|  |
| --- |
| 意达康通信科技 |
| **ixia高层API帮助手册** |
| IxLoad篇 v1.0 |







|  |
| --- |
| www.ixiacom.cn  2016/7/19 |

# 目录

[目录 3](#_Toc456733851)

[1 搭建自动化环境 4](#_Toc456733852)

[1.1 Tcl解释器 4](#_Toc456733853)

[1.2 Windows客户端 4](#_Toc456733854)

[1.3 Linux执行机 4](#_Toc456733855)

[2 开始编写脚本 5](#_Toc456733856)

[2.1 引用库目录 5](#_Toc456733857)

[3 全局方法 6](#_Toc456733858)

[3.1 连接本地IxLoad库 6](#_Toc456733859)

[3.2 断开本地IxLoad库 6](#_Toc456733860)

[3.3 加载配置文件 7](#_Toc456733861)

[3.4 重启端口 8](#_Toc456733862)

[图目录 9](#_Toc456733863)

[表格目录 9](#_Toc456733864)

# 搭建自动化环境

## Tcl解释器

Tcl 8.5及以上版本

## Windows客户端

IxLoad客户端或Windows 库文件压缩包

## Linux执行机

Linux独立安装包或压缩包

# 开始编写脚本

## 引用库目录

引用IxLoad高层API库的方法与引用普通tcl库的方式一致，需要将IxLoad高层API库所在的目录加入auto\_path；

库目录的下载地址<https://github.com/jundong/ixia-ixLoad-ixRepository-API>

### 语法

|  |
| --- |
|  |

package require IxRepository

### 参数

|  |
| --- |
|  |

无

### 示例

|  |
| --- |
|  |

假设你的库文件保存在”c:\temp\IxNetwork HLAPI”目录下；在tcl语言中，需要将”\”换为”/”，或使用”{}”；

**lappend auto\_path “C:/Ixia/Workspace/ixia-ixLoad-ixRepository-API”**

**package require IxRepository**

# 全局通用方法

## 连接本地IxLoad库

### 语法

|  |
| --- |
|  |

IXIA::connect

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

无

### 示例

|  |
| --- |
|  |

IXIA::connect

## 断开本地IxLoad库

### 语法

|  |
| --- |
|  |

IXIA::disconnect

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

无

### 示例

|  |
| --- |
|  |

IXIA::disconnect

## 加载配置文件

### 语法

|  |
| --- |
|  |

IXIA::loadRepository repPath

### 参数

|  |
| --- |
|  |

repPath：Repository文件的绝对路径

### 返回值

|  |
| --- |
|  |

成功：Repository的对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::loadRepository "[pwd]/configs/HTTP.rxf"

## 保存配置文件

### 语法

|  |
| --- |
|  |

IXIA::save repPath [overwrite]

### 参数

|  |
| --- |
|  |

repPath：保存的Repository文件的绝对路径

overwrite：可选，标识是否覆盖本地已有的同名文件

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::save"[pwd]/configs/HTTP.rxf" 1

## 应用配置

### 语法

|  |
| --- |
|  |

IXIA::apply

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::apply

## 运行测试

### 语法

|  |
| --- |
|  |

IXIA::run args

### 参数

|  |
| --- |
|  |

args: |key, value|

- resultDir : 测试结果存放路径，默认是当前脚本所在目录

- forceOwnership: 标识是否强制占用端口，默认是true

- resetPorts: 标识是否在测试前reset端口，默认是true

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::run

## 开始抓包

### 语法

|  |
| --- |
|  |

IXIA::startCapture args

### 参数

|  |
| --- |
|  |

args: |key, value|

- resultDir : 测试结果存放路径，默认是当前脚本所在目录

- bufferMemoryPercentage: 抓包文件大小，默认是测试端口内存的50%

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: startCapture

## 开始Hal抓包

### 语法

|  |
| --- |
|  |

IXIA::startHalCapture args

### 参数

|  |
| --- |
|  |

args: |key, value|

- mode: 抓包模式， 0：抓取所有报文， 1：只抓取过滤报文，默认是0，但是目前实现不支持

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: startHalCapture

## 停止Hal抓包

### 语法

|  |
| --- |
|  |

IXIA::stopHalCapture

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: stopHalCapture

## 保存Hal报文

### 语法

|  |
| --- |
|  |

IXIA::saveHalCapture args

### 参数

|  |
| --- |
|  |

args: |key, value|

- prefix: 保存的enc文件的前缀，最终得到的文件名字是：前缀\_机框ID\_板卡ID\_端口ID.enc

- resultDir: 保存enc文件的路径，默认为当前测试的结果目录

- user: 远程报文共享目录访问权限的用户

- password: 远程报文共享目录访问权限的密码

- pcap2xml: 标识是否直接把enc文件转换成xml文件，由于转换过程较慢，所以默认设置为false，保存的xml文件名为：前缀\_机框ID\_板卡ID\_端口ID.xml

- keep\_pcap: 是否保存当前抓包文件，默认值为true，默认保存的文件名为：前缀\_机框ID\_板卡ID\_端口ID.enc

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: saveHalCapture –prefix “Ixia1”

## 终止测试

### 语法

|  |
| --- |
|  |

IXIA::stop

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::stop

## 等待测试结束

### 语法

|  |
| --- |
|  |

IXIA::waitForTestStop

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::waitForTestStop

## 生成报告

### 语法

|  |
| --- |
|  |

IXIA::generateReport deltailedReport format

### 参数

|  |
| --- |
|  |

deltailedReport: 可选，标识是否生成详细报告，默认为1

format：可选，报告格式，默认为”PDF;HTML”

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: generateReport 1 “PDF;HTML”

## 选择测试统计数据

### 语法

|  |
| --- |
|  |

IXIA::selectStats statList portList [interval]

### 参数

|  |
| --- |
|  |

statList: 协议的数据统计列表，具体参考文件stats.tcl

interval：可选，数据统计上报时间间隔，最小时间是1秒

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

set statlist [list http\_client\_throughput]

IXIA::selectStats $statlist

## 获取协议统计数据

### 语法

|  |
| --- |
|  |

IXIA::getInstantStats proItem

### 参数

|  |
| --- |
|  |

proItem: 指定要取的数据统计的属性名字

### 返回值

|  |
| --- |
|  |

成功：返回统计数据值

失败：抛出异常

### 示例

|  |
| --- |
|  |

set statlist [list http\_client\_throughput]

IXIA::selectStats $statlist

………………………………….

IXIA::getInstantStats http\_client\_throughput

## 打开Log功能

### 语法

|  |
| --- |
|  |

IXIA::IxDebugOn [logPath]

### 参数

|  |
| --- |
|  |

logPath: 可选，测试log存放的路径，默认为：C:/Windows/Temp/ixlogfile

### 返回值

|  |
| --- |
|  |

成功：无

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::IxDebugOn C:/Windows/Temp/ixlogfile

## 关闭Log功能

### 语法

|  |
| --- |
|  |

IXIA::IxDebugOff

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

成功：无

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::IxDebugOff

## 重启端口

### 语法

|  |
| --- |
|  |

IXIA::reboot chasIndex portList [reset] [block]

### 参数

|  |
| --- |
|  |

chasIndex: 当前机框在机框链中的序号

portList：要重启的端口数组

reset：可选，标识是否重置至出厂默认配置

block：可选，标识是否同步模式重启端口

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::reboot 1 [list "172.16.174.133/1/1"]

## 配置Repository

### 语法

|  |
| --- |
|  |

IXIA:: configRepository args

### 参数

|  |
| --- |
|  |

-chassis: 当前机框IP或机框名字

-user：登录机框的用户名

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: configRepository –chassis 172.16.174.134 –user ixiaUser

## 清除机框链

### 语法

|  |
| --- |
|  |

IXIA:: clearChassisChain chassisChain

### 参数

|  |
| --- |
|  |

chassisChain: 要删除的机框对象

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

set chassisChain [ $IXIA::repository cget -chassisChain ]

IXIA:: clearChassisChain $chassisChain

## 获取当前Active测试对象

### 语法

|  |
| --- |
|  |

IXIA::getActiveTest

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

成功：返回当前Active测试对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActiveTest

## 获取Activity

### 语法

|  |
| --- |
|  |

IXIA:: getActivity actName

### 参数

|  |
| --- |
|  |

actName: IxLoad界面上配置的Activity的名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getActivity “Activity1”

## 获取DNSServerUrl对象

### 语法

|  |
| --- |
|  |

IXIA:: getDNSServerUrl

### 参数

|  |
| --- |
|  |

无

### 返回值

|  |
| --- |
|  |

成功：返回DNSServerUrl对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getDNSServerUrl

## 获取Network对象

### 语法

|  |
| --- |
|  |

IXIA::getNetwork networkName

### 参数

|  |
| --- |
|  |

networkName：IxLoad界面上显示的Network的名字

### 返回值

|  |
| --- |
|  |

成功：返回Network对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getNetwork “Network1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 配置Network对象

### 语法

|  |
| --- |
|  |

IXIA::configNetwork networkName args

### 参数

|  |
| --- |
|  |

networkName：IxLoad界面上显示的Network的名字

args: |key, value|

-auto\_nego : 标识网络端口是否自协商true/ false

-speed : 端口设置为非自协商模式下，可以选择的值："10M","100M"，”1000M”, “10G”

-port : 端口组列表，格式：chassIP/CardIndex/PortIndex

-media : 物理媒介类型，可选： "copper","fiber"或"auto"

-gratuitous\_arp: 标识是否接收免费ARP和发送ARP响应，true/false

-gateway : 网关IP

-gatewayincrby: 网关递增值，整型数据

-netmask : 网络长度，整型数据

-ipincrby : IP递增值，整型数据

-ipcount : IP数量

-ip : IP地址

-vlan\_id : Vlan ID，整型数据

-vlancount :Vlan数量

-vlanincrby: Vlan ID递增值，整型数据

-mac : Mac地址

-maccount : Mac数量，整型数据

-macincrby : Mac递增值，整型数据

-dns\_domain: DNS 域

-dns\_server: DNS 服务器地址/名字

-ipsec\_remote\_gateway: IPSec远端网关地址

-ipsec\_local\_gateway : IPSec本地网关地址

-enable : 使能Network, true/false：

-

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::configNetwork Network1 -mac 00:01:01:01:01:01 \

-vlan\_id 100 \

-ip 192.168.1.1 \

-ipcount 10 \

-ipincrby 0.0.0.1 \

-netmask 24 \

-gateway 192.168.1.10 \

-gratuitous\_arp true \

-dns\_domain www.huawei.com \

-dns\_server 10.70.80.90

## 获取RampupTime对象

### 语法

|  |
| --- |
|  |

IXIA::getActivityRampupTime actName args

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity RampupTime对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivityRampupTime “HTTPClient1”

## 获取Timeline Type对象

### 语法

|  |
| --- |
|  |

IXIA::getActivityTimelineType actName args

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity TimelineType对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivityTimelineType “HTTPClient1”

## 获取Sustain Time对象

### 语法

|  |
| --- |
|  |

IXIA:: getActivitySustainTime actName

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 配置Activity Timeline

### 语法

|  |
| --- |
|  |

IXIA::configActivityTimeline actName args

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

args： |key, value|

- rampupvalue：Ramp UP值，仅当Timeline Type是0才可设置

- rampuptype：取值为：0/1/2，含义是：<users interval >,<max pending users >and <Smooth users/ interval >，仅当Timeline Type是0才可设置

- rampdowntime：Ramp Down Time值，仅当Timeline Type是0才可设置

- rampdownvalue：Ramp Down Value值，仅当Timeline Type是0才可设置

- rampdowntime：Ramp Down Time值，仅当Timeline Type是0才可设置

- iterations：Iteration值，仅当Timeline Type是0才可设置

- rampupinterval：Ramp UP Interval 值，仅当Timeline Type是0才可设置

- sustaintime：Sustain Time值，仅当Timeline Type是0才可设置

-name：Timeline对象的名字（通过名字配置对象）

-timelinetype：取值：0/1，0代表基础模式，1代表高级模式

-segment0duration：Ramp UP Time值，仅当Timeline Type是1才可设置

-segment1duration：Ramp Sustain Time值，仅当Timeline Type是1才可设置

-segment2duration：Ramp Down Time值，仅当Timeline Type是1才可设置

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::configActivityTimeline “HTTPClient1” –rampupvalue 100 –timelinetype 0

## 配置Objective对象

### 语法

|  |
| --- |
|  |

IXIA::configObjective actName args

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 获取Sustain Time对象

### 语法

|  |
| --- |
|  |

IXIA:: getActivitySustainTime actName

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

args： |key, value|

-enableconstraint：使能Constraint选项

-constraintvalue：Constraint值，最小值为1

-userobjectivetype：Objective类型，可选：simulatedUsers，connectionRate，connectionAttemptRa，transactionRate，concurrentSessions，throughputKbps，throughputMbps，throughputGbps

-userobjectivevalue：Objective值

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 获取Sustain Time对象

### 语法

|  |
| --- |
|  |

IXIA:: getActivitySustainTime actName

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 获取Sustain Time对象

### 语法

|  |
| --- |
|  |

IXIA:: getActivitySustainTime actName

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 获取Sustain Time对象

### 语法

|  |
| --- |
|  |

IXIA:: getActivitySustainTime actName

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 获取Sustain Time对象

### 语法

|  |
| --- |
|  |

IXIA:: getActivitySustainTime actName

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 获取Sustain Time对象

### 语法

|  |
| --- |
|  |

IXIA:: getActivitySustainTime actName

### 参数

|  |
| --- |
|  |

actName：IxLoad界面上显示的Activity名字

### 返回值

|  |
| --- |
|  |

成功：返回Activity Sustain Time对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::getActivitySustainTime “HTTPClient1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

## 获取Traffic对象

### 语法

|  |
| --- |
|  |

IXIA::getNetTraffic trafficName

### 参数

|  |
| --- |
|  |

trafficName：IxLoad界面上显示的Traffic名字

### 返回值

|  |
| --- |
|  |

成功：返回Traffic对象

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA:: getNetTraffic “Traffic1”

# HTTP配置方法

## 配置HTTP Client Action

### 语法

|  |
| --- |
|  |

IXIA::configHttpClientAction actName clrflag args

### 参数

|  |
| --- |
|  |

actName：HTTP Activity的名字

clrflag：标识是否清楚Action List

args： [key, value]

- commandtype：命令类型，可以是："GET","DELETE","HEAD","POST"或"PUT"

- cmdname：命令的名字

- pageobject：Page对象，如：/128k.html

- destination：Server地址，如：http://1.1.1.1:90

- loopcount：循环次数

- thinkinterval：命令之间的等待时间

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::configHttpClientAction HTTPClient1 0 -commandType GET \

-cmdName GET\_0 \

-destination http://10.70.80.90:90 \

-pageObject /1k.html

## 配置HTTP Client Agent

### 语法

|  |
| --- |
|  |

IXIA::HttpClientAgent actName clrflag args

### 参数

|  |
| --- |
|  |

actName：HTTP Activity的名字

args： [key, value]

- enablessl：使能ssl，true/false

-certificate：ssl认证证书，完整证书字符串

-privatekey：ssl证书私有秘钥，完整的秘钥字符串

-enablecookie：使能cookie功能

- httpversion：指定HTTP版本号，可选值为：0/1

- maxpersistentrequests：Transaction/TCP, 默认为0

- keepalive：标识是否使能keepalive功能，true/false

- maxsessions：TCP/User，默认为1

- esm：有效的MSS值，默认为64

- maxpipeline：最大的PIP line值，默认1

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::configHttpClientAgent HTTPClient1 -enableSsl true \

-httpVersion 0 \

-maxPersistentRequests 0 \

-keepAlive true \

-maxSessions 44 \

-esm 128 \

-maxPipeline 33

## 配置HTTP Server WebPage

### 语法

|  |
| --- |
|  |

IXIA::configHttpServerWebPage actName args

### 参数

|  |
| --- |
|  |

actName：HTTP Activity的名字

clrflag：标识是否清除WebPage List

args： [key, value]

- page：页名字，如：/1k.html

- payloadSize：页的大小，默认为1024

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::configHttpServerWebPage HTTPServer1 0 -page /1k.html \

-payloadsize 1024

## 配置HTTP Server Agent

### 语法

|  |
| --- |
|  |

IXIA::configHttpServerAgent actName clrflag args

### 参数

|  |
| --- |
|  |

actName：HTTP Activity的名字

args： [key, value]

- httpport：HTTP 服务器端口，默认为80端口

- acceptsslconnections：使能ssl， true/false

-certificate：ssl认证证书，完整证书字符串

-privatekey：ssl证书私有秘钥，完整的秘钥字符串

### 返回值

|  |
| --- |
|  |

成功：0

失败：抛出异常

### 示例

|  |
| --- |
|  |

IXIA::configHttpServerAgent HTTPServer1 -httpPort 81 \

-acceptSslConnections true

# 图目录

[图 1 打开IxLoad服务 7](#_Toc389835380)

# 表格目录

[表格 1 端口配置参数列表 15](#_Toc389835363)