# websocket二进制流数据包分析

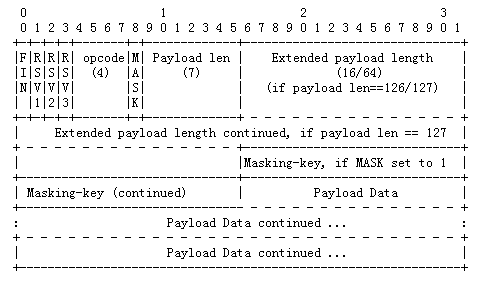
[HTTP网络](https://imweb.io/tab/network)

本文作者：IMWeb andenchen 原文出处：[IMWeb社区](http://imweb.io/topic/5b116b10d4c96b9b1b4c4e95) **未经同意，禁止转载**

WebSocket是为解决客户端与服务端实时通信而产生的技术。websocket协议本质上是一个基于tcp的协议，是先通过HTTP/HTTPS协议发起一条特殊的http请求进行握手后创建一个用于交换数据的TCP连接，此后服务端与客户端通过此TCP连接进行实时通信。

## **发送数据**

WebSocket中所有发送的数据使用帧的形式发送。客户端发送的数据帧都要经过掩码处理，服务端发送的所有数据帧都不能经过掩码处理。否则对方需要发送关闭帧。



**FIN**：标识是否为此消息的最后一个数据包，占 1 bit

**RSV1, RSV2, RSV3**: 用于扩展协议，一般为0，各占1bit

**Opcode**：数据包类型（frame type），占4bits

0x0：标识一个中间数据包

0x1：标识一个text类型数据包

0x2：标识一个binary类型数据包

0x3-7：保留

0x8：标识一个断开连接类型数据包

0x9：标识一个ping类型数据包

0xA：表示一个pong类型数据包

0xB-F：保留

**MASK**：占1bits，用于标识PayloadData是否经过掩码处理。

**Payload length**：Payload data的长度。

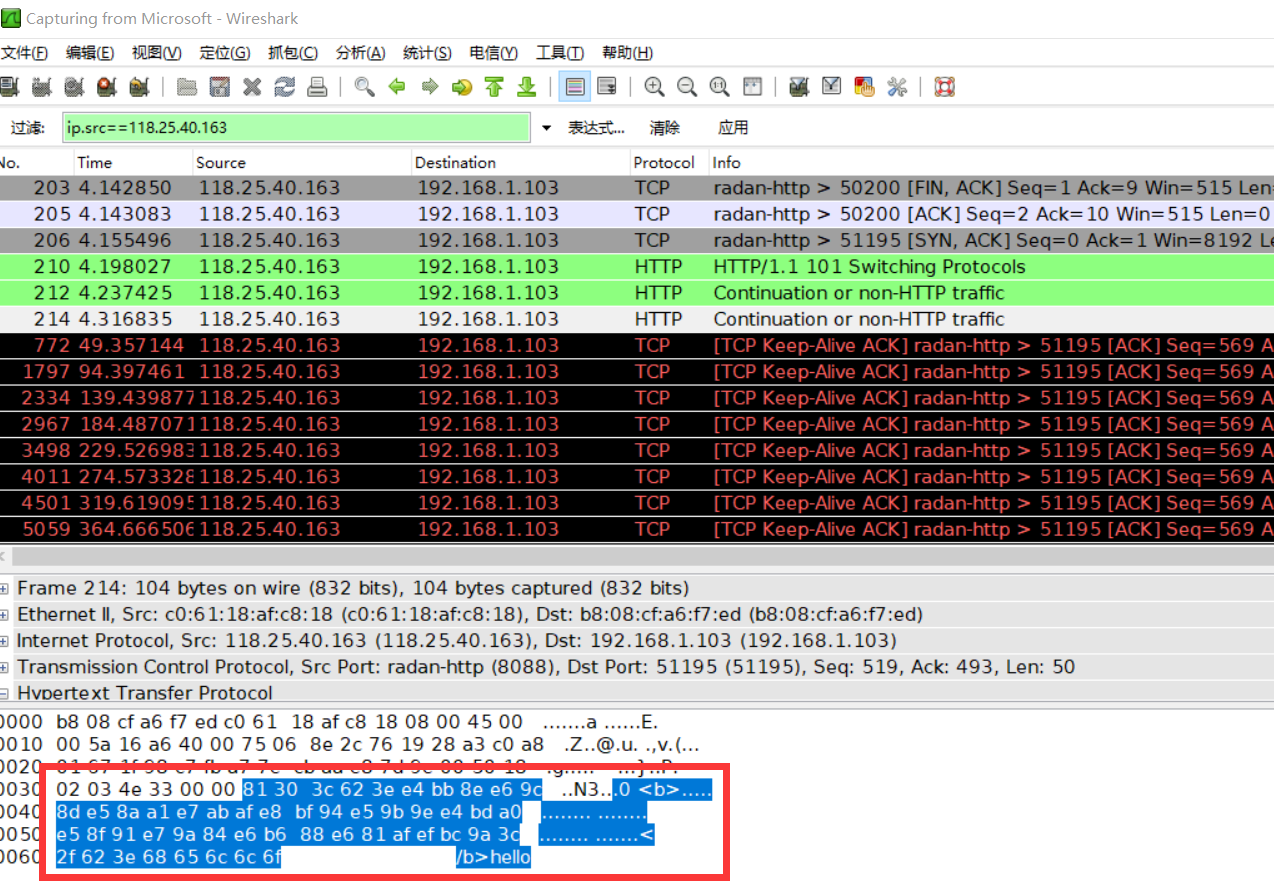
1、如果其值在0-125，则是payload的真实长度。

2、如果值是126，则后面2个字节形成的16bits无符号整型数的值是payload的真实长度。

3、如果值是127，则后面8个字节形成的64bits无符号整型数的值是payload的真实长度。

## **实例分析**

下面是通过websocket服务器返回的数据信息   
我们可以用wireshark抓取TCP包观察一下数据

  
抓取的二进制流是通过十六进制解析的，下面是具体的解包过程：

81（16进制）= 10000001（二进制）=> FIN(0) + RSV1(0) + RSV2(0) + RSV3(0) + Opcode(0x1)text类型数据包。

30（16进制）= 0110000（二进制）=> MASK(0) + Payload length(0x30)数据长度48bytes。

3c到6f 是具体的 text数据内容。（48bytes）