相关描述 |
| **2016-11-30** | V2.76 | Huan.you | 在GetWlanSettings及SetWlanSettings中新增参数WiFiOffTime. |
| **2016-12-28** | V2.77 | Xubo.jia | Add API 5.7.5 GetFDTimer  5.7.6 SetFDTimer |
| **2017-01-11** | V2.78 | Xubo.jia | 修改 API 5.4.6 GetNetworkSettings参数NetworkBand取值  增加API 5.4.7 SetNetworkSettings参数NetworkBand |
| **2017-04-25** | V2.79 | Jimin.zhong | 增加API 5.10.28 GetBindToPcSettings  增加API 5.10.29 SetBindToPcSettings |
| **2017-07-14** | V2.80 | Huan.you | GetSimStatus接口增加参数PLMN |

# 3. Descriptions

We provided one set of APIs which is JSON based (Json RPC2.0, HTTP post to dedicated URL). This method can be used for WebUI and 3rd party application which running on Windows/Mac/Android/IOS to manage Y858 device.

# 4. General Instructions

## HTTP Protocol

We use HTTP POST to dedicated URL (/jrd/webapi) to send protocol data. Please see the details of HTTP headers definition below:

### HTTP Request

Please see the http request header format below:

POST /jrd/webapi HTTP/1.1

Content-Type: application/json

Accept: application/json

Content-Length: {nnnn}

### HTTP Response

Please see the http response header format below:

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: {nnnn}

## Design principals

The communication between a client and device through this API is based on JSON RPC 2.0.

As major goal for the design should be the optimal performance and reliability.

For this purpose we have designed a schema, which includes type’s definitions for all objects used by API.

To be able to validate both request and response against the schema demands using as request method exclusively POST whether we want to get data from the device or do something else.

For better readability by human, the most values are defined as strings. Also enumerations have strings values. But through schema the range of possible string values are restricted.

### Request Messaging Patterns

A rpc call is represented by sending a Request object to a Server. The Request object has the following members:

1. jsonrpc

A String specifying the version of the JSON-RPC protocol. MUST be exactly "**2.0**".

1. method

A String containing the name of the method to be invoked.

1. params

A Structured value that holds the parameter values to be used during the invocation of the method.

1. id

An identifier established by the Client that MUST contain a String or NULL value if included. The Server MUST reply with the same value in the Response object if included. This member is used to correlate the context between the two objects.id format should be like : ModuleID.MethodID

#### Request without parameters

{

"jsonrpc": "2.0",

"method": "System.SetConnectionMode",

"params": {},

"id": "1.1"

}

#### Request with parameters

{

"jsonrpc": "2.0",

"method": "SetConnectionMode",

"params": {

"ConnectMode":1

},

"id": "1.1"

}

### Response Messaging Patterns

When a rpc call is made, the Server MUST reply with a Response, except for in the case of Notifications. The Response is expressed as a single JSON Object, with the following members:

1. jsonrpc

A String specifying the version of the JSON-RPC protocol. MUST be exactly "**2.0**".

1. result

This member is REQUIRED on success.  
This member MUST NOT exist if there was an error invoking the method.  
The value of this member is determined by the method invoked on the Server.

1. error

This member is REQUIRED on error.  
This member MUST NOT exist if there was no error triggered during invocation.  
The value for this member MUST be an Object as defined in section 4.2.2.1.

1. id

This member is REQUIRED.  
It MUST be the same as the value of the id member in the Request Object.  
If there was an error in detecting the id in the Request object (e.g. Parse error/Invalid Request), it MUST be Null.

Either the result member or error member MUST be included, but both members MUST NOT be included.

#### Error object

When a rpc call encounters an error, the Response Object MUST contain the error member with a value that is a Object with the following members:

1. code

A Number that indicates the error type that occurred.  
The value of error code MUST be as defined in section 6.1.

1. message

A String providing a short description of the error.  
The message SHOULD be limited to a concise single sentence.

#### Response without parameters

{

"jsonrpc": "2.0",

"result": {},

"id":"1.1"

}

#### Response with parameters

{

"jsonrpc": "2.0",

"result": {

"Manufacturer":"JRD",

"Model":"K1234",

"Imei":"35010000000000",

"Version": {

"HwVersion":"12345",

"SwVersion":"23456"

}

},

"id":"1.1"

}

#### Response with errors

{

"jsonrpc": "2.0",

"error": {

"code": -32601,

"message": "Method not found"

},

"id": "1.1"

}

### Full Example

#### Request:

POST /jrd /webapi HTTP/1.1

Content-Type: application/json

Accept: application/json

Content-Length: nnn

{

"jsonrpc": "2.0",

"method": "SetConnectionMode",

"params": {

"ConnectMode": "Auto"

},

"id": "1.1"

}

#### Response:

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: {nnnn}

{

"jsonrpc": "2.0",

"result": {

"Manufacturer":"JRD",

"Model":"K1234",

"Imei":"35010000000000",

"Version": {

"HwVersion":"12345",

"SwVersion":"23456"

}

},

"id":"1.1"

}

# 5. Basic Part API

## User Management Requests

### Login

#### POST Request

{

"jsonrpc": "2.0",

"method": "Login",

"params": {

"UserName": "admin",

"Password": "admin"

},

"id": "1.1"

}

#### Definition for result filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| UserName | String | Yes | The login user name. The default value is admin. |
| Password | String | Yes | The login user name. The default value is admin. |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"1.1"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "010101",

"message": "Username or Password is not correct."

},

"id": "1.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010101 | Username or Password is not correct. | Username or Password is not correct. |
| 010102 | Other user is login. | Other user is login. |
| 010103 | Login times is used out. | If the user login error 5 times, this user can login until the device reboot. |
| 010104 | Guest AP has no ui access right. | If disable guest ap ui access, guest ap can’t access webui. |

### Logout

#### POST Request

{

"jsonrpc": "2.0",

"method": "Logout",

"params": {},

"id": "1.2"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"1.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"010201",

"message": "Logoutfailed."

},

"id":"1.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010201 | Logout failed. | Logout failed. |

### GetLoginState

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetLoginState",

"params": {},

"id": "1.3"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"LockedRemainingTime":0,

"LoginRemainingTimes":5,

"State":0

},

"id":"1.3"

}

#### Definition for result filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| LockedRemainingTime | Integer | No | Remaining time, unit(second) |
| LoginRemainingTimes | Integer | No | Login Remaining Times |
| State | Integer | No | Login state  0: logout  1: login  2: the login times used out. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "010301",

"message": "Get login state failed."

},

"id": "1.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010301 | Get login state failed. | Get login state failed. |

### ChangePassword

#### POST Request

{

"jsonrpc": "2.0",

"method": "ChangePassword",

"params": {

"UserName":"admin",

"CurrPassword ":"",

"NewPassword":""

},

"id": "1.4"

}

#### Definition for result filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| UserName | String | NO | Current admin |
| CurrPassword | String | Yes | Current password |
| NewPassword | String | Yes | The newpassword |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"1.4"

}

#### Response with error

{

"jsonrpc": "2.0",

"error ":

{

"code":"010401",

"message": "Change passwordfailed."

},

"id":"1.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010401 | Change password failed. | Change password failed. |
| 010402 | The current password is wrong. | The current password is wrong. |

### HeartBeat

The web server check if the client is living when the client login. If web server can’t receive this method, web server will logout this client.

#### POST Request

{

"jsonrpc": "2.0",

"method": "HeartBeat",

"params": {},

"id": "1.5"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"1.5"

}

#### Response with error

{

"jsonrpc": "2.0",

"error ":

{

"code":"010501",

"message": "This method is wrong."

},

"id":"1.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010501 | This method is wrong. | This method is wrong. |
| 010502 | Other user login. | Other user login. |

### ForceLogin

#### POST Request

{

"jsonrpc": "2.0",

"method": "ForceLogin",

"params": {

"UserName": "admin",

"Password": "admin"

},

"id": "1.6"

}

#### Definition for result filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| UserName | String | Yes | The login user name. The default value is admin. |
| Password | String | Yes | The login user name. The default value is admin. |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"1.6"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "010601",

"message": "Username or Password is not correct."

},

"id": "1.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010601 | Username or Password is not correct. | Username or Password is not correct. |
| 010602 | Login times is used out. | If the user login error 5 times, this user can login until the device reboot. |

### SetPasswordChangeFlag

如果UI上开启提示修改默认密码功能，在提示修改默认密码的弹框中勾选不再提示后需要调用此接口并下发参数"change\_flag":1。修改密码成功后也需要修改此标示为1。

#### 5.1.7.1 Post Request:

{

"jsonrpc": "2.0",

"method": "SetPasswordChangeFlag",

"params":

{

"change\_flag":1

},

"id": "1.7"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"1.7"

}

#### Definition for each filed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | | **Type** | **Changable** | **Description** |
| change\_flag |  | |  | 0 :UI提示需要修改密码 /\*Firmware初始值为0，设备reset后重新置为0\*/  1:UI不提示修改密码 |

#### 5.1.7.3 Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"010701",

"message ":"setPasswordChangeFlag fail. ",

},

"id":"1.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010701 | Set PasswordChangeFlag fail. | Set PasswordChangeFlag fail. |

### 5.1.8 GetPasswordChangeFlag

#### 5.1.8.1 Post Request:

{

"jsonrpc": "2.0",

"method": "GetPasswordChangeFlag",

"params":

{

"change\_flag":0

},

"id": "1.8"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"1.8"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| change\_flag |  |  | 0 :UI提示需要修改密码 /\*Firmware初始值为0，设备reset后重新置为0\*/  1:UI不提示修改密码 |

#### 5.1.8.3 Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"010801",

"message ":"GetPasswordChangeFlag fail. ",

},

"id":"1.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 010801 | Get PasswordChangeFlag fail. | Get PasswordChangeFlag fail. |

## SIM Management Requests

### GetSimStatus

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSimStatus",

"params": {},

"id": "2.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"PinRemainingTimes":3,

"PinState":1,

"PukRemainingTimes":10,

"SIMLockRemainingTimes":1,

"SIMLockState":1,

"SIMState":1,

"PLMN":"46001"

},

"id":"2.1"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| PinRemainingTimes | Integer | Pin remaining times |
| PinState | Integer | 0: unknown  1: enable but not verified  2: PIN enable verified  3: PIN disable  4: PUK required  5: PUK times used out; |
| PukRemainingTimes | Integer | Puk remaining times |
| SIMLockRemainingTimes | Integer | SIMLockremaining times |
| SIMLockState | Integer | -1:no simlock/sim lock unlock  0: nck  1:nsck  2:spck  3:cck  4: pck  15,16,17,18,19:rck  30: rck forbid |
| SIMState | Integer | Possible values:  0: nown  1: SIM Card detected;  2 : PIN Required;  3 : PUK Required;  4 : SIM Lock required;  5 : PUK times used out;  6 : SIM card illegal;  7: SIM card ready;  11: SIM card is initing; |
| PLMN | String | Public Land Mobile Network (SimCard) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020101",

"message": "Get SIM statusfailed"

},

"id": "2.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020101 | Get SIM statusfailed | Get SIM statusfailed. |

### UnlockPin

#### POST Request

{

"jsonrpc": "2.0",

"method": "UnlockPin",

"params": {

"Pin":"1234"

},

"id": "2.2"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| Pin | String | Yes | Pin code |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020201",

"message": "Invalid Pin"

},

"id": "2.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020201 | Invalid Pin | Verify Pin code failed. |

### UnlockPuk

#### POST Request

{

"jsonrpc": "2.0",

"method": "UnlockPuk",

"params": {

"Puk":"12345678",

"Pin":"1234"

},

"id": "2.3"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| Puk | String | No | Puk code |
| Pin | String | Yes | pin code |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020301",

"message": "Invalid PUK."

},

"id": "2.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020301 | Invalid PUk | Verify PUK code failed. |

### ChangePINCode

#### POST Request

{

"jsonrpc": "2.0",

"method": "ChangePinCode",

"params": {

"CurrentPin":"1111",

"NewPin":"1234"

},

"id": "2.4"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| CurrentPin | String | No | The current PIN code. |
| NewPin | String | Yes | The target PIN Code that want to change. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020401",

"message": "Change PIN failed."

},

"id": "2.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020401 | Change PIN failed. | Change PIN code failed. |

### ChangePinState

#### POST Request

{

"jsonrpc": "2.0",

"method": "ChangePinState",

"params": {

"State":1,

"Pin":"1234"

},

"id": "2.5"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.5"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| State | Integer | Possible values:  0: PINDisabled;  1: PIN Enabled; |
| Pin | String | The current PIN code. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020501",

"message": "Change Pin state failed."

},

"id": "2.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020501 | Change Pin state failed. | Change Pin state failed. |

### GetAutoValidatePinState

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetAutoValidatePinState",

"params": {},

"id": "2.6"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"State": 0

},

"id":"2.6"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| State | Integer | Possible values:  0 : Disable  1 : Enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020601",

"message": "Get auto pin State failed."

},

"id": "2.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020601 | Get auto pin State failed. | Get auto pin State failed. |

### SetAutoValidatePinState

When the SIM Card status is required PIN, this function can verify the PIN code.

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetAutoValidatePinState",

"params":

{

"Pin":"1234",

"State": 0

},

"id": "2.7"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.7"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| Pin | String | Current PIN code. |
| State | Integer | Possible values:  0 : Disable  1 : Enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"020701",

"message": "Set auto pin State failed."

},

"id": "2.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020701 | Set auto pin State failed. | Set auto pin State failed. |

### UnlockSimlock

#### POST　Request

{

"jsonrpc": "2.0",

"method": "UnlockSimlock",

"params":

{

"SIMLockState":1,

"SIMLockCode": "0123456"

},

"id": "2.8"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.8"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| SIMLockCode | String | SIM lock code |
| SIMLockState | Integer | 0: nck  1:nsck  2:spck  3:cck  4: pck  15,16,17,18,19:rck |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020801",

"message": "Unlock SIM lock failed."

},

"id": "2.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020801 | Unlock SIM lock failed. | Unlock SIM lock failed. |

### ActiveSimlock

#### POST　Request

{

"jsonrpc": "2.0",

"method": "ActiveSimlock",

"params":

{

"State":0,

},

"id": "2.9"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.9"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| State | Integer | State:  0: active |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "020901",

"message": "Active SIM lock failed."

},

"id": "2.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 020901 | Active SIM lock failed. | Active SIM lock failed. |
| 020902 | PLMN code is null. | PLMN code is null. |

### EraseSimlock

#### POST　Request

{

"jsonrpc": "2.0",

"method": "EraseSimlock",

"params": {},

"id": "2.10"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"2.10"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "021001",

"message": "Erase SIM lock failed."

},

"id": "2.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 021001 | Erase SIM lock failed. | Erase SIM lock failed. |

## Connection

### GetConnectionState

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetConnectionState",

"params":{},

"id": "3.1"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"ConnectionStatus":"1",

"ConnectionTime":"",

"Conprofileerror":0,

"DlBytes":,

"DlRate":,

"IPv4Adrress":"",

"IPv6Adrress":"",

"Speed\_Dl":,

"Speed\_Ul":,

"UlBytes":,

"UlRate":

},

"id": "3.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ConnectionStatus | Integer | No | 0: disconnected  1: connecting  2:connected  3:disconnecting |
| ConnectionTime | String | No | ConnectionTime (s) |
| Conprofileerror | Integer | No | 0: ok  1: apn abnormal  2: pdp abnormal |
| DlBytes | Integer | No | Download bytes(B) |
| DlRate | Integer | No | Current download rate (bps) |
| IPv4Adrress | String | No | Connected ipv4 address |
| IPv6Adrress | String | No | Connected ipv6 address |
| Speed\_Dl | Integer | No |  |
| Speed\_Ul | Integer | No |  |
| UlBytes | Integer | No | Upload bytes(B) |
| UlRate | Integer | No | Current upload rate (bps) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "030101",

"message": "Get connection state failed. "

},

"id": "3.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 030101 | Get connection state failed. | Get connection state failed. |

### Connect

#### POST Request

{

"jsonrpc": "2.0",

"method": "Connect",

"params":{},

"id": "3.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "3.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "030201",

"message": "Connect failed. "

},

"id": "3.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 030201 | Connect failed | Connect to Internetfailed. |
| 030202 | No profile. | There is no default profile. |
| 030203 | It is not disconnect status. | It is not disconnect status. |
| 030204 | Data traffic is used out. | Data traffic is used out. |

### DisConnect

#### POST Request

{

"jsonrpc": "2.0",

"method": "DisConnect",

"params":{},

"id": "3.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "3.3"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"030301",

"message": "DisConnect failed. "

},

"id": "3.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 030301 | Disconnectfailed | Disconnect Internetfailed. |
| 030302 | It is not connection status. | It is not connection status. |

### GetConnectionSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetConnectionSettings",

"params":{},

"id": "3.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"ConnectMode":1,

"ConnOffTime":600,

"PdpType":0,

"RoamingConnect":0

},

"id": "3.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ConnectMode | Integer | Yes | 0: manual connect  1: auto connect |
| ConnOffTime |  |  |  |
| PdpType | Integer | Yes | PDP connection type:  PDP\_TYPE\_IPv4 = 0,  PDP\_TYPE\_PPP = 1,  PDP\_TYPE\_IPv6 = 2,  PDP\_TYPE\_IPv4v6 = 3 |
| RoamingConnect | Integer | Yes | Under Roaming status if can connect:  0: when roaming can not connect  1: when roaming can connect |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "030401",

"message": "Get connection settingsfailed. "

},

"id": "3.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 030401 | Get connection settingsfailed. | Get connection statefailed. |

### SetConnectionSettings

#### Request

{

"jsonrpc": "2.0",

"method": "SetConnectionSettings",

"params":

{

"PdpType":0,

"RoamingConnect":0,

"ConnectMode":1

},

"id": "3.5"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "3.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| PdpType | Integer | Yes | PDP connection type:  PDP\_TYPE\_IPv4 = 0,  PDP\_TYPE\_PPP = 1,  PDP\_TYPE\_IPv6 = 2,  PDP\_TYPE\_IPv4v6 = 3 |
| RoamingConnect | Integer | Yes | Under Roaming status if can connect:  0: when roaming can not connect  1: when roaming can connect |
| ConnectMode | Integer | Yes | 0: manual connect  1: auto connect |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "030501",

"message": "Set connection settingsfailed. "

},

"id": "3.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 030501 | Set connection settingsfailed. | Set connection statefailed. |

## Network Management Requests

### GetNetworkInfo

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetNetworkInfo",

"params": {},

"id": "4.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"Band":"",

"CellId":"",

"CenterFreq":"",

"CGI":"",

"DL\_channel":"",

"Domestic\_Roaming":1,

"EcIo":,

"eNBID":"",

"LAC":"",

"LTE\_state":"",

"mcc":"026",

"mnc":"333",

"NetworkName":"",

"NetworkType":1,

"PLMN":"26202",

"PLMN\_name":"china unicom",

"RncId":"",

"Roaming":0,

"RSCP":,

"RSRP":"",

"RSRQ":""

"RSSI":"",

"SignalStrength":1,

"SINR":"",

"SpnName":"",

"TxPWR":"",

"UL\_channel":""

},

"id":"4.1"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| Band | Integer |  |
| CellID | String | Cell ID as a decimal value  Reserved this filed, not implemented. Return value is empty string currently |
| CenterFreq | String |  |
| CGI | String |  |
| DL\_channel | string |  |
| Domestic\_Roaming |  |  |
| EcIo |  |  |
| eNBID | String |  |
| LAC | String | Location Area Code, a decimal value.  Reserved this filed, not implemented. Return value is empty string currently |
| LTE\_state | integer | 0: idle  1: searching |
| mcc | String |  |
| mnc | String |  |
| NetworkName | String | operator name |
| NetworkType | Integer | Network Type  0: NO SERVER  1: GPRS  2: EDGE  3: HSPA  4: HSUPA  5: UMTS  6: HSPA+  7: DC-HSPA+  8: LTE  9: LTE+  10:CDMA  11:GSM  12:EVDO  13:LTE-FDD  14:LTE-TDD 15: CDMA-->Ehrpd |
| PLMN | String | Public Land Mobile Network (Network) |
| PLMN\_name | string |  |
| RncID | String | RNC ID as a decimal value  Reserved this filed, not implemented. Return value is empty string currently |
| Roaming | Integer | Possible Values:  0: roaming  1: no roaming |
| RSCP |  |  |
| RSRP | String |  |
| RSRQ | string |  |
| RSSI | String |  |
| SignalStrength | Integer | RSSI value:  -1: No service  0: level 0  1: level 1  2: level 2  3: level 3  4: level 4 |
| SINR | String |  |
| SpnName | String | SPN name |
| TxPWR | String |  |
| UL\_channel | string |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "040101",

"message": "Get network information data error."

},

"id": "4.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040101 | Get network information data error. | Get network information data error. |

### SearchNetwork

#### POST Request

{

"jsonrpc": "2.0",

"method": "SearchNetwork",

"params":{},

"id": "4.2"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"4.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "040201",

"message": " Search network failed."

},

"id": "4.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040201 | Search network failed. | Search network failed. |

### SearchNetworkResult

#### POST Request

{

"jsonrpc": "2.0",

"method": "SearchNetworkResult",

"params":{},

"id": "4.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"ListNetworkItem":[{

"FullName":"",

"mcc":"",

"mnc":"",

"NetworkID":12,

"Numberic":"46001",

"Rat":"",

"ShortName":"",

"State":0,

},...],

" nw\_list\_len ":0,

"SearchState":0

},

"id":"4.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| FullName | String | Yes | network full name |
| mcc |  |  |  |
| mnc |  |  |  |
| NetworkID | Integer | Yes | Network ID |
| Numberic | String | Yes | Current network PLMN |
| Rat | Integer | Yes | 1: GSM  2: UMTS  3:LTE  4:CDMA  5:unknown |
| ShortName | String | Yes | network short name |
| State | Integer | Yes | 0: network unknown  1: network available  2: current register network  3: forbidden network |
| nw\_list\_len |  |  |  |
| SearchState | Integer | Yes | 0: search not network  1: searching  2: search successful  3: search failed |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "040301",

"message": "Get resultfailed."

},

"id": "4.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040301 | Get result failed. | Get result failed. |

### RegisterNetwork

#### POST Request

{

"jsonrpc": "2.0",

"method": "RegisterNetwork",

"params":

{

"NetworkID":"1",

},

"id": "4.4"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"4.4"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"040401",

"message": " Register network fail."

},

"id": "4.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040401 | Register network fail. | Register network fail. |

### GetNetworkRegisterState

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetNetworkRegisterState",

"params":{},

"id": "4.5"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

" regist\_state ":0,

},

"id":"4.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| regist\_state | Integer | Yes | 0: not regiseter  1: registting  2: register successful  3: registration failed |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "040501",

"message": "Getregister network state failed."

},

"id": "4.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040501 | Get register network state failed. | Get register network state failed. |

### GetNetworkSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetNetworkSettings",

"params":{},

"id": "4.6"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"DomesticRoam":0,

"DomesticRoamGuard":0

"NetselectionMode":"0",

"NetworkBand":"0",

"NetworkMode":"0"

},

"id":"4.6"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DomesticRoam | Integer | Yes | For Y856 Sprint  0: off;  1:on |
| DomesticRoamGuard | Integer | Yes | For Y856 Sprint  0: off;  1:on |
| NetselectionMode | Integer | Yes | 0: auto  1: manual |
| NetworkBand | Integer | No | 0: all band(1+6+10 Group)  1: GSM all band(2+3+4+5 Group)  2:GSM850  3:GSM900  4:GSM1800  5:GSM1900  6: WCDMA all band(7+8+9 Group)  7: WCDMA 900  8:WCDMA1800  9:WCDMA2100  10:LTE all band(11+12+13+14 Group)  11:LTE800(LTE Band20)  12:LTE900(LTE Band8)  13:LTE1800(LTE Band3)  14:LTE2600 (LTE Band7)  15:LTE2100 (LTE Band1)  16:LTE850(LTE Band26)  17:LTE1900(LTE Band25)  18:LTE2600(LTE Band41)  19:CDMA BC0  20:CDMA BC1  21:CDMA BC10  22:LTE\_Band25\_Band26\_Band41  23: LTE\_TDD\_BAND38\_2600  24: LTE\_TDD\_BAND40\_2300  25: LTE\_TDD\_BAND41\_2600  26: WCDMA850  27: WCDMA1900  28: LTE700(LTE BAND17)  29: LTE700(LTE BAND28)  30: LTE1700(LTE BAND4) |
| NetworkMode | Integer | Yes | 0: auto mode  1: 2G only  2: 3G only  3: LTE only  4: GSM\_LTE  5: UMTS\_LTE  6: GSM\_UMTS  7:CDMA\_EVDO : for Y856 Sprint  8:LTE\_CDMA\_EVDO: for Y856  9: EVDO ONLY  10: CDMA\_Ehrpd  11: CDMA Only(1x)  Sprint |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "040601",

"message": "Get network setting failed."

},

"id": "4.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040601 | Get network setting failed. | Getnetwork setting failed. |

### SetNetworkSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetNetworkSettings",

"params":

{

"NetworkMode":0,

"NetselectionMode":0

“NetworkBand”:0

},

"id": "4.7"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"4.7"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| NetworkMode | Integer | Yes | 0: auto mode  1: 2G only  2: 3G only  3: LTE only  4: GSM\_LTE  5: UMTS\_LTE  6: GSM\_UMTS  7:CDMA\_EVDO : for Y856 Sprint  8:LTE\_CDMA\_EVDO: for Y856  9: EVDO ONLY  10: CDMA\_Ehrpd  11: CDMA Only(1x)  Sprint |
| NetselectionMode | Integer | Yes | 0: auto  1: manual |
| NetworkBand | Integer | No | 0: all band(1+6+10 Group)  1: GSM all band(2+3+4+5 Group)  2:GSM850  3:GSM900  4:GSM1800  5:GSM1900  6: WCDMA all band(7+8+9 Group)  7: WCDMA 900  8:WCDMA1800  9:WCDMA2100  10:LTE all band(11+12+13+14 Group)  11:LTE800(LTE Band20)  12:LTE900(LTE Band8)  13:LTE1800(LTE Band3)  14:LTE2600 (LTE Band7)  15:LTE2100 (LTE Band1)  16:LTE850(LTE Band26)  17:LTE1900(LTE Band25)  18:LTE2600(LTE Band41)  19:CDMA BC0  20:CDMA BC1  21:CDMA BC10  22:LTE\_Band25\_Band26\_Band41  23: LTE\_TDD\_BAND38\_2600  24: LTE\_TDD\_BAND40\_2300  25: LTE\_TDD\_BAND41\_2600  26: WCDMA850  27: WCDMA1900  28: LTE700(LTE BAND17)  29: LTE700(LTE BAND28)  30: LTE1700(LTE BAND4) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "040701",

"message": "Set network setting failed."

},

"id": "4.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040701 | Set network setting failed. | Set network setting failed. |
| 040702 | Just can set when disconnected. | Just can set when disconnected. |

### GetNetworkModeListType————Firmware端实现，UI未调用接口

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetNetworkModeListType",

"params":{},

"id": "4.8"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

" NetworkModeListType ":0,

},

"id":"4.8"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| NetworkModeListType | Integer | Yes | 0: LTE\_CDMA-EVDO,CDMA-EVDO only,LTE only  1: CDMA-EVDO only  2: LTE only |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "040801",

"message": "Get NetworkModeListType failed."

},

"id": "4.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 040801 | Get NetworkModeListType failed. | Get NetworkModeListType failed. |

### GetDomRoamGuardForced————Firmware端实现，UI未调用接口

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetDomRoamGuardForced",

"params":{},

"id": "4.9"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

" DomRoamGuardForced ":0,

},

"id":"4.9"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DomRoamGuardForced | Integer | Yes | 0: off (Allow to configure domestic roam guard)  1: on (Not allow to configure domestic roam guard) |

## WLAN

### GetWlanState

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetWlanState",

"params":{},

"id": "5.1"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"WlanState":1,

},

"id": "5.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| WlanState | Integer | NO | 0: off  1: on  2: WPS |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050101",

"message": "Get WLANstatefailed."

},

"id": "5.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050101 | Get WlAN statefailed. | Get WlAN statefailed. |

### SetWlanOff

This function work when WIFI is on.

#### Request

{

"jsonrpc": "2.0",

"method": "SetWlanOff",

"params":{},

"id": "5.2"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "5.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050201",

"message": "Set WLAN off failed."

},

"id":"5.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050201 | Set WLAN off failed. | Set WLAN off failed. |
| 050202 | WLAN is off. | WLAN is off. |

### SetWlanOn

This function work when WIFI is off.

#### Request

{

"jsonrpc": "2.0",

"method": "SetWlanOn",

"params":{},

"id": "5.3"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "5.3"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050301",

"message": "Set WLAN on failed."

},

"id":"5.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050301 | Set WLAN on failed. | Set WLAN on failed. |
| 050302 | WLAN is on. | WLAN is on |

### GetWlanSettings

#### Request

{

"jsonrpc": "2.0",

"method": "GetWlanSettings",

"params":{},

"id": "5.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"APList":[{

"ApIsolation":1,

"ApStatus":1,

"Bandwidth":0,

"Channel":5,

"CountryCode":"it",

"CurChannel":4,

"curr\_num":3,

"max\_numsta":10,

"SecurityMode":1,

"Ssid":"",

"SsidHidden":1,

"WepKey":"",

"WepType":1,

"WlanAPID":0,

"WMode":1,

"WpaKey":"",

"WpaType":1

}],

"curr\_num":3,

"WlanAPMode":0,

"WiFiOffTime":0

},

"id": "5.4

}

1. Note: If only support one mode, the optional paramers are no required.

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ApIsolation | Integer | Yes | 0: disable  1: enable |
| ApStatus |  |  |  |
| Bandwidth | Integer | Yes | 0: Auto(20MHz/40MHz);  1: 20MHz;  2: 40MHz; |
| Channel | Integer | Yes | The current used 2.4G WIFI Channel:  0: Auto  1-14: Other channel |
| CountryCode | String | Yes | Country Code |
| CurChannel |  |  | Current channel |
| curr\_num | Integer | Yes | Current client count that connect to Main SSID |
| max\_numsta | Integer | Yes | 2.4 G WIFI max number client |
| SecurityMode | Integer | Yes | 0: disable  1: wep  2: WPA  3: WPA2  4: WPA/WPA2 |
| Ssid | String | Yes | SSID |
| SsidHidden | Integer | Yes | 0:disable  1: enable |
| WepKey | String | Yes |  |
| WepType | Integer | Yes | 0: OPEN  1: share |
| WlanAPID |  |  |  |
| WMode | Integer | Yes | 2.4G:  0: 802.11b;  1: 802.11b/g;  2: 802.11b/g/n  3: Auto |
| WpaKey | String | Yes |  |
| WpaType | Integer | Yes | 0: TKIP  1: AES  2: AUTO |
| WlanAPMode | Integer | Yes | Wlan running mode:  0: 2.4G  1: 5G  2: 2.4G and 5G |
| WiFiOffTime | Integer | Yes | WIFI sleep time:  0:disable  The other digits represent the time for sleep. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050401",

"message": "Get WLAN Settings failed."

},

"id": "5.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050401 | Get WLAN Settings failed. | Get WLAN Settings failed. |

### SetWlanSettings

#### Request

{

"jsonrpc": "2.0",

"method": "SetWlanSettings",

"params":

{

"WlanAPMode":0,

"WiFiOffTime":0,

"APList":[{

"WlanAPID":0,

"ApStatus":1,

"WMode":1,

"Ssid":"",

"SsidHidden":1,

"Channel":5,

"SecurityMode":1,

"WepType":1,

"WepKey":"",

"WpaType":1,

"WpaKey":"",

"CountryCode":"it",

"ApIsolation":1,

"max\_numsta":10

}]

},

"id": "5.5"

}

1. Note: If only support one mode, the optional paramers are no required.

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| WlanAPMode | Integer | Yes | Wlan running mode:  0: 2.4G  1: 5G  2: 2.4G and 5G |
| WlanAPID |  |  |  |
| ApStatus |  |  |  |
| WMode | Integer | Yes | 2.4G:  0:802.11b;  1: 802.11b/g;  2: 802.11b/g/n  3: Auto |
| Ssid | String | Yes | SSID |
| SsidHidden | Integer | Yes | 0:disable  1: enable |
| Channel | Integer | Yes | The current used 2.4G WIFI Channel:  0: Auto  1-14: Other channel |
| SecurityMode | Integer | Yes | 0: disable  1: wep  2: WPA  3: WPA2  4: WPA/WPA2 |
| WepType | Integer | Yes | 0: OPEN  1: share |
| WepKey | String | Yes |  |
| WpaType | Integer | Yes | 0: TKIP  1: AES  2: AUTO |
| WpaKey | String | Yes |  |
| CountryCode | String | Yes | Country Code |
| ApIsolation | Integer | Yes | 0: disable  1: enable |
| max\_numsta | Integer | Yes | 2.4 G WIFI max number client |
| WiFiOffTime | Integer | Yes | WIFI sleep time:  0:disable  The other digits represent the time for sleep. |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "5.5"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050501",

"message": "Set WLAN Settings failed."

},

"id":"5.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050501 | Set WLAN Settings failed. | Set WLAN Settings failed. |
| 050502 | WIFI is off, can not set. | WIFI is off, can not set. |
| 050503 | WPS is working | WPS is working,can not set |

### SetWPSPin

#### Request

{

"jsonrpc": "2.0",

"method": "SetWPSPin",

"params":{

"WpsPin":"1234"

},

"id": "5.6"

}

#### Response

{

"jsonrpc": "2.0",

"result": {}

"id": "5.6"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| WpsPin | String | Yes | The WPS PIN code. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050601",

"message": "Set WPS PIN mode failed."

},

"id":"5.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050601 | Set WPS PIN mode failed. | Set WPS PIN mode failed. |
| 050602 | WPS PIN not support | WPS PIN not support in WEP, WPA, WPA2 SKIP mode. |
| 050603 | WPS working | WPS is active |

### SetWPSPbc

#### Request

{

"jsonrpc": "2.0",

"method": "SetWPSPbc",

"params":{},

"id": "5.7"

}

#### Response

{

"jsonrpc": "2.0",

"result": {}

"id": "5.7"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050701",

"message": "Set WPS PBC failed."

},

"id":"5.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050701 | Set WPS PBC failed. | Set WPS PBC failed. |
| 050702 | WPS PBC not support | WPS PIN not support in WEP, WPA, WPA2 SKIP mode. |
| 050703 | WPS working | WPS is active |

### GetWlanSupportMode

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetWlanSupportMode",

"params":{},

"id": "5.8"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"WlanAPMode":0

},

"id": "5.8"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| WlanAPMode | Integer | NO | 0: Support 2.4G  1: Support 5G  2: Support 2.4G and 5G |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "050801",

"message": " Get wlan support mode failed."

},

"id": "5.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 050801 | Get wlan support mode failed. | Get wlan support mode failed. |

## SMS

### getSmsInitState

#### Request

{

"jsonrpc": "2.0",

"method": "getSmsInitState",

"params":{},

"id": "6.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"state":1

},

"id": "6.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| state | integer | Yes | The status of the SMS initing.  0:Complete;  1:Initing |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060101",

"message": "Get SMS init status failed."

},

"id": "6.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060101 | Get SMS init status failed. | Get SMS init status failed. |

### GetSMSContactList

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSMSContactList",

"params":

{

"Page": 1

},

"id": "6.2"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"Page":1,

"SMSContactList":[

{

"ContactId":0,

"PhoneNumber":["13713804698","13523459876"],

"ReportStatus":0,

"SMSContent":"Hello World",

"SMSId":"3",

"SMSTime":"2014-01-12",

"SMSType":0,

"TSMSCount":7,

"UnreadCount":4

},

……

]

"TotalPageCount":1

},

"id":"6.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| Page | Integer | Yes | Pages that SMS Contact list separate,  0: return all Contact SMS.  1: the first page ,every page has 10 contacts SMS list.  2….. |
| ContactId | Integer | Yes | The list ID. |
| PhoneNumber | Array | Yes | The SMS Contact info, may one and more. |
| ReportStatus |  |  |  |
| SMSContent | String | Yes | The first 40 characters of the latest SMS. |
| SMSId | Integer | Yes | The latest SMS id. |
| SMSTime | String | Yes | the latest SMS time.  format: YYYY-MM-DDhh:mm:ss |
| SMSType | Integer | Yes | 0 : read  1 : unread  2 : sent  3 : sent failed  4 : report  5 : flash  6: draft |
| TSMSCount | Integer | Yes | The total count of the SMS that the phone number has. |
| UnreadCount | Integer | Yes | The count of the unread SMS that the phone number has. |
| TotalPageCount |  |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060201",

"message": "Get SMS Contact List failed."

},

"id": "6.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060201 | Get SMS Contact List failed. | Get SMS Contact List failed. |

### GetSMSContentList

#### Request

{

"jsonrpc": "2.0",

"method": "GetSMSContentList",

"params": {

"Page": 0,

"ContactId":0

},

"id": "6.3"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"ContactId":0,

"Page":1,

"PhoneNumber":["13713804698","13523459876"],

"SMSContentList":[

{

"ReportStatus":0,

"SMSContent":"Hello World",

"SMSId":3,

"SMSTime":"2014-01-12",

"SMSType":0

},

……

]

"TotalPageCount":1

},

"id":"6.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ContactId | Integer | Yes | The id in 6.1 GetSMSContactList |
| Page | Integer | Yes | Pages that SMS Content list separate,  0: return allSMS content.  1 : the first page ,every page has 10 SMS content list.  2….. |
| PhoneNumber |  |  |  |
| ReportStatus |  |  |  |
| SMSContent | String | Yes | The all content of the SMS. |
| SMSId | Integer | Yes | The single SMS id. |
| SMSTime | String | Yes | The single SMS time.  format: YYYY-MM-DDhh:mm:ss |
| SMSType | Integer | Yes | 0 : read  1 : unread  2 : sent  3 : sent failed  4 : report  5 : flash  6: draft |
| TotalPageCount |  |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060301",

"message": "Get SMS contentlist failed."

},

"id": "6.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060301 | Get SMS content list failed. | Get SMS content list failed. |

### GetSMSStorageState

Apps need this api.

#### Request

{

"jsonrpc": "2.0",

"method": "GetSMSStorageState",

"params":{},

"id": "6.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"LeftCount": 20,

"MaxCount":100,

"TUseCount": 50,

"UnreadReport": 1,

"UnreadSMSCount": 20

},

"id": "6.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| LeftCount | Integer | Yes | The left space that store the SMS |
| MaxCount | Integer | Yes | The current max space (SMS Count) that store the SMS. |
| TUseCount | Integer | Yes | Current store SMS total count. |
| UnreadReport | Integer | Yes | 0： no new status report  1： exit new status report |
| UnreadSMSCount | Integer | Yes | Current store unread SMS count. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060401",

"message": " Get SMS Storage statefailed. "

},

"id": "6.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060401 | Get SMSStorage statefailed. | Get SMS Storage state failed. |

### DeleteSMS

#### Request

{

"jsonrpc": "2.0",

"method": "DeleteSMS",

"params": {

"DelFlag":1,

"ContactId":1,

"SMSId":1

},

"id": "6.5"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "6.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DelFlag | Integer | Yes | This flag means the SMS that want to delete.  0: delete all SMS  1: delete one record in Contact SMS list  2:delete one record in Content SMS list |
| ContactId | Integer | Yes | This id must be in Contact SMS list. |
| SMSId | Integer | Yes | This id must be Content SMS list. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060501",

"message": "Delete SMS failed."

},

"id": "6.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060501 | Delete SMS failed. | Delete SMS failed. |

### SendSMS

#### Request

{

"jsonrpc": "2.0",

"method": "SendSMS",

"params": {

"SMSId": -1,

"SMSContent":"Hello World",

"PhoneNumber":["+84912345"],

"SMSTime":"2014-09-11 18:06:00"

},

"id": "6.6"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "6.6"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SMSId | Integer | Yes | the ID of the SMS in draft, failed send t, and snt SMS, if the value is -1 means send a new SMS. |
| SMSContent | String | Yes | The send message content. |
| PhoneNumber | String | Yes | The sendPhoneNumber |
| SMSTime | String | Yes | Time when sending sms. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060601",

"message": "Send SMS failed."

},

"id": "6.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060601 | Send SMS failed. | Send SMS failed. |
| 060602 | Fail still sending last message. | Fail still sending last message. |
| 060603 | Fail with store space full. | Fail with store space full. |

### GetSendSMSResult

#### Request

{

"jsonrpc": "2.0",

"method": "GetSendSMSResult",

"params":{},

"id": "6.7"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"SendStatus":0

},

"id": "6.7"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SendStatus | Integer | Yes | 0 : none  1 : sending  2 : success  3: failstill sending last message  4 : fail with Memory full  5: fail |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060701",

"message": "Get send SMS status failed."

},

"id": "6.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060701 | Get send SMS status failed. | Get send SMS status failed. |

### SaveSMS

#### Request

{

"jsonrpc": "2.0",

"method": "SaveSMS",

"params":

{

"SMSId":1,

"SMSContent": "hi!",

"PhoneNumber": ["134544555"],

"SMSTime":"2014-09-11 18:06:00"

},

"id": "6.8"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "6.8"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SMSId | integer | Yes | the index of SMS, if savenew SMS is -1, else other. |
| SMSContent | String | Yes | The SMS content to save. |
| PhoneNumber | string | Yes | The phone number to save. |
| SMSTime | String | Yes | Time when saving sms |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060801",

"message": "Save SMS failed."

},

"id": "6.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060801 | Save SMS failed. | Save SMS failed. |
| 060802 | Fail with store space full. | Fail with store space full. |

### GetSMSSettings

#### Request

{

"jsonrpc": "2.0",

"method": "GetSMSSettings",

"params":{},

"id": "6.9"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"SMSCenter":"13888888888",

"SMSReportFlag":0,

"StoreFlag":0

},

"id": "6.9"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SMSCenter | String | Yes | the Center number of the SIM Card. |
| SMSReportFlag | Integer | Yes | SMS report is enable or disable  0 : Disable  1 : Enable |
| StoreFlag | Integer | Yes | SMS store space  0 : SIM Card  1 : ME |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "060901",

"message": "Get short message settings failed."

},

"id": "6.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 060901 | Get short message settings failed. | Get short message settings failed. |

### SetSMSSettings

#### Request

{

"jsonrpc": "2.0",

"method": "SetSMSSettings",

"params":

{

"SMSReportFlag":0,

"StoreFlag":0,

"SMSCenter":"13888888888"

},

"id": "6.10"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "6.10"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SMSReportFlag | Integer | Yes | SMS report is enable or disable  0 : Disable  1 : Enable |
| StoreFlag | Integer | Yes | SMS store space  0 : SIM Card  1 : ME |
| SMSCenter | String | Yes | the Center number of the SIM Card. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "061001",

"message": "Set SMS settings failed."

},

"id": "6.10"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 061001 | Set SMS settings failed. | Set SMS settings failed. |

### GetSingleSMS

#### Request

{

"jsonrpc": "2.0",

"method": "GetSingleSMS",

"params":{

"SMSId":3,

},

"id": "6.11"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"SMSContent":"Hello World"

},

"id": "6.11"

}

#### Definition for each filed

Notice：if a sms type is unread,once to get this sms by API GetSingleSMS ，this sms type must be change to read.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SMSContent | String | Yes | The all content of the SMS. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "061101",

"message": "Get single sms failed."

},

"id": "6.11"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 061101 | Get single sms failed. | Get single sms failed. |

## Statistics

### GetUsageRecord

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetUsageRecord",

"params":{

"current\_time": "2014-09-11 18:06:00"

},

"id": "7.1"

}

#### Response

{

"jsonrpc": "2.0",

"result":{

"CurrConnTimes": 2122,

"HCurrUseDL": 42344,

"HCurrUseUL": 2122,

"HUseData": 42342,

"MonthlyPlan ": 42344,

"RCurrUseDL": 42344,

"RCurrUseUL": 2122,

"RoamUseData": 42342,

"TConnTimes": 42342

},

"id": "7.1"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| CurrConnTimes | Integer | Current connected Internet times. |
| HCurrUseDL | Integer | The mothly use download data in home network. |
| HCurrUseUL | Integer | The mothly use upload data in home network. |
| HUseData | Integer | The mothly used data in home network. |
| MonthlyPlan | Integer | The Max usage date that the device should use, when the real usage is more than this data, the connection must disconnect. |
| RCurrUseDL | Integer | The mothly use download data in roaming network. |
| RCurrUseUL | Integer | The mothly use upload data in roaming network. |
| RoamUseData | Integer | The mothly used data in roaming network. |
| TConnTimes | Integer | The total connected Internet times. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "070101",

"message": "Get usage record failed."

},

"id":"7.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 070101 | Get usage record failed. | Get usage record failed. |

### SetUsageRecordClear

#### POST Request

{

"jsonrpc":"2.0",

"method":"SetUsageRecordClear",

"params":

{

"clear\_time":"2014-11-06 14:03:20"

},

"id":"7.2"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| clear\_time | string | Current time for clearing usage. |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "7.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "070201",

"message": "Clear all usage records failed."

},

"id":"7.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 070201 | Clear all usage records failed. | Clear all usage records failed. |

### GetUsageSettings

#### Request

{

"jsonrpc": "2.0",

"method": "GetUsageSettings",

"params":{},

"id": "7.3"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"AutoDisconnFlag": 1,

"BillingDay": 1,

"MonthlyPlan": 500,

"TimeLimitFlag": 1,

"TimeLimitTimes": 1,

"Unit"：0,

"UsedData": 300,

"UsedTimes": 1

},

"id": "7.3"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| AutoDisconnFlag | Integer | This flage control the disconnection when the usage settings get conditions.  0:disable, not auto disconnect  1:enable, auto disconnect |
| BillingDay | Integer | Billing day (Min:1, Max:31) |
| MonthlyPlan | Integer | The max data that on month could use. If this value =0, this means no limited. |
| TimeLimitFlag | Integer | The time limit function flage,  0: disable  1:enable |
| TimeLimitTimes | Integer | The time limit function open, must set the limit time. |
| Unit | Integer | The unit function flage,  0: MB  1: GB  2: KB |
| UsedData | Integer | The total roam and home used data in month. |
| UsedTimes | Integer | The used time that after open time limit function. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "070301",

"message": "Get usage settings failed."

},

"id":"7.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 070301 | Get usage settings failed. | Get usage settings failed. |

### SetUsageSettings

#### Request

{

"jsonrpc": "2.0",

"method": "SetUsageSettings",

"params":{

"BillingDay": 1,

"MonthlyPlan": 500,

"UsedData": 376989,

"TimeLimitFlag": 1,

"TimeLimitTimes": 1,

"UsedTimes": 88,

"AutoDisconnFlag": 1,

"Unit"：0

},

"id": "7.4"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| BillingDay | Integer | Billing day (Min:1, Max:31), not set on sprint project. |
| MonthlyPlan | Integer | The max data that on month could use. The max data that on month could use. If this value =0, this means no limited. |
| UsedData | Integer |  |
| TimeLimitFlag | Integer | The time limit function flage,  0: disable  1:enable |
| TimeLimitTimes | Integer | The time limit function open, must set the limit time. |
| **UsedTimes** | Integer |  |
| AutoDisconnFlag | Integer | This flage control the disconnection when the usage settings get conditions.  0:disable, not auto disconnect  1:enable, auto disconnect |
| Unit | Integer | The unit function flage,  0: MB  1: GB  2: KB |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "7.4"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "070401",

"message": "Set usage settings failed."

},

"id":"7.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 070401 | Set usage settings failed. | Set usage settings failed. |

### GetFDTimer

#### Request

{

1. "jsonrpc": "2.0",
2. "method": "GetFDTimer",
3. "params":{},
4. "id": "7.5"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| FDTimer | Integer | 0, 5, 10, 20, 30, 60, 90, 120 sec |

#### Response

{

"jsonrpc": "2.0",

"result": {

" FDTimer": 5,

},

"id": "7.5"

}

### SetFDTimer

#### Request

{

"jsonrpc": "2.0",

"method": "SetFDTimer",

"params":{

" FDTimer": 0,

},

"id": "7.6"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| FDTimer | Integer | 0, 5, 10, 20, 30, 60, 90, 120 sec |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "7.6"

}

## USSD

### SendUSSD

#### POST Request

{

"jsonrpc": "2.0",

"method": "SendUSSD",

"params":

{

"UssdContent": "\*#189",

"UssdType": 1

},

"id": "8.1"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| UssdContent | String | Send USSD content |
| UssdType | Number | 0:none;  1: done;  2: more;  3:abort  4:other;  5:NoSUP;  6:timeout  7:max; |

#### Response

{

"jsonrpc": "2.0",

"result": {}

"id": "8.1"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "080101",

"message": "Send USSD failed."

},

"id":"8.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 080101 | Send USSD failed. | Send USSD commandfailed. |

### GetUSSDSendResult

#### Request

{

"jsonrpc": "2.0",

"method": "GetUSSDSendResult",

"params":{},

"id": "8.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"SendState": 1,

"UssdContent": "",

"UssdContentLen": 1,

"UssdType": 1

}

"id": "8.2"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| SendState | Number | SendUSSD State  0:none;  1: sending;  2: complete; |
| UssdContent | String | ResponseUSSDcontent |
| UssdContentLen | Number | ResponseUSSD content length |
| UssdType | Number | 0:none;  1: done;  2: more;  3:abort  4:other;  5:NoSUP;  6:timeout  7:max; |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "080201",

"message": "Get send USSD result failed."

},

"id":"8.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 080201 | Get send USSD result failed. | Get send USSD result failed. |

### SetUSSDEnd

#### Request

{

"jsonrpc": "2.0",

"method": "SetUSSDEnd",

"params":{},

"id": "8.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "8.3"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "080301",

"message": "Set USSD end failed."

},

"id":"8.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 080301 | Set USSD end failed. | Set USSD end failed. |

## Update

### GetDeviceNewVersion

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetDeviceNewVersion",

"params":{},

"id": "9.1"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"State":1,

"Version": "",

"total\_size"：0

},

"id": "9.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| State | integer | No | 0:checking  1: New version  2: no new version  3: no connect  4: service not available  5: check error(time out etc…) |
| Version | string | No | The new software version number on the FOTA Sever. |
| total\_size |  |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"090101",

"message": "Get New version s failed."

},

"id":"9.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 090101 | Get New version failed. | Get New version failed. |

### SetFOTAStartDownload

此接口根据smartRelease配置表FotaUpgradeMode项的配置会有不同，需要注意：

FotaUpgradeMode通用版本默认配置项为1,代表SetFOTAStartDownload接口下载完成后自动进入升级

FotaUpgradeMode配置为0时，代表SetFOTAStartDownload接口下载完成后不会自动升级，需要调用5.9.6

SetDeviceStartUpdate来触发升级

#### Request

{

"jsonrpc": "2.0",

"method": "SetFOTAStartDownload",

"params":{},

"id": "9.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":{ },

"id": "9.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "090201",

"message": "Set device update failed."

},

"id":"9.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 090201 | Set Device start update failed. | Set Device start update failed. |

### GetDeviceUpgradeState

#### Request

{

"jsonrpc": "2.0",

"method": "GetDeviceUpgradeState",

"params":{},

"id": "9.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"Status":0,

"Process":0

},

"id": "9.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| Status | integer | No | 0: No start update(UI does not send the start update command)  1: updating (Download Firmware phase)  2: complete |
| Process | integer | No | If status =1,process:1- 100 , and stand for the updating(download)process. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "090301",

"message": "Get device update state failed."

},

"id":"9.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 090301 | Get device update state failed. | Get device update state failed. |

### SetDeviceUpdateStop

#### Request

{

"jsonrpc": "2.0",

"method": "SetDeviceUpdateStop",

"params":{},

"id": "9.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":{ },

"id": "9.4"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "090401",

"message": "Set device update stop failed."

},

"id":"9.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 090401 | Set device update stop failed. | Set device update stop failed. |

### SetCheckNewVersion

#### Request

{

"jsonrpc": "2.0",

"method": "SetCheckNewVersion",

"params":{},

"id": "9.5"

}

#### Response

{

"jsonrpc": "2.0",

"result":{ },

"id": "9.5"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "090501",

"message": "set check device new version failed ."

},

"id":"9.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 090501 | Set check device new version failed . | Set check device new version failed . |

### SetDeviceStartUpdate

**需要注意，软件downloaded完成之前不能调用此接口**

#### Request

{

"jsonrpc": "2.0",

"method": "SetDeviceStartUpdate",

"params":{},

"id": "9.6"

}

#### Response

{

"jsonrpc": "2.0",

"result":{ },

"id": "9.6"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "090601",

"message": "set device start update failed ."

},

"id":"9.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 090601 | set device start update failed . | set device start update failed . |

## Firewall

### GetMacFilterSettings

#### Request

{

"jsonrpc": "2.0",

"method": "GetMacFilterSettings",

"params":{},

"id": "10.1"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"filter\_policy":1,

"MacAllowList":["44:44:22:d5:21:24",…],

"MacDenyList":["22:44:44:a5:24:44",…]

},

"id": "10.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| filter\_policy | Integer | Yes | 0: Disable  1:allow  2:deny |
| MacAllowList | Array | Yes | The Allow Mac address. |
| MacDenyList | Array | Yes | The Deny Mac address. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100101",

"message": "Get MAC Filter Settingsfailed. "

},

"id": "10.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100101 | Get MAC Filter Settingsfailed. | Get MAC Filter Settingsfailed. |

### SetMacFilterSettings

#### Request

{

"jsonrpc": "2.0",

"method": " SetMacFilterSettings",

"params":{

"filter\_policy":1,

"MacAllowList":["44:44:22:d5:21:24",…],

"MacDenyList":["22:44:44:a5:24:44",…]

},

"id": "10.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| filter\_policy | Integer | Yes | 0: off  1:white  2:black |
| MacAllowList | Array | Yes | The Allow Mac address. |
| MacDenyList | Array | Yes | The Deny Mac address. |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100201",

"message": "Set MAC Filter settingsfailed. "

},

"id": "10.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100201 | Set MAC Filter settingsfailed. | Set MAC Filter settingsfailed. |
| 100202 | WPS is on, set failed. | WPS is on, set failed. |

### getPortFwding

#### Request

{

"jsonrpc": "2.0",

"method": "getPortFwding",

"params":{},

"id": "10.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"total\_num":1,

"portfwd\_list":[{

"private\_ip": "192.168.1.1",

"private\_port": "88",

"global\_port": "",

"fwding\_protocol": 6,

"port\_forward\_status": 1

},

},

"id": "10.3"

}

#### Definition for each filed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** | |
| portfwd\_list | Array | Yes | The port forwarding list. | |
| private\_ip | string | Yes |  | |
| private\_port | string | Yes |  | |
| global\_port | string | yes |  | |
| fwding\_protocol | number | yes | 6: TCP  17: UDP  253: TCP&UDP | |
|  |  |  |  | |
| port\_forward\_status | number |  | 0: disable  1: enable |
| total\_num | number | yes | The number of the port forwarding list. | |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100301",

"message": "Get port forwarding list failed. "

},

"id": "10.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100301 | Get port forwarding list failed. | Get port forwarding list failed. |

### addPortFwding

#### Request

{

"jsonrpc": "2.0",

"method": "addPortFwding",

"params":{

"private\_ip": "192.168.1.1",

"private\_port": "88",

"global\_port": "",

"fwding\_protocol": 6,

"port\_forward\_status": 1

},

"id": "10.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| private\_ip | string | Yes |  |
| private\_port | string | Yes |  |
| global\_port | string | yes |  |
| fwding\_protocol | number | yes | 6: TCP  17: UDP  253: TCP&UDP |
| port\_forward\_status | number |  | 0: disable  1: enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100401",

"message": "Add new port forwarding failed. "

},

"id": "10.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100401 | Add new port forwarding failed. | Add new port forwarding failed. |

### deletePortFwding

#### Request

{

"jsonrpc": "2.0",

"method": "deletePortFwding",

"params":{

"private\_ip": "192.168.1.1",

"private\_port": "88",

"global\_port": "",

"fwding\_protocol": 6

},

"id": "10.5"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| private\_ip | string |  |  |
| private\_port | string |  |  |
| global\_port | string |  |  |
| fwding\_protocol | number |  | 6: TCP  17: UDP  253: TCP&UDP |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100501",

"message": "Delete port forwarding failed. "

},

"id": "10.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100501 | Delete port forwarding failed. | Delete port forwarding failed. |

### editPortFwding

#### Request

{

"jsonrpc": "2.0",

"method": "editPortFwding",

"params":{

"old\_private\_ip": "192.168.1.1",

"old\_private\_port": "88",

"old\_global\_port": "192.168.1.1",

"old\_fwding\_protocol": 6,

"new\_private\_ip": "192.168.1.1",

"new\_private\_port": "88",

"new\_global\_port": "",

"new\_fwding\_protocol": 6,

},

"id": "10.6"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.6"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| old\_private\_ip | string |  |  |
| old\_private\_port | string |  |  |
| old\_global\_port | string |  |  |
| old\_fwding\_protocol | number |  | 6: TCP  17: UDP  253: TCP&UDP |
| new\_private\_ip | string |  |  |
| new\_private\_port | string |  |  |
| new\_global\_port | string |  |  |
| new\_fwding\_protocol | number |  | 6: TCP  17: UDP  253: TCP&UDP |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100601",

"message": "Edit port forwarding failed. "

},

"id": "10.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100601 | Edit port forwarding failed. | Edit port forwarding failed. |

### SetPortFwding————暂不支持

#### Request

{

"jsonrpc": "2.0",

"method": " SetPortFwding",

"params":{

"total\_num": 1,

"portfwd\_list":[{

"portfwd\_name": "name1",(optional)

"private\_ip": "192.168.1.1",

"private\_port": "88",

"global\_port": "",

"fwding\_protocol": 6,

"fwding\_status": 1

},…]

},

"id": "10.7"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.7"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| portfwd\_name | string | Yes | Recored port forward item name,for Y856sprint is Mandatory,other is optional |
| private\_ip | string | Yes |  |
| private\_port | string | Yes |  |
| global\_port | string | yes |  |
| fwding\_protocol | number | yes | 6: TCP  17: UDP  253: TCP&UDP |
| fwding\_status | number | yes | 0: off;  1: on |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100701",

"message": "Set port forwarding failed. "

},

"id": "10.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100701 | Set port forwarding failed. | Set port forwarding failed. |

### getDMZInfo

#### Request

{

"jsonrpc": "2.0",

"method": "getDMZInfo",

"params":{},

"id": "10.8"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

" dmz\_status":1,

" dmz\_ip":"192.168.1.60"

},

"id": "10.8"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| dmz\_status | number |  | 0: DMZ off  1: DMZ on |
| dmz\_ip | string |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100801",

"message": "Get DMZ Settings failed. "

},

"id": "10.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100801 | Get DMZ Settings failed. | Get DMZ Settings failed. |

### setDMZInfo

#### Request

{

"jsonrpc": "2.0",

"method": "setDMZInfo",

"params":{

"dmz\_status":1,

"dmz\_ip":"192.168.1.60"

},

"id": "10.9"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| dmz\_status | number |  | 0: DMZ off  1: DMZ on |
| dmz\_ip | string |  |  |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.9"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "100901",

"message": "Set DMZ settings failed. "

},

"id": "10.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100901 | Set DMZ settings failed. | Set DMZ settings failed. |

### getIPFilterList

#### Request

{

"jsonrpc": "2.0",

"method": "getIPFilterList",

"params":{},

"id": "10.10"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"total\_num":1,

"ipFilter\_list":[{

"list\_id": 0,

"lan\_ip": "192.168.1.1",

"lan\_port": "80",

"wan\_ip": "20.12.12.5",

"wan\_port": "85",

"ip\_protocol": 6,

"ip\_status": 0

},…]

},

"id": "10.10"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ipFilter\_list | Array | Yes | The ip filter list. |
| list\_id | number |  | record id |
| lan\_ip | string |  |  |
| lan\_port | string |  |  |
| wan\_ip | string |  |  |
| wan\_port | string |  |  |
| ip\_protocol | number |  | 6: TCP  17: UDP  253: TCP&UDP |
| ip\_status | number |  |  |
| total\_num | number |  | the count of record |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101001",

"message": "Get ip filter list failed. "

},

"id": "10.10"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101001 | Get ip filter list failed. | Get ip filter list failed. |

### addIPFilter

#### Request

{

"jsonrpc": "2.0",

"method": "addIPFilter",

"params":{

"lan\_ip": "192.168.1.1",

"lan\_port": "80",

"wan\_ip": "20.12.12.5",

"wan\_port": "85",

"ip\_protocol": 6,

"ip\_status": 0

}

,

"id": "10.11"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.11"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| lan\_ip | string |  |  |
| lan\_port | string |  |  |
| wan\_ip | string |  |  |
| wan\_port | string |  | End IP Address |
| ip\_protocol | number |  | 6: TCP  17: UDP  253: TCP&UDP |
| ip\_status | number |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101101",

"message": "Add new ip filter failed. "

},

"id": "10.11"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101101 | Add new ip filter failed. | Add new ip filter failed. |

### deleteIPFilter

#### Request

{

"jsonrpc": "2.0",

"method": "deleteIPFilter",

"params":{

"ip\_id": 1

},

"id": "10.12"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.12"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ip\_id | number |  | the count of record |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101201",

"message": "Delete ip filter failed. "

},

"id": "10.12"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101201 | Delete ip filter failed. | Delete ip filter failed. |

### editIPFilter

#### Request

{

"jsonrpc": "2.0",

"method": "editIPFilter",

"params":{

"list\_id": 0,

"lan\_ip": "192.168.1.1",

"lan\_port": "80",

"wan\_ip": "20.12.12.5",

"wan\_port": "85",

"ip\_protocol": 6,

"ip\_status": 0

},

"id": "10.13"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.13"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| list\_id | number |  | record id |
| lan\_ip | string |  |  |
| lan\_port | string |  |  |
| wan\_ip | string |  |  |
| wan\_port | string |  | End IP Address |
| ip\_protocol | number |  | 6: TCP  17: UDP  253: TCP&UDP |
| ip\_status | number |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101301",

"message": "Edit ip filter failed. "

},

"id": "10.13"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101301 | Edit ip filter failed. | Edit ip filter failed. |

### getFirewallSwitch

#### Request

{

"jsonrpc": "2.0",

"method": "getFirewallSwitch",

"params":{},

"id": "10.14"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"firewall\_status": 0,

"ipflt\_status": 0,

"wan\_ping\_status": 0

},

"id": "10.14"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| firewall\_status | number |  | 0: disable  1: enable |
| ipflt\_status | number |  | 0: disable  1: enable |
| wan\_ping\_status | number |  | 0: disable  1: enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101401",

"message": "Get Firewall Switch failed. "

},

"id": "10.14"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101401 | Get Firewall Switch failed. | Get Firewall Switch failed. |

### setFirewallSwitch

#### Request

{

"jsonrpc": "2.0",

"method": "setFirewallSwitch",

"params":{

"firewall\_status": 0,

"ipflt\_status": 0,

"wan\_ping\_status": 0

},

"id": "10.15"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| firewall\_status | number |  | 0: disable  1: enable |
| ipflt\_status | number |  | 0: disable  1: enable |
| wan\_ping\_status | number |  | 0: disable  1: enable |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.15"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101501",

"message": "Set Firewall Switch failed. "

},

"id": "10.15"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101501 | Set Firewall Switch failed. | Set Firewall Switch failed. |

### SetIPFilter————暂不支持

#### Request

{

"jsonrpc": "2.0",

"method": "SetIPFilter",

"params":{

"total\_num": 1,

"ipFilter\_list":[{

"list\_id": 0,

"lan\_ip": "192.168.1.1",

"lan\_port": "80",

"wan\_ip": "20.12.12.5",

"wan\_port": "85",

"ip\_protocol": 6,

"ip\_status": 0

},…

]

},

"id": "10.16"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.16"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| list\_id | number |  | record id |
| lan\_ip | string |  |  |
| lan\_port | string |  |  |
| wan\_ip | string |  |  |
| wan\_port | string |  | End IP Address |
| ip\_protocol | number |  | 6: TCP  17: UDP  253: TCP&UDP |
| ip\_status | number |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101601",

"message": "Set ip filter failed. "

},

"id": "10.13"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101601 | Set ip filter failed. | Set ip filter failed. |

### GetVPNPassthrough————暂不支持

#### Request

{

"jsonrpc": "2.0",

"method": " GetVPNPassthrough",

"params":{},

"id": "10.8"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

" status":1,

" vpn\_server ":"192.168.1.60"

},

"id": "10.17"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| status | number |  | 0: VPN pass-through off  1: VPN pass-through on |
| vpn\_server | string |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101701",

"message": "Get VPN pass-through Settings failed. "

},

"id": "10.17"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101701 | Get VPN pass-through Settings failed. | Get VPN pass-through Settings failed. |

### SetVPNPassthrough————暂不支持

#### Request

{

"jsonrpc": "2.0",

"method": " SetVPNPassthrough ",

"params":{

"status":1,

"vpn\_server":"192.168.1.60"

},

"id": "10.18"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| status | number |  | 0: VPN pass-through off  1: VPN pass-through on |
| vpn\_server | string |  |  |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.9"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101801",

"message": "Set VPN pass-through settings failed. "

},

"id": "10.18"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101801 | Set VPN pass-through settings failed. | Set VPN pass-through settings failed. |

### getUrlFilterSettings

#### Request

{

"jsonrpc": "2.0",

"method": "getUrlFilterSettings",

"params":{},

"id": "10.19"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"filter\_policy":1,

"UrlDenyList":["http://www.taobao.com",…]

},

"id": "10.19"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| filter\_policy | Integer | Yes | 0: Disable  1:allow  2:deny |
| UrlDenyList | Array | Yes | The Deny Url address. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "101901",

"message": "Get URL Filter Settings failed. "

},

"id": "10.19"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 101901 | Get URL Filter Settings failed. | Get URL Filter Settings failed. |

### SetUrlFilterSettings

#### Request

{

"jsonrpc": "2.0",

"method": "SetUrlFilterSettings",

"params":{

"filter\_policy":1,

"UrlDenyList":["http://www.taobao.com",…]

},

"id": "10.20"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| filter\_policy | Integer | Yes | 0: off  1:white  2:black |
| UrlDenyList | Array | Yes | The Deny Url address. |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.20"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "102001",

"message": "Set URL Filter settings failed. "

},

"id": "10.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 102001 | Set URL Filter settings failed. | Set URL Filter settings failed. |

### GetWanAccess

#### Request

{

"jsonrpc": "2.0",

"method": "GetWanAccess",

"params":{},

"id": "10.14"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"disableWanAcess": 1

},

"id": "10.21"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| disableWanAcess | number |  | 0:开启wan侧访问  1:关闭wan侧访问 |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "102101",

"message": "Get Wan Access setting failed. "

},

"id": "10.21"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 102101 | Get Wan Access setting failed. | Get Wan Access setting failed. |

### SetWanAccess

#### Request

{

"jsonrpc": "2.0",

"method": "SetWanAccess ",

"params":{

"disableWanAcess ": 0

},

"id": "10.22"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| disableWanAcess | number |  | 0:开启wan侧访问  1:关闭wan侧访问 |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "10.22"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "102201",

"message": "Set Wan Access setting failed. "

},

"id": "10.22"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 102201 | Set Wan Access setting failed. | Set Wan Access setting failed. |

* + 1. **GetBindToPcSettings**
       1. **Request**

{

"jsonrpc": "2.0",

"method": "GetBindToPcSettings",

"params":{},

"id": "10.23"

}

* + - 1. **Response**

{

"jsonrpc": "2.0",

"result":

{

"MacAllowList":["44:44:22:d5:21:24"]

},

"id": "10.23"

}

* + - 1. **Definition for each filed**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| MacAllowList | Array | Yes | The Allow Mac address. |

* + - 1. **Response with error**

{

"jsonrpc": "2.0",

"error": {

"code": "102301",

"message": "Get Bind To Pc Settingsfailed. "

},

"id": "10.23"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 100101 | Get Bind To Pc Settingsfailed | Get Bind To Pc Settingsfailed |

* + 1. **SetBindToPcSettings**
       1. **Request**

{

"jsonrpc": "2.0",

"method": " SetBindToPcSettings",

"params":{

"MacAllowList":["44:44:22:d5:21:24"]

},

"id": "10.24"

}

* + - 1. **Definition for each filed**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| MacAllowList | Array | Yes | The Allow Mac address. |

* + - 1. **Response**

{

"jsonrpc": "2.0",

"result":{},

"id": "10.24"

}

* + - 1. **Response with error**

{

"jsonrpc": "2.0",

"error": {

"code": "102401",

"message": "Set Bind To Pc Settingsfailed. "

},

"id": "10.24"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 102401 | Set Bind To Pc Settingsfailed | Set Bind To Pc Settingsfailed. |

## LAN

### GetLanSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetLanSettings",

"params":{},

"id": "11.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"DHCPLeaseTime":12，

"DHCPServerStatus":0，

"DNSAddress1":""，

"DNSAddress2":""，

"DNSMode":0，

"EndIPAddress":"192.168.1.99 "，

"host\_name": "plus10.home",

"IPv4IPAddress": "192.168.1.1"，

"MacAddress":"",

"StartIPAddress": "192.168.1.3"，

"SubnetMask":"255.255.255.0 "

},

"id": "11.1"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DHCPLeaseTime | integer | Yes | DHCP Lease Time (hours):1,6,12,24 |
| DHCPServerStatus | integer | Yes | DHCP server status  0:disable  1:enable |
| DNSAddress1 | string | Yes |  |
| DNSAddress2 | string | Yes |  |
| DNSMode | integer | Yes | DNS mode:  0:off  1:off |
| EndIPAddress | String | Yes | End IP Address |
| host\_name | String |  |  |
| IPv4IPAddress | String | Yes | IPv4 IP address, Gateway |
| MacAddress | string | No | The LAN MAC Address |
| StartIPAddress | String | Yes | Start IP Address |
| SubnetMask | String | Yes | Subnet Mask |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "110101",

"message": "Get LAN Settings failed."

},

"id":"11.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 110101 | Get LAN Settings failed. | Get LAN Settings failed. |

### SetLanSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetLanSettings",

"params":{

"IPv4IPAddress": "192.168.1.1",

"SubnetMask":"255.255.255.0",

"DHCPServerStatus":0,

"StartIPAddress": "192.168.1.3",

"EndIPAddress":"192.168.1.99",

"DHCPLeaseTime":12

},

"id": "11.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| IPv4IPAddress | String | Yes | IPv4 IP Address |
| SubnetMask | String | Yes | Subnet Mask |
| DHCPServerStatus | integer | Yes | DHCP Server Status  0:disable  1:enable |
| StartIPAddress | String | Yes | Start IP Address |
| EndIPAddress | String | Yes | End IP Address |
| DHCPLeaseTime | integer | Yes | DHCP Lease Time(Hours) |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "11.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "110201",

"message": "Set LAN settings failed."

},

"id":"11.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 110201 | Set LAN settings failed. | Set LAN settings failed. |

## ConnectionDevices

### GetConnectedDeviceList

#### Request

{

"jsonrpc": "2.0",

"method": "GetConnectedDeviceList",

"params":{},

"id": "12.1"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"ConnectedList":[

{

"id":1,

"DeviceName":"d1-orange-link",

"MacAddress":"98:FC:D2:99:EF",

"IPAddress":"192.168.1.100",

"AssociationTime":100,

"DeviceType":1,

"ConnectMode":0

},

……

]

},

"id": "12.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| id | Integer | No | The list element only flag. |
| DeviceName | string | No | The connected device name |
| MacAddress | String | No | The connected device MAC address. |
| IPAddress | string | No | The connected device IP address. |
| AssociationTime | Integer | No | (Option) The device connected times. (s) |
| DeviceType | Integer | No | 0: use the web UI and login device;  1: just connected to device but not login. |
| ConnectMode | Integer | No | 0:USB connect  1:WIFI connect |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "120101",

"message": "Get Connected device list failed."

},

"id":"12.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 120101 | Get Connected device list failed. | Get Connected device list failed. |

### GetBlockDeviceList

#### Request

{

"jsonrpc": "2.0",

"method": "GetBlockDeviceList",

"params":{},

"id": "12.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"BlockList":[{

"AssociationTime":100,

"ConnectMode":0,

"DeviceName":"d1-orange-link",

"DeviceType":1,

"id":1,

"IPAddress":"192.168.1.100",

"MacAddress":"98:FC:D2:99:EF"

},…

]

},

"id": "12.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| AssociationTime | Integer | No | (Option) The device connected times. (s) (Option) |
| ConnectMode | Integer | No | 0:USB connect  1:WIFI connect |
| DeviceName | string | No | The connected device name |
| DeviceType | Integer | No | This parameter as (Option).  0: use the web UI and login device;  1: just connected to device but not login. |
| id | Integer | No | The list element only flag. |
| IPAddress | string | No | The connected device IP address.(Option) |
| MacAddress | String | No | The connected device MAC address. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "120201",

"message": "Get block device list failed."

},

"id":"12.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 120201 | Get block device list failed. | Get block device list failed. |

### SetConnectedDeviceBlock

#### Request

{

"jsonrpc": "2.0",

"method": "SetConnectedDeviceBlock",

"params":{

"DeviceName":"d1-orange-link",

"MacAddress":"98:FC:D2:99:EF",

},

"id": "12.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DeviceName | string | No | The connected device name |
| MacAddress | String | No | The connected device MAC address. |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "12.3"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "120301",

"message": "Set device block failed."

},

"id": "12.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 120301 | Set device block failed. | Set device block failed. |
| 120302 | Device is not exist. | Device id it not exist. |

### SetDeviceUnblock

#### Request

{

"jsonrpc": "2.0",

"method": "SetDeviceUnlock",

"params":{

"DeviceName":"d1-orange-link",

"MacAddress":"98:FC:D2:99:EF",

},

"id": "12.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DeviceName | string | No | The connected device name |
| MacAddress | String | No | The connected device MAC address. |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "12.4"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "120401",

"message": "Set device unblock failed."

},

"id": "12.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 120401 | Set device unblock failed. | Set device unblock failed. |
| 120402 | Unblock device is not exist. | Unblock device is not exist. |

### SetDeviceName

#### POST request

{

"jsonrpc": "2.0",

"method": "SetDeviceName",

"params": {

"DeviceName":"d1-orange-link",

"MacAddress":"98:FC:D2:99:EF",

"DeviceType":1

},

"id": "12.5"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"12.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DeviceName | string | Yes | Then new connected device name. |
| MacAddress | String | No | The connected device MAC address. |
| DeviceType | Integer | No | 0:Block device  1:connected device |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "120501",

"message": "Set connected device name failed."

},

"id": "12.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 120501 | Set device name failed. | Set device name failed. |

## System Information Requests

### GetSystemInfo

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSystemInfo",

"params": {},

"id": "13.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"AppVersion":"",

"DeviceName":"y858",

"HttpApiVersion":"1.0",

"HwVersion":"ABCD",

"ICCID":"",

"IMEI":"123456789012",

"IMSI":"",

"MacAddress":"",

"MSISDN":"",

"MsisdnMark":0,

"sn":,

"SwVersion":"",

"WebUiVersion":""

},

"id":"13.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| AppVersion | String |  |  |
| DeviceName | String | No | Device name. |
| HttpApiVersion | String | No | Http API Version used by the device |
| HwVersion | String | No | Hardware Version of device |
| ICCID | String |  |  |
| IMEI | String | No | IMEI of device |
| IMSI | String | No |  |
| MacAddress | String | No | Mac address |
| MSISDN | String | No | My number |
| MsisdnMark | Integer |  | 0: Need to get my number  1: No need to get my number |
| sn |  |  |  |
| SwVersion | String | No | Firmware Version of device |
| WebUiVersion | String | No | Web ui version |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "130101",

"message": "Get items failed."

},

"id": "13.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130101 | Get items failed. | Get some or all sub-items failed. |

### SetLanguage

#### Post Request:

{

"jsonrpc": "2.0",

"method": "SetLanguage",

"params":

{

"Language":"en"

},

"id": "13.2"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"13.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| Language | String | No | en,fr,jp… |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"130201",

"message ":"set current language setting fail. ",

},

"id":"13.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130201 | Set current language setting fail. | Set current language setting fail. |

### GetCurrentLanguage

#### Post Request:

{

"jsonrpc": "2.0",

"method": "GetCurrentLanguage",

"params": {},

"id": "13.3"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {

"Language":"en",

},

"id":"13.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| Language | String | Yes | en,fr,jp… |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"130301",

"message ":"Get current language setting fail ",

},

"id":"13.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130301 | Get currentlanguage settingfail. | Get current language setting fail. |

### GetSystemStatus

#### Post Request:

{

"jsonrpc": "2.0",

"method": "GetSystemStatus",

"params":{},

"id": "13.4"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {

"bat\_cap":50,

“bat\_level”:1,

"chg\_state":1,

"ConnectionStatus":1,

"Conprofileerror"：0,

"curr\_num": 4,

"Domestic\_Roaming"：1,

"NetworkName": "CMCC",

"NetworkType":1,

"Roaming": 0,

"SignalStrength":1,

"SmsState":1,

"TotalConnNum": 5,

"WlanState":1

},

"id":"13.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| bat\_cap | Integer | Yes | Current battery capacity  0~100 |
| bat\_level | Integer | Yes | 0: Grade 0  1: Grade 1  2: Grade 2  3: Grade 3  4: Grade 4  5: low volt |
| chg\_state | Integer | Yes | 0: Charging  1: Charge done  2: Charge remove  3: error occur  4: Charging out for Power Bank |
| ConnectionStatus | Integer | Yes | 0: disconnected  1: connecting  2:connected  3:disconnecting |
| Conprofileerror | Integer |  |  |
| curr\_num | Integer | Yes | Current WIFI client |
| Domestic\_Roaming | Integer |  |  |
| NetworkName | String |  | operator name |
| NetworkType | Integer | Yes | Network Type  0: NO SERVER  1: GPRS  2: EDGE  3: HSPA  4: HSUPA  5: UMTS  6: HSPA+  7: DC-HSPA+  8: LTE  9: LTE+  10:WCDMA  11:CDMA  12:GSM |
| Roaming |  |  | 0: Roaming  1: no |
| SignalStrength | Integer | Yes | RSSI value:  -1: No service  0: level 0  1: level 1  2: level 2  3: level 3  4: level 4 |
| SmsState | Integer | Yes | 0: no SMS 1: SMS full 2: Noral 3: New SMS |
| TotalConnNum | Integer |  | The total connected device number, USB and WiFi |
| WlanState | Integer | Yes | Wlan status  0: off  1: on  2:WPS |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"130401",

"message ":"Get system Status fail.",

},

"id":"13.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130401 | Get system Status fail. | Get system Status fail. |

### SetDeviceReboot

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetDeviceReboot",

"params":{},

"id": "13.5"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "13.5"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "130501",

"message": "Set device reboot failed."

},

"id":"13.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130501 | Set device reboot failed. | Set device reboot failed. |

### SetDeviceReset

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetDeviceReset",

"params":{},

"id": "13.6" 实际为13.5,与上一节重复，应改为13.6。

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "13.6"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "130601",

"message": "Set device reset failed."

},

"id":"13.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130601 | Set device reset failed. | Set device reset failed. |

### SetDeviceBackup

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetDeviceBackup",

"params":{},

"id": "13.7" 实际显示为13.5，应改为13.7。

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "13.7"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "130701",

"message": "Set device back up failed."

},

"id":"13.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130701 | Set device back up failed. | Set device back up failed. |

Note: after that API complete success, we should get the backup file from the path: /cfgbak/configure.bin

### 5.13.8 SetDeviceRestore————已取消使用

#### 5.13.8.1 POST Request

{

"jsonrpc": "2.0",

"method": "SetDeviceRestore",

"params":{

"filename ": " "

},

"id": "13.8"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| filename | String | Yes | The upload file name. |

#### Response

1. {
2. "jsonrpc": "2.0",
3. "result": {},
4. "id": "13.8"，
5. }

#### 5.13.8.3 Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "130801",

"message": "Set device restore failed."

},

"id":"13.8"

}



|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130801 | Set device restore failed. | Set device restore failed. |

### SetDevicePowerOff

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetDevicePowerOff",

"params":{},

"id": "13.9"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "13.9"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "130901",

"message": "Set device power off failed."

},

"id":"13.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 130901 | Set device power off failed. | Set device power off failed. |

### SetAppBackup——————非UI接口

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetAppBackup",

"params":{},

"id": "13.10"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "13.10"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "131001",

"message": "Set APP backup failed."

},

"id":"13.10"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131001 | Set APP backup failed. | Set APP backup failed. |

### SetAppRestoreBackup——————非UI接口

#### 5.13.11.1 POST Request

{

"jsonrpc": "2.0",

"method": "SetAppRestoreBackup",

"params":{},

"id": "13.11"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"RestoreErr": 0，

},

"id": "13.11"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| RestoreErr | Integer | Yes | 0: successful  1: no backup file  2: other error |

#### 5.13.11.3 Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "131101",

"message": "Set APP restore failed."

},

"id":"13.11"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131101 | Set APP restore failed. | Set APP restore failed. |

### GetWizardStatus——————非UI接口

This function first for W800 Yota. And this function to show if the Wizard Settings should show when first use the devices, and this value should be set as default after restore.

#### 5.13.12.1Post Request:

{

"jsonrpc": "2.0",

"method": "GetWizardStatus",

"params": {},

"id": "13.12"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {

"setWizard":1

},

"id":"13.12"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| setWizard | String | No | The Operation of set wizard.  0:disable  1:enable |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"131201",

"message ":"Get Wizard status fail. ",

},

"id":"13.12"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131201 | Get Wizard status fail. | Get Wizard status fail. |

### SetWizardStatus——————非UI接口

#### Post Request:

{

"jsonrpc": "2.0",

"method": "SetWizardStatus",

"params": {"setWizard":0},

"id": "13.13"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"13.13"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| setWizard | integer | No | This vaule must be 0; |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"131301",

"message ":"Set Wizard status failed. ",

},

"id":"13.13"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131301 | Set Wizard status failed. | Set Wizard status failed. |

### GetRedirectSwitch——————非UI接口

#### Post Request:

{

"jsonrpc": "2.0",

"method": "GetRedirectSwitch",

"params": {},

"id": "13.14"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {

"redirect\_switch":1

},

"id":"13.14"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| redirect\_switch | Integer | No | The DNS redirect status.  0:disable  1:enable |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"131401",

"message ":"Get Redirect switch fail. ",

},

"id":"13.14"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131401 | Get Redirect switch fail. | Get Redirect switch fail. |

### SetRedirectSwitch——————非UI接口

#### Post Request:

{

"jsonrpc": "2.0",

"method": "SetRedirectSwitch",

"params": {"redirect\_switch":0},

"id": "13.15"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"13.15"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| redirect\_switch | integer | Yes | 0:disable  1:enable |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"131501",

"message ":"Set DNS Redirect switch failed. "

},

"id":"13.15"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131501 | Set DNS Redirect switch failed. | Set DNS Redirect switch failed. |

### GetUpnpSettings

#### Request:

{

"jsonrpc": "2.0",

"method": "GetUpnpSettings",

"params": {},

"id": "13.16"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {

"upnp\_switch":1

},

"id":"13.16"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| upnp\_switch | Integer | No | 0:disable  1:enable |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"1316401",

"message ":"Get UPNP settings fail. ",

},

"id":"13.16"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131601 | Get UPNP settings fail. | Get UPNP settings fail. |

### SetUpnpSettings

#### Post Request:

{

"jsonrpc": "2.0",

"method": "SetUpnpSettings",

"params": {"upnp\_switch":0},

"id": "13.17"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"13.17"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| upnp\_switch | integer | Yes | 0:disable  1:enable |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"131701",

"message ":"Set UPNP settings failed. "

},

"id":"13.17"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131701 | Set UPNP settings failed. | Set UPNP settings failed. |

### GetDebugInfo——————未实现接口

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetDebugInfo",

"params": {},

"id": "13.18"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"MSID":"",

"MEID":"",

"UICC":"",

"PRLVersion":"",

"ActiveDate":"",

"ActiveMIPProfile":"",

"NAI":"",

"RefurbishDate":"",

"LTEAPINI":"",

"LifeTimeTotalTransferred":"",

"QLIC":"",

"LastError":"",

"FreeMemory":6546,

"TotalMemory":65466

},

"id":"13.18"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| MSID | String | No | Device MSID |
| MEID | String | No | Only allow to view the MEID. Display both  HEX and Decimal values |
| UICC | String | No |  |
| PRLVersion | String | No | Current Device PRL |
| ActiveDate | String | No |  |
| ActiveMIPProfile | String | No |  |
| NAI | String | No |  |
| RefurbishDate | String | No |  |
| LTEAPINI | String | No |  |
| LifeTimeTotalTransferred | String | No |  |
| QLIC | String | No | States wither QLIC is supported and currently enabled or disabled. |
| LastError | String | No | Include the last received MIP error code |
| FreeMemory | Integer | No | Display the free memory values of the device's RAM in MB |
| TotalMemory | Integer | No | Display the total memory values of the device's RAM in MB |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "130101",

"message": "Get items failed."

},

"id": "13.18"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 131801 | Get debug info failed. | Get some or all sub-items failed. |

### uploadBackupSettings

#### URL:/goform/uploadBackupSettings

#### POST Request

Content-Disposition: form-data;

name="iptUpload";

filename="configure.bin"

Content-Type: application/octet-stream

#### Response

{

"error": "0"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| error | number |  | Error code  0:success  1: failed  2: call API wrong  3: UI parameter wrong  4: data too long, no memory to use |

### GetHfaActivationSatus——————未实现接口

#### Request:

{

"jsonrpc": "2.0",

"method": "GetHfaActivationStatus",

"params": {},

"id": "13.20"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {

"LoginState ":0,

"HfaActivationStatus":0,

"HfaRetryWaitSec":60,

"HfaRetryTotalNum":5,

"HfaRetryTime":0

},

"id":"13.20"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| LoginState | Integer | No | 0: logout  1: login  2: the login times used out. |
| HfaActivationStatus | Integer | No | 0:None  1:Contacting network  2:waiting for retry  3:activated  4:failed  5:Cancel  6:Check PRL Update  7: PRL Upgraded  8: No PRL Update available  9:Check For firmware Update  10: No Firmware Update available  11:canceling  12: completed |
| HfaRetryWaitSec | Integer | No | Next retry left time |
| HfaRetryTotalNum | Integer | No | Total HFA retry number |
| HfaRetryTime | Integer | No | Current HFA retry times |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"132001",

"message ":"Get HFA Activation Status fail. ",

},

"id":"13.20"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 132001 | Get UPNP settings fail. | Get HFA Activation Status fail |

### CancelHfaActivation——————未实现接口

#### Post Request:

{

"jsonrpc": "2.0",

"method": "CancelHfaActivation",

"params": {},

"id": "13.21"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"13.21"

}

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"132101",

"message ":"Cancel Hfa Activation failed. "

},

"id":"13.21"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 132001 | Cancel Hfa Activation failed | Cancel Hfa Activation failed |

### getNewPageFlagSemaphore

#### Request:

{

"jsonrpc": "2.0",

"method": "getNewPageFlagSemaphore",

"params": {},

"id": "13.22"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {

"flag\_state":0

},

"id":"13.22"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| flag\_state | Integer | No | 0(default is 0 and when device is reboot need to set the flag\_state to 0) |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"1322401",

"message ":"Get open new page flag fail. ",

},

"id":"13.22"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 132201 | Get open new page flag fail. | Get open new page flag fail. |

### setNewPageFlagSemaphore

#### Post Request:

{

"jsonrpc": "2.0",

"method": "setNewPageFlagSemaphore",

"params": {"flag\_state ":0},

"id": "13.23"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"13.23"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| flag\_state | integer | Yes | 0(default is 0 and when device is reboot need to set the flag\_state to 0) |

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"132301",

"message ":"Set open new page flag fail. "

},

"id":"13.23"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 132301 | Set open new page flag fail. | Set open new page flag fail. |

### SendPingURL——————Firmware端未实现接口

#### Response

{

"jsonrpc": "2.0",

"method": " SendPingURL",

"result": {}

"id": "13.24"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "132401",

"message": "Send Ping failed."

},

"id":"13.24"}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 132401 | Send Ping failed. | Send Ping failed. |
| 132402 | Ping operation fails. | Ping operation fails. |

### SetMyNumber——————Firmware端未实现接口

#### Request

{

"jsonrpc": "2.0",

"method": "SetMyNumber",

"params":{

"MSISDN":,

"MsisdnMark":1

} ,

"id": "13.25"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "13.25"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| msisdn | string |  | My number |
| MsisdnMark | Integer |  | 0: Need to get my number  1: No need to get my number |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "132501",

"message": "Set my number failed. "

},

"id": "13.25"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 132501 | Set my number failed. | Set my number failed. |

## Sharing

### GetDLNASettings——————Firmware端未实现接口

#### 5.14.1.1 POST Request

{

"jsonrpc": "2.0",

"method": "GetDLNASettings",

"params":{},

"id": "14.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"DlnaStatus": 0,

"DlnaName":"Y858"

},

"id": "14.1"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| DlnaStatus | Integer | Yes | 0: disable;  1: enable |
| DlnaName | String | Yes | The device name |

#### 5.14.1.3Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140101",

"message": "Get DLNA Settings failed."

},

"id":"14.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140101 | Get DLNA Settings failed. | Get DLNA Settings failed. |

### SetDLNASettings——————Firmware端未实现接口

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetDLNASettings",

"params":{

"DlnaStatus": 0,

"DlnaName":"P10KB"

},

"id": "14.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "14.2"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140201",

"message": "Set DLNA Settings failed."

},

"id":"14.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140201 | Set DLNA Settings failed. | Set DLNA Settings failed. |

### GetSambaSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSambaStatus",

"params":{},

"id": "14.3"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"SambaStatus": 0

},

"id": "14.3"

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SambaStatus | Integer | No | 0: disable  1: enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140301",

"message": " Get Samba Settings failed."

},

"id":"14.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140301 | Get Samba Settings failed. | Get Samba Settings failed. |

### SetSambaSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetSambaStatus",

"params":{

"SambaStatus":1,

},

"id": "14.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "14.4"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140401",

"message": " Set Samba Settings failed."

},

"id":"14.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140401 | Set Samba Settings failed. | Set Samba Settings failed. |

### GetFtpSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetFtpStatus",

"params":{},

"id": "14.5"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"FtpStatus": 0

},

"id": "14.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| FtpStatus | Integer | No | 0: disable  1: enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140501",

"message": "Set Ftp Settings failed."

},

"id":"14.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140501 | Get Ftp Settings failed. | Get Ftp Settings failed. |

### SetFtpSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetFtpStatus",

"params":{

"FtpStatus":1,

},

"id": "14.6"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "14.6"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140601",

"message": " Set Ftp Settings failed."

},

"id":"14.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140601 | Set Ftp Settings failed. | Set Ftp Settings failed. |

### GetSDCardSpace

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSDCardSpace",

"params":{},

"id": "14.7"

}

#### Response

{

"jsonrpc": "2.0",

"result":{

"TotalSpace": 645656,

"UsedSpace": 6456

},

"id": "14.7"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| TotalSpace | String | No | The unit is (GB) |
| UsedSpace | String | No | The unit is (GB) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140701",

"message": "Get SD Card Space failed."

},

"id":"14.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140701 | Get SD Card Space failed. | Get SD Card Space failed. |

### GetSDFileList

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSDFileList",

"params":{

"Path": "/",

"Page":1

},

"id": "14.8"

}

#### Response

{

"jsonrpc": "2.0",

"result":{

"FileList":[

{

"FileName":"",

"FileType":1,

"CreateTime":"2014-01-12",

"Size":55545,

},

……

]

"Page":1,

"TotalPage":1,

"Path":"/"

},

"id": "14.8"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| Path | string | Yes | The path of the SD files. |
| FileName | String | No | The name of the files that under the current path. |
| FileType | Integer | No | 0: file;  1:folder |
| CreateTime | string | No | The time to create the file or folder |
| Size | Integer | No | The size of the file, the folder’size is 0.  The unit is KB. |
| Page | Integer | Yes | Every page have 20 files, web UI get the files one page by one page; |
| TotalPage | Integer | No | The total page that under the path. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140801",

"message": "Set SD Card settings failed."

},

"id":"14.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140801 | Set SD Card settings failed. | Set SD Card settings failed. |

### GetSDcardStatus

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSDcardStatus",

"params":{},

"id": "14.9"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"SDcardStatus": 1

},

"id": "14.9"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SDcardStatus | Integer | No | 0: no Sdcard device  1: Sdcard device |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "140901",

"message": "Get SD Card status failed."

},

"id":"14.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 140901 | Get SD Card status failed. | Get SD Card status failed. |

### 5.14.10 GetUsbcardSettings

#### 5.14.10.1 POST Request

{

"jsonrpc": "2.0",

"method": "GetUsbcardStatus",

"params":{},

"id": "14.10"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"UsbcardStatus": 1

},

"id": "14.10"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| UsbcardStatus | Integer | No | 0: disable  1: enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "141001",

"message": "Get Usbcard Settings failed."

},

"id":"14.10"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 141001 | Get Usbcard Settings failed. | Get Usbcard Settings failed. |

### SetUsbcardSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetUsbcardStatus",

"params":{

"UsbcardStatus":1,

},

"id": "14.11"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "14.11"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "141101",

"message": " Set Ftp Settings failed."

},

"id":"14.11"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 141101 | Set Usbcard Settings failed. | Set Usbcard Settings failed. |

## Profile

### GetProfileList

#### 5.15.1.1 POST Request

{

"jsonrpc": "2.0",

"method": "GetProfileList",

"params":{},

"id": "15.1"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"ProfileList " ：[

{

"ProfileID":1,

"ProfileName":"China Unicom",

"APN":"3gnet",

"Password":"1234",

"AuthType":0,

"DailNumber":"\*99#",

"UserName":"china",

"Default":1,

"IsPredefine":1,

"IPAdrress":"",

"PdpType":1

},….

]，

"data\_len": 1

},

"id": "15.1"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ProfileID | Integer | Yes | Profile index |
| ProfileName | String | Yes | Profile name |
| APN | String | Yes | APN name |
| Password | String | Yes | password |
| AuthType | Integer | Yes | Auth type:  0: None  1: PAP  2: CHAP  3: PAP& CHAP |
| DailNumber | String | Yes | Dail number |
| UserName | String | Yes | User name |
| Default | Integer | Yes | Current Default profile:  0:not Default  1:Default |
| IsPredefine | Integer | Yes | Is predefine or user create profile:  0: Predefine profile  1: User Createprofile |
| IPAddress | string | Yes | If empty the dynamic IP, else static IP. |
| PdpType | Integer | Yes | 0:IPV4  1:IPV6  2:IPV4V6  (Only for sprint is different)  1:IPV4  2:IPV6  3:IPV4V6 |
| data\_len |  |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"150101",

"message": "Get profile listfailed. "

},

"id": "15.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 150101 | Get profile listfailed. | Get profile listfailed. |

### AddNewProfile

#### Request

{

"jsonrpc": "2.0",

"method": "AddNewProfile",

"params":

{

"ProfileName":"China Unicom",

"APN":"3gnet",

"UserName":"china",

"Password":"1234",

"AuthType":0,

"DailNumber":"\*99#",

"IPAdrress":"",

"PdpType":1

},

"id": "15.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "15.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ProfileName | String | Yes | Profile name |
| APN | String | Yes | APN name |
| UserName | String | Yes | User name |
| Password | String | Yes | password |
| AuthType | Integer | Yes | Auth type:  0: None  1: PAP  2: CHAP  3: PAP& CHAP |
| DailNumber | String | Yes | Dail number |
| IPAddress | string | Yes | If empty the dynamic IP, else static IP. |
| PdpType | Integer | Yes | 0:IPV4  1:IPV6  2:IPV4V6  (Only for sprint is different)  1:IPV4  2:IPV6  3:IPV4V6 |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "150201",

"message": "Add newprofile failed. "

},

"id": "15.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 150201 | Add newprofile failed. | Add newprofile failed. |

### EditProfile

#### Request

{

"jsonrpc": "2.0",

"method": "EditProfile",

"params":

{

"ProfileID":1,

"ProfileName":"China Unicom",

"APN":"3gnet",

"UserName":"china",

"Password":"1234",

"AuthType":"",

"DailNumber":"\*99#",

"IPAdrress":"",

"PdpType":1

},

"id": "15.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "15.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ProfileID | Integer | Yes | Profile id must in the profile list. |
| ProfileName | String | Yes | Profile name |
| APN | String | Yes | APN name |
| UserName | String | Yes | User name |
| Password | String | Yes | password |
| AuthType | Integer | Yes | Auth type:  0: None  1: PAP  2: CHAP  3: PAP& CHAP |
| DailNumber | String | Yes | Dail number |
| IPAddress | string | Yes | If empty the dynamic IP, else static IP. |
| PdpType | Integer | Yes | 0:IPV4  1:IPV6  2:IPV4V6  (Only for sprint is different)  1:IPV4  2:IPV6  3:IPV4V6 |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "150301",

"message": "Edit profile failed. "

},

"id": "15.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 150301 | Edit profile failed. | Edit profile failed. |
| 150302 | Profile is not exit. | Profile is not exit. |

### DeleteProfile

#### Request

{

"jsonrpc": "2.0",

"method": "DeleteProfile",

"params":

{

"ProfileID":1

},

"id": "15.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "15.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ProfileID | Integer | Yes | Profile id must in the profile list. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"150401",

"message": "Delete profile failed. "

},

"id": "15.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 150401 | Delete profile failed. | Delete profile failed. |
| 150402 | Profile is not exist. | Profile is not exist. |

### SetDefaultProfile

#### Request

{

"jsonrpc": "2.0",

"method": "SetDefaultProfile",

"params":

{

"ProfileID":1

},

"id": "15.5"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "15.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ProfileID | Integer | Yes | Profile id |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "150501",

"message": "Set default profile failed."

},

"id": "15.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 150501 | Set default profile failed. | Set default profile failed. |
| 150502 | Profile is not exist. | Profile is not exist. |

### getCurrentProfile

#### Request

{

"jsonrpc": "2.0",

"method": "getCurrentProfile",

"params": {},

"id": "15.6"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"ProfileID": 1,

"ProfileName": "China Unicom",

"APN": "3GNET",

"Password": "",

"AuthType": -1,

"DailNumber": "\*99#",

"UserName": "",

"IsPredefine": 0,

"PdpType": -1,

"IPAdrress": ""

},

"id": "15.6"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| ProfileID | Integer | Yes | Profile index |
| ProfileName | String | Yes | Profile name |
| APN | String | Yes | APN name |
| Password | String | Yes | password |
| AuthType | Integer | Yes | Auth type:  0: None  1: PAP  2: CHAP  3: PAP& CHAP |
| DailNumber | String | Yes | Dail number |
| UserName | String | Yes | User name |
| IsPredefine | Integer | Yes | Is predefine or user create profile:  0: Predefine profile  1: User Createprofile |
| IPAddress | string | Yes | If empty the dynamic IP, else static IP. |
| PdpType | Integer | Yes | 0:IPV4  1:IPV6  2:IPV4V6  (Only for sprint is different)  1:IPV4  2:IPV6  3:IPV4V6 |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "150601",

"message": "get connected profile error"

},

"id": "15.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 150601 | get connected profile error | get connected profile error |

## Power Management

### GetBatteryState

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetBatteryState",

"params":{},

"id": "16.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"chg\_state": 0,

"bat\_cap": 100,

"bat\_level": 4,

"BatteryLevel": 50

},

"id": "16.1"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| chg\_state | Integer | No | 0: Charging  1: Charge done  2: Charge remove  3: error occur  4: Charging out for Power Bank |
| bat\_cap | Integer |  |  |
| bat\_level | Integer |  |  |
| BatteryLevel | Integer | No | BatteryLevel (Min: 1 Max:100) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "160101",

"message": "Get battery state failed."

},

"id":"16.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 160101 | Get battery state failed. | Get battery state failed. |

### GetPowerSavingMode

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetPowerSavingMode",

"params":{},

"id": "16.2"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"SmartMode": 0,

"WiFiMode": 1, /\*[Optional]\*/

"ConnAutoOff": 1

},

"id": "16.2"，

}

Note: If only support smart mode, the optional paramers are not required.

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SmartMode | Integer | Yes | 0: disable;  1: enable |
| WiFiMode | Integer | Yes | 0: disable;  1: enable |
| ConnAutoOff | Integer |  |  |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "160201",

"message": "Get power saving mode failed."

},

"id":"16.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 160201 | Get power saving mode failed. | Get power saving mode failed. |

### SetPowerSavingMode

#### POST Request

{

"jsonrpc": "2.0",

"method": " SetPowerSavingMode",

"params":{

"SmartMode": 0,

"WiFiMode": 1 /\*[Optional]\*/

},

"id": "16.3"

}

Note: If only support smart mode, the optional paramers are not required.

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SmartMode | Integer | Yes | 0: disable;  1: enable |
| WiFiMode | Integer | Yes | 0: disable;  1: enable |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "16.3"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "160301",

"message": "Set power saving mode failed."

},

"id":"16.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 160301 | Set power saving mode failed. | Set power saving mode failed. |

## WIFI Extender——————未实现接口

### GetWIFIExtenderSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetWIFIExtenderSettings",

"params":{},

"id": "17.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"StationEnable": 0

},

"id": "16.1"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| StationEnable | Integer | No | 0:disable  1:enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "170101",

"message": "Get WIFI extender settings failed."

},

"id":"17.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170101 | Get WIFI extender settings failed. | Get WIFI extender settings failed. |

### SetWIFIExtenderSettings

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetWIFIExtenderSettings",

"params":{},

"id": "17.2"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"StationEnable": 0

},

"id": "17.2"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| StationEnable | Integer | Yes | 0: disable;  1: enable |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "170201",

"message": "Get WIFI extender settings failed."

},

"id":"17.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170201 | Get WIFI extender settings failed. | Get WIFI extender settings failed. |
| 170202 | WPS is on. | When WPS is on, can not enable. |

### GetWIFIExtenderCurrentStatus

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetWIFIExtenderCurrentStatus ",

"params":{},

"id": "17.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":{

"HotspotConnectStatus": 0,

"HotspotId": 1,

"HotspotSSID":"",

"UploadRate":43242, /\*[Optional]\*/

"DownloadRate":423424 /\*[Optional]\*/

},

"id": "17.3"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| HotspotConnectStatus | Integer | Yes | Is there hotspot in on;  0: disconnected  1: connected |
| HotspotId | Integer | Yes | The only id for single hotspot in the hotspot list.  From 0 to max value; |
| HotspotSSID | String | Yes | The connected hotspot SSID, if no hotspot connected, this value is "". |
| UploadRate | Integer | Yes | The upload rate of the hotspot. (the unit is second.) |
| DownloadRate | Integer | Yes | The download rate of the hotspot. (the unit is second.) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "170301",

"message": "Get current status failed."

},

"id":"17.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170301 | Get current status failed. | Get current status failed. |

### GetHotspotList

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetHotspotList",

"params":{},

"id": "17.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"HotspotList " ：[

{

"HotspotId":1,

"ConnectState ":"",

"SSID ":0,

" Signal ":"",

"SecurityMode ":0,

"EncryptionType ":0,

"Key":""

},….

]

},

"id": "17.4"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| HotspotId | Integer | No | The only flag that stored hotspot list info. |
| ConnectState | String | Yes | The hotspot current status  0:disconnect  1:connected |
| Signal | String | Yes | Current signal of the hotspot.  The signal of searched hotspot.  0: no signal;  1:  2:  3:  4:  5: full |
| SecurityMode | Integer | Yes | 0:No Security;  1:WEP；  2:WPA  3: WPA2  4: WPA/WPA2 |
| EncryptionType | Integer | Yes | The WEP/WPA/WPA2 encryption type.  0:open  1:share  2:AES  3:TKIP |
| Key | Integer | Yes | The password store for the hotspot, if the hotspot has no security, this key ="". |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"170401",

"message": "Get hotspot listfailed. "

},

"id": "17.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170401 | Get hotspot list failed. | Get hotspot list failed. |

### EditHotspotInfo

#### Request

{

"jsonrpc": "2.0",

"method": "EditHotspotInfo",

"params":{

"HotspotId":1,

"SSID ":0,

"SecurityMode ":0,

"EncryptionType ":0,

"Key":""

},

"id": "17.5"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "17.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| HotspotId | Integer | No | The only flag that the hotspot that to edit. |
| SSID | string | No | The SSID of the hotspot that to edit. |
| SecurityMode | Integer | Yes | 0:No Security;  1:WEP；  2:WPA  3: WPA2  4: WPA/WPA2 |
| EncryptionType | Integer | Yes | The WEP/WPA/WPA2 encryption type.  0:open  1:share  2:AES  3:TKIP |
| Key | Integer | Yes | The password store for the hotspot, if the hotspot has no security, this key ="". |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "170501",

"message": "Edit hotspot failed. "

},

"id": "17.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170501 | Edit hotspot failed. | Edit hotspot failed. |
| 170502 | hotspot is not exit. | hotspot is not exit. |
| 170503 | This hotspot is working, can not edit. | This hotspot is working, can not edit. |

### DeleteHotspot

#### Request

{

"jsonrpc": "2.0",

"method": "DeleteHotspot",

"params":

{

"HotspotId":1,

},

"id": "17.6"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "17.6"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| HotspotId | Integer | No | HotspotId id must in the store hotspot list. |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"170601",

"message": "Delete hotspot failed. "

},

"id": " 17.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170601 | Delete hotspot failed. | Delete hotspot failed. |
| 170602 | hotspot is not exist. | hotspot is not exist. |
| 170603 | This hotspot is working, can not delete. | This hotspot is working, can not delete. |

### AddHotspot

#### Request

{

"jsonrpc": "2.0",

"method": "AddHotspot",

"params":

{

"SSID":"",

"SecurityMode":0,

"EncryptionType":0,

"Key":""

},

"id": "17.7"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "17.7"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SSID | string | No | The SSID of the hotspot that to add. |
| SecurityMode | Integer | Yes | 0:No Security;  1:WEP；  2:WPA  3: WPA2  4: WPA/WPA2 |
| EncryptionType | Integer | Yes | The WEP/WPA/WPA2 encryption type.  0:open  1:share  2:AES  3:TKIP |
| Key | Integer | Yes | The password store for the hotspot, if the hotspot has no security, this key ="". |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "170701",

"message": "Add hotspot failed. "

},

"id": "17.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170701 | Add hotspot failed. | Add hotspot failed. |

### SearchHotspot

#### POST Request

{

"jsonrpc": "2.0",

"method": "SearchHotspot",

"params":{},

"id": "17.8"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id":"17.8"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "170801",

"message": "Search hotspot failed."

},

"id": "17.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170801 | Search hotspot failed. | Search hotspot failed. |

### GetSearchHotspotList

#### POST Request

{

"jsonrpc": "2.0",

"method": "GetSearchHotspotList",

"params":{},

"id": "17.9"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"SearchState":0,

"SearchedHotspotList":[{"ConnectState":0,"SSID":"","Signal":0,"SecurityMode":0,"EncryptionType":0},...]

},

"id":"17.9"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SearchedHotspotList | Array | No | The hotspot list that searched. |
| SearchState | Integer | Yes | Search result:  0:no search  1:searching  2:success  3:failed |
| ConnectState | Integer | Yes | This hotspot is disconnected or connected.  0:disconnected  1: connected |
| SSID | Integer | Yes | 0: network unknown  1: network available  2: current register network  3: forbidden network |
| Signal | Integer | Yes | The signal of searched hotspot.  0: no signal;  1:  2:  3:  4:  5: full |
| SecurityMode | Integer | Yes | 0:No Security;  1:WEP；  2:WPA  3: WPA2  4: WPA/WPA2 |
| EncryptionType | Integer | Yes | The WEP/WPA/WPA2 encryption type.  0:open  1:share  2:AES  3:TKIP |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "170901",

"message": "Get search hotspot list failed."

},

"id": "17.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 170901 | Get search hotspot list failed. | Get search hotspot list failed. |

### ConnectHotspot

#### POST Request

{

"jsonrpc": "2.0",

"method": "ConnectHotspot",

"params":{

"HotspotId":1

},

"id": "17.10"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| HotspotId | Integer | No | Hotspot ID that want to connect. |

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "17.10"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "171001",

"message": "Connect hotspot failed. "

},

"id": "17.10"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 171001 | Connect hotspot failed. | Connect hotspot failed. |

### DisconnectHotspot

#### POST Request

{

"jsonrpc": "2.0",

"method": "DisConnectHotspot",

"params":{},

"id": "17.11"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "17.11"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"171101",

"message": "Disconnect hotspot failed."

},

"id": "17.11"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 171101 | Disconnect hotspot failed. | Disconnect hotspot failed. |
| 171102 | Not hotspot connect. | The hotspot is not connected. |

## OMA——————未实现接口

### SetPRLUpdate

#### POST Request

{

"jsonrpc": "2.0",

"method": "SetPRLUpdate",

"params":{},

"id": "18.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "11.1"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "180101",

"message": "Set PRL update failed."

},

"id":"18.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180101 | Set PRL update failed. | Set PRL update failed. |

### GetPRLUpdateProcess

#### Request

{

"jsonrpc": "2.0",

"method": "GetPRLUpdateProcess",

"params":{},

"id": "18.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"PRL\_update\_status":1,

},

"id": "18.2"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| PRL\_update\_status | Integer | No | 0: None  1:Checking  2:success  3: failed  4:Canceling  5:Cancel completed |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"180201",

"message": "Get PRL update process failed."

},

"id":"18.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180201 | Get PRL update process failed. | Get PRL update process failed. |

### SetPRLUpdateCancel

#### Request

{

"jsonrpc": "2.0",

"method": "SetPRLUpdateCancel",

"params":{},

"id": "18.3"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "18.3"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"180301",

"message": "Set PRL update cancel failed."

},

"id":"18.3"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180301 | Set PRL update cancel failed. | Set PRL update cancel failed. |

### SetProfileUpdate

#### Request

{

"jsonrpc": "2.0",

"method": "SetProfileUpdate",

"params":{},

"id": "18.4"

}

#### Response

{

"jsonrpc": "2.0",

"result":{},

"id": "18.4"

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "180401",

"message": "Set profile update failed."

},

"id":"18.4"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180401 | Set profile update failed. | Set profile update failed. |

### GetProfileUpdateProcess

#### Request

{

"jsonrpc": "2.0",

"method": "GetProfileUpdateProcess",

"params":{},

"id": "18.5"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"profile\_update\_status":1,

},

"id": "18.5"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| profile\_update\_status | Integer | No | 0:None  1:updating, executing update settings  2: Completed update settings  3：Not Completed update setings  4:Canceling  5:Cancel completed |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code":"180501",

"message": "Get profile update process failed."

},

"id":"18.5"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180501 | Get profile update process failed. | Get profile update process failed. |

### SetProfileUpdateCancel

#### Post Request:

{

"jsonrpc": "2.0",

"method": "SetProfileUpdateCancel",

"params": {},

"id": "18.6"

}

#### Response:

{

"jsonrpc": "2.0",

"result": {},

"id":"18.6"

}

#### Response with error

{

"jsonrpc": "2.0",

"error ": {

"code":"180601",

"message ":"Set Profile Update Cancel failed. ",

},

"id":"18.6"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180601 | Set Profile Update Cancel failed. | Set Profile Update Cancel failed. |

### SetFactoryReset

This operation, also called Reverse Logistics, will display provisioning information and reset

the modem to the factory default values.

#### Request

{

"jsonrpc": "2.0",

"method": "SetFactoryReset",

"params":{},

"id": "18.7"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "18.7"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "180701",

"message": "Set Factory reset failed."

},

"id":"18.7"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180701 | Set Factory reset failed. | Set Factory reset failed. |

### SetDeviceRenew

Use this to clear all account information. You will need to re-activate in order to connect to

the mobile netowrk.

#### Request

{

"jsonrpc": "2.0",

"method": "SetDeviceRenew",

"params":{},

"id": "18.8"

}

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "18.8"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "180801",

"message": "Set Device Renew failed. "

},

"id":"18.8"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180801 | Set Device Renew failed. | Set Device Renew failed. |

### GetServerIndicator

Use this to get the OMA server indicator which is use currently.

#### Request

{

"jsonrpc": "2.0",

"method": "GetServerIndicator",

"params":{},

"id": "18.9"

}

#### Response

{

"jsonrpc": "2.0",

"result": {"serverId":1},

"id": "18.9"，

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| serverId | Integer | No | 1:Sprint Production  2:Sprint OMATest  3: Sprint RTB 1  4：Sprint RTB 2  5: Sprint ORT  6: Sprint ST  7: Sprint Dev  99: Sprint OEM Test |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "180901",

"message": "Get OMA Server Indicator failed. "

},

"id":"18.9"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180901 | Get OMA Server Indicator failed. | Get OMA Server Indicator failed. |

### SetServerIndicator

#### Request

{

"jsonrpc": "2.0",

"method": "SetServerIndicator",

"params":{"serverId":1},

"id": "18.10"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| serverId | Integer | No | 1:Sprint Production  2:Sprint OMATest  3: Sprint RTB 1  4：Sprint RTB 2  5: Sprint ORT  6: Sprint ST  7: Sprint Dev  99: Sprint OEM Test |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "18.10"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "181001",

"message": "Set OMA Server Indicator failed. "

},

"id":"18.10"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 181001 | Set OMA Server Indicator failed. | Set OMA Server Indicator failed. |

### SetOMAOEMServer

Use this API to set Sprint OEM Test server settings, if the OMA server indicator is 99, and should set this parameters.

#### Request

{

"jsonrpc": "2.0",

"method": "SetOMAOEMServer",

"params":{

"AppID": " ",

"ServerID":"",

"Name": "",

"SrvAddr":"",

"PortNbr":"",

"SrvAAuthType":"",

"SrvAAuthSecret":"",

"SrvAAuthData":"",

"SrvAAuthName":"",

"CltAAuthType":"",

"CltAAuthSecret":"",

"CltAAuthData":"",

"CltAAuthName":"",

"ProxyAddr":"",

"ProxyPort":""

},

"id": "18.11"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| AppID | String | Yes |  |
| ServerID | String | Yes |  |
| Name | String | Yes |  |
| SrvAddr | String | Yes |  |
| PortNbr | String | Yes |  |
| SrvAAuthType | String | Yes |  |
| SrvAAuthSecret | String | Yes |  |
| SrvAAuthData | String | Yes |  |
| SrvAAuthName | String | Yes |  |
| CltAAuthType | String | Yes |  |
| CltAAuthSecret | String | Yes |  |
| CltAAuthData | String | Yes |  |
| CltAAuthName | String | Yes |  |
| ProxyAddr | String | Yes |  |
| ProxyPort | String | Yes |  |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "18.11"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "181101",

"message": "Set OMA Sprint OEM Test failed. "

},

"id":"18.11"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 181101 | Set OMA Sprint OEM Test failed. | Set OMA Sprint OEM Test parameters failed. |

### GetOMALinkInfo

Use this API to get some link info form OMA.

#### Request

{

"jsonrpc": "2.0",

"method": "GetOMALinkInfo",

"params":{

"CheckUsageURL": " ",

"UserGuidURL":"",

"GetCustomerCare":""

},

"id": "18.12"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| CheckUsageURL | String | No | come from the Oma customization  node in GTR-DDF-00346 -> ./Customization/Browser/CheckUsage |
| UserGuidURL | String | No | come from the Oma customization node in GTR-DDF-00346 -> ./Customization/Browser/UserManual |
| GetCustomerCare | String | No | come from the Oma customization node in << GTR\_MBB\_CHMLN\_SPEC\_Ver1-2-0\_2014-10-31.pdf >>  GTR-MBB\_CHMLN-00018  ./Customization/Contacts/CustomerCare |

#### Response

{

"jsonrpc": "2.0",

"result": {},

"id": "18.12"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "181201",

"message": "Get OMA Link Info failed. "

},

"id":"18.12"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 181201 | Get OMA Link Info failed. | Get OMA Link Info failed. |

### GetOMABrandAlpha

#### Request

{

"jsonrpc": "2.0",

"method": " GetOMABrandAlpha",

"params":{},

"id": "18.13"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| BrandAlpha | String | No | OMA Brand Alpha |

#### Response

{

"jsonrpc": "2.0",

"result": {

"BrandAlpha": " "

},

"id": "18.13"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "181301",

"message": "Get OMA Brand Alpha failed. "

},

"id":"18.13"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 181301 | Get OMA Brand Alpha failed | Get OMA Brand Alpha failed |

### GetSimLockStatus

#### Request

{

"jsonrpc": "2.0",

"method": " GetSimLockStatus",

"params":{},

"id": "18.14"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| SimLockStatus | Integer | No | 0: the device must only recognize and enable an inserted Sprint SIM.  1: the device must recognize and enable any inserted foreign SIM  2: the device must recognize any inserted UICC, International or Domestic  -1: not enable SIM LOCK |

#### Response

{

"jsonrpc": "2.0",

"result": {

"SimLockStatus": 0

},

"id": "18.14"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "181401",

"message": " Get SIM Lock status failed. "

},

"id":"18.13"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 181401 | Get SIM Lock status failed | Get SIM Lock status failed |

### GetOMACarrierID

#### Request

{

"jsonrpc": "2.0",

"method": "GetOMACarrierID",

"params":{},

"id": "18.15"

}

#### Definition for each filed

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Changable** | **Description** |
| CarrierID | String | No | OMA Subscriber CarrierID |

#### Response

{

"jsonrpc": "2.0",

"result": {

"CarrierID ": " "

},

"id": "18.15"，

}

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "181501",

"message": "Get OMA Subscriber CarrierID failed. "

},

"id":"18.15"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 181501 | Get OMA Subscriber CarrierID failed | Get OMA Subscriber CarrierID failed |

## TR069

### 5.19.1 GetClientConfiguration——————Firmware端未实现接口

#### POST Request

{

"jsonrpc": "2.0",

"method": " GetClientConfiguration",

"params":{},

"id": "19.1"

}

#### Response

{

"jsonrpc": "2.0",

"result": {

"AcsUrl": "",

"AcsUserName":"" ,

"AcsUserPassword": "",

"ConReqAuthent":1,

"ConReqUserName": "",

"ConReqUserPassword": "",

"Inform":1,

"InformInterval": 60

}

"id": "19.1"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| AcsUrl | String | Acs Url |
| AcsUserName | String | Acs User Name |
| AcsUserPassword | String | Acs User Password |
| ConReqAuthent | Number | 0:Disable;  1:Enable; |
| ConReqUserName | String | Connection Request User Name |
| ConReqUserPassword | String | Connection Request User Password |
| Inform | Number | 0:Disable;  1:Enable; |
| InformInterval | Number | Inform Interval(s) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "190101",

"message": " Get client configuration failed."

},

"id":"19.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 190101 | Get client configuration failed. | Get client configuration failed. |

### SetClientConfiguration——————Firmware端未实现接口

#### Request

{

"jsonrpc": "2.0",

"method": " GetClientConfiguration",

"params":

{

"AcsUrl": "",

"AcsUserName":"" ,

"AcsUserPassword": "",

"ConReqAuthent":1,

"ConReqUserName": "",

"ConReqUserPassword": "",

"Inform":1,

"InformInterval": 60

},

"id": "19.2"

}

#### Response

{

"jsonrpc": "2.0",

"result":

{

"AcsUrl": "",

"AcsUserName":"" ,

"AcsUserPassword": "",

"ConReqAuthent":1,

"ConReqUserName": "",

"ConReqUserPassword": "",

"Inform":1,

"InformInterval": 60

}

"id": "19.2"

}

#### Definition for each filed

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Decription** |
| AcsUrl | String | Acs Url |
| AcsUserName | String | Acs User Name |
| AcsUserPassword | String | Acs User Password |
| ConReqAuthent | Number | 0:Disable;  1:Enable; |
| ConReqUserName | String | Connection Request User Name |
| ConReqUserPassword | String | Connection Request User Password |
| Inform | Number | 0:Disable;  1:Enable; |
| InformInterval | Number | Inform Interval(s) |

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "190201",

"message": " Set client configuration failed."

},

"id":"19.2"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 190201 | Set client configuration failed. | Set client configuration failed. |

## FeatureGuide

### Get feature list

Getting the Firmware supported APIs list.

#### Post Request:

{

"jsonrpc": "2.0",

"method": "GetFeatureList",

"params": {},

"id": "18.1"

}

#### Possible Response:

{

"jsonrpc": "2.0",

"result": {

"DeviceName": "Y900",

"HttpApiVersion": "1.0",

"Features": {

"User": [

"Login", "Logout",

"GetLoginState",

"ChangePassword",

"HeartBeat"

],

"SIM": [

"GetSimStatus",

"UnlockPin",

"UnlockPuk",

"ChangePinCode",

"ChangePinState",

"GetAutoValidatePinState",

"SetAutoValidatePinState",

"UnlockSimlock",

"ActiveSimlock",

"EraseSimlock"

],

"Connection": [

"GetConnectionState",

"Connect",

"DisConnect",

"GetConnectionSettings",

"SetConnectionSettings"

],

"Network": [

"GetNetworkInfo",

"SearchNetwork",

"SearchNetworkResult",

"RegisterNetwork",

"GetNetworkRegisterState",

"GetNetworkSettings",

"SetNetworkSettings"

],

"Wlan": [

"GetWlanState",

"SetWlanOff",

"SetWlanOn",

"GetWlanSettings",

"SetWlanSettings",

"SetWPSPin",

"SetWPSPbc",

"GetWlanSupportMode"

],

"SMS": [

"getSmsInitState",

"GetSMSContactList",

"GetSMSContentList",

"GetSMSStorageState",

"DeleteSMS",

"SendSMS",

"GetSendSMSResult",

"SaveSMS",

"GetSMSSettings",

"SetSMSSettings",

"GetSingleSMS"

],

"Statistics": [

"GetUsageRecord",

"SetUsageRecordClear",

"GetUsageSettings",

"SetUsageSettings"

],

"USSD": [

"SendUSSD",

"GetUSSDSendResult",

"SetUSSDEnd"

],

"Update": [

"GetDeviceNewVersion",

"SetDeviceStartUpdate",

"GetDeviceUpgradeState",

"SetDeviceUpdateStop"

],

"Firewall": [

"GetMacFilterSettings",

"SetMacFilterSettings" ,

"GetPortFwding",

"AddPortFwding",

"DeletePortFwding",

"EditPortFwding"

],

"LAN": [

"GetLanSettings",

"SetLanSettings"

],

"ConnectionDevices": [

"GetConnectedDeviceList",

"GetBlockDeviceList",

"SetConnectedDeviceBlock",

"SetDeviceUnlock",

"SetDeviceName"

],

"System": [

"GetSystemInfo",

"SetLanguage",

"GetCurrentLanguage",

"GetSystemStatus",

"SetDeviceReboot",

"SetDeviceReset",

"SetDeviceBackup",

"SetDeviceRestore" ,

"GetWizardStatus",

"SetWizardStatus"

],

"Sharing": [ /\*[Optional]\*/

"GetDLNASettings", /\*[Optional]\*/

"SetDLNASettings", /\*[Optional]\*/

"GetSambaStatus", /\*[Optional]\*/

"SetSambaStatus", /\*[Optional]\*/

"GetFtpStatus", /\*[Optional]\*/

"SetFtpStatus", /\*[Optional]\*/

"GetSDCardSpace", /\*[Optional]\*/

"GetSDFileList" /\*[Optional]\*/

],

"Profile": [

"GetProfileList",

"AddNewProfile",

"EditProfile",

"DeleteProfile",

"SetDefaultProfile"

],

"PowerManagement": [

"GetBatteryState",

"GetPowerSavingMode",

"SetPowerSavingMode"

] ，

"WifiExtender": [

"GetWIFIExtenderSettings",

"SetWIFIExtenderSettings",

"GetWIFIExtenderCurrentStatus",

"GetHotspotList",

"EditSingleHotspotInfo",

"DeleteHotspot",

"AddHotspot",

"SearchHotspot",

"GetSearchHotspotList",

"ConnectHotspot",

"DisconnectHotspot"

]

}

},

"id": "18.1"

}

1. Note: If only support SD card, the optional paramers are no required.

DeviceName is the device type name, such as Y900/Y800/W800/H850

HttpApiVersion is the current Http API Version.

Now feature list has 6 feature group: System,User,SIM,Network,SMS,Wlan,Statistics,.

Every feature group has multiple feature function and the feature function name will be returned.

#### Response with error

{

"jsonrpc": "2.0",

"error": {

"code": "180101",

"message": "Get feature listfailed"

},

"id": "18.1"

}

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| 180101 | Get feature listfailed | Get feature listfailed. |

# 6. Appendix

## Error Definition

### 关于各模块Error Code的定义

用六位数来表示错误码XX XX 01，前面2位为模块名称，中间2位为函数名称，最后2位为每个函数定义的私有错误信息,这个需要在每个接口函数定义的时候给出。

错误码的范围时XX XX 01 to XX XX 99。

例如1：错误码为01 03 01意思是User Management模块的第三个函数GetLoginState出错的原因为Get login state failed。

### JSON RPC保留的Error code

The error codes from and including -32768 to -32000 are reserved for pre-defined errors. Any code within this range, but not defined explicitly below is reserved for future use. The error codes are nearly the same as those suggested for XML-RPC at the following url: <http://xmlrpc-epi.sourceforge.net/specs/rfc.fault_codes.php>

### Common Error Tables

|  |  |  |
| --- | --- | --- |
| Error Code | Error Message | Error Description |
| -32700 | Parse error | Invalid JSON was received by the server. An error occurred on the server while parsing the JSON text. |
| -32600 | Invalid Request | The JSON sent is not a valid Request object. |
| -32601 | Method not found | The method does not exist / is not available. |
| -32602 | Invalid params | Invalid method parameter(s). |
| -32603 | Internal error | Internal JSON-RPC error. |
| -32604 | Access is denied | The request of access is not permitted and denied |
| -32000 to -32099 | Server error | Reserved for implementation-defined server-errors. |

The remainder of the space is available for application defined errors.