# Linux系统下Apache2.4.17的安装过程

## 工具/原料

* Linux系列系统
* Apache Server2.4.17源码包
* APR源码包
* APR-Util源码包
* PCRE源码包

## 方法/步骤

### 1、环境要求

首先来看官方文档对环境的要求：

The following requirements exist for building Apache httpd:

APR and APR-Util

Make sure you have APR and APR-Util already installed on your system.

Perl-Compatible Regular Expressions Library (PCRE)

This library is required but not longer bundled with httpd.

Disk Space

Make sure you have at least 50 MB of temporary free disk space available.

ANSI-C Compiler and Build System

Make sure you have an ANSI-C compiler installed. The GNU C compiler (GCC) from the Free Software Foundation (FSF) is recommended.

Accurate time keeping

Elements of the HTTP protocol are expressed as the time of day. So, it's time to investigate setting some time synchronization facility on your system. Usually the ntpdate or xntpd programs are used for this purpose which are based on the Network Time Protocol (NTP).

翻译：

要编译Apache httpd，存在以下要求：

APR and APR-Util

确保你的系统中已经安装了APR和APR-Util。

Perl-Compatible Regular Expressions Library (PCRE)

这个库是必需的，但它不再与httpd捆绑在一起。

Disk Space

确保你有至少50MB的临时空闲磁盘空间可以使用。

ANSI-C Compiler and Build System

确保你已经安装了一个ANSI-C编译器。推荐使用由自由软件基金会（FSF）编写的GNU C编译器（GCC）。

Accurate time keeping

HTTP协议的元素会以一天的时间进行表示。所以，它的时间会参照你系统中的一些时钟同步设备的设置。通常基于网络时间协议（NTP）的程序ntpdate或xntpd就是用来做这样的事情。

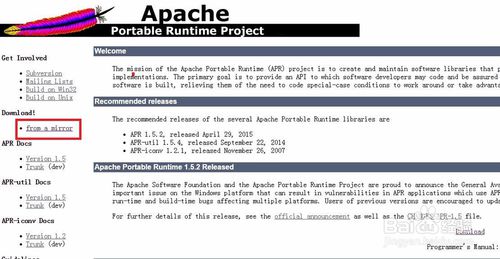
通过官方文档我们了解到，在安装Apache2.4之前我们要做的是：安装APR、APR-Util和PCRE。

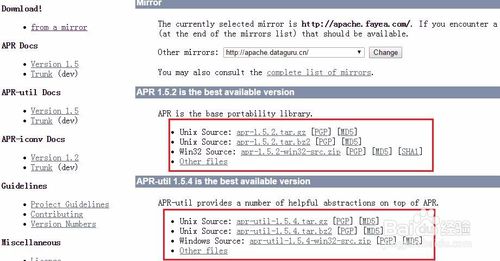
### 2、按照ANSI-C编译器

无论你是要安装APR、APR-Util、PCRE还是Apache Server或者是其他的一些软件，总之，只要是源码安装，那么就需要一个ANSI-C编译器。如果你没有，那就用yum装一个吧，既然推荐使用GCC，那就装个GCC吧。执行命令：yum  install  -y  gcc。如果，你闲得发慌，你可以不依赖yum，自己进行安装GCC试试，相信你会爽歪歪的！

### 3、下载APR与APR-Util

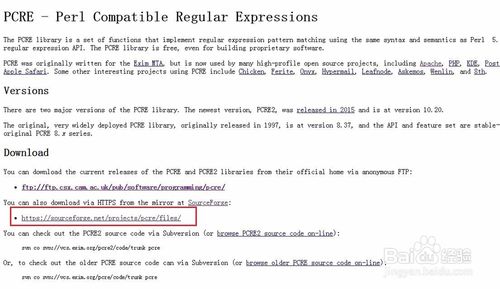
接下来，就是要安装APR、APR-Util和PCRE。安装之前，来看下怎么下载它们。先来看下APR与APR-Util的下载，大家可以去http://apr.apache.org进行下载，看下面两张图：

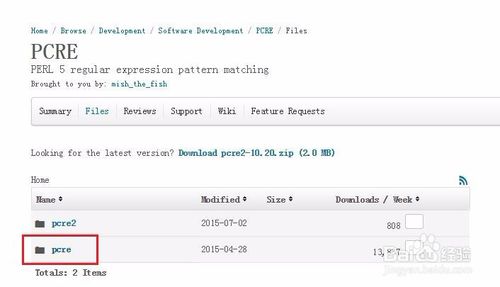
[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=1)

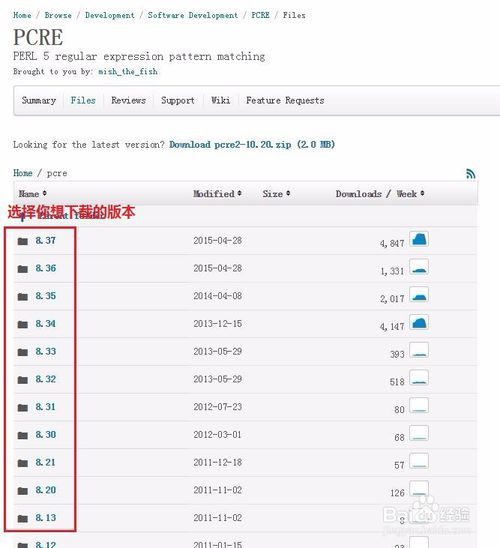
[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=2)

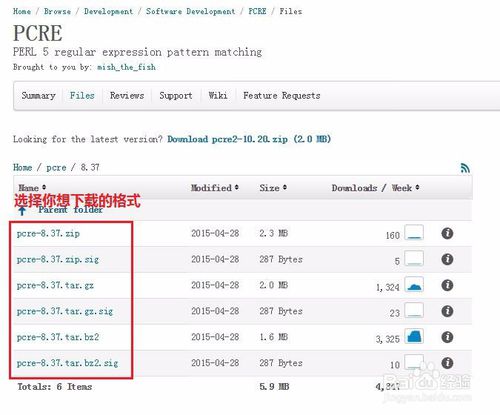
### 4、下载PCRE

再来看下怎么下载PCRE。大家可以去http://www.pcre.org进行下载，看下面四张图：

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=3)

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=4)

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=5)

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=6)

### 5、解压APR

将下载的3个压缩包上传到你的Linux系统中去。我上传到了我Linux系统下/mysoft目录中。进入这个目录：cd  /mysoft。我们先解包APR，并查看/mysoft目录下的文件，依次执行命令：

gzip  -d  apr-1.5.2.tar.gz

tar  -xf  apr-1.5.2.tar

ls  -l

可以看到/mysoft目录下多了一个目录apr-1.5.2。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=7)

### 6、APR的编译和安装

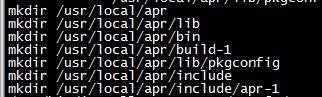
进入apr-1.5.2目录：cd  ./apr-1.5.2/。然后，进行APR的源码安装，依次执行命令：

./configure

make

make install

安装完成后，我们可以在make install过程中看到如下图所示信息，我们可以看到APR的安装路径/usr/local/apr。当然，你也可以使用  --prefix=目录  选项来指定一个安装目录。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=8)

### 7、解压APR-Util

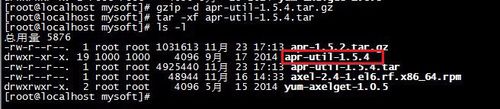
接着，我们来解包APR-Util，并查看/mysoft目录下的文件，依次执行命令：

gzip  -d  apr-util-1.5.4.tar.gz

tar  -xf  apr-util-1.5.4.tar

ls  -l

可以看到/mysoft目录下多了一个目录apr-util-1.5.4。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=9)

### 8、APR-Util的编译和安装

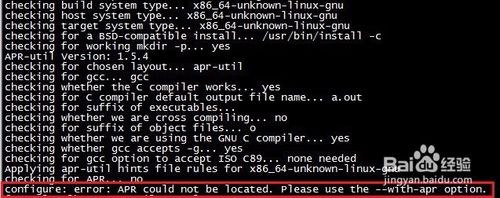
进入apr-util-1.5.4目录：cd  ./apr-util-1.5.4/。然后，进行APR-Util的源码安装，依次执行命令：

./configure  --with-apr=/usr/local/apr

make

make install

在安装APR-Util时，需要指定APR的安装路径，所以需要带上选项--with-apr=/usr/local/apr，如果你不带上这个选项，在configure的时候会看到下图的错误提示。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=10)

解压完apr-util后，执行make报错，如下：

[root@localhost apr-util-1.6.0]# make  
make[1]: Entering directory `/root/Downloads/apache\_svn/apr-util-1.6.0'  
/bin/sh /usr/local/apr/build-1/libtool --silent --mode=compile gcc -g -O2 -pthread -DHAVE\_CONFIG\_H -DLINUX -D\_REENTRANT -D\_GNU\_SOURCE -I/root/Downloads/apache\_svn/apr-util-1.6.0/include -I/root/Downloads/apache\_svn/apr-util-1.6.0/include/private -I/usr/local/apr/include/apr-1 -o xml/apr\_xml.lo -c xml/apr\_xml.c && touch xml/apr\_xml.lo  
xml/apr\_xml.c:35:19: error: expat.h: No such file or directory  
xml/apr\_xml.c:66: error: expected specifier-qualifier-list before ‘XML\_Parser’  
xml/apr\_xml.c: In function ‘cleanup\_parser’:  
xml/apr\_xml.c:364: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:365: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c: At top level:  
xml/apr\_xml.c:384: error: expected ‘;’, ‘,’ or ‘)’ before ‘\*’ token  
xml/apr\_xml.c: In function ‘apr\_xml\_parser\_create’:  
xml/apr\_xml.c:401: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:402: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:410: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:411: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:412: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:424: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:424: error: ‘default\_handler’ undeclared (first use in this function)  
xml/apr\_xml.c:424: error: (Each undeclared identifier is reported only once  
xml/apr\_xml.c:424: error: for each function it appears in.)  
xml/apr\_xml.c: In function ‘do\_parse’:  
xml/apr\_xml.c:434: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:438: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c:442: error: ‘apr\_xml\_parser’ has no member named ‘xp\_err’  
xml/apr\_xml.c:442: error: ‘apr\_xml\_parser’ has no member named ‘xp’  
xml/apr\_xml.c: In function ‘apr\_xml\_parser\_geterror’:  
xml/apr\_xml.c:500: error: ‘apr\_xml\_parser’ has no member named ‘xp\_err’  
xml/apr\_xml.c:500: error: ‘apr\_xml\_parser’ has no member named ‘xp\_err’  
make[1]: \* [xml/apr\_xml.lo] Error 1  
make[1]: Leaving directory `/root/Downloads/apache\_svn/apr-util-1.6.0'  
make: \* [all-recursive] Error 19

解决方法：安装expat库：yum install expat-devel

### 9、解压PCRE

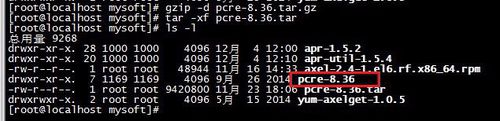
下面，我们来安装PCRE。首先，解包PCRE，并查看/mysoft目录下的文件，依次执行命令：

gzip  -d  pcre-8.36.tar.gz

tar  -xf  pcre-8.36.tar

ls  -l

可以看到/mysoft目录下多了一个目录pcre-8.36。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=11)

### 10、PCRE的编译和安装

进入pcre-8.36目录：cd  ./pcre-8.36/。然后，进行pcre-8.36源码安装，但是，当你在执行./configure时，会看到下图所示提示：

configure: error: You need a C++ compiler for C++ support

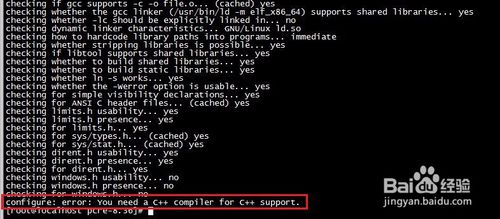
不用担心，yum又派上用场了，执行命令：yum  install  -y  gcc  gcc-c++ 。

gcc-c++安装完成后，我们继续我们的pcre-8.36源码安装，依次执行命令：

./configure

make

make  install

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=12)

### 11、下载Apache2

至此，准备工作就绪，开始准备安装Apache Server2.4，首先，来看下怎么下载Apache2.4，我们可以去http://httpd.apache.org进行下载，看下图。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=13)

### 12、Apache2的一些配置信息

安装Apache之前，我们再来看一小段官方文档：

 --enable-mods-shared=MODULE-LIST

Defines a list of modules to be enabled and build as dynamic shared modules. This mean, these module have to be loaded dynamically by using the LoadModule directive.MODULE-LIST is a space separated list of modulenames enclosed by quotation marks. The module names are given without the preceding mod\_. For example:

--enable-mods-shared='headers rewrite dav'

Additionally you can use the special keywords reallyall, all, most, few and none. For example,

--enable-mods-shared=most will compile most modules and build them as DSO modules

--enable-mods-shared=few will only compile a very basic set of modules

The default set is most.

The LoadModule directives for the chosen modules will be automatically generated in the main configuration file. By default, all those directives will be commented out except for the modules that are either required or explicitly selected by a configure --enable-foo argument. You can change the set of loaded modules by activating or deactivating the LoadModule directives in httpd.conf. In addition the LoadModule directives for all built modules can be activated via the configure option --enable-load-all-modules.

--enable-mods-static=MODULE-LIST

This option behaves similar to --enable-mods-shared, but will link the given modules statically. This mean, these modules will always be present while running httpd. They need not be loaded with LoadModule.

--enable-modules=MODULE-LIST

This option behaves like to --enable-mods-shared, and will also link the given modules dynamically. The special keyword none disables the build of all modules.

翻译：

 --enable-mods-shared=MODULE-LIST

 定义一个要启用的模块列表，并且这些模块会以动态共享模块进行编译。这就意味着，这些模块需要通过使用LoadModule指令进行动态加载。MODULE-LIST是一个由引号引起来且由空格进行分隔的模块名列表。这些模块名要以去掉前面的mod\_的形式给出。例如：

 --enable-mods-shared='headers rewrite dav'

 你还可以使用特殊关键字reallyall,all,most,few和none。例如：

 --enable-mods-shared=most 编译大部分模块并将它们作为动态共享模块进行编译

 --enable-mods-shared=few 只编译一组比较基本的模块

默认关键字是most。

对于已选择的模块来说，LoadModule指令会自动在主要配置文件中生成。默认情况下，所有这些指令会被注释掉，除了一些必需的或是通过一个--enable-foo参数明确指定的模块。你可以在httpd.conf文件中通过激活或注销LoadModule指令来改变已加载模块的设置。另外，可以通过配置选项--enable-load-all-modules来将所有已编译模块的LoadModule指令激活。

--enable-mods-static=MODULE-LIST

这个配置选项的行为与--enable-mods-shared类似，但会静态链接给出的模块。这就意味着，在运行httpd的时候，这些模块总是存在。它们不需要用LoadModule指令进行加载。

--enable-modules=MODULE-LIST

这个配置选项的行为与--enable-mods-shared类似，也会动态链接给出的模块。可以使用特殊关键字none来禁用所有要编译的模块。

### 13、解压Httpd

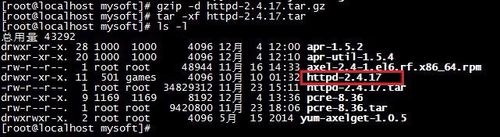
更多的内容大家可以自己去官网看，这里我就不再多讲了。进入正题，解包httpd，并查看/mysoft目录下的文件，依次执行命令：

gzip  -d  httpd-2.4.17.tar.gz

tar  -xf  httpd-2.4.17.tar

ls  -l

可以看到/mysoft目录下多了一个目录httpd-2.4.17。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=14)

### 14、Apache的编译和安装

进入httpd-2.4.17目录：cd  ./httpd-2.4.17/。然后，进行Apache源码安装，依次执行命令：

./configure  --enable-mods-shared=all

make

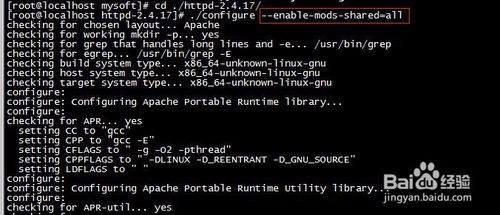
make  install

Apache默认安装在/usr/local/apache2目录下。同样，这里你也可以使用

--prefix=目录名

来指定一个安装目录，例如：

./configure  --prefix=/usr/mydir  --enable-mods-shared=all

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=15)

在安装apache的时候出现错误：

 error: mod\_deflate has been requested but can not be built due to prerequisite failures

解决方法：是因为那是没有安装zlib包。

centos下可以采用yum install zlib-devel, ubuntu下可以采用apt-get install zlib1g-dev

执行make时报错，如下：

make[2]: \*\*\* [exports.lo] Error 1

make[2]: Leaving directory `/usr/local/src/httpd-2.4.7/server'

make[1]: \*\*\* [all-recursive] Error 1

make[1]: Leaving directory `/usr/local/src/httpd-2.4.7/server'

make: \*\*\* [all-recursive] Error 1

解决方法：

在configure后加上 “--with-included-apr”。再重新编译（./configure）， make, make install. 即可。

问题又来了，加上--with-included-apr之后，编译，报错如下：

configure: error: Bundled APR requested but not found at ./srclib/. Download and unpack the corresponding apr and apr-util packages to ./srclib/.

错误为：apr，apr-util缺失，需要下载并解压到./srclib/目录下  
解决方法：  
# cd /usr/local/src/

# cp -r apr-1.5.2 /usr/local/src/httpd-2.4.7/srclib/apr

# cp -r apr-util-1.5.4 /usr/local/src/httpd-2.4.7/srclib/apr-util  
再次执行./configure就不会报错，make，make install也不会报错；

### 15、尝试启动Apache

Apache已经安装成功，现在启动httpd服务。执行命令：

/usr/local/apache2/bin/apachectl  -k  start。

大家会发现，httpd服务并没有启动，而是出现了一条如下图所示的错误信息：

AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using localhost.localdomain. Set the 'ServerName' directive globally to suppress this message

[Linux系统下Apache2.4.17的安装过程](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=16)

### 16、修改配置文件信息

出现上述错误信息，是因为配置文件httpd.conf中的ServerName没有修改。

我们用Vim打开httpd.conf配置文件，执行命令：

vi  /usr/local/apache2/conf/httpd.conf

设置Vim在执行搜索时不区分大小写，执行命令：

:set ic

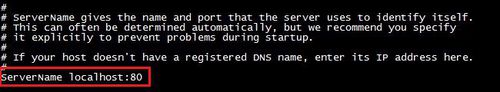
搜索"servername"字符串，执行命令：

/servername

找到servername后，按A键进行编辑。将ServerName前的#号去掉，将ServerName后面的字符串改为：localhost:80，如下图所示。

修改完成后，按Esc键由编辑模式切换到命令模式，然后执行保存退出命令：

:wq

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=17)

### 17、启动Apache服务

在启动httpd之前，我们先为其创建一个软链接（相当于windows下的快捷方式），否则每次重启Apache都要输那么一长串，很费劲。

为httpd创建软链接，并查看一下软链接是否创建成功，依次执行命令：

ln  -s  /usr/local/apache2/bin/apachectl  /mylinks/httpd。

ls  -l  /mylinks

来吧，启动httpd服务，执行命令：/mylinks/httpd  -k  start

[Linux系统下Apache2.4.17的安装过程](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=18)

### 18、查看防火墙状态

在测试Apache是否已经正常工作之前，我们先看下防火墙的状态：

systemctl  status  firewalld

如果处于running运行状态，就把它关掉：

systemctl  stop  firewalld

然后，查看一下防火墙是否已经关掉：

systemctl  status  firewalld

如果处于dead状态，就说明已经关掉了。

如果你希望每次系统启动的时候都不要启动防火墙，可以执行命令：

systemctl  disable  firewalld

同样，如果你希望每次系统启动的时候都启动防火墙，可以执行命令：

systemctl  enable  firewalld

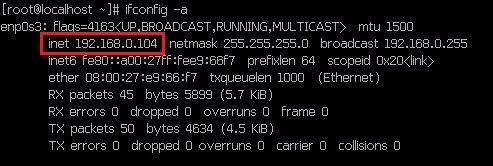
[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=19)

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=20)

[Linux系统下Apache2.4.17的安装过程](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=21)

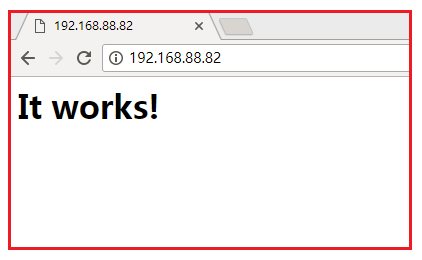
### 19、查看IP地址

防火墙关闭后，接着来看下你系统的IP地址：ifconfig  -a。可以看到我的IP是192.168.0.104。

[](http://jingyan.baidu.com/album/7e4409533d7f0f2fc0e2ef91.html?picindex=22)

### 20、浏览器访问

打开浏览器，在地址栏中输入：192.168.0.104，回车。可以看到It  works!



## 注意事项

* 确保你的Linux系统中已安装了gcc和gcc-c++
* 确保你的Linux系统是连网的