

基于代码的网站内外网访问^①

李振华, 郑琳川, 刘 睿

(西华师范大学, 南充 637009)

摘 要: 从某事业单位对内外网差异化访问的需求出发, 分析了多种访问控制方法的特点及适用范围, 最终选用代码方法通过编写服务器端程序达到区别内外网访问的目的。

关键词: 访问控制; PHP; Javascript

Inside and Outside Website Access Control Based on Code

LI Zhen-Hua, ZHENG Lin-Chuan, LIU Rui

(China West Normal University, Nanchong 637009, China)

Abstract: Based on the different needs for inside and outside network access in a certain institution, this paper analyzes the features of several access control methods and their applicable range. Through the study, we finally decide on the code method for writing server-side program to distinguish the access for inside and outside network.

Key words: access control; PHP; javascript

1 引言

目前, 事业单位的信息化程度普遍提高, 绝大多数单位都建立了自己的网站系统, 并利用网站发布和交换信息。但随着《中华人民共和国政府信息公开条例》的出台, 国家对信息公开的内容和范围进行了详细的界定, 并不是所有的信息都能在网络上发布。因此迫切需要网站系统能够区分内外网的访问用户, 还能对外网用户访问内网资源进行控制。以笔者所在的某高校为例, 学校大部分二级部门的网站系统开发时并未考虑分离内外网访问, 网站信息对内外网都是开放的, 而且各部门的网站系统差别较大, 网站代码质量参差不齐, 让所有部门重新开发网站系统不太现实。因此需要找到一种方案能够以最低代价实现内外网差异化访问。

2 内外网差异化访问的需求

笔者所在高校的每个二级部门都有各自的部门网站, 而且各网站所用的服务器端技术多样化, 采用比较多的技术有 ASP、php、java, 使用的数据库以 Access 和 MySQL 为主。由于开发时间都比较早, 而且开发时

没有分离内外网访问的需求, 所以绝大部分的系统并不具有区分内外网访问的功能, 不能对内外网访问进行有效控制, 换句话说就是不管是校内用户还是校外用户都能浏览所有的页面。如果某部门要发布一些只能是校内用户才能阅读的内部信息, 在现有系统上是根本不能实现的。要实现网站内外网差异化访问有很多技术方案可以做到, 但是问题在于大多数方案一般都会对系统从底层开始做大面积的改动或者是增加额外的设备来达到效果, 很显然要让所有部门接受系统的重新架构或者增加成本的改造是不太现实的。而且一些部门的网站系统是委托学生开发的, 学生毕业后网站就处于无人维护的状态, 要改造系统根本无从谈起。因此需要找到一种方案在实现分离内外网访问的同时让改动最少并对各个部门现有的网站系统影响最小。

3 常用区分内外网访问的方法

3.1 智能 DNS 技术

智能 DNS 技术是相对于普通 DNS 技术来说的, 采用普通 DNS 技术的服务器只负责为用户解析出 IP 记录, 而不去判断用户从哪里来, 这样会造成所有用

^① 收稿时间:2013-05-23;收到修改稿时间:2013-06-27

户都只能解析到固定的 IP 地址上. 而智能 DNS 会判断用户的来路, 而做出一些智能化的处理, 然后把智能化判断后的 IP 返回给用户^[1]. 通过智能 DNS 技术可以让内外网用户获得不同的解析结果, 从而达到内外网用户访问不同的内容的目的. 但是一般情况下单位的域名都是由运营商或者是教育科研网管理, 也就是说如果要采用智能 DNS 技术就必须和运营商和教育科研网的管理部门沟通, 所以在操作上有很大的难度. 而且并不是所有的二级部门都拥有独立的域名, 没有域名的部门网站大多都是通过 IP 地址访问, 所以智能 DNS 也不能完全解决问题.

3.2 地址重写技术

地址重写就是将访问地址进行截取, 然后经过一系列的地址分析处理以后让访问者重新定位到新的地址上去的方法^[2]. 依据这种技术的特点, 需要将原有的网站系统作成两个版本, 一个内网版本, 一个外网版本, 例如可将内网版本的 url 发布成 `http://www.xxx.com/`, 外网版本的 url 发布成 `http://www.xxx.com/outer/`, 当外网用户访问某内网文件 `http://www.xxx.com/A.html` 时, 通过服务器端处理后就将其访问重定向到 `http://www.xxx.com/outer/A.html` 上去了, 而内网用户访问 `http://www.xxx.com/A.html` 则不会重定向. 这种方法虽然能够达到分离内外网访问者的目的, 但是需要在每台服务器上做设置, 由于不少部门都有自己的服务器, 要在所有服务器上做设置其工作量太大, 而且有些部门没有固定的技术维护人员无法完成两个网站版本的制作, 所以实施起来比较困难.

3.3 服务器端代码技术

服务器端代码技术通过编写服务器端程序来判断访问者的来源, 然后有区别的让访问者重定向到指定的网站地址上去. 由于程序部署方便, 一般主流的服务器端程序, 例如 ASP, ASP.net, Java, PHP 等都有相应的函数和方法来判断访问者来源, 功能实现上相对要容易, 而且程序复制方便, 即便在多个服务器上部署也不用耗费太多人力, 改造的成本相对要小得多. 所以本文选用编写代码的方法来实现区分内外网的功能.

4 具体思路及实现

由于校园网内采用的是固定 IP 上网, 且学校 ip 地址段也比较集中, 再加上服务器端程序能够很方便地获取访问者的 ip 地址, 因此可通过 ip 地址来区分内外

网用户. 服务器端开发语言如 ASP、PHP、JSP 等都具有获取访问者 IP 的函数, 可以根据各个网站系统采用的开发语言单独编写相应的处理程序来实现, 但是由于网站数量较多为了减少工作量也为后期统一管理, 可以只编写一个处理程序然后让每个网页进行载入的方法来实现, 该方法的工作方式如图 1 所示.

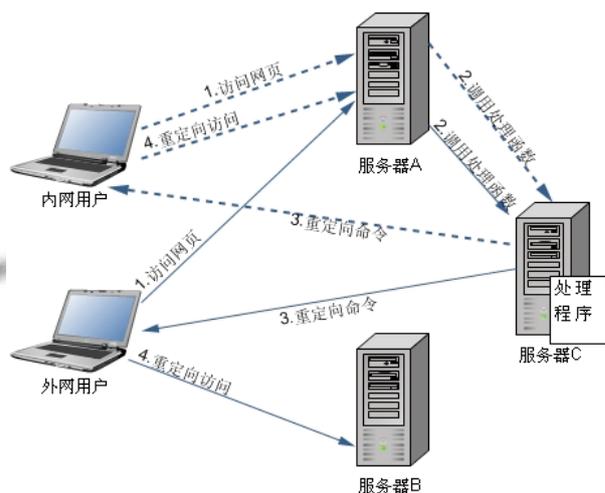


图 1 内外网用户访问

图中所示服务器 A 代表校内某部门服务器, 里面存放所有对内网用户公开的信息, 服务器 B 存放对外网用户公开的信息, 当然对外信息不一定要单独存放于一台服务器, 可以和内网信息放置于同一服务器下的不同目录里, 服务器 C 存放的处理程序用于判断访问者是来自外网还是内网, 然后进行相应的处理操作让内网用户重定向访问服务器 A 里的内容, 外网用户重定向访问服务器 B 里的内容. 这种方案的好处就是能够大幅减少工作量, 不需要更改各部门的服务器的设置, 只需要编写一个处理程序, 然后在发布信息时在网站内容页面将这个处理程序载入即可, 而且只有一个处理程序也很方便日后进行修改和管理.

一般情况下获取客户端 ip 可以通过 `$_SERVER["REMOTE_ADDR"]` 实现, 其功能是获取最后一个和服务器建立连接的 IP 地址, 以 PHP 编程语言为例具体代码如下^[3].

```
<?php
$ip = $_SERVER["REMOTE_ADDR"]; //获取访问用户的 IP 地址
?>
```

但是如果访问者使用了代理, REMOTE_ADDR 将

显示代理服务器的 IP, 这会使检测的 IP 不准确, 为了获取到客户端的真实 IP 就必须再利用 `$_SERVER["HTTP_CLIENT_IP"]`、`$_SERVER["HTTP_X_FORWARDED_FOR"]` 共同完成。`$_SERVER["HTTP_CLIENT_IP"]` 是获取代理服务器发回的 HTTP

header, `$_SERVER["HTTP_X_FORWARDED_FOR"]` 简称 XFF 头, 代表客户端, 即 HTTP 的请求端真实的 IP, 只有在通过了 HTTP 代理(比如 APACHE 代理)或者负载均衡服务器时才会添加该项。三个变量的取值情况如表 1。

表 1 各种代理情况下参数取值表

	REMOTE_ADDR	HTTP_CLIENT_IP	HTTP_X_FORWARDED_FOR
没有代理服务器	客户端 IP	没数值或不显示	没数值或不显示
使用透明代理服务器	最后一个代理服务器 IP	代理服务器 IP	客户端 IP
使用普通匿名代理服务器	最后一个代理服务器 IP	代理服务器 IP	代理服务器 IP
使用欺骗性代理服务器	代理服务器 IP	代理服务器 IP	代理服务器 IP
使用高匿名代理服务器	代理服务器 IP	没数值或不显示	没数值或不显示

根据表 1 的取值情况可以编写出程序来获取访问者的 IP 或者是代理服务器的 IP, 程序如下:

```
<?php
function getIP() {
if ($_SERVER["HTTP_X_FORWARDED_FOR"]) {
    if ($_SERVER["HTTP_CLIENT_IP"]) {
        $proxy = $_SERVER["HTTP_CLIENT_IP"];
    } else {
        $proxy = $_SERVER["REMOTE_ADDR"];
    }
    $ip=$_SERVER["HTTP_X_FORWARDED_FOR"];
} else {
    if ($_SERVER["HTTP_CLIENT_IP"]) {
        $ip = $_SERVER["HTTP_CLIENT_IP"];
    } else {
        $ip = $_SERVER["REMOTE_ADDR"];
    }
}
return $ip;
if (isset($proxy)) {
return $proxy;
}
} //function
?>
```

获取了 IP 之后就需要对其是否属于内网进行判断, 然后进行相应的处理, 程序如下:

Ipcheck.php:
<?php

```
function isIntranet($str_ip) {
//设置内网 IP 地址段
$arr_iparranges=array();
$arr_iparranges[0]=array("210.41.192.1","210.41.192.254");
$arr_iparranges[1]=array("210.41.194.254","210.41.200.4");
$tag=false;
foreach ($arr_iparranges as $k=>$iparranges) {
    $arr_beginIP=explode(".", $iparranges[0]);
    $arr_endIP=explode(".", $iparranges[1]);
    $arr_IP=explode(".", $str_ip);
    $lng_beginIP=(integer)$arr_beginIP[0]*pow(2,24)+(integer)$arr_beginIP[1]*pow(2,16)+(integer)$arr_beginIP[2]*pow(2,8)+(integer)$arr_beginIP[3]*pow(2,0);
    $lng_endIP=(integer)$arr_endIP[0]*pow(2,24)+(integer)$arr_endIP[1]*pow(2,16)+(integer)$arr_endIP[2]*pow(2,8)+(integer)$arr_endIP[3]*pow(2,0);
    $lng_arrIP=(integer)$arr_IP[0]*pow(2,24)+(integer)$arr_IP[1]*pow(2,16)+(integer)$arr_IP[2]*pow(2,8)+(integer)$arr_IP[3]*pow(2,0);
    (($lng_arrIP>=$lng_beginIP)&&($lng_arrIP<=$lng_endIP))? $tag=( $tag||true):$tag=( $tag||false); //判断 IP 是否在内网 IP 地址段内
} //foreach
Return $tag;
} //function
.....
```

```
?>
```

完成 IP 地址的判断后就可以对访问者进行相应的跳转或限制等操作了。因为该程序是独立程序,要实现跳转操作必须要知道跳转前访问者访问的 url 地址,实现该功能就需要在各部门网站信息内容的页面里通过<script>将处理程序连接进来,然后将本页的 url 通过参数的形式传给处理程序,处理程序对 url 进行重新计算后在<script>中让网页页面进行跳转或者停止,具体实现代码如下:

各部门页面:

```
<html>
<head>
.....
<script src='http://xxx.xxx.xx/ipcheck.php' language=
Javascript'></script>
</head>
.....
</html>
```

Ipcheck.php:

```
<?php
function getIP(){
.....
}
function isIntranet($str_ip){
.....
}
$visitorIP=getIP();
$newurl="URL"; //需要跳转的新的 IP 地址
if(isIntranet($visitorIP)==false){
echo "window.location='url'";
```

```
}
```

```
?>
```

由于<script>放置在<head>中,所以在正文内容还未被加载的时候浏览器运行脚本会跳转到其他的页面中去,文章内容是不会被显示出来的,因为脚本是通过服务器端程序动态生成的,所以即便禁止客户端执行脚本也不会影响脚本的运行。

5 结论

代码方法最大的优势在于灵活性很强,可以根据具体需要进行设计,而且在实现内外网差异化访问时代价最小,在保留原各部门网站系统的情况下只需要做很少的改动就可以实现。但是,代码方法也有不足,访问者可以通过伪造 HTTP 头部信息来达到“蒙混过关”的目的。就像没有万能药一样,很难去找到一种方案能够与我们的需求完美匹配,由于事业单位实现内外网访问的主要需求在隔离内外网用户访问这方面并没有非常严格的要求,所以该方案还是可接受的。

参考文献

- 1 besthome. 智能 DNS. [2013-03-22]. <http://baike.baidu.com/view/1572287.htm>
- 2 曲俊华,朱海涛,赵波.URL 重写技术在动态网站优化中的应用研究.网络安全技术与应用,2011,(9):60-61.
- 3 周小科,管兆宁.利用 ASP 获取客户端真实 IP 地址.苏州大学学报(工科版),2003,23(5):67-69.
- 4 Lerdorf R, Tatroe K, MacIntyre P. 陈浩,胡丹,等译.PHP 程序设计.北京:电子工业出版社,2009.
- 5 Flanagan D.李强译.Javascript 权威指南.北京:机械工业出版社,2007.