liaohui

上海交通大学 | 上海市闵行区东川路800号电院3-129

网络性能测试平台

version3

目录

[1 项目背景 2](#_Toc404533012)

[2 建设目标与内容 3](#_Toc404533013)

[2.1 建设目标 3](#_Toc404533014)

[2.2建设内容 3](#_Toc404533015)

[2.2.1 主动测量 3](#_Toc404533016)

[2.2.2被动测量 5](#_Toc404533017)

[2.2.3其他测量 6](#_Toc404533018)

[2.3性能指标 6](#_Toc404533019)

[3 系统总体设计方案 8](#_Toc404533020)

[3.1总体设计 8](#_Toc404533021)

[3.1.1系统总体框架 8](#_Toc404533022)

[3.1.2系统网络拓扑 9](#_Toc404533023)

[3.2系统功能设计 10](#_Toc404533024)

[3.2.1功能模块划分 10](#_Toc404533025)

[3.2.2主动测试功能模块 12](#_Toc404533026)

[3.2.3被动测量功能模块 17](#_Toc404533027)

[3.2.4其他测量功能模块 17](#_Toc404533028)

[3.2.5前端显示功能模块 18](#_Toc404533029)

[3.3技术路线 18](#_Toc404533030)

[3.3.1运行平台 18](#_Toc404533031)

[3.2.2 代码实现 19](#_Toc404533032)

[4系统功能显示 23](#_Toc404533033)

[4.1主动测试功能显示 23](#_Toc404533034)

[4.2被动测量功能显示 26](#_Toc404533035)

[4.3其他测量方法功能显示 27](#_Toc404533036)

# 项目背景

上海教育宽带网以各高校和17个区县教育网为主，主干构架在教育专用光缆之上，拥有自主管理权，可以根据设备不同、应用需求的变化、以及发展的需要，方便灵活地进行提升带宽。一方面，上海教育宽带网使用静态路由的方式以20Gbps的带宽实现与中国教育科研网的互通；通过2Gbps带宽实现与上海交互中心互通，1Gbps带宽与上海电信运营商互联，1Gbps带宽与上海移动互联，1Gbps带宽与东方有线互联，另一方面，宽带网上目前开展了各类应用，如MPLS VPN，视频会议，跨校认证，无线通等，不同的业务对网络的需求目标也不一样，有些业务需要低延迟，有些业务需要高带宽，宽带网的性能是否足够是支撑各种业务是否能够顺利展开的前提条件，因此迫切的需要一套网络性能测量矩阵平台，对全网的延迟、抖动、QoS、可用带宽等进行测量，依照平台的部署点构建测量矩阵，以便运维人员和宽带网用户及时方便的了解网络性能，为提高业务的服务质量提供相应的依据。

# 建设目标与内容

## 建设目标

为实现上海教育宽带网的网络性能测量矩阵平台，达到以下目标：

1. 设计一套能够测试上海教育宽带网当前网络状况的测试工具；
2. 能够对全网的延迟、抖动、QoS以及可用带宽等标量进行测试；
3. 能够利用矩阵形式显示各种网络状况，保证运维人员和宽带用户及时方便的了解网络性能；
4. 保证平台的功能完善、数据安全以及性能稳定；
5. 系统总体框架的设计合理性，保证拥有良好的扩展性，对系统功能的增减等不需要做大量的修改；
6. 拥有友好的前台界面，操作简单易懂，保证良好的人机交互性；
7. 平台可以支持分布式部署，可以按照规模发展进行灵活；本期项目需要支持4个以上测量点的部署；
8. 平台必须支持中文界面；需提供可编程接口；支持可读性良好的故障报告和趋势报告。

## 2.2建设内容

本系统必须支持主动测量和被动测量等多种测量方式。

### 2.2.1 主动测量

主动测量方法需要通过主动向网络中注入探测数据包，并收集、分析探测包在网络中传输后发现的状态变化，从而计算出要测量的网络性能参数。例如，可以向网络发送数据包并不断提高发送速率直至网络饱和，以此来测量网络的最大负载能力。

主动测量的主要优点是不依赖于被测对象的测量能力。但另一方面，这种测量会给网络

增加额外的通信流量，这在一定程度上也可能影响测量的结果。所以应该考虑试图进行的测

量对测量结果产生的影响，并尽量使这种影响降到最低。

主动测量方法的工作原理图如图2-1所示。

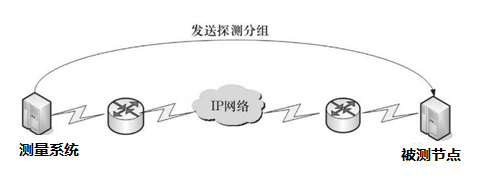


图2-1 主动测量工作原理示意图

**主动测量需要实现的主要内容包括：**

**1. 基于ICMP的方法**

通过向网络中发送某种特定消息类型的ICMP探测包，根据网络做出的不同反应，对网络状态进行测量和判断。常见的ICMP的消息类型如表2-1所示。

表2-1 常见ICMP的消息类型

|  |  |
| --- | --- |
| 消息类型 | 消息功能 |
| 类型0，8 | 询问和询问应答 |
| 类型3 | 目标不可达 |
| 类型4 | 阻塞源 |
| 类型5 | 重定向 |
| 类型11 | 超时 |
| 类型12 | 参数错误 |
| 类型13,14 | 时间戳和时间戳应答 |
| 类型15,16 | 信息请求和信息应答 |

**2.基于TCP的方法**

TCP协议是面向连接的，通过测试TCP的性能能够反映发送端与接收端的端与端之间的性能参数，如：

1).重传个数

2).建立和关闭TCP连接的时间

3).平均段大小

4).吞吐率

利用TCP主动探测过程如下：

1).搜索活跃的IP设备，以确定探测目的地，然后从源端向被探测目的设备上的一个端口发送TCP SYN包，进行TCP握手。

2).目标设备接受TCP SYN包，如果目的端口未开放，目的设备将返回一个RST包，断开连接；否则会进行握手，向源端发送SYN|ACK包。

3).源端接收到SYN|ACK 包后，并不建立一个完整的连接，而是发送RST包断开这次TCP连接，完成一次测量。

**3.基于UDP的方法**

基于UDP的主动测量过程中，测量主机需要向目的主机发送探测数据包，但通信双方之间的传输类似于传递邮件，消息从一方发送到另一方，两者之间没有明确的连接，通信双发是对等的，单词传输的最大数据量取决于具体的网络。

基于UDP协议的主动测量对网络中的各项性能指标进行监督，主要有：

1).单向时延

2).往返时延

3).丢包率

4).带宽

### 2.2.2被动测量

被动测量是在网络的测量点处布置数据采集器，收集流经该点的网络流量。测量软件或系统周期性地轮询被动监测设备并采集信息，以判断网络性能和状态。收集到的数据可以用于进行各种流量分析，包括流量中各种应用成分的分析，报文长度分布分析，按报文到达的时间分析流量的规律，分析网络的利用率以及各种出错报文的大小分布等。被动测量的工作原理如图2-2所示。

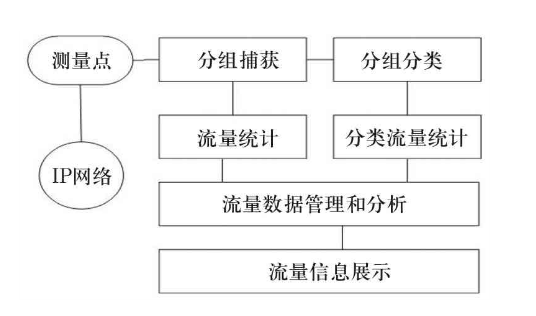


图2-2 被动测量的工作原理

被动测量可以通过三种方式获得：

服务器端测量：通常是在服务器端安装测试代理，实时监测服务器的性能、资源使用等状况。

用户端测量：将监测功能封装到客户应用中，从特定用户的角度实时监测相关

的业务性能。

利用网络探针：网络探针可用于监测网络传输状态，分析捕获的数据包，以实现对网络及相关业务的测量。

**被动测量方法需要实现的主要内容包括：**

**1.基于SNMP的方法**

SNMP是TCP/IP协议簇中的网络管理协议，定义了传送管理信息的协议消息格式及管理站和设备代理之间进行消息传送的规程，能对IP网络中不同类型的设备进行监控和管理。

SNMP协议定义了五种操作原语，通过这几种操作原语就可以实现对MIB（管理信息库）中的数据进行读写等操作，获得网络的相关信息。主要入表2-2所示。

表2.2 SNMP管理者对MIB中的信息执行的操作原语

|  |  |
| --- | --- |
| 原语 | 含义 |
| GetRequest | 管理站请求从被管理对象的MIB中读取一个信息 |
| GetNextRequest | 管理站请求从被管对象的MIB中读取几个相邻信息 |
| SetRequest | 管理站用于对被管对象的MIB中的消息进行设置 |
| GetResponse | 管理代理对GetRequest、GetNextRequest、SetRequest的相应 |
| Trap | 管理代理用于向管理站报告被管对象的状态变化和故障 |

在基于SNMP的被动测量方法中，每个管理代理统计测量网络局部性能数据，然后管理者通过轮询或陷阱方式从代理处获取全网的性能状况。

**2.基于数据包捕获的方法**

基于数据包捕获的被动测量方法是运用特定的软件或硬件捕获流经网络设备或者端口的数据包，并根据网络数据包所属的协议的格式对数据包进行解析和分类，然后按一定的格式进行统计、汇总，得到测量结果。

## 2.2.3其他测量

**1.下载测试**

在每个点部署三种大小的文件，供用户进行TCP下载测试。用于测量网络的下行速度。

**2.上传测试**

支持用户进行文件的上传测试。用测试当前网络的上行速度。

## 2.3性能指标

**1.主动测量**

1).带宽。单位时间内物理通路理论上所能传送的最大比特数。

2).拥塞。当前网络的拥塞情况

3).延迟。数据包离开源点的时间与到达目的点的时间的时间间隔，即数据包在网络传输中的延时时间。

4).抖动。数据流中不同数据包时延的变化。

5).丢包。某一时间段内被测网络传输及处理中丢失或出错的数据包数量。

6).可用性。当前网络是否可用。即各网络组件间的互连通性，是网络性能的基本指标，反映了数据能否在各网络组件之间互相传送的属性。

**2.被动测量**

1).往返时延。即RTT，表示从发送端发送数据开始，到发送端收到来自接收端的确认（接收端收到数据后便立即发送确认），总共经历的时延。

2).端到端丢包。发送端到接收端的整体丢包情况

3).带宽。最大传输比特数。

4).吞吐量。单位时间内传送通过网络中给定点的数据量。

5).CPU及内存状况

**3.其他测试**

1).上行速率

2).下行速率

以上几个指标也是我们系统实现过程中需要重点关注并呈现的指标。如表2-3所示。

表2-3 测试性能指标

|  |  |  |  |
| --- | --- | --- | --- |
|  | 主动测试 | 被动测试 | 上传下载测试 |
| 性能指标 | 带宽 | 往返时延 | 上行速率 |
| 拥塞 | 端到端丢包 | 下行速率 |
| 延迟 | 带宽 |  |
| 抖动 | 吞吐率 |  |
| 丢包 | CPU及内存状况 |  |
| 可用性 |  |  |

# 3 系统总体设计方案

## 3.1总体设计

网络测量矩阵平台应该能支持主动测量和被动测量，并对连通性、吞吐率、带宽、时延、时延抖动、丢包率等多项指标进行测试，同时呈现给用户。

### 3.1.1系统总体框架

整体框架如图3-1所示。如图所示，系统可调用主动或被动测量的方法，对测量指标如吞吐率、带宽、时延等进行测量，并且用户可在前端（如Web端）选择相应的形式（如矩阵形式，柱形图形式等）进行数据显示，从而获知目前的网络状况。

用户登录系统时，系统通过与数据库数据比对，通过相应的身份认证判定用户所属的范围（如交大）。用户登录之后，可发起主动测试，被动测试，上传下载测试等等，用户所属的子服务器将代其进行相应的操作，如向所有相连的自服务器发送测试包，或者向单个服务器发送测试包等等。



图3-1 系统总体框架图

### 3.1.2系统网络拓扑

网络拓扑图如图3-2所示。网络主要可有一下几部分组成：

**1).用户**

系统用户通过互联网访问服务器(B/S结构)，服务器返回用户相应的请求信息（如请求某两个子节点之间链路的网络状况）。

**2).中心服务器**

保存当前所有链路的网络信息（如交大子节点到复旦子节点的网络状况），用户在联网的条件下可以通过浏览器对服务器进行访问，获取相应数据。

**3).身份认证**

用户登录时需进行身份认证，判断该用户属于哪个子节点服务范围。如当前用户数据交大，在登录过程中与身份认证子系统进行比对，验证成功后可与各个学校的账号进行关联（如交通大学的JAccount账号）。

**4).子节点路由**

子节点路由分别部署在各个高校，如交大、复旦等等，用于主动发起网络状况测试（发送探测包进行网络测试），或者被动测试（定时收集网络状况信息并进行数据的保存）。属于网络的关键节点。

**5).子服务器**

子服务器属于各个高校，每个高校拥有一个对应的自服务器用于保存当前网络的信息，同时定时的与中心服务器进行数据同步，保证子服务器与中心服务器数据的一致性。



图3-2系统拓扑图

## 3.2系统功能设计

### 3.2.1功能模块划分

根据功能划分，系统主要可以分为两个模块：数据显示功能模块和测量功能模块，如图3-3所示。

**1.数据显示功能模块**

**1).主动测试子模块**

i).ICMP测试模块

利用ICMP协议的方法进行主动测量。

ii).TCP测试模块

利用TCP协议的方法进行主动测量。

iii).UDP测试模块

利用UDP协议的方法进行主动测量。

**2).被动测试子模块**

i).SNMP测试模块

利用SNMP协议的方法进行被动测量。

ii).数据包捕获模块

利用数据包捕获的方法进行被动测量。

**3).其他测试子模块**

i).上传测试模块

利用文件上传进行网络测试（上传速率等）。

ii).下载测试模块

利用文件下载进行网络测试（下载速率等）。

**2.数据显示功能模块**

**1).矩阵显示模块**

利用矩阵的方式进行前端数据显示。

**2).其他显示模块**

利用其他的显示方法进行前端数据显示，如柱状图、折线图等对最近一段时间内网络状况变化进行显示。

系统的功能模块图如图3-3所示。



图3-3 系统功能模块图

### 3.2.2主动测试功能模块

用户可主动发起测试，从而获取当前网络状况，如端到端的时延，丢包等情况。主要可分为以下几种情况。

**1).当前子节点到其他所有节点的网络状况信息**

用户通过互联网访问中心服务器，申请测试当前所属子节点到其它所有子节点的网络状况信息。中心服务器向指定的子服务器发送消息，该子节点利用主动测试方法向网络中其他所有子节点发起网络性能测试，测试结束后将获取的网络状况信息发送至中心服务器进行数据保存，最终用户可获取所需的信息或数据。如图3-4所示。



图3-4主动测试当前子节点到其它节点测试图

该种情况测试时的流程可由下图所示，即时序图如图3-5



图3-5 主动测试当前子节点到其它节点时序图

对应图中的时序，可以见以下描述。

1.用户发起测试，申请测试当前所属子节点（假设当前所属子服务器为子服务器1）到网络中其它所有子节点（假设当前网络共有四个子节点）间的网络状况；

2.中心服务器通知对应的子服务器，令其发起主动测试；

3.所属子服务器发送数据包测试子服务器1与子服务器2之间的网络状况；

4. 获取子服务器1与子服务器2之间的网络状况信息；

5.所属子服务器发送数据包测试子服务器1与子服务器3之间的网络状况；

6. 获取子服务器1与子服务器3之间的网络状况信息；

7.所属子服务器发送数据包测试子服务器1与子服务器4之间的网络状况；

8. 获取子服务器1与子服务器4之间的网络信息；

9.服务器1将获取的网络状况信息发送至中心服务器进行保存；

10.中心服务器返回用户相应的网络状况数据并在WEB前端进行显示。

**2).任意两个节点之间的网络状况信息**

用户能够获取任意两个指定子服务器（可手动设置指定服务器的IP地址）之间的网络状况。用户可申请测试任意两个节点之间的网络状况信息（如子节点1和子节点2之间的网络信息），中心服务器发送消息至对应的子服务器进行网络测试，当子节点1测试完与子节点2之间的网络状况之后，发送对应的网络状况信息至中心服务器进行保存，同时中心服务器返返回用户相应的数据并在前端进行显示，可如图3-6所示。



图3-6主动测试任意两点间的网络状况

该情况下进行的网络测试流程可如下图所示，即时序图如图3-7所示。



图3-7主动测试任意两子节点间网络测试时序图

该时序图可见以下描述。

1.用户发起测试，申请测试任意两子节点间（假设为子节点1和子节点2）的网络状况；

2.中心服务器通知对应的子服务器1，令其发起主动测试；

3.所属子服务器发送数据包测试子服务器1与子服务器2之间的网络状况；

4. 获取子服务器1与子服务器2之间的网络状况信息；

5.服务器1将获取的网络状况信息发送至中心服务器进行保存；

6.中心服务器返回用户相应的网络状况数据并在WEB前端进行显示。

**3).全局的网络状况信息**

用户能够获取当前网络所有子节点之间的网络状况，并且形成矩阵图。用户发起主动测试申请，测试当前网络中所有子节点（假设当前网络存在四个子节点）之间的网络状况信息。中心服务器发送请求至所有子服务器，测试与其相连的子服务器之间的网络状况。各个子服务器将测试完的网络状况信息发送至中心服务器进行保存，最后中心服务器将相应的网络状况信息发送至用户并在前端进行显示，如图3-8 所示。



图3-8主动测试网络所有子节点间网络状况图

该情况下进行的网络测试流程可如下图所示，即时序图如图3-9所示。



图3-9主动测试所有节点间网络状况信息时序图

对应图中的时序，可以见以下描述。

1.用户发起测试，申请测试当前网络中所有子节点之间的网络状况（假设当前网络共有四个子节点，子节点1，子节点2，子节点3，子节点4）；

2.中心服务器通知所有的子服务器，令其发起主动测试；

3.子服务器1发送数据包测试子服务器1与子服务器2之间的网络状况；

4. 获取子服务器1与子服务器2之间的网络状况信息；

5.子服务器1发送数据包测试子服务器1与子服务器3之间的网络状况；

6. 获取子服务器1与子服务器3之间的网络状况信息；

7.子服务器1发送数据包测试子服务器1与子服务器4之间的网络状况；

8. 获取子服务器1与子服务器4之间的网络信息；

9.服务器1将获取的网络状况信息发送至中心服务器进行保存；

10.子服务器2发送数据包测试子服务器2与子服务器3之间的网络状况；

11. 获取子服务器2与子服务器3之间的网络状况信息；

12.子服务器2发送数据包测试子服务器2与子服务器4之间的网络状况；

13. 获取子服务器2与子服务器4之间的网络状况信息；

14.子服务器2将获取的网络状况信息发送至中心服务器进行保存；

15.子服务器3发送数据包测试子服务器3与子服务器4之间的网络状况；

16. 获取子服务器1与子服务器4之间的网络信息；

17.服务器3将获取的网络状况信息发送至中心服务器进行保存；

18.中心服务器返回用户相应的网络状况数据并在WEB前端进行显示。

### 3.2.3被动测量功能模块

在无人工干预情况下，各个子节点定时测试当前网络状况，并讲收集到的数据发送到中心服务器的数据库中进行保存（包括往返时延，端到端丢包，带宽，吞吐率，CPU及内存状况），目前初定于只保存每天每个整点时刻（12:00，1:00，2:00等等）的网络状况，当用户利用被动测量获取网络状况时，则直接从数据库中对应信息。

1).定时数据收集功能

每天整天时刻子服务器利用被动测试的相应方法被动测试当前网络状况，并保存在子服务器中，同时定时与中心服务器中数据进行同步。

2).用户获取数据功能

用户若需要进行被动测试，则可以直接从服务器中获取已经保存的网络状况信息。如图3-10所示。



图3-10 被动测量当前网络状况图

### 3.2.4其他测量功能模块

**1).上传功能模块**

系统提供用户上传功能。并根据当前网络情况实时的监听上传速率，并在前端提供给用户显示。

用户可以在系统网站中选择文件上传，系统可以通过监听文件上传，获取相应的网络状况（如上传速率等等），最终将获取的网络状况信息返回用户并在前端进项显示。

**2).下载功能模块**

系统在各个节点部署了3个不同大小的文件（1MB，10MB，100MB）的文件供用户进行下载测试。

用户可在系统网站中选择文件下载，系统则通过监听文件的下载，获取相应的网络状况（如下载速率等等），最终将获取的网络状况信息返回用户并在前端进项显示。如图3-11所示。



图3-11 上传下载测量图

### 3.2.5前端显示功能模块

根据测试方法的不同以及测量性能指标的不同，提供不同的显示方式。即设计多功能的前端显示。

1).矩阵显示功能

2).列表显示功能

3).其他显示功能

# 3.3技术路线

## 3.3.1运行平台

运行平台及要求主要如表3-1所示。

表3-1

|  |  |  |
| --- | --- | --- |
|  | 方案一 | 方案二 |
| 操作系统 | Windows | Linux |
| Web服务器 | ISS | Apache |
| 前端开发语言 | HTML5+JS+CSS | HTML5+JS+CSS |
| 后端开发语言 | C#+SQL | PHP+SQL |
| 数据库 | SQL Server | MySQL |

## 3.2.2 代码实现

在系统的网络测试中，我们利用socket进行网络通信，以下php代码实现了简单的网络通信，作为系统实现的参考。

**1.服务器端**：

<?php

//确保在连接客户端时不会超时

set\_time\_limit(0);

$ip = '127.0.0.1';

$port = 1935;

/\*

+-------------------------------

\* @socket通信整个过程

+-------------------------------

\* @socket\_create

\* @socket\_bind

\* @socket\_listen

\* @socket\_accept

\* @socket\_read

\* @socket\_write

\* @socket\_close

+--------------------------------

\*/

/\*-----------------------------------\*/

if(($sock = socket\_create(AF\_INET,SOCK\_STREAM,SOL\_TCP)) < 0) {

echo "socket\_create() 失败的原因是:".socket\_strerror($sock)."\n";

}

if(($ret = socket\_bind($sock,$ip,$port)) < 0) {

echo "socket\_bind() 失败的原因是:".socket\_strerror($ret)."\n";

}

if(($ret = socket\_listen($sock,4)) < 0) {

echo "socket\_listen() 失败的原因是:".socket\_strerror($ret)."\n";

}

$count = 0;

do {

if (($msgsock = socket\_accept($sock)) < 0) {

echo "socket\_accept() failed: reason: " . socket\_strerror($msgsock) . "\n";

break;

} else {

//发到客户端

$msg ="测试成功！\n";

socket\_write($msgsock, $msg, strlen($msg));

echo "测试成功了啊\n";

$buf = socket\_read($msgsock,8192);

$talkback = "收到的信息:$buf\n";

echo $talkback;

if(++$count >= 5){

break;

};

}

//echo $buf;

socket\_close($msgsock);

} while (true);

socket\_close($sock);

?>

**2.客户端：**

<?php

error\_reporting(E\_ALL);

set\_time\_limit(0);

echo "<h2>TCP/IP Connection</h2>\n";

$port = 1935;

$ip = "127.0.0.1";

/\*

+-------------------------------

\* @socket连接整个过程

+-------------------------------

\* @socket\_create

\* @socket\_connect

\* @socket\_write

\* @socket\_read

\* @socket\_close

+--------------------------------

\*/

$socket = socket\_create(AF\_INET, SOCK\_STREAM, SOL\_TCP);

if ($socket < 0) {

echo "socket\_create() failed: reason: " . socket\_strerror($socket) . "\n";

}else {

echo "OK.\n";

}

echo "试图连接 '$ip' 端口 '$port'...\n";

$result = socket\_connect($socket, $ip, $port);

if ($result < 0) {

echo "socket\_connect() failed.\nReason: ($result) " . socket\_strerror($result) . "\n";

}else {

echo "连接OK\n";

}

$in = "Ho\r\n";

$in .= "first blood\r\n";

$out = '';

if(!socket\_write($socket, $in, strlen($in))) {

echo "socket\_write() failed: reason: " . socket\_strerror($socket) . "\n";

}else {

echo "发送到服务器信息成功！\n";

echo "发送的内容为:<font color='red'>$in</font> <br>";

}

while($out = socket\_read($socket, 8192)) {

echo "接收服务器回传信息成功！\n";

echo "接受的内容为:",$out;

}

echo "关闭SOCKET...\n";

socket\_close($socket);

echo "关闭OK\n";

?>

# 4系统功能显示

前端界面主要可分为连通性显示、带宽显示、吞吐率显示、时延显示、丢包率显示等几个主要界面。

## 4.1主动测试功能显示

1.网络带宽情况。如图4-1所示



图4-1主动测量-网络带宽

2.网络延迟情况，如图4-2所示。



图4-2 网络延迟情况

3.网络可用性情况，如图4-3所示。



图4-3 网络可用性图

4.丢包情况如图4-4所示。



图4-4 网络丢包情况

5.当前节点到其他子节点的网络状况情况如图4-5所示。



图4-5当前节点到其余子节点网络情况

6.主动测试端到端的网络情况。如交大子节点到复旦子节点的网络状况，如图4-6所示。



图4-6 主动测量端到端网络情况

## 4.2被动测量功能显示

被动测量网络状况如图4-7所示。



图4-7被动测量网络状况

## 4.3其他测量方法功能显示

上传下载测试如图4-8所示。

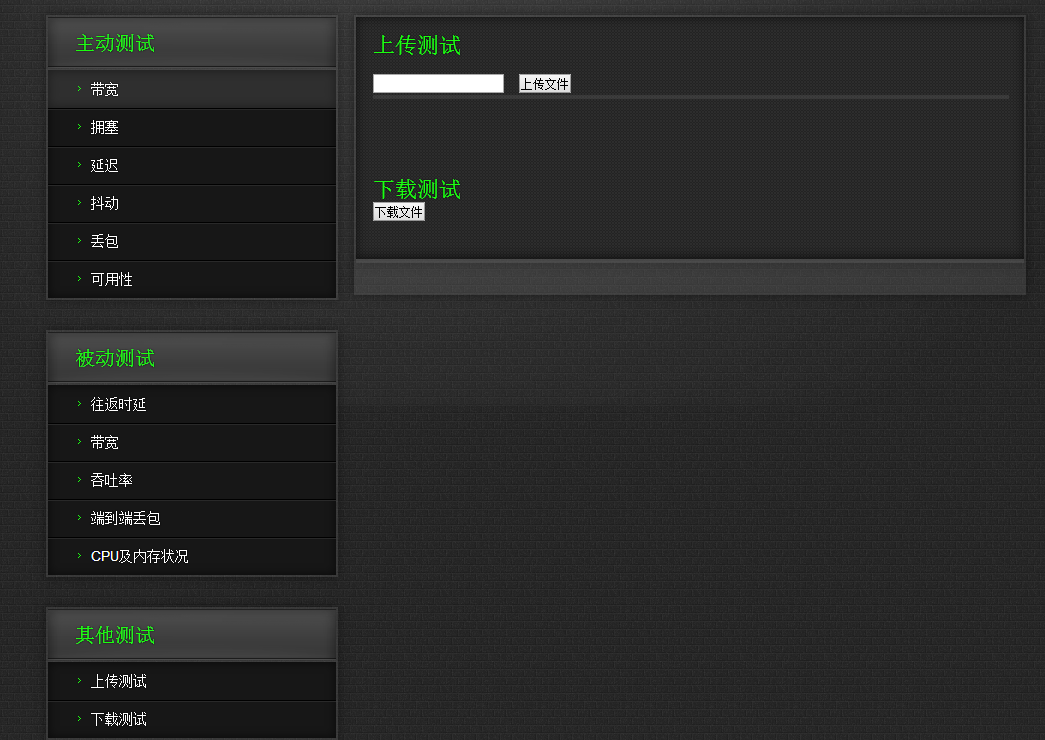


图4-8 上传下载测量