



# Nuking The Competition

By icarus

This article copyright [Melonfire](#) 2000-2002. All rights reserved.

# Table of Contents

<b><u>Plug And Play Technology</u></b> .....	<b>1</b>
<b><u>Getting Started</u></b> .....	<b>2</b>
<b><u>Building Blocks</u></b> .....	<b>4</b>
<b><u>Managing Things</u></b> .....	<b>6</b>
<b><u>All Wired Up</u></b> .....	<b>8</b>
<b><u>Doing The Funky Chameleon</u></b> .....	<b>9</b>
<b><u>Commercial Break</u></b> .....	<b>10</b>
<b><u>Conclusions</u></b> .....	<b>11</b>

# Plug And Play Technology

If you're a fan of open-source technology, you're probably familiar with Slashdot (<http://www.slashdot.org/>), one of the pre-eminent news sites for everything Linux and open-source. Part of the reason for Slashdot's popularity is its open publishing system – absolutely anyone can register and post news items and comments to the Web site. In addition to frequently updated articles (and comments that range from intelligent to idiotic), the site also offers news, online polls, FAQs, discussion forums, user-based personalization, a search engine, links to related Web sites and much, much more.

Of course, Slashdot isn't the only site offering visitors these capabilities – its brethren include Squishdot (<http://www.squishdot.org/>), Midgard (<http://www.midgard-project.org/>), and some aspects of Zope (<http://www.zope.org/>). And though setting up such a site may appear complex and time-consuming at first glance, it's actually not that difficult – and the rewards can be immense, especially if the finished product catches the imagination of Joe Netizen.

Over the next few pages, I'm going to discuss deployment of just such a content publishing system, in order to illustrate how painless the process can be. However, instead of building it from scratch, I'm going to be using the free PHP-Nuke system as the basis for this development effort, as it includes all of the features I need...and quite a few more besides!

These features include a powerful and flexible content management system, online polls and user registration support. Users can post articles to the Web site (which may optionally be moderated before appearing in the site index), post comments on these articles (a threaded discussion board is also available), search for specific articles, and personalize the interface to their requirements. An integrated banner rotator makes it easy to plug in commercial advertising, while freely available "themes" make it easy to alter the appearance of the user interface. Finally, numerous free add-on modules make adding new features, such as a guestbook, a calendar, an email reader and multi-language support a snap!

This article copyright [Melonfire](#) 2001. All rights reserved.

# Getting Started

The first order of business is – obviously – to download and install PHP–Nuke. You can download a copy from <http://www.phpnuke.org/>, the official site; since this site itself runs on PHP–Nuke, this is also an opportunity to see the features available first–hand. A couple of different versions are available – I'll be using version 4.4.1a throughout this article, although you should also take a look at version 5.0 (in beta at the time of writing).

PHP–Nuke requires a Web server running PHP, and a mySQL database server; I've tested it on both Windows and Linux, and it works fine on both operating systems. If you've got these pre–requisites, unpack the distribution into a directory off your Web server root (say, "/nuke/"), and create a new mySQL database to hold your PHP–Nuke data (say, "nuke").

---

```
mysql> CREATE DATABASE nuke;  
Query OK, 1 row affected (0.00 sec)
```

---

Once the database has been created, you can import the default database schema from the included SQL file, "nuke.sql".

---

```
$ mysql -D nuke -u root -p < nuke.sql
```

---

Next, you need to update the system configuration to reflect the database, username and password. These variables can be found in the "config.php" global configuration file in PHP–Nuke's "html/" directory; update them to reflect your system configuration.

---

```
$dbhost = "localhost";  
$dbuname = "root";  
$dbpass = "hf843i2";  
$dbname = "nuke";
```

---

You may also have to specify a "theme" for PHP–Nuke (more on this later). The default theme is OpenMind, but if your distribution of PHP–Nuke doesn't include this theme (mine didn't), peek in the "themes/" directory, pick a theme and let "config.php" know.

## Nuking The Competition

```
$Default_Theme = "Sunset";
```

---

At this point, the system is ready to go – point your browser to the location where you installed the files (say, "http://localhost/nuke/html/"), and you should see something like this.



This is the default configuration and layout – but it's by no means cast in stone. PHP-Nuke offers extensive customization options, allowing you to alter every aspect of the default GUI. In order to illustrate, point your browser to the administration section (say, "http://localhost/nuke/html/admin.php") and log in as user "God", password "Password".

This article copyright [Melonfire](#) 2001. All rights reserved.



# Building Blocks

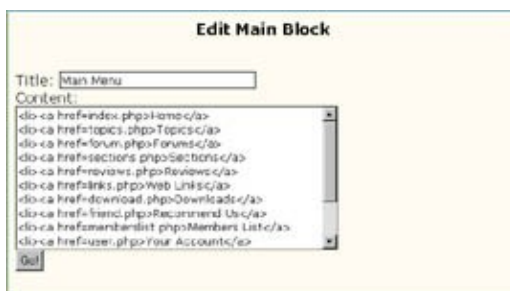
The administration section contains everything you need to configure the PHP-Nuke system – however, the first thing you should do is visit the "Edit Admins" section and alter the super-user password for greater security.



At this point, you may also add new administrators to the system – this comes in handy if you expect a great deal of activity on your portal and would like to assign specific individuals or editors to moderate different sections of the site. Administrator rights can be assigned either for specific sections – articles, topics, downloads, links, discussion boards – or for the complete system.

If required, you can also edit registered user accounts – this includes altering the account password and assigning the user moderator privileges.

The administration section is also your starting point if you need to customize the PHP-Nuke interface. For example, let's suppose you'd like a shorter main menu on your front page. The administration section offers the "Main Block" item, which allows you to customize the menu items which appear in the main menu – delete existing items or add your own, the process is as simple as filling out a text box.



You can do the same with the other blocks which appear on the right and left borders – simply select the "Left Blocks" and "Right Blocks" options in the administration menu to edit or delete existing blocks, or add new ones. The changes you make will be immediately reflected throughout the site.

This is probably one of the better features of the PHP-Nuke system. Most of the time, the default blocks are not what you will want, and you will need to alter them to reflect your own preferences. Of the various options available, I try to always make sure that the login block, the news block and the survey are always visible; additionally, the last time I deployed this system, I added a couple of new blocks containing useful information or instructions for visitors, a link to the "site of the day", and promotional blocks containing sponsor messages.

## Nuking The Competition

The "Preferences" section is the place to go when you need to alter site-specific information – the name of your site, the URL and logo, the theme you'd like to use, the language, the text which appears at the bottom of every page, and default configuration options for different sections of the site. This is one of the more important administration sections, and I'd suggest you spend some time on the options available.



**General Site Info**

Site Name:

Site URL:

Site Logo:

Site Slogan:

Site Start Date:

Administrator Email:

Number of Items in TOP Page:

Stories Number in the Home:

Stories in Old:

Articles Box:

This article copyright [Melonfire](#) 2001. All rights reserved.

# Managing Things

Moving deeper into the guts of the system, let's take a brief look at the other administration options, and what they mean to a site administrator.

Every story published through PHP–Nuke must be linked to a specific "topic" – this is a fundamental feature of the system, and also makes it easier to categorize posted items. The default PHP–Nuke configuration contains a list of topics, all of them related to software and operating systems. Since this is obviously not going to work for everyone – on the two occasions I've deployed the system, neither of the sites was even remotely connected to the software industry – PHP–Nuke offers a "Topics Manager" to help you alter the default topics. You can use this to either edit or delete existing topics, or add new ones.



Similar to the "Topics Manager" is the "Section Manager", which allows you to create special sections on the site. Personally, I've never found a use for this feature – the topics are usually more than sufficient, and the distinction between "topics" and "sections" is fine enough to confuse the average user – but it's good to have the additional flexibility.

Finally, a "Forum Manager" makes it possible to create new discussion boards in the "Forums" section – note that you'll need to create forum categories before you can begin adding new boards. The manager allows you to specify a topic for the forum, a moderator (if you'd like to filter posts), and the access level (anyone/registered users only/administrators only). These settings can be specified on a per–forum basis.

In particular, these kinds of discussion boards would be valuable to businesses in the service industry – for example, Web hosting. A Web hosting company could set up discussion boards devoted to different aspects or features of their service offering – email, Perl/CGI, PHP, databases – and invite their customers to use this forum to communicate with other customers, and with administrators. Properly supported and promoted, and backed by a good search engine (PHP–Nuke comes with a fairly good one), this kind of community–supported support system would end up being a valuable resource to both the business and its customers.

The "Forum Ranking" and "Forum Configuration" items allow you to rank your discussion boards and set up basic configuration parameters; the defaults are usually fine here.

Finally, the "New Article" section is designed to allow administrators to post new stories to the site – this content appears on the main page of the site. Most of the options are self–explanatory, and the online help provides more than enough information on what each of the fields represents.

An interesting feature here is the "Auto Articles" option. This allows administrators to enter new articles into the system, but delay their publication until a specific date. Again, if you have a complete editorial team



## Nuking The Competition

working on your site, or you don't update it on a regular basis, this option has limited value; however, it comes in handy if you run your Web site as a hobby and work a five-day week (you can spend the weekend putting together new content and automatically release the new material through the week!)

### When do you want to publish this story?

Now is: May 17, 2001 @ 12:44:44

Day:  Month:  Year:

Hour:  :  : 00

This article copyright [Melonfire](#) 2001. All rights reserved.



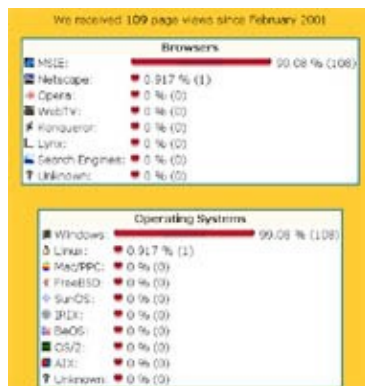
# All Wired Up

The other administration menu items are very specific – the "Download", "Web Links", "FAQ" and "Surveys/Polls" sections allow administrators to add new items to the respective areas of the Web site, and you will need to visit them whenever you decide to update your Web site.

The "Headlines" section allows your site to read news articles from other Web sites using an RDF or XML source. This is a pretty powerful feature, and it's a great way to make the front page dynamic – I'd recommend using this if you can. A number of XML sources are available by default, including PHPBuilder, LinuxPreview and PHP–Nuke itself, and you can add your own to the list as well.

Another interesting (though infrequently–used) feature is "Ephemerids", which allows you to remind users of specific events that occurred in the past. For example, if April 2 is your birthday, you can enter the date into the "Ephemerids" system, together with a short description of the event, and the system will pop up a notification of the event to every visitor on April 2 every year. It's fun and easy to do; however, you should activate this feature only if it fits in with the overall tone of your site.

As visitors browse your PHP–Nuke site, the system is keeping itself busy as well, tracking which stories are being read and storing browser information. All this information is then compiled and made available to administrators and users – the "Stats" section displays pageviews per month, together with a graph indicating the various browser types and operating systems preferred by visitors, while the "Top 10" section keeps track of the most active categories, articles, threads and polls. This information is very valuable, and is one of the reasons why I like PHP–Nuke so much – it makes tracking visitor activity far simpler than it usually is.



This article copyright [Melonfire](#) 2001. All rights reserved.

# Doing The Funky Chameleon

While the default administration interface does allow access to most of the things an administrator would need, it's not the only option available. The variables you set through the various control panels correspond to the variables in the "config.php" configuration file, and to the records in PHP-Nuke's extensive database. If you're a power user, you should spend some time studying the configuration file and the database schema to understand how the various tables are linked together; the information's bound to come in handy if things go wrong.

It's also sure to come in handy when you decide that the default PHP-Nuke configuration no longer works for you. As I've already stated, there are a large number of PHP-Nuke themes available for download – like WinAmp skins, these themes allow administrators (and registered users) to personalize the PHP-Nuke system extensively, even to the extent of completely altering colours, page layout and displayed articles. While the default themes which ship with PHP-Nuke are fine to start out with, you're going to have to learn to make your own – especially if you plan to deploy this system on a commercial basis.

PHP-Nuke themes are relatively easy to build, once you have a sound understanding of the architecture. Every theme consists of three main files – "header.php", "theme.php" and "footer.php" – together with required images.

Each of these files has a specific function in the PHP-Nuke system. "header.php" sets up the header which appears at the top of every page, together with the blocks on the left side; "theme.php" sets up the UI for the articles and news items; and "footer.php" sets up the blocks on the right side, and a page footer. The order in which the various pre-defined and custom blocks appear can be controlled as well – each of the blocks is represented by a function call, as the following snippet from "header.php" demonstrates:

---

```
mainblock();
adminblock();
leftblocks();
ephemblock();
headlines();
online();
```

---

As you might imagine, if you know HTML, slapping together your own theme is not very hard at all. However, rather than starting from scratch, I'd recommend that you take an existing theme file and modify it to reflect your requirements (at least the first couple of times that you attempt this exercise). This will help you get to grips with the function of each file, and also provide some understanding of which parts of the system can be customized easily, and which ones require more work.

If you'd like to learn more about building your own themes, I'd recommend that you also take a look at the excellent tutorial at <http://www.dinerminor.com/> for a detailed, step-by-step description of the process.

This article copyright [Melonfire](#) 2001. All rights reserved.



# Commercial Break

PHP–Nuke also comes with a pretty good built–in banner advertising system, so that webmasters can begin offering advertisers banner placement facilities immediately. The banner administration system, available from the administration menu, allows administrators to create accounts for different advertisers, and specify the number of impressions purchased by each. Advertisers can provide one or more banners, and PHP–Nuke will take care of rotating through and displaying the banners as visitors browse through the site.

The image shows two forms from the PHP-Nuke administration interface. The top form, titled 'Add a New Banner', includes fields for 'Client Name' (with a dropdown menu), 'Impressions Purchased' (with a value of 0 and 'Unlimited' selected), 'Image URL', and 'Click URL'. Below these fields is an 'Add Banner' button. The bottom form, titled 'Add a New Client', includes fields for 'Client Name', 'Contact Name', 'Contact Email', 'Client Login', and 'Client Password'. There is also a large text area for 'Extra Info' and an 'Add Client' button at the bottom.

A neatly compiled table allows administrators and advertisers to immediately view statistics per banner – number of views, number of clicks, and percentage of views converted into clicks. This information is also available to the advertiser via email.

The image shows three tables from the PHP-Nuke administration interface. The first table, 'Current Active Banners', has columns for ID, Impressions, Imp. Left, Clicks, % Clicks, Client Name, and Functions. The second table, 'Finished Banners', has columns for ID, Imp, Clicks, % Clicks, Date Started, Date Ended, Client Name, and Functions. The third table, 'Advertising Clients', has columns for ID, Client Name, Active Banners, Contact Name, Contact Email, and Functions.

Current Active Banners						
ID	Impressions	Imp. Left	Clicks	% Clicks	Client Name	Functions

Finished Banners							
ID	Imp	Clicks	% Clicks	Date Started	Date Ended	Client Name	Functions
1	5	2	40%	2001-04-20 11:24:35	2001-04-20 11:29:02	Melonfire	

Advertising Clients					
ID	Client Name	Active Banners	Contact Name	Contact Email	Functions
1	Melonfire	0	John Doe	john@isomesite.com	

While the system is pretty basic, it works well, and should satisfy most requirements...and, of course, you can always invest in a more professional system once the green stuff starts rolling in!

This article copyright [Melonfire](#) 2001. All rights reserved.

# Conclusions

There are many uses for the PHP–Nuke system – a budding webmaster could use it as the basis for his or her community portal, a developer or consultant could use it to rapidly build a portal for a client, and a hobbyist could take it apart to understand how it works.

As a tool for rapid development and delivery, it offers numerous advantages – it's easy to install and configure, the built–in theme support makes it easy to customize the interface, and the active community support (take a look at <http://www.nukeaddon.com/> for a bunch of great add–ons, toys and utilities built especially for this system) and open license make bug fixes and modifications a breeze. The XML/RDF support is also a nice touch, and makes it easy to integrate other content sources into the system.

That said, there are a few things that I'd like to see changed as well. A greater separation of interface code from application logic would be first on my wish list, as this would make it easier to build themes for the system (this is apparently expected in the next release of the software). I'd also look for improvements in the administration user interface, especially in the menus and the aesthetics, and better documentation for future releases.

That's about it for the moment. I hope this article offered some insight into the features available in the PHP–Nuke system, and some guidelines on how to go about deploying and customizing it to your own needs. If you have comments, or simply want to share your experiences with PHP–Nuke, drop me a line – and till next time, stay healthy!

This article copyright [Melonfire](#) 2001. All rights reserved.