

# Setting Up Virtual Hosts for XAMPP

Running a Web server on your desktop computer is a great way to speed up Web site development. By previewing your Web pages directly through a Web server on your own computer you can test out server-side programming like server-side include files, form processing scripts, or database-driven Web pages. However, there's one problem associated with running a server on your own computer: by default, you only have a single "domain"—`http://localhost/`—so if you work on more than one Web site you put them all under "localhost" in different directories. This isn't very real world and can cause problems when using server-side includes and root-relative links. Fortunately there's a better way.

For information on setting up XAMPP, [visit the XAMPP instructions page](#).

## Virtual Hosts

**Virtual Hosts** give you the ability to "host" more than one Web site and domain on your computer. With a virtual host you can have separate local domain names for each of your Web sites: for example, `http://clientA/` for one site and `http://clientB/` for another. When you type the URL for the Virtual Host in your Web browser, the browser doesn't go out onto the internet to find the site, but instead asks for the proper file from the Web server running on your computer. Not only does a Virtual Host let you run multiple Web sites on your computer, but it also lets you store the files for those sites anywhere on your computer and not just in the `C:\XAMPP\htdocs` folder.

Adding a Virtual Host is a 2-step process:

### 1. Add a new entry to your computer's *hosts* file.

A *hosts* file can be used to point requests for a domain to a particular IP address in other words, it lets you re-direct communications to a particular domain. In the case of a virtual host, it can tell the computer to NOT go out on the internet when you type a particular URL like `http://clientA/`, but instead look for that particular domain on your own computer.

### 2. Edit the Apache configuration file to accept Virtual Hosts and define the particular Virtual Hosts you want to setup on your computer.

The first step above, merely redirects requests from a particular domain to your computer, but this step prepares the Web server on your computer for handling those requests. In this step, you not only provide the name of the Virtual Host, but also tell Apache where the files for the site are located on your computer.

## Detailed Steps

### 1. Launch Notepad and open the *hosts* file located at `C:\windows\system32\drivers\etc\hosts`. (You may not be able to see the windows folder—some files are hidden by default under Windows. Here are [instructions to make those files visible](#).)

On Vista, you'll also need to have access to change the hosts file. To do that, launch Notepad by right clicking on Notepad from the Start menu and choosing "Run As Administrator." This will give you permission to edit and save the file.

### 2. At the end of that file type:

## Recent Stuff

Eric Meyer on Non-Breaking Space Show

The #1 Reason to Use Sass

Nonbreaking Space Podcast : Interview with Aaron Walter

Kristina Halvorson on Non-Breaking Space Show

Testing on Mobile Devices with Adobe Shadow

CSS Sizing Advice: %, REMS, and EMs

New CSS3 Cursor Styles

HTML5 / CSS3 Browser Support

Keep an eye out for these HTML5 elements

Dreamweaver CS5.5: The Missing Manual Is Here!

### 127.0.0.1 clientA.local

127.0.0.1 is how a computer refers to itself—it's an IP address that points back to the computer, kind of like a computer's way of saying "ME." The second part (clientA.local) is the "domain" of the virtual host. To visit this domain in a Web browser you'd type `http://clientA.local`. You don't have to add the .local part to the hosts files—you could just as easily add **127.0.0.1 clientA** and access the site in your Web browser with `http://clientA`—but I find it helpful for differentiating between a real Web site out on the Internet like **clientA.com**, and the test sites I have running on my own computer.

#### 3. Save and close the *hosts* file.

That finishes the first part of this task. You've prepared your computer to handle requests to `http://clientA.local`. Now you need to tell the Web server, Apache, how to handle those requests.

#### 4. In Notepad open the Apache configuration file located at `C:\xampp\apache\conf\extra\httpd-vhosts.conf`

#### 5. At the bottom of that file add:

```
NameVirtualHost *
<VirtualHost *>
    DocumentRoot "C:\xampp\htdocs"
    ServerName localhost
</VirtualHost>
<VirtualHost *>
    DocumentRoot "C:\Documents and Settings\Me\My Documents\clientA\website"
    ServerName clientA.local
    <Directory "C:\Documents and Settings\Me\My Documents\clientA\website">
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

The first five lines of code turn on the Virtual Host feature on Apache, and set up the `C:\xampp\htdocs` folder as the default location for `http://localhost`. That's important since you need to be able to access the XAMPP web pages at `http://localhost/` so that you can use PHPMyAdmin.

The stuff in yellow represents a single Virtual Host. You'll add one chunk of code just like this for each Virtual Host (or Web site) on your computer

You'll need to modify the stuff highlighted in blue. The first item — `DocumentRoot` — indicates where the files for this site are located on your computer. The second part—`ServerName` — is the name you provided in step 2 above: the virtual host name. For example, **clientA.local**. The third item — the `<Directory>` part — is the same path you provided for the `DocumentRoot`. This is required to let your Web browser have clearance to access these files.

#### 6. Save and close the Apache configuration file, and restart Apache from the XAMPP control panel.

#### 7. Start a Web browser and type a URL for the virtual host. For example:

`http://clientA.local/`.

You should now see the home page for your site.

## More Virtual Hosts

If you want to add additional Virtual hosts add the proper entry to the hosts file and add another block of text like that in yellow above to the Apache configuration file. For example, say you had another Web site for ClientB. You'd add **127.0.0.1 clientB.local** in the hosts file and the `C:\xampp\apache\conf\extra\httpd-vhosts.conf` would look like this:

```
NameVirtualHost *
<VirtualHost *>
    DocumentRoot "C:\xampp\htdocs"
    ServerName localhost
</VirtualHost>
<VirtualHost *>
    DocumentRoot "C:\Documents and Settings\Me\My Documents\clientA\website"
    ServerName clientA.local
    <Directory "C:\Documents and Settings\Me\My Documents\clientA\website">
        Order allow,deny
        Allow from all
    </Directory>
```

```
</VirtualHost>
<VirtualHost *>
    DocumentRoot "C:\Documents and Settings\Me\My Documents\clientB\website"
    ServerName clientB.local
    <Directory "C:\Documents and Settings\Me\My Documents\clientB\website">
        order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

## More Information

---

[This site also has instructions for installing XAMPP.](#)