广告SDK消息通道合并软件设计方案

**（仅供内部使用）**

|  |  |
| --- | --- |
| 拟 制 |  |
| 审 核 |  |
| 批 准 |  |

**目 录**

[广告SDK消息通道合并软件设计方案 1](#_Toc128082727)

[1 引言 4](#_Toc128082728)

[1.1 目的 4](#_Toc128082729)

[1.2 背景 4](#_Toc128082730)

[1.3 范围 4](#_Toc128082731)

[2 通信架构 5](#_Toc128082732)

[2.1 术语定义 5](#_Toc128082733)

[2.2 通信拓扑设计 5](#_Toc128082734)

[2.3 WESocket SDK与 WebSocket Server之间通信约定 5](#_Toc128082735)

[2.3.1 WebSocket 连接url格式 5](#_Toc128082736)

[2.3.2 WebSocket 子协议名称 6](#_Toc128082737)

[2.3.3 认证鉴权 6](#_Toc128082738)

[2.3.4 消息格式 6](#_Toc128082739)

[2.3.5 业务类型定义 7](#_Toc128082740)

[2.3.6 节点id定义 7](#_Toc128082741)

[2.3.7 通信过程约定 8](#_Toc128082742)

[2.3.8 心跳通信维护 8](#_Toc128082743)

[2.3.9 详细业务描述 9](#_Toc128082744)

[2.3.9.1 心跳（HeartBeat） 9](#_Toc128082745)

[2.3.9.2 下行消息（DnMessage） 9](#_Toc128082746)

[2.3.9.3 上行消息（UpMessage） 10](#_Toc128082747)

[3 Business Client与Business Server之间业务通讯约定 11](#_Toc128082748)

[3.1 广告SDK与广告平台业务相关 11](#_Toc128082749)

[3.1.1 架构拓扑设计 11](#_Toc128082750)

[3.1.2 消息格式 11](#_Toc128082751)

[3.1.3 业务类型定义 13](#_Toc128082752)

[3.1.4 详细业务描述 13](#_Toc128082753)

[3.1.4.1 广告任务下发（PlayTask） 13](#_Toc128082754)

[4 兼容性设计 14](#_Toc128082755)

[5 安全和可靠设计 14](#_Toc128082756)

[5.1 数据安全 14](#_Toc128082757)

[5.2 可靠性设计 14](#_Toc128082758)

# 引言

## 目的

本文档编写的目的是为了WebSocket SDK与WebSocket Server之间的通信方式、鉴权和消息格式约定以及介绍广告SDK消息通道合并的框架设计以及核心业务流程，为编码、维护、测试、迭代优化提供参考。

## 背景

广告SDK与广告平台之间为保持消息推送和数据上报，使用Netty搭建了一套C/S的长连接通道，因云平台已存在基于WebSocket的C/S长连接消息分发通道，为便于统一维护，需把广告SDK中的长连接替换为WebSocket的长连接。

**本文档主要包括两部分内容：**

1. WebSocket SDK与WebSocket Server之间的通信方式、鉴权和消息格式约定；
2. 广告SDK集成WebSocket SDK后与广告平台的通讯交互细节约定。

**涉及主要工作事项如下：**

1. WebSocket SDK开发，包含：

* 连接、重连管理模块；
* 心跳维护与空闲检测模块；
* 数据收发回调管理模块；

1. WebSocket Server端开发：协议适配开发，与业务后台数据互传功能开发；
2. 广告SDK集成WebSocket SDK后与广告后台数据交互适配开发。

## 范围

系统设计人员、软件开发人员、软件测试人员和项目评审人员。

# 通信架构

## 术语定义

* **WebSocket Server**

WebSocket服务端，长连接管理平台，提供长连接消息分发通道，透传消息。

* **WebSocket SDK**

与WebSocket服务端进行连接，提供重连，心跳维护，数据发送与接收的能力。

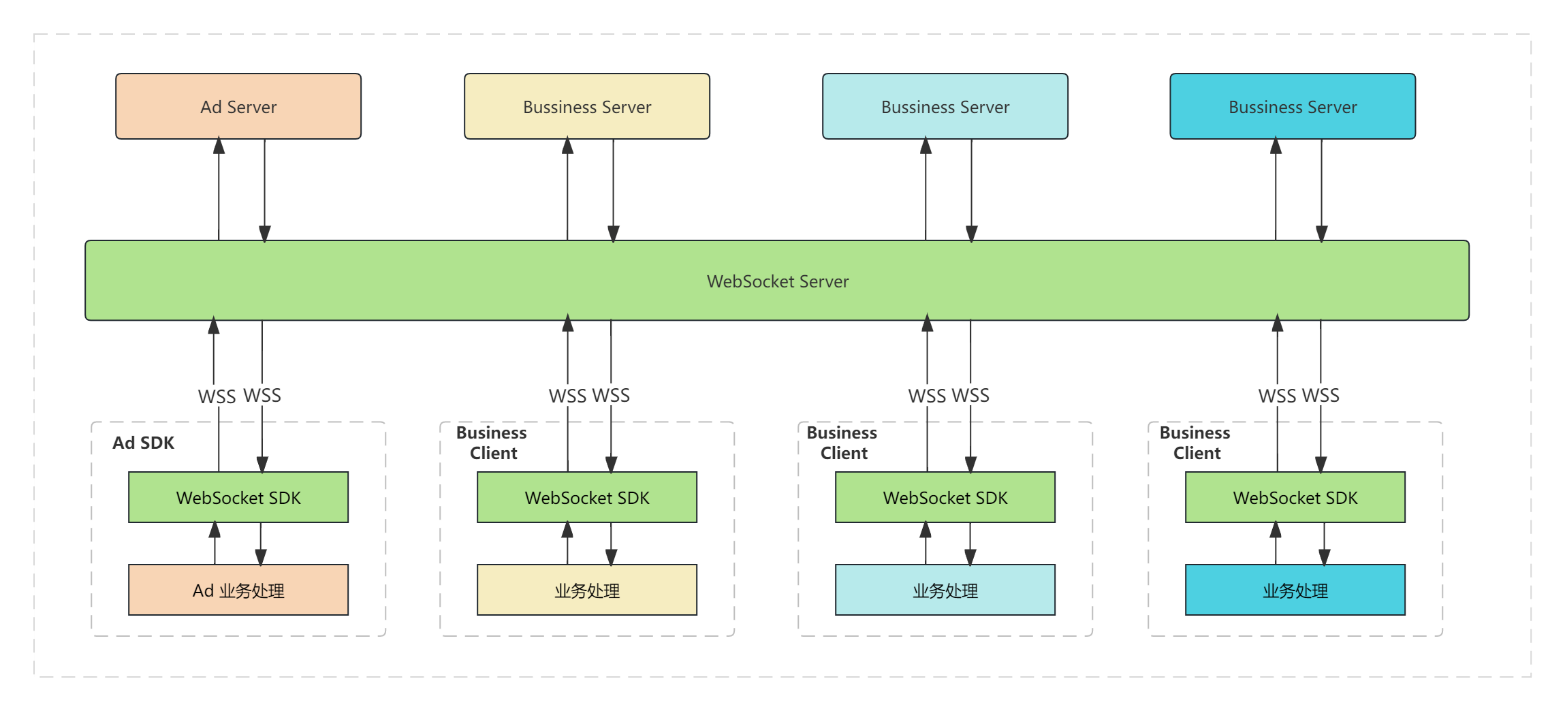
* **Business Client**

业务客户端，集成WebSocket SDK，通过长连接通道与业务服务端，进行业务数据收发。

* **Business Server**

业务服务端，与WebSocket服务端通信，通过长连接通道与业务客户端，进行业务数据收发。

## 通信拓扑设计



## WebSocket SDK与 WebSocket Server之间通信约定

### WebSocket 连接url格式

wss://{domain}/ws/webSocket?sn={SN}

1. SN：设备唯一标识，充电桩为桩序列码，广告SDK可使用广告后台下发的playCode；
2. Domain域名：参考[《平台各环境地址》](https://confluence.autel.com/pages/viewpage.action?pageId=50932057)

### WebSocket 子协议名称

1. apmp1.0 (注：Autel Push Manager Protocol 首字母简写)
2. 连接时请求头需带上协议：Sec-WebSocket-Protocol: apmp1.0

### 认证鉴权

1. 因 Token 获取与鉴权相关，还无通用开放平台能力，临时使用游客接口获取Token：

https://{domin}/api/base-uc-app/portal/visitor/token?loginFrom=web

1. Domain域名：参考[《平台各环境地址》](https://confluence.autel.com/pages/viewpage.action?pageId=50932057)
2. 连接时请求头需带上Token：Authorization: xxxxxxxx

### 消息格式

协议版本：apmp1.0

1. **请求消息格式**



|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | 参考《业务类型定义》 |
| seq | String | 是 | 消息序列，递增，到最⼤值后归零 |
| source | int | 否 | 源地址，表明发送方，参考《节点ID定义》 |
| target | int | 否 | 目的地址，表明接收方，参考《节点ID定义》 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

1. **响应消息格式**



|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | 参考《业务类型定义》 |
| seq | String | 是 | 消息序列，回传接收到的请求消息中的seq |
| source | int | 否 | 源地址，表明发送方，参考《节点ID定义》 |
| target | int | 否 | 目的地址，表明接收方，参考《节点ID定义》 |
| code | int | 是 | 状态码，200表明成功 |
| message | String | 否 | 状态信息描述 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

### 业务类型定义

|  |  |
| --- | --- |
| **cmd 业务类型** | **说明** |
| HeartBeat | 心跳消息，与WebSocket Server进行心跳收发 |
| DnMessage | 主动下行消息，用于业务服务端向业务客户端下发消息 |
| UpMessage | 主动上行消息，用于业务客户端向业务服务端上传消息 |

### 节点id定义

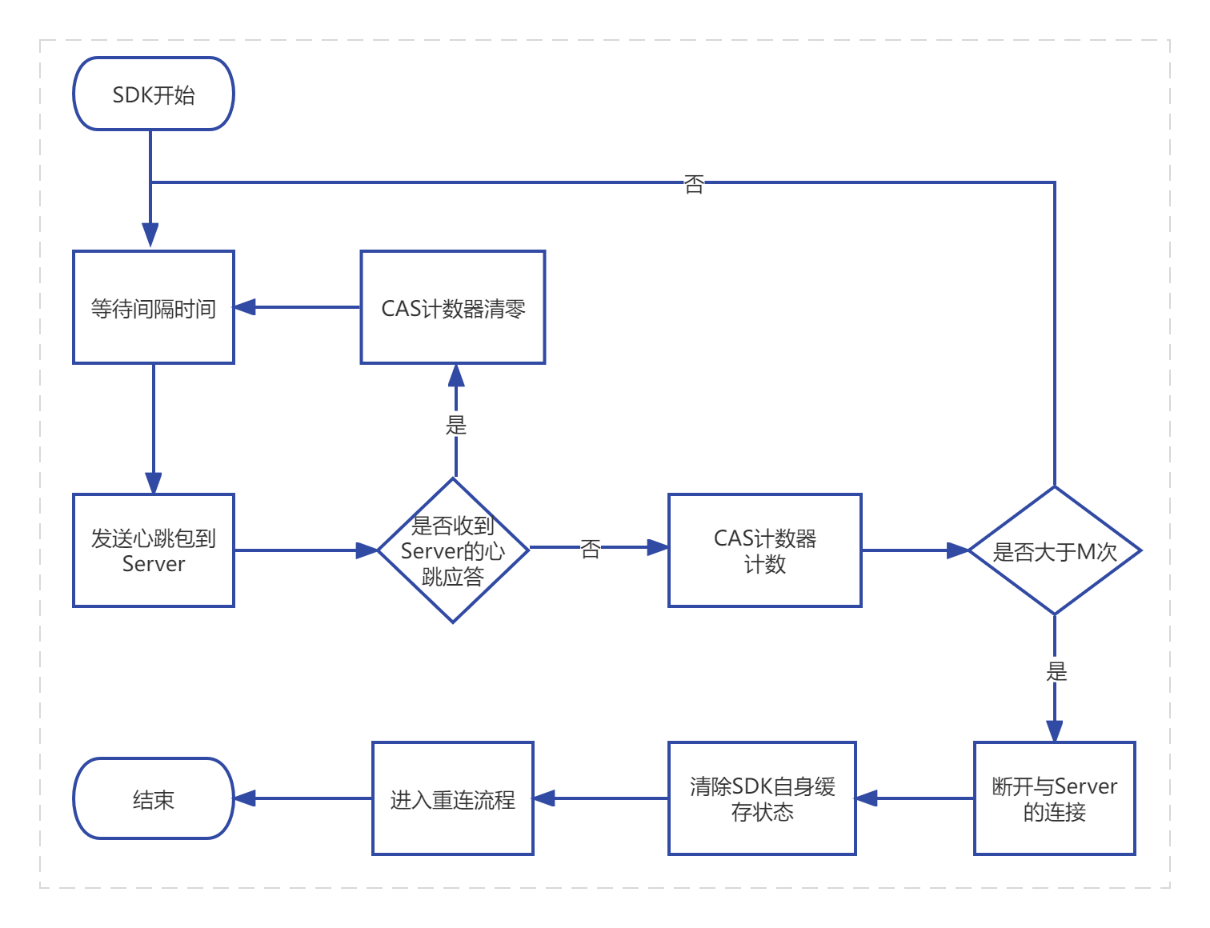
|  |  |
| --- | --- |
| **节点ID定义** | **说明** |
| 1001 | 业务客户端：广告SDK |
| 1002 | 业务服务端：广告平台 |

### 通信过程约定

1. WebSocket SDK与WebSocket Server之间采用Client/Server的TCP通信方式；
2. 客户端与服务端通信前需进行Token鉴权；
3. 连接由客户端主动发起，客户端和服务端都有权利断开连接，xx秒内未连接成功按连接超时处理，客户端将进入重连流程（SDK连接超时默认30s，可在初始化时，配置指定超时时间）；
4. 客户端需维护通信连接状态的有效性，在连接断开后，需自动进行重连尝试，直到连接恢复。
5. 所有通信消息除特别说明外都采用问答方式，有问必有答。数据发起方发送数据后，xx秒内未得到回复按通讯超时处理。（SDK通讯超时默认60s，可在初始化时，配置指定超时时间）；
6. 通讯超时后需重发数据（心跳等不影响业务的报文不需重发），默认重发次数为3，连续重发后仍然得不到回应，则认为存在异常，停止重发。

### 心跳通信维护

1. WebSocket SDK与WebSocket Server建立连接后，应及时发送心跳包；
2. SDK应该每n秒发送一次心跳包 （定值设置，默认10s，可在初始化时，配置指定时间间隔），且SDK需统计心跳包应答情况；
3. SDK 统计丢失m次心跳包后，应主动断开与服务端的通讯连接，进入重连流程 （定值设置，默认5次，可在初始化时，配置指定次数）。



### 详细业务描述

#### 心跳（HeartBeat）

1. 心跳请求

WebSocket SDK -> WebSocket Server

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | HeartBeat |
| seq | String | 是 | 消息序列，递增，到最⼤值后归零 |

1. 心跳响应

WebSocket Server -> WebSocket SDK

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | HeartBeat |
| seq | String | 是 | 消息序列，回传接收到的请求消息中的seq |
| code | int | 是 | 状态码，200表明成功 |
| message | String | 否 | 状态信息描述 |

#### 下行消息（DnMessage）

1. 下行消息传输请求

Business Server -> Business Client

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | DnMessage |
| seq | String | 是 | 消息序列，递增，到最⼤值后归零 |
| source | int | 是 | 源地址，表明发送方，参考《节点ID定义》 |
| target | int | 是 | 目的地址，表明接收方，参考《节点ID定义》 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

1. 下行消息传输响应

Business Clinet -> Business Server

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | DnMessage |
| seq | String | 是 | 消息序列，回传接收到的请求消息中的seq |
| source | int | 是 | 源地址，表明发送方，参考《节点ID定义》 |
| target | int | 是 | 目的地址，表明接收方，参考《节点ID定义》 |
| code | int | 是 | 状态码，200表明成功 |
| message | String | 否 | 状态信息描述 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

#### 上行消息（UpMessage）

1. 上行消息传输请求

Business Clinet -> Business Server

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | UpMessage |
| seq | String | 是 | 消息序列，递增，到最⼤值后归零 |
| source | int | 是 | 源地址，表明发送方，参考《节点ID定义》 |
| target | int | 是 | 目的地址，表明接收方，参考《节点ID定义》 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

1. 上行消息传输响应

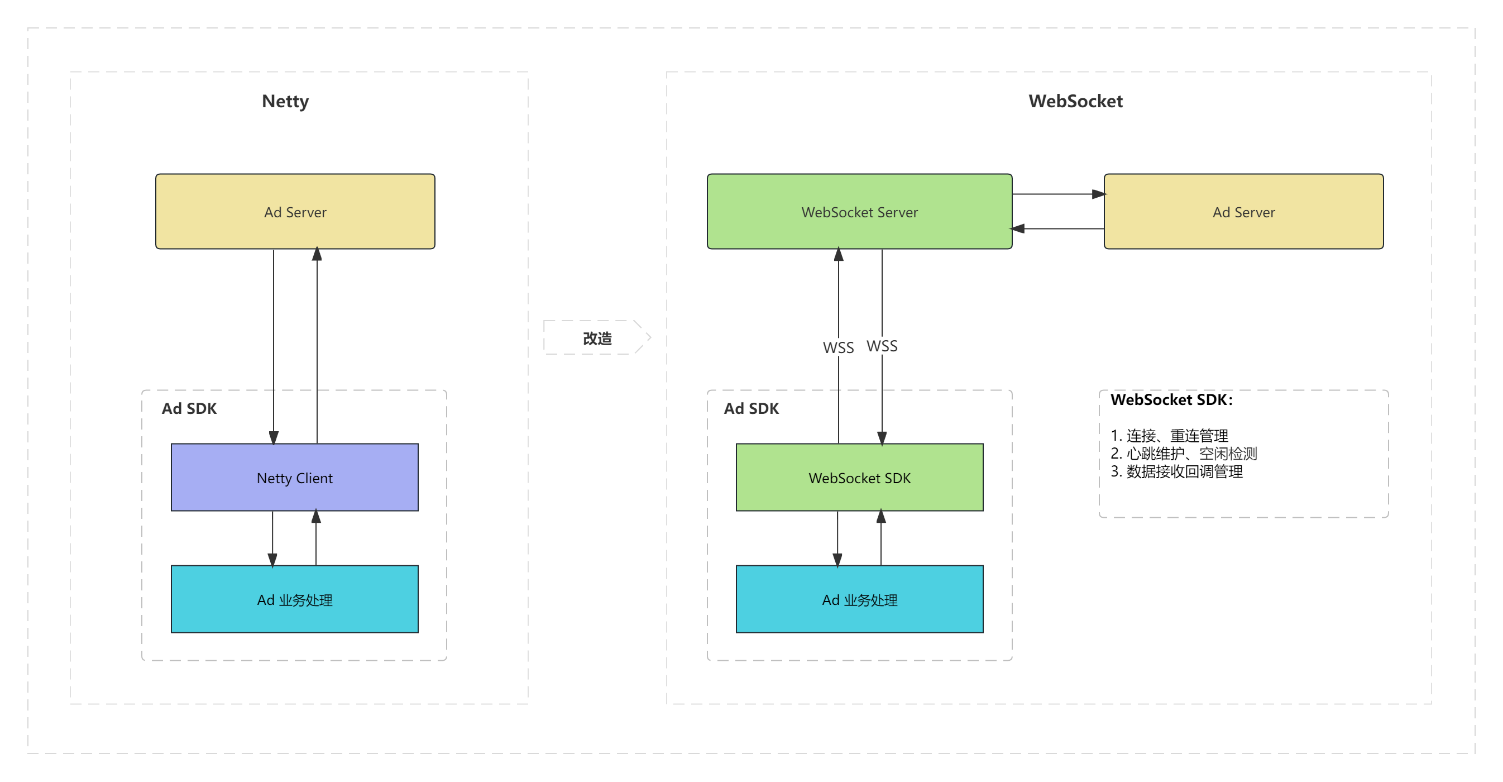
Business Server -> Business Clinet

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | UpMessage |
| seq | String | 是 | 消息序列，回传接收到的请求消息中的seq |
| source | int | 是 | 源地址，表明发送方，参考《节点ID定义》 |
| target | int | 是 | 目的地址，表明接收方，参考《节点ID定义》 |
| code | int | 是 | 状态码，200表明成功 |
| message | String | 否 | 状态信息描述 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

# Business Client与Business Server之间业务通讯约定

## 广告SDK与广告平台业务相关

### 架构拓扑设计



1. 广告SDK集成 WebSocket SDK与WebSocket Server保持长连接，收发广告平台的数据，广告SDK接收到广告平台消息后，需回复广告平台；
2. 设备在线与否，广告平台从WebSocket Server端查询，广告SDK不用定时发送心跳包数据到广告平台。

### 消息格式

此消息数据定义在WebSocket的data数据中，如下：



1. **请求消息格式**



|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | 参考下面的《业务类型定义》 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

1. **响应消息格式**



|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | 参考下面的《业务类型定义》 |
| code | int | 是 | 状态码，200表明成功 |
| message | String | 否 | 状态信息描述 |
| data | Object | 否 | 数据，json格式，若⽆特别说明可为空 |

### 业务类型定义

|  |  |
| --- | --- |
| **cmd 业务类型** | **说明** |
| PlayTask | 广告任务下发，用于广告平台向广告SDK下发广告任务 |
|  |  |

### 详细业务描述

#### 广告任务下发（PlayTask）

走DnMessage下行消息通道

1. 广告任务下发请求

Business Server -> Business Client

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | PlayTask |
| data | Object | 是 | 下发的PlayTask数据，json格式 |

下发的PlayTask数据结构

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| id | long | 是 | 消息唯一id |
| taskId | long | 是 | 任务唯一id |
| type | short | 是 | 任务类型 |

1. 广告任务下发响应

Business Client -> Business Server

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数据类型** | **是否必填** | **说明** |
| cmd | String | 是 | PlayTask |
| code | int | 是 | 状态码，200表明成功 |
| message | String | 否 | 状态信息描述 |

# 兼容性设计

# 安全和可靠设计

## 数据安全

## 可靠性设计