

数字资源Web应用服务器的失效检测^①

余静涛, 王醒宇

(浙江工业大学图书馆 杭州 310032)

摘要: 图书馆数字资源 Web 服务器是访问数字资源的门户, 是用户使用后台数字资源的前提条件. 在研究了 HTTP 和 HTTPS 协议的基础上, 利用 HttpClient 客户端编程工具包和 Ajax 技术设计并实现了数字资源 Web 应用服务器的失效检测系统, 该系统能够准确快速的发现无法提供服务的数字资源 Web 服务器, 并提供服务器出错的信息, 使得网络管理员能够很快做出响应.

关键词: 数字资源 Web 服务器; HTTP 协议; HTTPS 协议; HttpClient 工具包; Ajax 技术

Failure Detection of Digital Resources Web Server

SHE Jing-Tao, WANG Xing-Yu

(Zhejiang University of Technology, Hangzhou 310032, China)

Abstract: Digital Resources Web server is a gateway to access the digital resources, and the prerequisite for using a background of digital resources. Based on the study of the HTTP and HTTPS protocols, this thesis uses HttpClient client programming toolkits and Ajax technology to design and implement digital resources Web application server failure detection system which can quickly and accurately find digital resources Web sever that can not provide service and provide the server error message, enables network administrators to respond quickly.

Key words: digital resources web server; HTTP protocol; HTTPS protocol; HttpClient toolkits; Ajax technology

1 引言

随着图书馆信息资源的数字化、网络化, 各种数字资源的 Web 网站正逐渐成为提供各种信息和服务的重要窗口. 图书馆的用户也越来越多的通过 Web 应用服务器来获取数字资源, 因此, 数字资源 Web 服务器作为访问资源的门户, 是用户使用后台数字资源的前提条件.

目前, 很多图书馆还是通过人工点击的方式来检查数字资源 Web 服务器是否可用, 这种方式不仅效率低下, 而且不能及时发现无法提供服务的 Web 服务器. 也有不少图书馆利用网络监控产品来检测, 但是这种方式只能检测 Web 服务器的硬件状态和 Web 服务的状态, 无法检测数字资源是否能够正常获取. 因此设计一个高效的数字资源 Web 应用服务器失效检测系统用来准确快速的发现无法正常提供服务的 Web 服务器是

非常有必要的.

2 技术概述

数字资源 Web 应用服务器的失效检测系统是在研究 HTTP 和 HTTPS 协议的基础上, 利用 HttpClient 客户端编程工具包和 Ajax 技术来开发完成的.

2.1 HTTP 协议

超文本传输协议 (HTTP-Hypertext transfer protocol) 是一种详细规定了浏览器和万维网服务器之间互相通信的规则, 通过因特网传送万维网文档的数据传送协议. 它是一个基于请求与响应模式的, 无状态的, 应用层的协议, 通常基于 TCP 连接. 目前大多数的 Web 应用程序开发都是构建在 HTTP 协议上的^[1].

2.2 HTTPS 协议

HTTPS(全称: Hypertext Transfer Protocol over Secure

① 基金项目:2012年浙江省图书馆学会课题项目(Ztx2012 A-12)

收稿时间:2013-12-04;收到修改稿时间:2014-01-20

Socket Layer), 是以安全为目标的 HTTP 通道, 简单讲是 HTTP 的安全版^[2], 即 HTTP 下加入 SSL 层, HTTPS 的安全基础是 SSL, 因此加密详细内容就需要 SSL^[3]. SSL (Secure Socket Layer)是由 Netscape 所研发, 用以保障在 Internet 上数据传输之安全.

2.3 HttpClient 开发包

HttpClient 是 Apache Jakarta Common 下的子项目, 可以用来提供高效的、最新的、功能丰富的支持 HTTP 协议的客户端编程工具包, 并且它支持 HTTP 协议最新的版本和建议^[4]. 以下列出的是 HttpClient 提供的主要的功能, 要知道更多详细的功能可以参见 HttpClient 的主页.

(1)实现了所有 HTTP 的方法(GET,POST,PUT, HEAD 等)

(2)支持自动转向

(3)支持 HTTPS 协议

(4)支持代理服务器等

2.4 Ajax 技术

Ajax即“Asynchronous JavaScript and XML”(异步 JavaScript和XML), 是指一种创建交互式网页应用的网页开发技术. 与传统的Web应用不同, Ajax技术在请求/响应模式中, 不需要重载用户填写的数据, 也不需要整个页面重载, 因为Ajax技术在用户与服务器之间引入了一个中间层Ajax引擎, 使用户操作与服务器响应异步化. 这样就可以利用客户端的处理能力来减轻服务器和带宽的负担, 使Web应用程序如同桌面应用程序一样反应灵敏, 从而提升了用户体验^[5].

Ajax 技术实际上是几项技术的整合, XMLHttpRequest是最核心的技术, 它可以实现向服务器发送请求并接受返回数据的全过程, 是实现异步交互的关键所在. JavaScript则是XMLHttpRequest和DOM间的桥梁, 利用XMLHttpRequest的响应返回值来更新本地页面的DOM对象.

3 需求及系统设计

图书馆的数字资源 Web 服务器是否能够正常提供服务不仅要看网站网页是否能正常访问, 同时也要看需要的数字资源是否能够获取, 所以数字资源 Web 应用服务器的失效检测系统需要从两个方面来检测, 一方面是利用 HTTP、HTTPS 协议的响应状态代码来检测 Web 服务器的状态. 响应状态代码由三

位数字组成, 第一个数字定义了响应的类别, 且有五种可能取值:

1xx: 指示信息——表示请求已接收, 继续处理.

2xx: 成功——表示请求已被成功接收、理解、接受.

3xx: 重定向——要完成请求必须进行更进一步的的操作.

4xx: 客户端错误——请求有语法错误或请求无法实现.

5xx: 服务器端错误——服务器未能实现合法的请求.

另一方面是通过判断数字资源下载链接是否用来

检测数字资源是否能够获取. 具体的系统架构如图 1 所示.

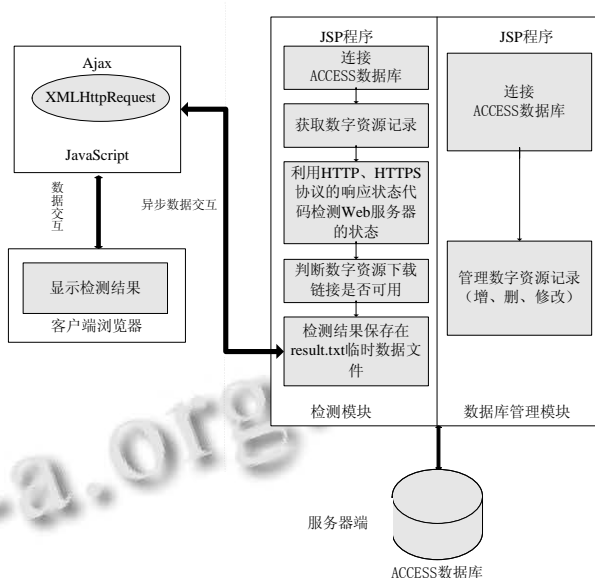


图 1 数字资源 Web 应用服务器失效检测系统架构

数字资源 Web 应用服务器的失效检测系统采用 B/S(浏览器/服务器)架构, 这种架构方便管理员在任意的机器上获取检测信息. 系统使用 JAVA 语言来编写, 采用 ACCESS 数据库保存数字资源记录. 系统的功能模块主要有检测模块、数据库管理模块和检测结果显示模块. 检测模块采用开源的 HttpClient 客户端编程工具包对数字资源 Web 服务器进行检测, 数据库管理模块对数字资源记录进行管理, 检测结果显示模块是采用 Ajax 技术将检测结果显示在浏览器页面上.

检测系统的工作流程大致如下:

(1)网络管理员将数字资源记录通过数据库管理模块保存到 ACCESS 数据库中, 每条数字资源记录包括数字资源的名字, Web 服务器的地址和 Web 服务器上任意一个资源的下载链接.

(2)检测系统每隔一个指定的时间读取保存在 ACCESS 数据库里的所有数字资源记录的信息, 然后根据这些信息反复调用 HttpClient 对象的成员函数来完成检测的工作. 检测工作分为两步, 第一步利用 HTTP 或者 HTTPS 协议的响应状态代码对 Web 服务器的状态进行检测判断, Web 服务器的异常状态包括拒绝服务, 地址错误, 目标不可访问等等, 响应状态代码和对应的文本描述由 HttpClient 开发包获取. 第二步对资源下载的连接进行检测, 判断资源是否可以下载. 在检测过程中, 如果发现第一步的检测异常后就不再继续进行第二步的检测.

(3)系统将数字资源 Web 服务器的响应状态代码的文本描述保存到 result.txt 临时数据文件中, 将数字资源名、检测时间和 Web 服务器异常的响应状态代码文本描述保存在 log.txt 日志文件中.

(4)利用 Ajax 技术读取 ACCESS 数据库中的数字资源记录和临时数据文件 result.txt 中的信息, 并在浏览器页面上显示, 一旦检测到 Web 服务器状态异常或者资源无法下载的错误, 就以弹窗的方式通知网络管理员.

4 系统实现

数字资源 Web 应用服务器的失效检测系统的功能模块主要有检测模块、数据库管理模块和检测结果显示模块.

4.1 检测模块

检测模块是数字资源 Web 应用服务器的失效检测系统的核心模块, 主要负责在读取数字资源记录后利用 HttpClient 开发包对 Web 服务器状态和资源下载链接进行检测, 并将检测结果保存到临时的数据文件中. 检测模块的流程图如图 2.

图 2 中的 check 函数实现的核心代码如下:

```
public int check(String name,String address,String downlink) throws Exception
{ //生成一个HttpClient对象
```

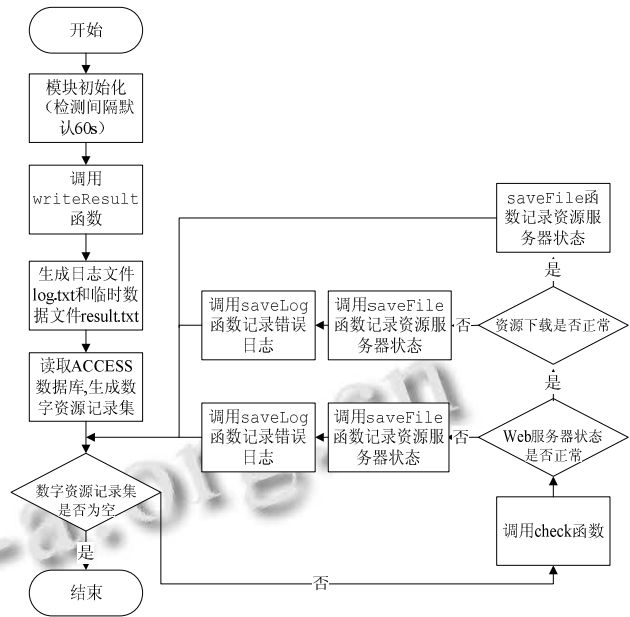


图 2 检测模块的流程图

```
HttpClient httpClient = new HttpClient();
//判断协议类型 HTTP或者HTTPS
if(address.substring(0,5).equalsIgnoreCase("https"))
{/* MySecureProtocolSocketFactory这个自定义的类必须实现接口org.apache.commons.httpclient.protocol.
SecureProtocolSocketFactory, 该类中调用自定义的TrustAnyTrustManager, 这个类必须实现接javax.net.
ssl.X509TrustManager */
Protocol myhttps = new Protocol("https", new
MySecureProtocolSocketFactory(), 443);
Protocol.registerProtocol("https", myhttps); }
//创建GET方法的实例
GetMethod getMethod = new GetMethod(address);
//使用系统提供的默认的恢复策略, 在发生异常时候
将自动重试3次
getMethod.getParams().setParameter(HttpMethodParams
.RETRY_HANDLER,
new DefaultHttpMethodRetryHandler());
//数字资源Web应用服务器响应状态代码
int statusCode = 0;
int downCode=0; //文档下载链接状态代码
try { statusCode =
httpClient.executeMethod(getMethod); //执行Get方法
// 判断数字资源Web应用服务器的状态, 如果状态异常,
则在日志文件中记录服务器名, 异常的描述以及
```

异常发生时间

```
if (statusCode != HttpStatus.SC_OK) { //保存检测结果
saveFile(getMethod.getStatusText()+"\r\n");
.....
//日志文件中记录服务器名, 异常的描述以及异常发生时间
saveLog(name+" "+getMethod.getStatusText()+"
"+Str+"<br/>");}
//如果Web应用服务器状态正常, 就检测文档下载
的链接, 下载成功返回零, 失败返回非零
else{ downCode=checkLink(downlink)
//保存检测结果
if (downCode!=0){ saveFile( "Failed to download
"+" \r\n");
.....}
Else { saveFile(getMethod.getStatusText()+"\r\n");
System.out.println(" The Server link is available ");}}
} catch (HttpException e)
{ //发生致命的异常, 可能是协议不对或者返回的内
容有问题..... }
finally {getMethod.releaseConnection();//释放连接}
return statusCode; }
```

代码中的 saveLog 函数将数字资源 Web 服务器的名字、检测时间和异常的响应状态代码文本描述保存在 log.txt 日志文件中, 方便管理员查询。saveFile 函数将数字资源 Web 服务器的响应状态代码文本描述保存在 result.txt 临时数据文件中, 该文件将作为显示模块的数据源。

4.2 数据库管理模块

数据库管理模块的功能比较简单, 主要就是对数字资源记录进行增、删、修改的操作, 数字资源记录表的结构如表 1:

表 1 数字资源记录表结构

| 字段名称 | 数据类型 | 描述 |
|----------|------|---------------------|
| id | 自动编号 | 序号 |
| name | 文本 | 数字资源 Web 服务器名称 |
| address | 文本 | 数字资源 Web 服务器地址 |
| downlink | 文本 | Web 服务器上任意一个资源的下载链接 |

4.3 检测结果显示模块

检测结果显示模块的主要功能是利用 Ajax 技术

读取 ACCESS 数据库中的数字资源记录和存有检测结果的临时数据文件 result.txt, 并将结果显示在客户端的浏览器上。实现的核心代码如下:

```
<script type="text/javascript">
var xmlHttp;
function createXMLHttpRequest() {
//判断浏览器类型
if(window.ActiveXObject) {
xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
//定义xmlHttp对象, 只对老版本IE有效}
else if(window.XMLHttpRequest) {
xmlHttp=new XMLHttpRequest();
//定义xmlHttp对象, 其他浏览器有效}
if(xmlHttp) {
//getData.jsp对临时数据文件进行读取并返回数据
xmlHttp.open("POST","/admin/checkdatabase/getData.js
p",true);
xmlHttp.onreadystatechange=setCallback;
//调用回调函数
xmlHttp.send(null); }}
function getProgress() {createXMLHttpRequest();}
function setCallback() //定义回调函数
{ if(xmlHttp.readyState==4)
{ if(xmlHttp.status==200) {.....}
//处理getData.jsp返回的数据
// 将处理过后的字符串str显示在浏览器页面上
document.getElementById('content').innerHTML=str;
var flag=0;
//flag用来判断Web服务器是否能够正常提供服务
.....if(flag) { alert("数据库链接出错");}
//弹窗方式报错
setTimeout("getProgress()",60000);
//设置获取数据频率
}}
</script>
```

浏览器显示结果如图 3 所示。

图中表格的每行记录包括数字资源的名称、数字资源 Web 服务器地址、资源下载路径和该 Web 服务器的状态。一旦服务器的状态异常的话, 浏览器就

弹窗显示相应的错误信息。



图 3 失效检测系统显示界面

5 小结

本论文在研究 HTTP 和 HTTPS 协议的基础上, 利用支持 HTTP, HTTPS 协议的 HttpClient 客户端编程工具和 Ajax 技术开发一个图书馆数字资源 Web 应用服务器的失效检测系统. 系统的主要功能是准确快速的发现无法提供服务的数字资源 Web 服务器, 并提供

数字资源 Web 服务器出错的信息, 使得网络管理员能够很快做出响应, 该系统已经在浙江工业大学图书馆的数字资源实际管理工作中得到应用, 效果良好.

参考文献

- 1 祝瑞, 车敏. 基于 HTTP 协议的服务器程序分析. 现代电子技术, 2012, 35(4): 117-119.
- 2 Huang Q, Frahime J. SSL Remote Access VPNs. Cisco Press, 2009.
- 3 Cameron RW, Neil R. Juniper Networks Secure Access Ssl Vpn Configuration Guide. Elsevier Science Ltd, 2005.
- 4 金发华, 陈障洪. HttpClient 入门. http://www.ibm.com/developerworks/cn/opensource/os-httpclient/. [2005-11-10].
- 5 姚文声. Ajax 技术在在线考试系统中的应用. 电脑编程技巧与维护, 2011(16): 69-70.