

基于minilzo的socket数据压缩技术

基于minilzo的socket数据压缩技术

问题分析

常用快速无损压缩算法比较

Snappy

FastLZ

LZO/miniLZO

测试数据

最终方案

实际数据

问题分析

诸神游戏服务器使用二进制流传输，协议描述采用了两种方式的混合，其一是struct(采用1字节的内存对齐)，其二是使用google protobuf，我们把每个协议的实例化称之为一个网络包，实际使用中同一个网络包会大量存在着相同的字节。诸神玩家落地策略采用的是序列化二进制 chunk，并存储在mysql中，那么每一玩家数据的大小关系到数据库容量的大小，以及加载的速度。上述两例都有相同数据压缩需求，

常用快速无损压缩算法比较

Snappy

Snappy是在谷歌内部生产环境中被许多项目使用的压缩库，包括BigTable，MapReduce和RPC等。官方网站：
<http://code.google.com/p/snappy/>

FastLZ

FastLZ是一个高效的轻量级压缩解压库，其官方测试数据如下表：

LZO/miniLZO

LZO是一个开源的无损压缩C语言库，其优点是压缩和解压缩比较迅速占用内存小等特点（网络传输希望的是压缩和解压缩速度比较快，压缩率不用很高），其提供了比较全的LZO库和一个精简版的miniLZO库。官方网站：
<http://www.oberhumer.com/opensource/lzo/>

测试数据

算法	压缩比	加密速度	解密速度
FastLZ	13.4%	21MB/s	118MB/s
Snappy	22.2%	172MB/s	410MB/s
LZO	20.5%	135MB/s	409MB/s

最终方案

经过综合考量，选择了minilzo这种兼顾cpu和压缩率的方式，其中针对网络包只压缩数据大小超过64字节的，如果压缩后大小超过原始大小则放弃，在数据包的首字节使用标志位来标示是否压缩。网络包支持压缩和非压缩两种方式，玩家存档数据也采用类似方案。

```
if(pack->Size>64){
    minilzo_compress(pack->Data,pack->Size,buf,buflen);
}
```

实际数据

针对MSG_ASKOBJ等频繁发送的大包，压缩率高达70%，针对玩家数据压缩率高达85%。