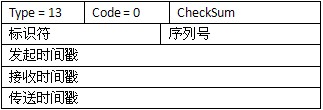
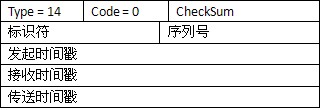
## ICMP 时间戳

时间戳请求:



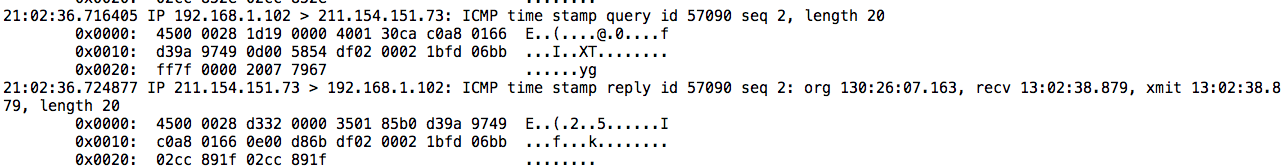
          请求端填写发起时间戳

  时间戳应答:

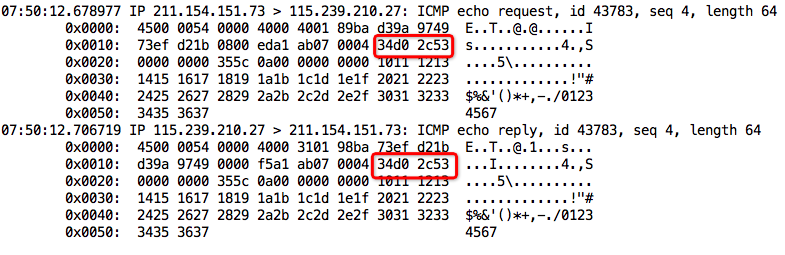


        应答系统收到请求报文时填写接收时间戳；发送应答时填写发送时间戳

 ICMP时间戳请求允许系统向另一个系统查询当前的时间，返回的**值是自午夜开始计算的毫秒数**，提供了毫秒级的分辨率。  
   请求端填写发起时间戳， 接收时间戳、传送时间戳置0  
   应答端填写接收时间戳、传送时间戳，发起时间戳与请求报文中一致



ping不存在时间同步，它只是将发起请求时间加入到每个消息包，通过他来计算网络时延。

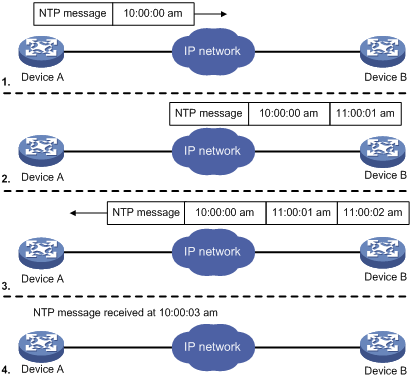


## NTP时间同步

### NTP工作原理

NTP的基本工作原理如下图所示。Device A和Device B通过网络相连，它们都有自己独立的系统时钟，需要通过NTP实现各自系统时钟的自动同步。为便于理解，作如下假设：

* 在Device A和Device B的系统时钟同步之前，Device A的时钟设定为10:00:00am，Device B的时钟设定为11:00:00am。
* Device B作为NTP时间服务器，即Device A将使自己的时钟与Device B的时钟同步。
* NTP报文在Device A和Device B之间单向传输所需要的时间为1秒。



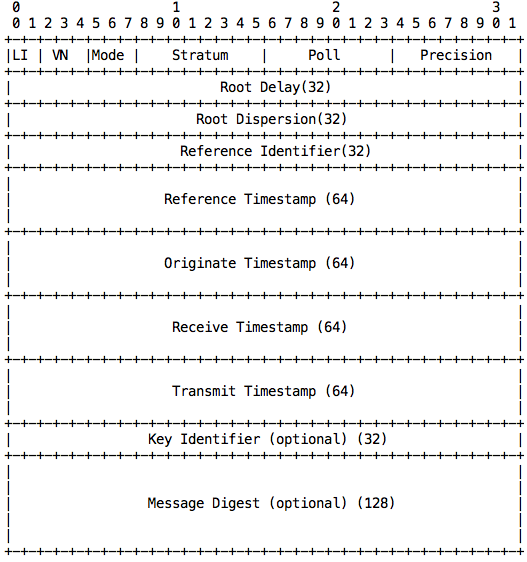
* Device A发送一个NTP报文给Device B，该报文带有它离开Device A时的时间戳，该时间戳为10:00:00am（T1）。
* 当此NTP报文到达Device B时，Device B加上自己的时间戳，该时间戳为11:00:01am（T2）。
* 当此NTP报文离开Device B时，Device B再加上自己的时间戳，该时间戳为11:00:02am（T3）。
* 当Device A接收到该响应报文时，Device A的本地时间为10:00:03am（T4）。

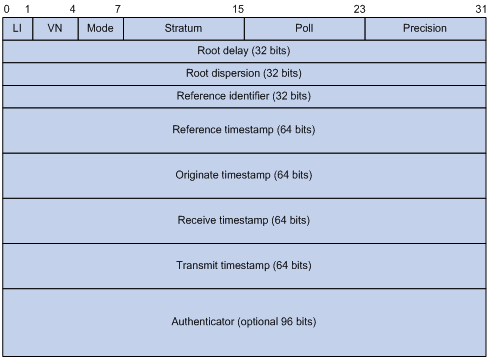
至此，Device A已经拥有足够的信息来计算两个重要的参数：

* **NTP报文的往返时延Delay=（T4-T1）-（T3-T2）=2秒。**
* **Device A相对Device B的时间差offset=（（T2-T1）+（T3-T4））/2=1小时。**

NTP协议

NTP基于UDP报文进行传输，使用的UDP端口号为123；时钟同步报文封装在UDP报文中，其格式如下图所示。





主要字段的解释如下：

* LI（Leap Indicator）：长度为2比特，值为“11”时表示告警状态，时钟未被同步。为其他值时NTP本身不做处理。
* VN（Version Number）：长度为3比特，表示NTP的版本号，目前的最新版本为3。
* Mode：长度为3比特，表示NTP的工作模式。不同的值所表示的含义分别是：0未定义、1表示主动对等体模式、2表示被动对等体模式、3表示客户模式、4表示服务器模式、5表示广播模式或组播模式、6表示此报文为NTP控制报文、7预留给内部使用。
* Stratum：系统时钟的层数，取值范围为1～16，它定义了时钟的准确度。层数为1的时钟准确度最高，准确度从1到16依次递减，层数为16的时钟处于未同步状态，不能作为参考时钟。
* Poll：轮询时间，即两个连续NTP报文之间的时间间隔。
* Precision：系统时钟的精度。
* Root Delay：本地到主参考时钟源的往返时间。
* Root Dispersion：系统时钟相对于主参考时钟的最大误差。
* Reference Identifier：参考时钟源的标识。
* Reference Timestamp：系统时钟最后一次被设定或更新的时间。
* Originate Timestamp：NTP请求报文离开发送端时发送端的本地时间。
* Receive Timestamp：NTP请求报文到达接收端时接收端的本地时间。
* Transmit Timestamp：应答报文离开应答者时应答者的本地时间。
* Authenticator：验证信息。

## 游戏中时间同步

解决三个问题：

* 服务器需要的数据：

rtt(round trip time)

* 客户端需要的数据：

time offset

* 心跳长连接维持， 如每5分钟次做一次time offset /rtt统计

数据包：

【-----包头-----】

原始orig timestamp,客户端发起时间 （uint64\_t）

接收recv timestamp,服务器接收时间 （uint64\_t）

传送xmit timestamp,服务器下发时间 （uint64\_t）(transmit)

rtt (uint64\_t) 单位 0.1毫秒

* rtt需要在第三路握手才可以提交到服务器。
* 平常的心跳连接，数据都为空。
* 该数据需要在与服务器交互时第一个提交到服务器。