

# 一个模拟的负载均衡系统的实现

## 软件使用说明书

项目负责人：××××

文件状态：	项目名称	一个模拟的负载均衡系统的实现	文档名称	使用说明书
【 <input type="checkbox"/> 】草稿	文件标识		当前版本	V1.0
【 <input checked="" type="checkbox"/> 】正式发布	作者	××××	完成时间	2013-08-03
【 <input type="checkbox"/> 】正在修改	页数	10	密级	

2013 年 8 月 3 日

# 文档控制

## 修改记录

起止日期	修改类型	作者	参与者	版本	备注
2013-07-14 — 2013-07-14	编写	××××		V1.0	
2013-08-03 — 2013-08-03	增加	××××		V2.0	

## 审阅人

姓名	职位	审阅签字

## 存档

存档号	地点/位置	备注

# 目 录

一、 概述.....	4
1.1 背景 .....	4
1.2 应用领域与使用对象.....	4
1.3 参考资料 .....	4
1.4 术语与缩写解释.....	4
二、 系统综述.....	5
2.1 系统结构 .....	5
2.2 系统功能简介 .....	5
2.3 性能 .....	5
2.4 版权声明 .....	5
三、 运行环境.....	6
3.1 运行要求 .....	6
四、 客户端操作说明 .....	7
4.1 软件配置 .....	7
4.2 软件运行 .....	7
4.2.1 启动参数说明.....	7
4.2.2 快捷启动.....	7
4.2.3 操作说明.....	7
五、 服务端操作说明 .....	9
5.1 软件配置 .....	9
5.2 软件运行 .....	9
5.2.1 启动参数说明.....	9
5.2.2 快捷启动.....	9
5.2.3 操作说明.....	9
5.2.4 详细信息显示.....	9
5.2.5 统计信息显示.....	10
六、 LB 端操作说明.....	11
6.1 软件配置 .....	11
6.2 软件运行 .....	12
6.2.1 启动说明.....	12
6.2.2 操作说明.....	12
6.2.3 统计信息显示.....	12
6.2.4 详细信息显示.....	13
6.2.5 运行日志.....	13
6.2.6 健康检测.....	13
6.2.7 会话保持.....	14
6.2.8 负载均衡.....	15
七、 附录.....	17
7.1 BAT 脚本语法简介 .....	17
7.2 VBS 脚本语法简介 .....	17

# 一、概述

## 1.1 背景

为了构建可伸缩的，高可用的网络服务，很多大型网站都采用了负载均衡技术。  
利用负载均衡技术，可以将多台廉价的、低性能的服务器，组合成一台性能强劲的，高可用的虚拟服务器。

## 1.2 应用领域与使用对象

2013 年“中兴捧月”杯校园赛事嘉年华程序设计大赛参赛作品。  
作者：××××

## 1.3 参考资料

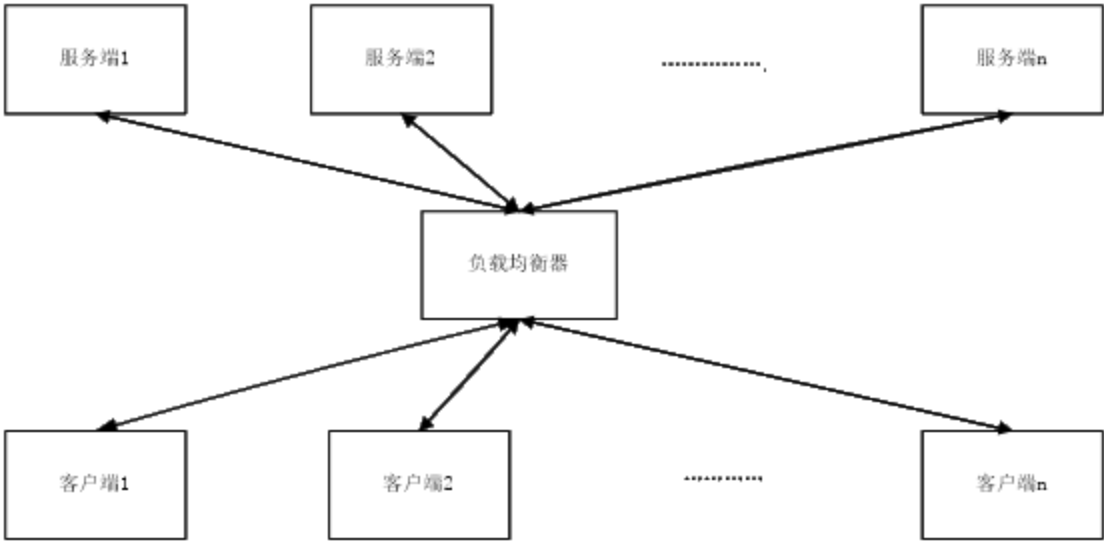
RFC-1305 文档

## 1.4 术语与缩写解释

缩写、术语	解释
NTP	Network Time Protocol

## 二、 系统综述

### 2.1 系统结构



### 2.2 系统功能简介

- 服务端 (server.exe)——辅助程序，通过 UDP 端口，提供时间查询服务。
- 负载均衡器(LB.exe)——核心程序，用于实现负载均衡功能。
- 客户端 (client.exe)——辅助程序，通过 UDP 端口，访问时间查询服务。

### 2.3 性能

经作者本人在虚拟机中测试，负载均衡程序的并发能力大概在 300 左右，如果将客户端和服务端分散在非负载均衡端的其他计算机上，并发能力应该还会提高。

### 2.4

## 三、 运行环境

### 3.1 运行要求

CPU:	1GHz 以上
内存:	可用内存 128M 以上
硬盘:	可用空间 1GB 以上
操作系统:	Windows XP 以上
开发环境:	VC6
网络:	TCP/IP(IPv4)
数据库:	不需要

## 四、 客户端操作说明

### 4.1 软件配置

无需配置。

### 4.2 软件运行

#### 4.2.1 启动参数说明

客户端需要带参数启动，一共有 6 个启动参数，依次为：

- ①客户端的 ID
- ②客户端的 `usr_id`
- ③要发送到的接收端的 ID（按题目描述应该为 LB 端的 ID）
- ④要发送的数据包的个数
- ⑤点分十进制的要发送到的接收端的 IP（按题目描述应该为 LB 端的 IP）
- ⑥要发送到的接收端的端口号（按题目描述应该为 LB 端的端口）

#### 4.2.2 快捷启动

为省去每次启动都要指定启动参数的麻烦，系统提供了两种**脚本启动**方式，脚本语法已经在脚本文件中配置好，使用者只需改动需要改变的参数即可。

##### ①BAT 脚本：

文件名称：运行\_client(显示窗口).bat

使用方法：双击

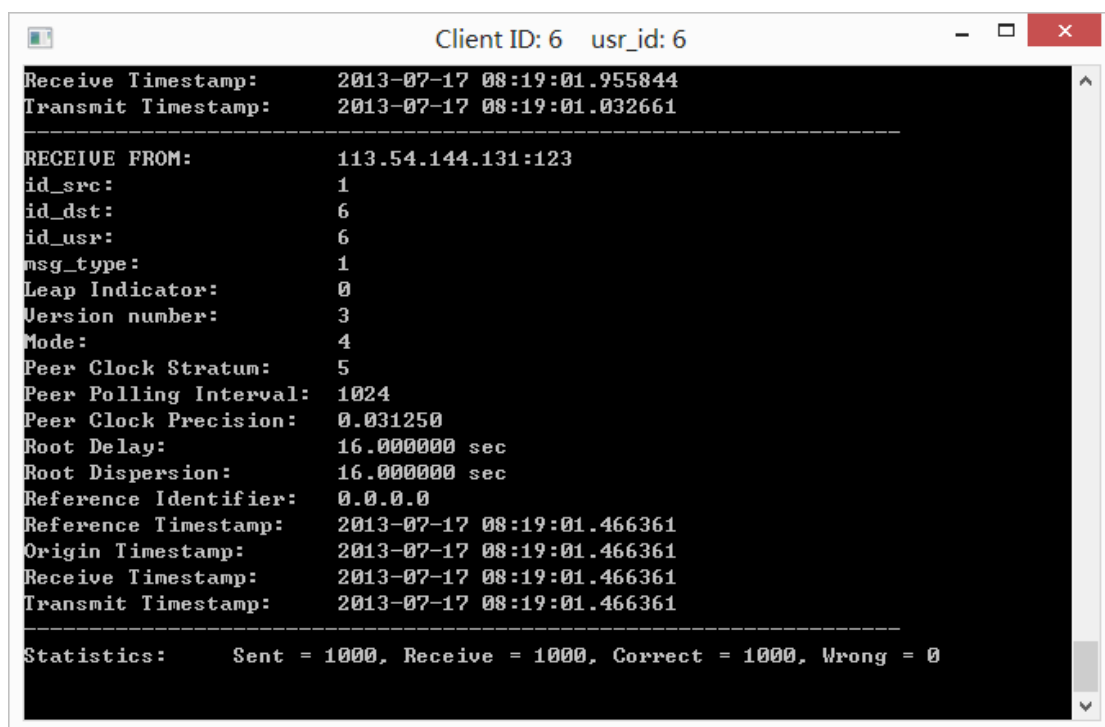
##### ②VBS 脚本：

文件名称：运行\_client(最小化窗口).vbs

使用方法：双击

#### 4.2.3 操作说明

运行客户端后，软件会按照参数向指定 IP 的指定端口发送指定数量的数据包，并将自己接收/发送的每一个消息，都实时显示在屏幕上。发送完所有数据后，显示相关的统计信息后即退出。统计信息包括本进程发送了多少条消息，接收了多少条消息（正确的多少条，错误的多少条）。



The screenshot shows a window titled "Client ID: 6 usr\_id: 6" with a black background and white text. The text displays network protocol details and statistics. The details are organized into two main sections separated by dashed lines. The first section shows the receive and transmit timestamps. The second section shows the receive details, including the source IP, various IDs, message type, and various indicators and timestamps. The statistics section at the bottom shows the number of sent, received, correct, and wrong packets.

```
Client ID: 6  usr_id: 6
Receive Timestamp: 2013-07-17 08:19:01.955844
Transmit Timestamp: 2013-07-17 08:19:01.032661
-----
RECEIVE FROM: 113.54.144.131:123
id_src: 1
id_dst: 6
id_usr: 6
msg_type: 1
Leap Indicator: 0
Version number: 3
Mode: 4
Peer Clock Stratum: 5
Peer Polling Interval: 1024
Peer Clock Precision: 0.031250
Root Delay: 16.000000 sec
Root Dispersion: 16.000000 sec
Reference Identifier: 0.0.0.0
Reference Timestamp: 2013-07-17 08:19:01.466361
Origin Timestamp: 2013-07-17 08:19:01.466361
Receive Timestamp: 2013-07-17 08:19:01.466361
Transmit Timestamp: 2013-07-17 08:19:01.466361
-----
Statistics: Sent = 1000, Receive = 1000, Correct = 1000, Wrong = 0
```



## 五、 服务端操作说明

### 5.1 软件配置

无需配置。

### 5.2 软件运行

#### 5.2.1 启动参数说明

服务端需要带参数启动，一共有 3 个启动参数，依次为：

- ①服务端的 ID
- ②LB 端的 ID
- ③要绑定的本地端口

#### 5.2.2 快捷启动

为省去每次启动都要指定启动参数的麻烦，系统提供了两种脚本启动方式，脚本语法已经在脚本文件中配置好，使用者只需改动需要改变的参数即可。

##### ①BAT 脚本：

文件名称：运行\_server(显示窗口).bat

使用方法：双击

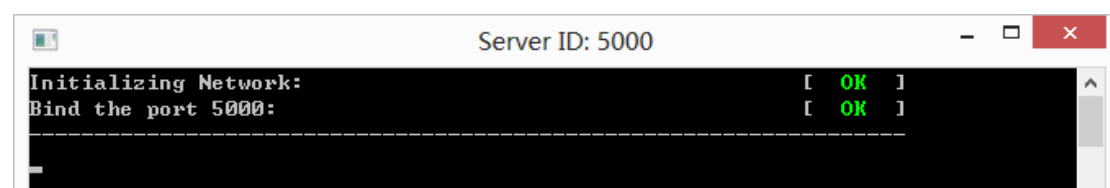
##### ②VBS 脚本：

文件名称：运行\_server (最小化窗口).vbs

使用方法：双击

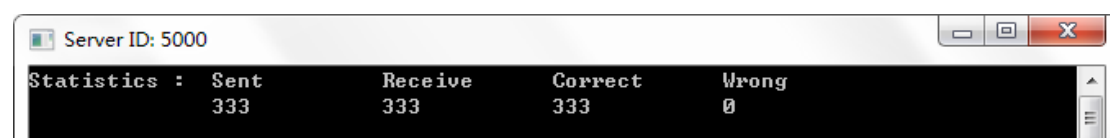
#### 5.2.3 操作说明

运行服务端后，软件初始化完成即进入工作状态，会接收 dst\_id 与自己的 ID 相匹配的时间请求数据包，并返回应答包。



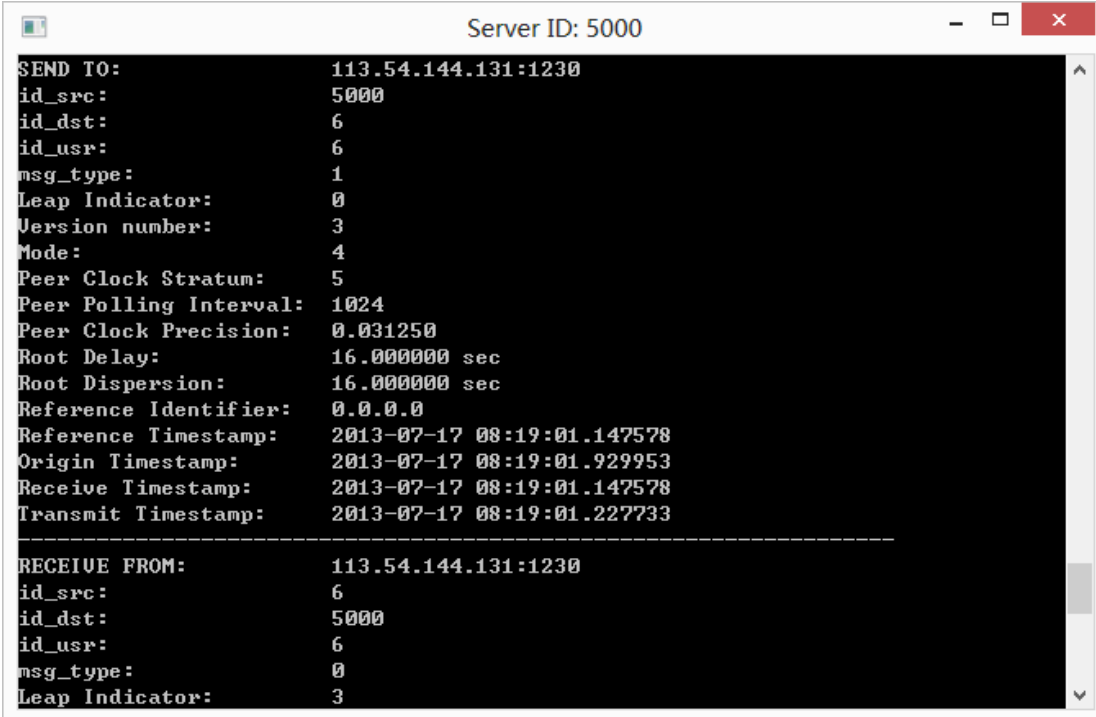
#### 5.2.4 统计信息显示

按下 Alt+S，服务端会显示相关的统计信息。统计信息包括本进程发送了多少条消息，接收了多少条消息（正确的多少条，错误的多少条）。



## 5.2.5 详细信息显示

按下 Alt+D，服务端会将自己接收/发送的每一个消息，都实时显示在屏幕上。



```
Server ID: 5000
SEND TO:          113.54.144.131:1230
id_src:           5000
id_dst:           6
id_usr:           6
msg_type:         1
Leap Indicator:   0
Version number:   3
Mode:             4
Peer Clock Stratum: 5
Peer Polling Interval: 1024
Peer Clock Precision: 0.031250
Root Delay:       16.000000 sec
Root Dispersion:  16.000000 sec
Reference Identifier: 0.0.0.0
Reference Timestamp: 2013-07-17 08:19:01.147578
Origin Timestamp:  2013-07-17 08:19:01.929953
Receive Timestamp: 2013-07-17 08:19:01.147578
Transmit Timestamp: 2013-07-17 08:19:01.227733
-----
RECEIVE FROM:     113.54.144.131:1230
id_src:           6
id_dst:           5000
id_usr:           6
msg_type:         0
Leap Indicator:   3
```

## 六、 LB 端操作说明

### 6.1 软件配置

在运行前需要设置与可执行文件同目录的配置文件 LB.cfg。

日志文件的格式如下（--号开头的是注释）：

```
-- LB [config file]

-- LB 的 ID （如 ID 为 1）
LB_id = 1

-- 开放给 Client 的端口（如端口 123）
client_udp_port = 123

-- 与 Server 通信的端口（如端口 1230）
server_udp_port = 1230

-- 负载均衡 DLL 文件路径
-- 三种方式："Circular.dll"、"Prorate.dll"、"Fast.dll"
-- DLL 加载失败时使用内置轮转算法
Balance = "Circular.dll"

-- 会话保持方式
-- 两种方式："src_id"、"usr_id"
-- 设置为其他字符串可关闭此功能
Keep = "off"

-- 以下重复指明每一个 Server 的 ID、IP、通信端口、权重
-- 增减服务器时添加/删除一项或多项服务器信息即可
-- 服务器信息是由大括号括起的一组 ID、IP、通信端口、权重
servers = {
    {
        id      = 5000,
        ip      = "127.0.0.1",
        port    = 5000,
        weight  = 3,
    },
}
```

```
{
    id   = 5001,
    ip    = "127.0.0.1",
    port  = 5001,
    weight = 4,
},
{
    id   = 5002,
    ip    = "127.0.0.1",
    port  = 5002,
    weight = 5,
},
.....
}
```

## 6.2 软件运行

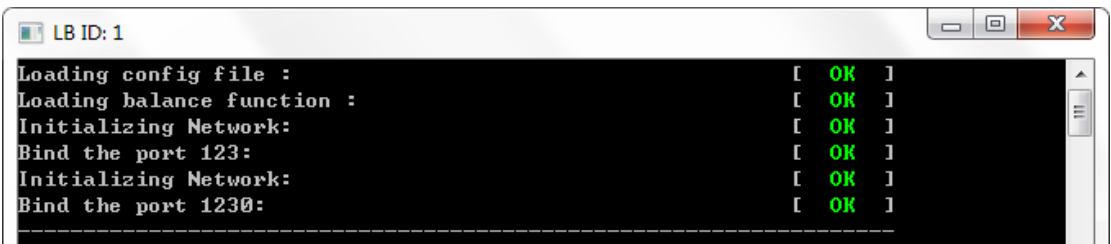
### 6.2.1 启动说明

LB 端不需要启动参数，软件运行后会读取同目录的 LB.cfg 文件并进行初始化工作，请确保 LB.cfg 文件已经配置正确。

### 6.2.2 操作说明

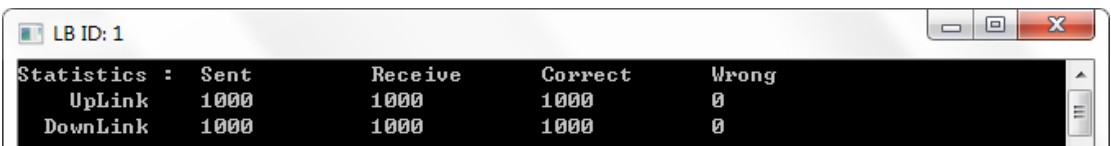
运行 LB 端后，软件初始化完成即进入工作状态，会接收 dst\_id 与自己的 ID 相匹配的时间请求数据包，并按照轮转算法选出一个服务端，将时间请求消息中的 dst\_id 改成此服务端的 id 后，将消息通过 server\_udp\_port 分发给该服务端处理。

当 LB 端通过 server\_udp\_port 接收到客户端的“时间应答”消息后，将消息中的 src\_id 改成自己的 ID，然后将消息通过 client\_udp\_port 发送给消息中的 dst\_id 所指示的客户端。



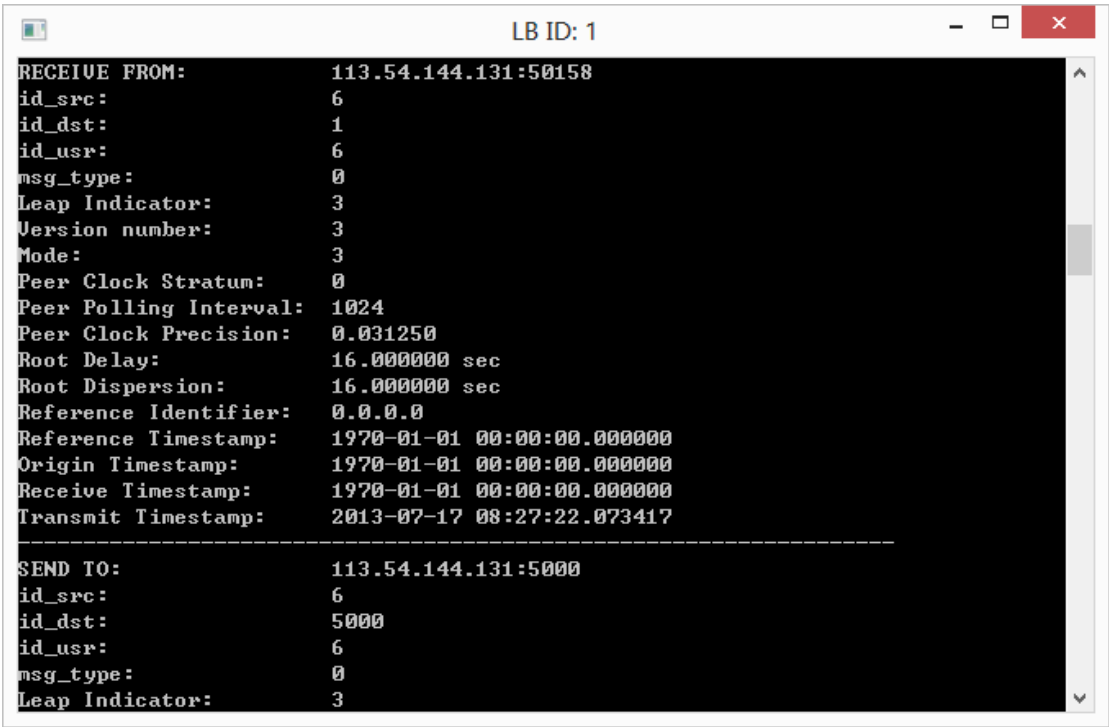
### 6.2.3 统计信息显示

按下 Alt+S，LB 端会显示本进程从客户端接收了多少条消息（正确的多少条，错误的多少条），向客户端发送了多少条消息，从服务端接收了多少条消息（正确的多少条，错误的多少条），向服务端发送了多少条消息。



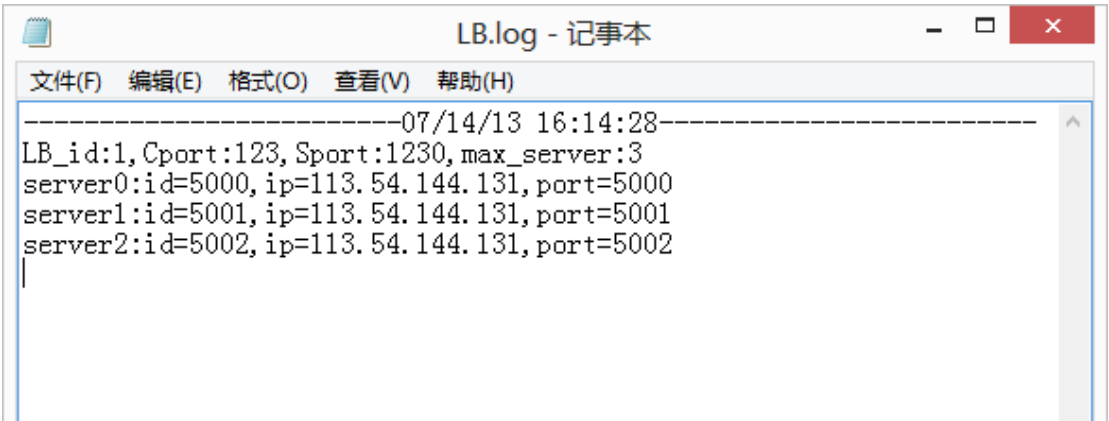
### 6.2.4 详细信息显示

按下 Alt+D, LB 端会将自己接收/发送的每一个消息, 都实时显示在屏幕上。



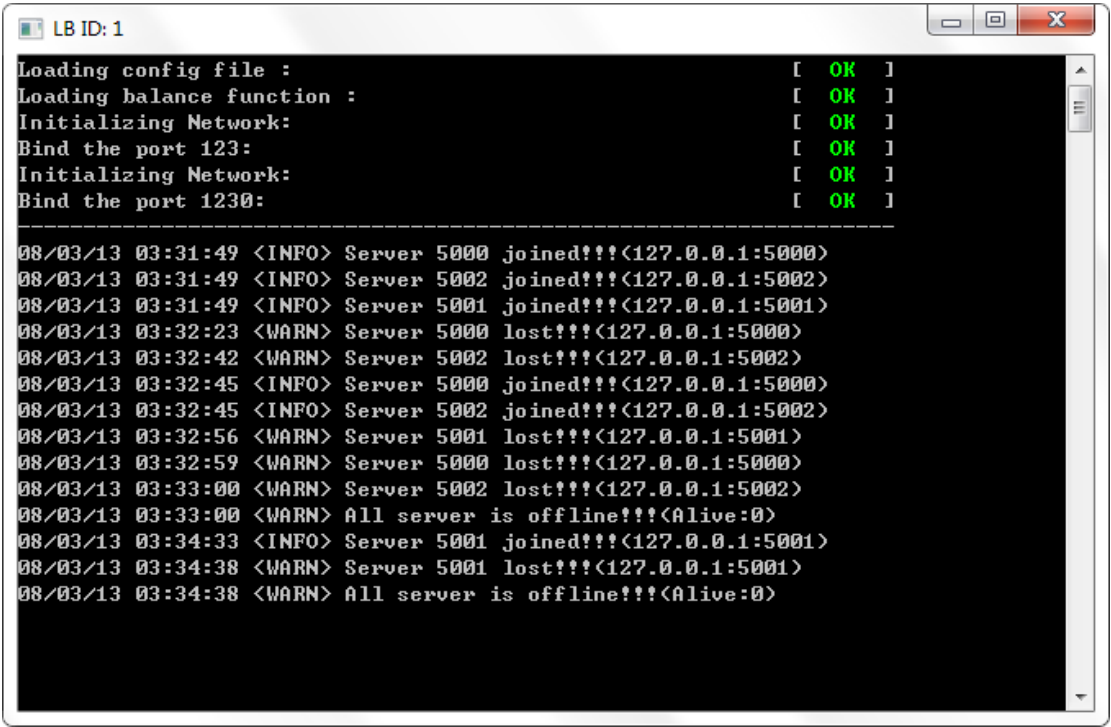
### 6.2.5 运行日志

在运行过程中, 如果出现异常事件 (如 UDP 接收、发送失败等), LB 端将会记录详细的日志信息。



### 6.2.6 健康检测

负载均衡器每隔 0.5 秒,向每个服务端发送一个心跳报文。如果连续 4 次收不到某个服务端的心跳响应,就认为服务端出现了故障。后续对时间请求消息做负载均衡时,就不再分发给此服务端处理。



### 6.2.7 会话保持

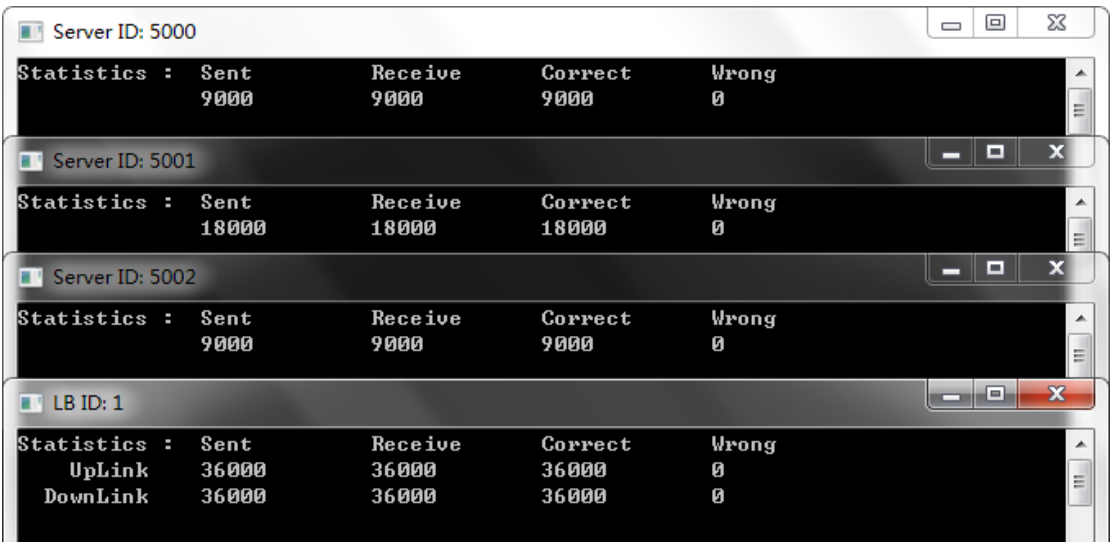
#### 基于 src\_id/usr\_id 的会话保持

启用此功能后，收到来自某个客户端的某条时间请求消息时：

如果发现没有与此客户端对应的会话条目，则认为此消息开启了一条全新的会话，并且此消息是该会话的第一条消息，此时需要将本条时间请求消息按照当前使用的负载均衡算法分发到某个服务端 x 上，同时创建一个新的会话条目，在其中记录此消息的 src\_id/usr\_id 及服务端 x 的 id；

如果发现有与此客户端对应的会话条目，则认为此消息是该会话上的后续消息，此时将此消息分发到会话条目标记的服务端上即可。

如果一条会话连续 30 秒没有任何时间请求消息，则删除此会话条目。



## 6.2.8 负载均衡

### 轮转算法

Server ID: 5000	Statistics : Sent	Receive	Correct	Wrong
	12000	12000	12000	0
Server ID: 5001	Statistics : Sent	Receive	Correct	Wrong
	12000	12000	12000	0
Server ID: 5002	Statistics : Sent	Receive	Correct	Wrong
	12000	12000	12000	0
LB ID: 1	Statistics : Sent	Receive	Correct	Wrong
UpLink	36000	36000	36000	0
DownLink	36000	36000	36000	0

### 按比例分发的负载均衡算法

由于不同的服务器，配置不同，处理能力不同。因此，有时候将请求消息，根据各服务端的处理能力，按照一定的比例分发到各个服务端，会有更好的整体表现。

按比例分发的负载均衡算法，为每个服务端配置一个权重值（1 到 10 之间的一个数字），在配置文件进行配置。

如果启用了此负载均衡算法，负载均衡程序，将按权重值比例，将时间请求消息分发到相应的服务端。

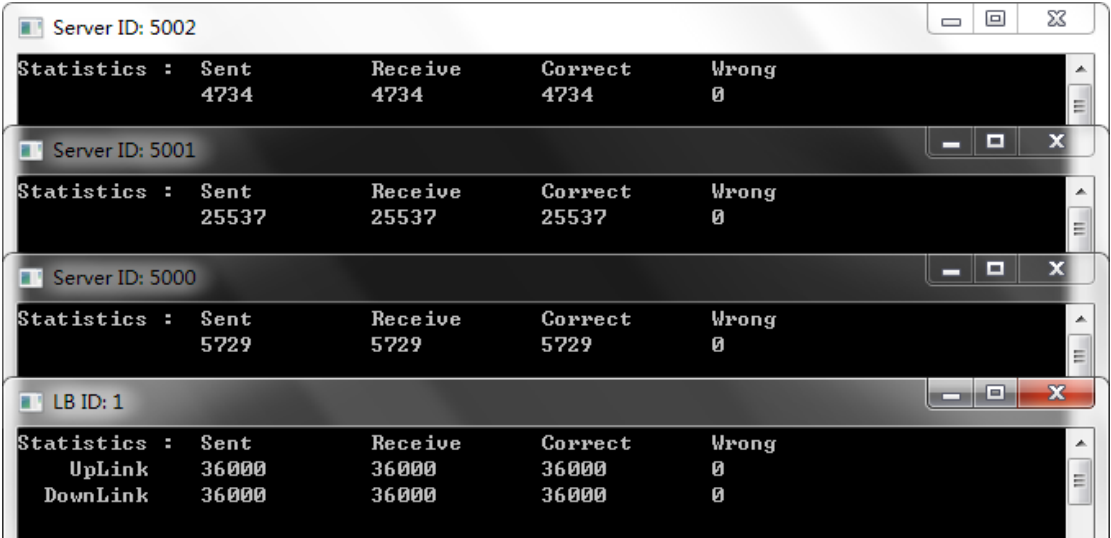
Server ID: 5000	Statistics : Sent	Receive	Correct	Wrong
	9000	9000	9000	0
Server ID: 5001	Statistics : Sent	Receive	Correct	Wrong
	12000	12000	12000	0
Server ID: 5002	Statistics : Sent	Receive	Correct	Wrong
	15000	15000	15000	0
LB ID: 1	Statistics : Sent	Receive	Correct	Wrong
UpLink	36000	36000	36000	0
DownLink	36000	36000	36000	0

### 基于最快响应的负载均衡算法

某些情况下，对服务请求的响应速度非常重要。在这种情况下，将服务请求分发给响应速度最快的服务端处理比较合适。

为了支持该功能，负载均衡进程，需要对各个服务端响应心跳请求的速度进行跟踪，从而能够找出响应最快的服务端。

如果启用了此负载均衡算法，负载均衡程序，总是将时间请求消息分发到响应最快的服务端。





# 七、 附录

## 7.1 BAT 脚本语法简介

i 的取值从 6 到 10, 步进为 1  
即要运行 5 个该程序的实例

要运行的程序路径

依次指定参数  
%%i 就是把 i 的值作为参数

```
cd Release  
for /l %%i in (6 1 10) do start client.exe %%i %%i 1 10 192.168.1.1 123
```

## 7.2 VBS 脚本语法简介

i 的取值从 6 到 6  
即要运行 1 个该程序的实例

要运行的程序路径

指定参数

运行方式  
1 为正常运行

```
for i = 6 to 6  
createobject("wscript.shell").run "Release\server.exe 5000 1 5000",1  
Next
```

i 的取值从 6 到 10

要运行的程序路径

指定参数  
(&是 vbs 的字符串连接符)

运行方式  
2 为最小化运行

```
for i = 6 to 10  
createobject("wscript.shell").run "Release\client.exe "&i&" "&i&" 1 10 192.168.1.1 123",2  
Next
```