

Table of Contents

Web Server Guide.....	1
Apache 2.2.X.....	2
ActivePerl 5.8.....	3
PHP 5.....	4
MySQL 4.1.....	5
Server Location.....	8
FAQ	10

Apache 2.2.X

1. Download & Unpack

Go to <http://www.apache.org/dist/httpd/binaries/win32/> and choose a mirror. You want the latest **win32-x86-no_ssl.msi** "MSI Installer Package" release.

My file was named: *apache_2.2.6-win32-x86-no_ssl.msi*

NOTE:

Apache apparently removed the "quick download" link to the win32 binary. The long way to officially find it is [Download](#) > **Other files** > **binaries** folder (not the "Binary Releases" info link at the top) > **win32** folder > then the latest **win32-x86-no_ssl.msi** release

2. Install

When you install Apache, you'll get a prompt for "Server Information." Here is the settings I used:

Network Domain: localhost
Server Name: localhost
Admin Email: (your any email)

[**checked**]: for All Users, on Port 80, as a Service

3. Starting/Stopping Apache

After installing, Apache2 automatically starts. The  icon in the System Tray means it started. The  icon means the "Monitor Apache Servers" is running, but Apache2 isn't started.

You can easily start/stop/restart Apache and Apache2 via that icon in your System Tray. If you get "The requested operation has failed!" error while starting apache use the "Test Configuration" shortcut in the Start Menu to find the error (if the text window pops up then closes before you can read it, your config file is fine).

4. Testing

Now the ultimate test. To see if it's serving. Open your browser and head to: <http://127.0.0.1/> or <http://localhost/>

If it shows the ***It works!*** you have your server software installed and running.

5. Making Apache point to your files

Using Notepad open [E:\Apache2_2\conf\httpd.conf](#) (in the start-menu there should also be a "Edit the Apache HTTP Server httpd.conf Configuration File" shortcut) and search for **DocumentRoot**. Change it from something like **DocumentRoot "E:/Apache2_2"** to the location where your HTML files and site are located. In my case: **DocumentRoot "E:/public_html"**

NOTE:

The first time i installed Apache2, i accidentally changed ServerRoot. Make sure you change the correct line which is **DocumentRoot** and which is down about line 150.

6. Then scroll down about one page and change: **<Directory "E:/Apache2_2/htdocs">** to point to the same location you set DocumentRoot to in the last step. Restart apache to make the changes take effect

Introduction: Installing Perl and/or CGI is completely optional and none of the other parts of this tutorial require it to be installed. The only reason to install ActivePerl is if you will be running Perl based CGI scripts.

ActivePerl 5.8

1. Download

Download the latest version of ActivePerl from <http://www.activestate.com/store/activeperl/download> (if it asks you to register, you can just leave the form blank and hit "Continue").

Get the Windows (x86) MSI version. My file was named: *ActivePerl-5.8.8.822-MSWin32-x86-280952.msi*

2. Install

On the "Setup" screen (where you choose what features to install -- should be about the 2nd or 3rd screen in. I tend to miss it every time and have to go back) change the install location (hit the little "browse" button). I highly recommend that you install perl to a directory like:

c:\usr\

In someone else's words (very good advice)

If you will be using Perl CGI programs and want to maintain some level of portability between both Linux machines and Windows machines, you will want to install Perl to the same location on your Windows machine that it is on most Linux machines.

For example, on a standard Linux machine, Perl is located at /usr/bin/perl and so every Perl program that I write begins with #!/usr/bin/perl. So, when I install Perl on a Windows machine, instead of installing it in the default location (which is E:\perl for ActivePerl) I install it in E:\usr so that the Perl executable is located at /usr/bin/perl. This allows me to write code on my Windows machine, then move it (without making any changes) to a Linux machine and have it run there. And vice versa.

[Check] Add Perl to the PATH environment variable

[Check] Create Perl file extension association

The rest should be grayed out and read-only, but if not, leave them unchecked

3. Activating CGI

Using Notepad (or other text editor) open *E:\Apache2_2\conf\httpd.conf* (also should be start-menu shortcut called "Edit Apache HTTP httpd.conf File") and search for Options Indexes FollowSymLinks (about line 190) when you find it add ExecCGI to the end so it looks like Options Indexes FollowSymLinks ExecCGI

4. [OPTIONAL] Enabling CGI in any directory

If you want to use CGI outside the *E:/Apache2_2/cgi-bin/ ScriptAliased* directory, you will need to uncomment the following line: #AddHandler cgi-script .cgi becomes AddHandler cgi-script .cgi (remove the #) I also added .pl behind .cgi so 'perl'

extension is also treated as cgi files.

If you will be creating your own cgi-bin, you will want to comment out: `ScriptAlias /cgi-bin/ "E:/Apache2/cgi-bin/"` so it becomes `#ScriptAlias /cgi-bin/ "E:/Apache2/cgi-bin/"`

5. Finding your location to perl

If you do not know where your perl.exe installed to, go to *Start -> Search* and type in a search for **perl.exe** This location is the path to perl you put on the top of all your cgi scripts. If you listened to my advice in the "Install" step, the path should be close to: `E:/usr/bin/perl.exe`

Some notes

For the perl path **E:/usr/bin/perl.exe** all of these are/were valid. I prefer the last one because of the Linux <<>> Windows portability.

`#!E:/usr/bin/perl.exe`

`#!E:/usr/bin/perl`

`#!/usr/bin/perl.exe`

`#!/usr/bin/perl`

6. Testing CGI

If you if you uncommented (removed the # symbol) the line *AddHandler cgi-script .cgi* in step #4, then create a file in your document_root called **hello.cgi** and put these three lines in it (if you did not comment/disable it, put the CGI file in

`E:/Apache2_2/cgi-bin/`):

`#!/usr/bin/perl`

`print "Content-type:text/html\n\n";`

`print "hello world";`

7. Restart Apache if it is already running. Now go to <http://localhost/cgi-bin/hello.cgi> (or wherever you put the file) and run the script.
8. If you get a *hello world* in your browser, CGI is running. If you get a 500 error, go to the last entry in E:/Apache2_2/logs/error.log (or the *Review Error Log* in the start menu) to see exactly what caused this error.

PHP 5

1. Download & Unpack

Download and install PHP from <http://www.php.net/downloads.php>, you should grab the newest 5.x.x Windows Binary zip package that will work on apache.

My file was named: *php-5.2.4-Win32.zip*

2. Unzip php. In my case, I unzipped to:
e:\php\
3. Rename **e:\php\php.ini-dist** it to **php.ini**
4. **Edit your php.ini**

Open php.ini in a text editor and scroll down about halfway through the file and look for doc_root then change it to point to whatever your Apache DocumentRoot is set to. In my case: doc_root = "e:\public_html"

Scroll down about 7 more lines and change the extension_dir from extension_dir = "." to the location of the ext directory after you unzipped PHP. in my case:
extension_dir = "e:\php\ext"

If you are going to be using your server for testing, i recommend (this is optional) you search and change the error reporting to give you info on any errors, notices, or bad coding. If you want to enable this type of stuff, search for error_reporting for and change:

```
error_reporting = E_ALL & ~E_NOTICE & ~E_STRICT  
to  
error_reporting = E_ALL | E_NOTICE | E_STRICT
```

5. Editing Apache Conf File

Using Notepad open *e:\Apache2_2\conf\httpd.conf* (in the start-menu there should also be a "Edit the Apache HTTP Server httpd.conf Configuration File" shortcut). Either at the very beginning or end of the file add the following lines: (**NOTE:** be sure to change BOTH e:/php parts to the directory you installed your php to)

```
LoadModule php5_module "e:/php/php5apache2_2.dll"  
AddType application/x-httpd-php .php  
PHPIniDir "e:/php"
```

Note: If you installed the older Apache 2.0, instead of the above lines, you will need to use the lines listed on the bottom step of the [Apache 2.0 tutorial](#).

6. [OPTIONAL] Editing Apache Conf File (part 2)

To get apache to automatically look for an index.php, search httpd.conf for **DirectoryIndex** (about line 212) and add the files you want apache to look for when a directory is loaded (if it doesn't find any of these files, it displays folder contents).

Mine looks like:

```
<IfModule dir_module>  
    DirectoryIndex index.php index.html default.html  
</IfModule>
```

7. Testing

Restart Apache if it is already running (if it doesn't start or you get errors, use your Apache "Test Configuration" shortcut in the Start Menu to see why).

To test your PHP simply create a **test.php** file in your Apache "DocumentRoot" folder (E:\public_html\ in my case). In your test.php file, type these 3 lines and then load the file in your browser like <http://localhost/test.php> (you should get a whole long list of php variables, settings, etc):

```
<?php
phpinfo();
?>
```

8. Documentation Suggestion

One weird thing I have noticed about PHP is that it does not come with documentation of any kind. If you are a php developer/programmer, I suggest you download the documentation. Downloads can be found on <http://www.php.net/download-docs.php> and I personally recommend the "English - Windows HTML Help" (chm) version as the search is so handy, although they all contain the same information.

MySQL 4.1

1. Download MySQL v4.1

Download MySQL from <http://dev.mysql.com/downloads/mysql/4.1.html> Just make sure you get a *Windows Essentials (x86)* binary version. (after you click the "pick a mirror," if it asks you to login, scroll down a little and click the "*No thanks, just take me to the downloads!*" link under the login form)

My file was named: *mysql-essential-4.1.22-win32.msi*

2. Install MySQL v4.1

Run the MSI file and use the following settings

- Typical Setup
- Skip Sign-Up
- make sure "Configure the MySQL Server now" is checked
- "Detailed Configuration"
- "Developer Machine"
- "Multifunctional Database"
- "InnoDB Tablespace Settings" - leave everything default
- "Decision Support (DSS)/OLAP"
- make sure "Enable TCP/IP Networking" is checked and leave the port number at 3306 (at this point, if you have a firewall, it will usually try to access itself on the localhost)
- "Standard Character Set"
- check "Install As Windows Service"
- enter your root password and I would recommend leaving "Enable root access from remote machines" unchecked
- then hit "execute" and it'll install and set it up.

3. Getting PHP5 to work with MySQL - [Official Info](#)

Unfortunately PHP5 removed built-in support for MySQL. To get it to work, the easiest way is to copy the mysql library file by hand. Open the folder you unzipped your PHP to. Copy the **libmysql.dll** file (should be located like E:\php\libmysql.dll) into your Window's System folder (usually C:\Windows\System32\ although might be C:\WinNT\System\ or something).

Then open up your php.ini in a text editor and search for **;extension=php_mysql.dll** and remove the ; in front of that line.

Restart Apache and see if you get any errors. If it complains about "php_mysql.dll" either your extension directory isn't correct or windows can't find libmysql.dll

4. Testing MySQL

Testing MySQL is not exactly easy. However, here are the common connection strings for PHP and CGI. I recommend downloading [phpMyAdmin](#) and using it to create and manage your databases, etc.

PHP Connection test

```
<?php
// hostname or ip of server (for local testing, localhost should
work)
$dbServer='localhost';

// username and password to log onto db server
$dbUser='root';
$dbPass='';

// name of database
$dbName='test';

$link = mysql_connect("$dbServer", "$dbUser", "$dbPass") or
die("Could not connect");
print "Connected successfully<br>";
mysql_select_db("$dbName") or die("Could not select database");
print "Database selected successfully<br>";

// close connection
mysql_close($link);
?>
```

CGI Connection test (Must have DBI module installed)

```
#!/usr/bin/perl

print "Content-type: text/html\n\n";

# DBI is perl module used to connect to the database
use DBI;

# hostname or ip of server (for local testing, localhost should work)
$config{'dbServer'} = "localhost";

# username and password to log onto db server
$config{'dbUser'} = "root";
$config{'dbPass'} = "";

# name of database
$config{'dbName'} = "test";

# MySQL driver (shouldn't need to change)
$config{'dataSource'} = "DBI:mysql:$config{'dbName'}";
$config{'dbServer'}";

my $dbh = DBI->connect($config{'dataSource'}, $config{'dbUser'},
$config{'dbPass'}) or
die "Can't connect to $config{'dataSource'}<br>$DBI::errstr";
print "Connected successfully<br>";
$dbh->disconnect();
```

Some notes for CGI users

If you do not know how to install DBI here's how.

On your start menu there should be a *ActiveState ActivePerl 5.8 > Perl Package Manager* shortcut. If it's the old console version, type **install dbi** and when it finishes type **install DBD-mysql** command. If it's the newer version, look for the packages **DBI** and **DBD-mysql**. Find each one and "Mark for install" then "Run marked actions" to install them. If you can't find one of the packages listed, make sure you have "View all packages" button selected on the top left.

Server Location

1. Getting your Server Location

Unfortunately unless you have a static IP with your internet connection, your IP address (and server URL) will change every time you connect to the internet, or your modem gets reset etc. To get your current IP (Internet Protocol) number do one of the following:

Windows 95, 98:

Start -> Run -> type in **WINIPCFG** when the window pops up use the top pull down menu to select the adapter you use to connect to the internet. I had an Ethernet [network] and PPP [dial-up] adapter listed, and by default everytime it's sitting on the network adapter.

Windows 2000, XP:

Start -> Run -> type in **cmd** (this starts DOS/Command prompt). Type in **ipconfig** and you should get a short list of settings with your IP address displayed. (if you want to see a complete list: **ipconfig /all** but you don't usually need that much info)

Now open the browser and type in <http://your.number> (like <http://203.213.7.131>). Now anybody can access your site by that number till your IP number changes (usually when you get disconnected from the internet it will change).

Now open the browser and type in <http://your.number> (like <http://203.213.7.131>). Now anybody can access your site by that number till your IP number changes.

Shortcut:

Well, unless you are using a proxy, your IP and server is 203.213.7.131

2. Getting a Dynamic DNS Provider

If your IP changes each time you log onto the internet, you might consider getting a free Dynamic DNS provider. They work by giving you a static name like <http://yourname.DNSprovidername.com>. Then each time you connect to the internet, the provider will update so your URL will point to your ever changing IP number. This is a great service, and check out the [Dynamic DNS primer](#) for a better explanation of dynamic DNS, what it is, and how it works.

3. LAN IP

If "winipcfg" or "ipconfig" from above returned a number that was in the 192.168.x.x range, that IP is your LAN (Local Area Network) IP. Only computers that are sharing your internet connection can access your server using that number. In which case you'll need to setup your router/modem to forward your "outside" IP like 203.213.7.131 to your 192.168.x.x LAN ip.

4. Router Issues

If you have a router or share your high-speed connection with multiple computers, you will likely need to log into your router and configure it to forward Port 80 to your computer. How you do that depends on your router. The basic info to get you started:

Manufacturer	Login IP	Default username	Default Password
Netgear	192.168.0.1	admin	password
D-Link	192.168.0.1	admin	
Linksys	192.168.1.1		admin

Look for something under "Virtual Servers" or "Forwarding." Type of info it typically asks:

Private IP: (enter your LAN ip. should be something like 192.168.x.x)

Protocol Type: TCP

Private Port: 80

Public Port: 80

If you still need help, check google or your router manual as I usually do not answer questions relating to this.

FAQ

- **PHP not keeping sessions active between pages**

This can usually be solved by editing php.ini and setting the session.use_trans_sid parameter to 1 so the line looks like session.use_trans_sid = 1

- **When Apache launches, it gives error: *No installed service named "Apache2"***

For some reason Apache doesn't always install the service. What you need to do is go into command prompt (start > run > type in cmd) then type in: c:\apache2\bin\apache -k install

Make sure change that path to the location you installed Apache to. That should put the service you need back in.

A reader told me they also had the same problem and to fix it they went into the "Custom" option during Apache Install and changed the install location. He thought the problem was due to the old "space-in-the-name" problem. Reinstalling to a path without spaces fixed the problem for him. On the other hand, i've installed it many times with the spaces and never had a problem.

- **Apache gives the error *Only one usage of the socket address (protocol/network address/port) is normally permitted.***

This message means that something else is already using your port 80 that apache wants to use. Check to make sure you don't have something like Microsoft's IIS (Internet Information Server), some file sharing program with a 'remote login' option, or some other type of server that might be using port 80.

- **MySQL won't start as a service**

A reader with this problem said he completely uninstalled MySQL then reinstalled changing the MySQL service name from the default to **mysql41** (or any other name different than default) and it started right up after that.

- **How do I install a module for PHP like GD2**

In your php.ini file, make sure the "extension_dir" setting is correct (so php knows where to look for the GD extension), then just search for extension you want like: ;extension=php_gd2.dll

and remove the ; in front of it so apache knows to load the module. Simply restart apache and it should load the module and work.

- **How do I install phpMyAdmin**

Because of frequent requests, i'm going to give a simple 1-2-3 on how to install [phpMyAdmin](#). If you want to install it "correctly" you should [follow their official docs](#). If you want to get it working fast, do this:

1. **Download** the sucker (there are "quick downloads" links at the top left)
2. **Extract** it somewhere in your document_root (the folder where your html "web files" are). It creates it's own sub-folder which you might want to rename after extracting

3. **Rename** the ugly long-tailed *phpMyAdmin-2.9.2-all-languages-utf-8-only* folder it created to something shorter like: *phpmyadmin*
4. **Edit** the file *phpmyadmin/libraries/config.default.php* ignoring all the "do not edit" wrnings and scroll down to about line 150. The lines you want to fill out are:
 5. `$cfg['Servers'][$i]['user']` = `'root';` // MySQL user
 6. `$cfg['Servers'][$i]['password']` = `'';` // MySQL password
7. **Load** the sucker up in your browser like <http://localhost/phpmyadmin/> (or wherever you installed it to)