linux的Apache用户认证、域名跳转、Apache访问日志介绍

[原创](javascript:;)[Gary\_加油涛](http://blog.51cto.com/taoxie)[2017-12-20 21:32:03](javascript:;)[评论(0)](http://blog.51cto.com/taoxie/2052678#comment)[321人阅读](javascript:;)

**Apache用户认证（针对目录）**

这个功能就是在用户访问网站的时候，需要输入用户密码才能顺利访问。一些比较重要的站点或者网站后台通常会加上用户认证，目的是保证安全。

1. 虚拟主机的配置文件：

**编辑配置文件**

[root@gary-tao **local**]*# vim /usr/local/apache2.4/conf/extra/httpd-vhosts.conf*

**更改111.com的虚拟主机认证内容如下：**

<VirtualHost \*:80>

DocumentRoot "/data/wwwroot/111.com"

ServerName 111.com

ServerAlias www.example.com

<Directory /data/wwwroot/111.com> *//指定认证的目录*

AllowOverride AuthConfig *//这个相当于打开认证的开关*

AuthName "111.com user auth" *//自定义认证的名字，作用不大*

AuthType Basic *//认证的类型，一般为Basic，其他类型阿铭没用过*

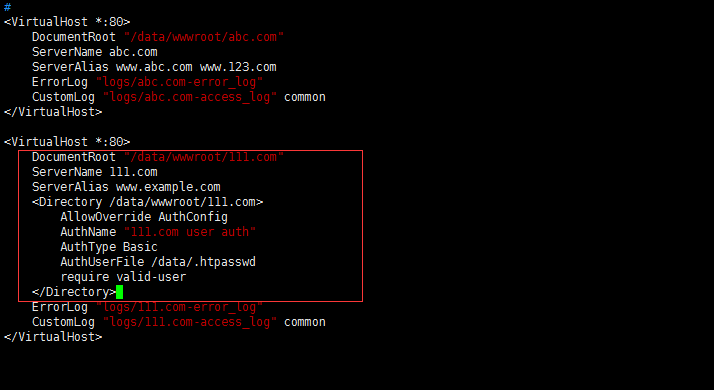
AuthUserFile /data/.htpasswd *//指定密码文件所在位置*

**require** valid-user *//指定需要认证的用户为全部可用用户*

</Directory>

</VirtualHost>

**最终保存文件，示例如下：**



2.Apache自带命令htpasswd创建密码文件

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/htpasswd -c -m /data/.htpasswd xie //创建用户密码文件*

New password: //新建密码

Re-type new password: //新建密码

Adding password **for** user xie

[root@gary-tao **local**]*# ls /data/.htpasswd //查看密码文件*

/data/.htpasswd

[root@gary-tao **local**]*# cat /data/.htpasswd //查看生成用户密码*

xie:$apr1$h/QEC7nC$hNNV080nvhSI2jWCQLt7M0

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/htpasswd -m /data/.htpasswd aming //再增加一个用户*

New password:

Re-type new password:

Adding password **for** user aming

[root@gary-tao **local**]*# cat /data/.htpasswd*

xie:$apr1$h/QEC7nC$hNNV080nvhSI2jWCQLt7M0

aming:$apr1$At/pBlDA$4IYzNISYUew9ELrea5dP7.

**说明：**

* -c：是创建;
* -m：是指定md5加密类型;
* 指定用户为xie（PS：如果再次新增用户，就不需要再加-c ，因为已经创建过密码文件了）;

3.测试语法和加载配置文件

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl -t*

Syntax OK

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl graceful*

4.测试配置是否成功

* 访问111.com，出现401状态码，说明访问的这个域名需要用户认证。

[root@gary-tao local]# curl -x127.0.0.1:80 111.com

**<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">**

<html><head>

<title>401 Unauthorized</title>

</head><body>

<h1>Unauthorized</h1>

<p>This server could not verify that you

are authorized to access the document

requested. Either you supplied the wrong

credentials (e.g., bad password), or your

browser doesn't understand how to supply

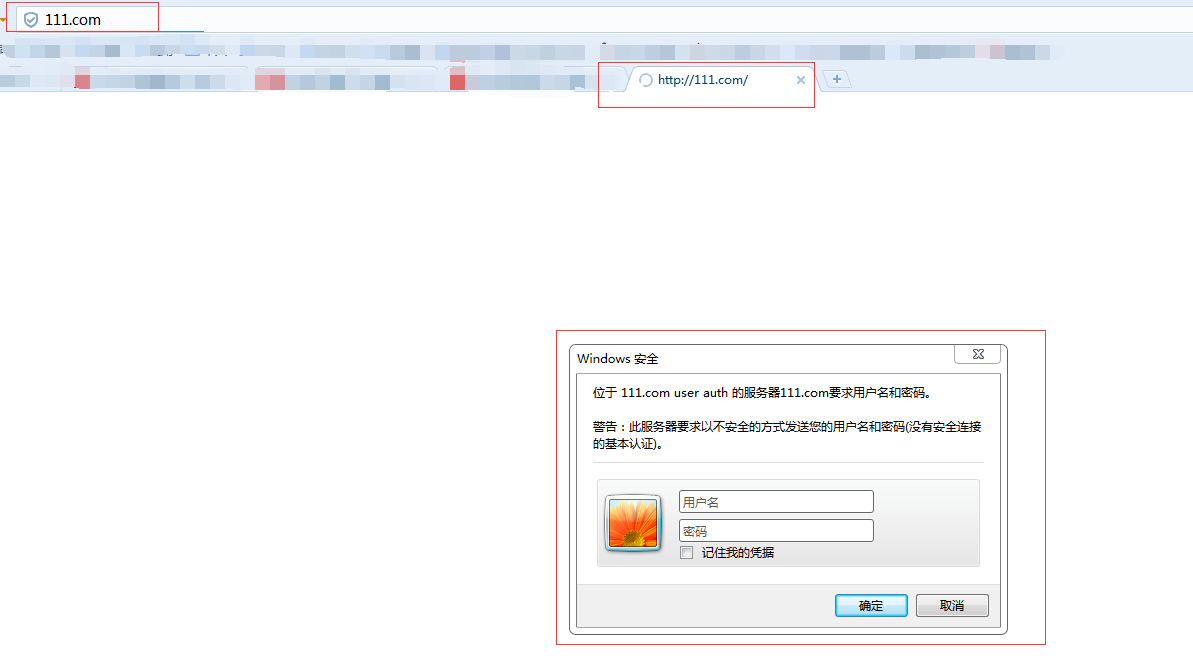
the credentials required.</p>

</body></html>

* 在本地windows系统里做hosts解析111.com

路径：C:\Windows\System32\drivers\etc  
格式：172.16.111.100 111.com

* 定义完本地hosts后，用浏览器访问111.com网站时就会出现用户认证，用户密码就是刚才增加的用户和设置的密码



5.使用curl -x输入用户名密码访问。

**用法**

[root@gary-tao local]*# curl -x127.0.0.1:80 -uxie:xie 111.com -I*

HTTP/1.1 200 OK

**Date: Wed, 20 Dec 2017 10:51:28 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**X-Powered-By: PHP/7.1.6**

**Content-Type: text/html; charset=UTF-8**

**说明：状态码变成200了，就是正常的，-u的作用是指定用户和密码。**

6.还可以针对单个文件进行认证（针对文件）

示例内容：

<VirtualHost \*:80>

DocumentRoot "/data/wwwroot/www.123.com"

ServerName www.123.com

<FilesMatch admin.php> //跟上面的不同的是这行，上面是指定认证的目录，这里是指定单个文件。

AllowOverride AuthConfig

AuthName "123.com user auth"

AuthType Basic

AuthUserFile /data/.htpasswd

require valid-user

</FilesMatch> //这行也不同

</VirtualHost>

* 在配置文件修改成以下：

**<VirtualHost \*:80>**

DocumentRoot "/data/wwwroot/111.com"

ServerName 111.com

ServerAlias www.example.com

*#<Directory /data/wwwroot/111.com>*

**<FilesMatch 123.php>**

AllowOverride AuthConfig

AuthName "111.com user auth"

AuthType Basic

AuthUserFile /data/.htpasswd

require valid-user

**</FilesMatch>**

*#</Directory>*

ErrorLog "logs/111.com-error\_log"

CustomLog "logs/111.com-access\_log" common

**</VirtualHost>**

* 更改完成后测试语法及重新加载配置文件：

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl -t*

Syntax OK

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl graceful*

* 在111.com目录下编辑创建测试文件123.PHP。

[root@gary-tao **local**]*# vim /data/wwwroot/111.com/123.php*

* 用curl -x访问：

[root@gary-tao **local**]*# curl -x127.0.0.1:80 111.com -I //不用-u加用户和密码了，也可以访问，出现200状态码*

HTTP/1.1 200 OK

Date: Wed, 20 Dec 2017 11:04:06 GMT

Server: Apache/2.4.29 (Unix) PHP/7.1.6

X-Powered-By: PHP/7.1.6

Content-Type: text/html; charset=UTF-8

[root@gary-tao **local**]*# curl -x127.0.0.1:80 111.com/123.php -I //但是访问文件123.php时就出现401了，说明需要用户认证了*

HTTP/1.1 401 Unauthorized

Date: Wed, 20 Dec 2017 11:04:17 GMT

Server: Apache/2.4.29 (Unix) PHP/7.1.6

WWW-Authenticate: Basic realm="111.com user auth"

Content-Type: text/html; charset=iso-8859-1

[root@gary-tao **local**]*# curl -x127.0.0.1:80 -uxie:xie 111.com/123.php -I //只有用-u加用户和密码才能正常访问123.php。*

HTTP/1.1 200 OK

Date: Wed, 20 Dec 2017 11:04:38 GMT

Server: Apache/2.4.29 (Unix) PHP/7.1.6

X-Powered-By: PHP/7.1.6

Content-Type: text/html; charset=UTF-8

[root@gary-tao **local**]*# curl -x127.0.0.1:80 -uxie:xie 111.com/123.php //进入到文件里。*

123.php[root@gary-tao **local**]*#*

[root@gary-tao **local**]*#*

**域名跳转**

**域名跳转的作用有两点：**

1. 如果某个域名不再使用了，但是搜索引擎还留着之前的老域名的链接，这意味着用户可能会搜到我们的网站并且点击老的域名，固需要把老域名做个跳转跳到新域名，这样用户搜的时候，也可以访问网站。
2. 一个站点有多个域名会对SEO的排名有影响，如果把多个域名全部跳转到一个指定的域名，这样以这个域名为中心，就可以把权重集中在这个域名上，并给定义一个状态码为301，301叫作永久重定向。

**需求，把123.com域名跳转到www.123.com.**

1. 编辑配置文件

[root@gary-tao **local**]*# vim /usr/local/apache2.4/conf/extra/httpd-vhosts.conf*

2.修改增加如下内容：

<VirtualHost \*:80>

DocumentRoot "/data/wwwroot/www.123.com"

ServerName www.123.com

ServerAlias 123.com

<IfModule mod\_rewrite.c> *//需要mod\_rewrite模块支持*

RewriteEngine **on** *//打开rewrite功能*

RewriteCond %{HTTP\_HOST} !^www.123.com$ *//定义rewrite的条件，主机名（域名）不是www.123.com满足条件*

RewriteRule ^/(.\*)$ http:*//www.123.com/$1 [R=301,L] //定义rewrite规则，当满足上面的条件时，这条规则才会执行*

</IfModule>

</VirtualHost>

**修改示例如下：**



3.检测语法及重新加载配置：

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl -t*

Syntax OK

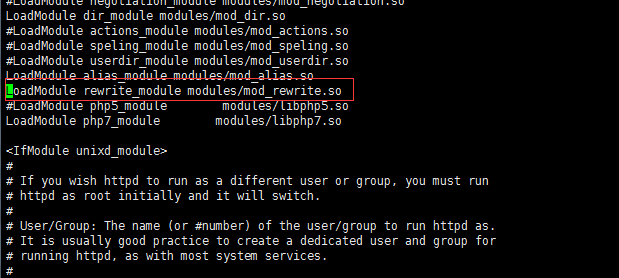
[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl graceful*

4.检测apache是否加载了rewrite模块。

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl -M|grep -i rewrite //若无该模块，需要编辑配置文件httpd.conf，删除rewrite\_module (shared) 前面的#*

[root@gary-tao **local**]*# vi /usr/local/apache2.4/conf/httpd.conf //进入配置文件，搜索rewrite，把前面#去掉*

**示例如下：**



5.检测语法及重新加载配置，查看加载模块：

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl -t*

Syntax OK

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl graceful*

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl -M|grep -i rewrite //查看加载模块*

rewrite\_module (shared)

6.测试

[root@gary-tao local]*# curl -x 127.0.0.1:80 -I 2111.com.cn*

HTTP/1.1 301 Moved Permanently

**Date: Wed, 20 Dec 2017 12:31:50 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**Location: http://111.com/**

**Content-Type: text/html; charset=iso-8859-1**

[root@gary-tao local]*# curl -x 127.0.0.1:80 2111.com.cn //看内容*

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">

<html><head>

<title>301 Moved Permanently</title>

</head><body>

<h1>Moved Permanently</h1>

<p>The document has moved <a href="http://111.com/">here</a>.</p>

</body></html>

[root@gary-tao local]*# curl -x 127.0.0.1:80 2111.com.cn/adfjadfa/adfdafadfaf -I*

HTTP/1.1 301 Moved Permanently

**Date: Wed, 20 Dec 2017 12:34:05 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**Location: http://111.com/adfjadfa/adfdafadfaf**

**Content-Type: text/html; charset=iso-8859-1**

[root@gary-tao local]*# curl -x 127.0.0.1:80 http://111.com/adfjadfa/adfdafadfaf -I*

HTTP/1.1 404 Not Found

**Date: Wed, 20 Dec 2017 12:35:08 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**Content-Type: text/html; charset=iso-8859-1**

[root@gary-tao local]*# curl -x 127.0.0.1:80 http://111.com/123.php -I*

HTTP/1.1 200 OK

**Date: Wed, 20 Dec 2017 12:36:35 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**X-Powered-By: PHP/7.1.6**

**Content-Type: text/html; charset=UTF-8**

[root@gary-tao local]*# vi /usr/local/apache2.4/conf/httpd.conf*

[root@gary-tao local]*# /usr/local/apache2.4/bin/apachectl -t*

Syntax OK

[root@gary-tao local]*# /usr/local/apache2.4/bin/apachectl graceful*

[root@gary-tao local]*# curl -x 127.0.0.1:80 http://111.com/123.php -I*

HTTP/1.1 403 Forbidden

**Date: Wed, 20 Dec 2017 12:39:23 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**Content-Type: text/html; charset=iso-8859-1**

**说明：**

* -I 不显示访问内容，只看状态码
* 404 这个页面不存在
* 301 永久跳转
* 401 用户密码验证,密码验证不对就401,验证对了就200
* 403 把granted改成denied就会403

**Apache访问日志**

访问日志的作用很大,不仅可以记录网站的访问日志,还可以在网站有异常发生时帮助我们定位问题,比如有攻击时,是可以通过查看日志看到一些规律的.日志记录了很多系统的信息，通过读日志，可以找到系统问题的原因。而日志有不同的格式，分为common和combined，combined可以记录更多的信息。

1. 查看默认配置文件日志

[root@gary-tao **local**]*# ls /usr/local/apache2.4/logs/*

111.com-access\_log 111.com-error\_log abc.com-access\_log abc.com-error\_log access\_log error\_log httpd.pid

[root@gary-tao **local**]*# ls /usr/local/apache2.4/logs/111.com-access\_log*

/usr/**local**/apache2.4/logs/111.com-access\_log

[root@gary-tao **local**]*# cat /usr/local/apache2.4/logs/111.com-access\_log*

172.16.111.1 - xie [20/Dec/2017:20:09:54 +0800] "GET / HTTP/1.1" 200 12

127.0.0.1 - - [20/Dec/2017:20:31:50 +0800] "HEAD HTTP://2111.com.cn/ HTTP/1.1" 301 -

127.0.0.1 - - [20/Dec/2017:20:32:53 +0800] "GET HTTP://2111.com.cn/ HTTP/1.1" 301 223

127.0.0.1 - - [20/Dec/2017:20:34:05 +0800] "HEAD HTTP://2111.com.cn/adfjadfa/adfdafadfaf HTTP/1.1" 301 -

127.0.0.1 - - [20/Dec/2017:20:35:08 +0800] "HEAD http://111.com/adfjadfa/adfdafadfaf HTTP/1.1" 404 -

127.0.0.1 - - [20/Dec/2017:20:36:35 +0800] "HEAD http://111.com/123.php HTTP/1.1" 200 -

127.0.0.1 - - [20/Dec/2017:20:39:23 +0800] "HEAD http://111.com/123.php HTTP/1.1" 403 -

127.0.0.1 - - [20/Dec/2017:20:40:16 +0800] "HEAD http://111.com/123.php HTTP/1.1" 200 -

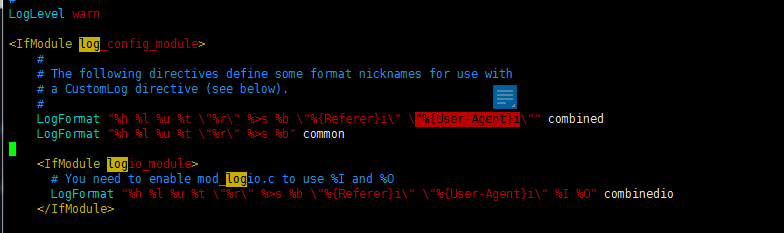
2.介绍日志配置文件格式

[root@gary-tao **local**]*# vim /usr/local/apache2.4/conf/httpd.conf*

LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined

LogFormat "%h %l %u %t \"%r\" %>s %b" common

**内容示例如下：**



**访问日志记录用户的每一个请求说明如下：**

* %h:为访问网站的IP;
* %l:为访问远程登录名,这个字段基本上为"-";
* %u:为用户名,当使用用户认证时,这个字段为认证的用户名;
* %t:为时间;
* %r:为请求的动作(比如用ctrl-I是就为HEADE);
* %s:为请求的状态,写成%>s为最后的状态码;
* %b:为传输数据大小;
* %{Referer}i:为referer信息(请求本次地址上一次的地址就为referer,比如在百度中搜索阿铭linux,然后通过百度的搜索结果页面点击然后到了阿名的论坛,那访问阿铭的论坛的这次请求的referer就是baidu,当然那个地址肯定是很长的);
* %{User-Agent}i:为浏览器标识,比如你用Firefox或者Chrome浏览器,则该字段显示内容不一样,是带有浏览器的标识的。

3.定义虚拟主机配置文本日志格式：

[root@gary-tao **local**]*# vim /usr/local/apache2.4/conf/extra/httpd-vhosts.conf //进入配置文件*

**把common日志格式格式改成comdined日志格式，示例如下：**



4.测试语法及重新加载配置

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl -t*

Syntax OK

[root@gary-tao **local**]*# /usr/local/apache2.4/bin/apachectl graceful*

5.随便做几个操作命令，然后查看日志。

[root@gary-tao local]*# !curl*

curl -x 127.0.0.1:80 http://111.com/123.php -I

HTTP/1.1 200 OK

**Date: Wed, 20 Dec 2017 13:10:16 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**X-Powered-By: PHP/7.1.6**

**Content-Type: text/html; charset=UTF-8**

[root@gary-tao local]*# curl -x 127.0.0.1:80 http://111.com/123.php -I*

HTTP/1.1 200 OK

**Date: Wed, 20 Dec 2017 13:10:31 GMT**

**Server: Apache/2.4.29 (Unix) PHP/7.1.6**

**X-Powered-By: PHP/7.1.6**

**Content-Type: text/html; charset=UTF-8**

[root@gary-tao local]*# tail /usr/local/apache2.4/logs/111.com-access\_log*

127.0.0.1 - - [20/Dec/2017:20:34:05 +0800] "HEAD HTTP://2111.com.cn/adfjadfa/adfdafadfaf HTTP/1.1" 301 -

127.0.0.1 - - [20/Dec/2017:20:35:08 +0800] "HEAD http://111.com/adfjadfa/adfdafadfaf HTTP/1.1" 404 -

127.0.0.1 - - [20/Dec/2017:20:36:35 +0800] "HEAD http://111.com/123.php HTTP/1.1" 200 -

127.0.0.1 - - [20/Dec/2017:20:39:23 +0800] "HEAD http://111.com/123.php HTTP/1.1" 403 -

127.0.0.1 - - [20/Dec/2017:20:40:16 +0800] "HEAD http://111.com/123.php HTTP/1.1" 200 -

127.0.0.1 - - [20/Dec/2017:21:10:16 +0800] "HEAD http://111.com/123.php HTTP/1.1" 200 - "-" "curl/7.29.0"

127.0.0.1 - - [20/Dec/2017:21:10:31 +0800] "HEAD http://111.com/123.php HTTP/1.1" 200 - "-" "curl/7.29.0"

172.16.111.1 - xie [20/Dec/2017:21:10:38 +0800] "GET / HTTP/1.1" 200 12 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.221 Safari/537.36 SE 2.X MetaSr 1.0"

172.16.111.1 - xie [20/Dec/2017:21:10:38 +0800] "GET / HTTP/1.1" 200 12 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.221 Safari/537.36 SE 2.X MetaSr 1.0"

172.16.111.1 - xie [20/Dec/2017:21:10:39 +0800] "GET / HTTP/1.1" 200 12 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.221 Safari/537.36 SE 2.X MetaSr 1.0"

版权声明：原创作品，如需转载，请与作者联系。否则将追究法律责任