



..... A

Definitive

GUIDE

to

**WordPress Optimization,
SEO, Security**

.....

Techglimpse.com



A Definitive Guide to
WordPress Optimization, SEO, Security

Techglimpse.com

Table of Contents

Introduction.....	4
WordPress Security.....	5
Change WordPress table prefix (during install)	5
Don't create username as 'admin'	7
Install coming soon plugin while you do development at the backend	8
Display name should not be username	9
Disable Plugin and Theme Editor in wp-admin.....	12
Disable XML-RPC	14
Change WordPress Login error message	15
Change WordPress admin URL	16
Enable Two-Step Authentication for WordPress Admin	18
Install User lockout Plugin.....	20
Delete unwanted themes and plugins	22
Install iThemes Security Plugin	24
Change those default Security Keys.....	25
Rename Theme folder	26
WordPress SEO Optimization.....	27
Disable dashboard on Multi-author site	27
Remove unwanted Meta tags.....	29
Remove Query strings from URLs	30
Remove or hide Secondary feed URLs	31
Remove extra JS file added to support Emoji	32
Prevent search engine from Indexing XML Sitemap.....	35
Install WP-Optimize to optimize database	36
Install W3 Total Cache to speed your site.....	38
Prioritize Critical Content above the fold	39
Install Best SEO Plugin.....	42
Analyze and Speed up your website	43
Cache Static contents by using ExpiresByType	44
Use CloudFlare Free Service.....	45

Optimize Images automatically	46
Manage 301 redirects easily	47
Prevent search engines from Indexing core files	48
WordPress Optimization	49
Remove HTML in WordPress Comments	49
Limit Comments Length	50
Move WordPress Uploads directory	51
Disable Post Revisions	52
Increase the interval of Posts Autosave	53
Disable Admin bar	54
Remove WordPress Footer notes	55
Disable Automatic Updates	56
Increase WP upload size	57
Limit number of posts in RSS feed	58
Use Distraction-free mode for writing	59
Stay logged-in to WordPress for longer duration	61
Remove unused CSS styles using Chrome Developer tool	62
Handle Adblockers smartly	64
References	66

Introduction

WordPress is an amazing platform to build your website or a blog for free. Originally, WordPress was developed as blogging software, but at later stages, WordPress being so robust, Millions of people have used it to build beautiful websites. It's powerful, easy to install, loaded with features, active community, millions of plugins etc., WordPress is completely customizable and there is no limit to what type of website you can create using WordPress. WordPress being SEO friendly is more popular among the CMS powered websites. Started off in 2003 with fewer users, today it has grown to be the largest self-hosted blogging tool in the world, used on millions of sites and seen by tens of millions of people every day.

Because of growing use and popularity of WordPress it's also the platform loved by hackers. A recent investigation has concluded that WordPress sites were attacked 24.1 percent more than websites running on all other CMS platforms combined and that it suffers 60 percent more cross-site scripting incidents than all other CMS-backed sites combined. According to security researchers, the vast majority of WordPress-related—and other CMS-related—security problems arise through vulnerabilities in plugins. To date, WordPress features 36,547 different plugins available for download. It's a huge attack surface, and one which poses problems frequently.

You can install WordPress in just 2 minutes, but your job should not stop there. There are plenty of things to do after installing WordPress – it ranges from security, SEO, accessibility, performance, optimization etc...Let me share what I learned over the years with WordPress and the outcome is this eBook **“A definitive guide to WordPress Optimization, SEO, Security”**.

Note: I'll be using WordPress version 4.2.2 for all the below tips. However, it's advised to confirm compatibility with other WordPress versions. I also assume that the WordPress is running on a Linux operating system with Apache HTTP server.

The below tips are categorized into Security, SEO and Optimization.

WordPress Security

If you are using Content Management System (CMS) like WordPress. Make sure you always run the latest version. Also, if you are using plugins to enable any security mechanisms, WordPress customization or enhancing your website performance, use the Best and trusted Plugins. Here is the list of [Top security plugins for WordPress](#)¹.

Change WordPress table prefix (during install)

Every information on your site is going to be stored in a WordPress database thus making it hacker's favorite target. Hence it's very important to change the default table prefix (wp_) to something else, because the spammers and hackers out there knows what the default table prefix is (wp_) and that allows them to perform SQL injection kind of attacks. Unfortunately, many people would install WordPress with default settings which include default database table prefix. This would lead for hackers to plan a mass attack by targeting the default prefix wp_.

It's always advised to change the prefix while you install WordPress, thus making a smartest way to protect your database. WordPress database need not be called as 'wordpress' or 'wp'. It's a good practice to create database with unique name and avoid using 'wordpress' or 'wp' or your website name, thus protecting your site by mere hiding these details from the bots and the lazy hackers. This is security by obscurity.

Below you should enter your database connection details. If you're not sure about these, contact your host.

Database Name	<input type="text" value="wordpress"/>	The name of the database you want to run WP in.
User Name	<input type="text" value="username"/>	Your MySQL username
Password	<input type="text" value="password"/>	...and your MySQL password.
Database Host	<input type="text" value="localhost"/>	You should be able to get this info from your web host, if localhost does not work.
Table Prefix	<input type="text" value="site_"/>	If you want to run multiple WordPress installations in a single database, change this.

Note: If you are planning to change the table prefix after installing the WordPress, then here's the [guide](#)² which provides you few steps to follow without messing the things already in place.

Don't create username as 'admin'

The hackers will love the default usernames! Don't create a WordPress account with default username as 'admin' or any other common username. The hackers have a huge collection of common username and password, which is later used for Brute force method to attack wordpress blogs. Recently, a security firm reported that more than [ninety thousand wordpress sites have been attacked](#)³ using Brute force method. It's highly recommended to change your default usernames. You can also change the default username on existing WordPress installations by following few simple steps listed below:

Step 1: Login to your MySQL database via command-line or using PhpMyAdmin.

Step 2: Connect to your wordpress database.

Step 3: Run the below query to confirm that you are using the default username 'admin'.

```
mysql> select * from wp_users where user_login='admin';
```

If you were using default username 'admin', then the above query will display a table containing various information about the queried account.

Step 4: Now run the below query to modify the default 'admin' username to something else.

```
mysql> UPDATE wp_users SET user_login='myaccount' WHERE user_login='admin';
```

That's it!

Now any opened session for 'admin' user will be killed and wordpress will redirect you to login page. Try login with your new username and confirm whether everything works fine.

Note: Avoid using Author's original name as usernames.

Note: You might have strong passwords, but change it at least once in six months.

Install coming soon plugin while you do development at the backend

This point might surprise many, but it's really important. Whether you are launching a new website or if you are carrying out some developmental activities, or simply performing a bit of routine maintenance, then you may not want to leak certain information to the public. These are the times coming soon page can come in real handy. For example, if you look at the HTML source of the default theme, you will find certain unwanted Meta tags added by WordPress – leaking version information and an archive page with a default “Hello World” post displaying the username of the published account (mostly the first account that you created during the installation). So until you remove all those (just keep scrolling down to see how), it's better to [install a coming soon plugin](#)⁴ through which you can create a landing page or coming soon page in as little as 5 minutes without any programming or design skills.

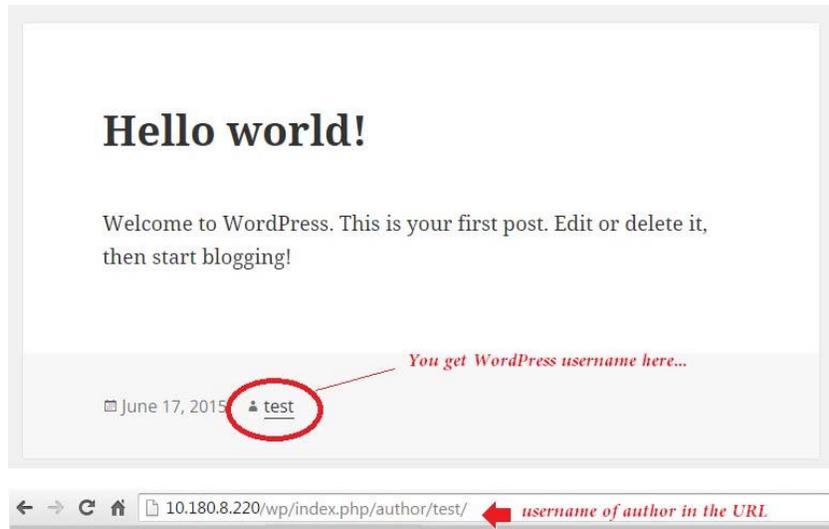
Even on the landing page and coming soon page, you can do following things like:

- Collect emails of your visitors with Aweber, Mailchimp, and other email list management software
- Create sales funnels
- Maintenance mode so you can notify your users while you service your site.
- Shortcode to place your custom email form anywhere on your landing or coming soon page
- Custom Analytics to track your visitors
- Create video Landing and coming soon page
- Many more....

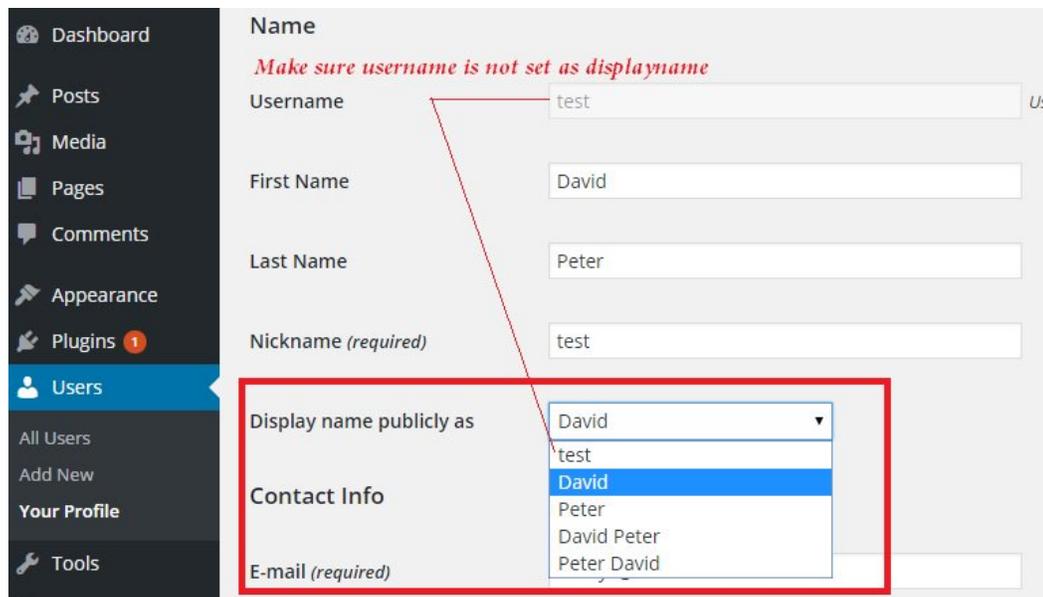
Tips: *You don't need a lot of crazy features to get a nice landing page or coming soon page. Keep your landing page or coming soon page with a narrowly focused message.*

Display name should not be username

By default username is set as display name and that's seen in author URL as well. For example, have a look at the below snapshot and you will see the circled word 'test', which is a WordPress account that posted the article. Using the username as display name you are giving a hint to the hackers about the user accounts on your site. The hackers now do a Brute force attack on your WordPress site.



To change that, click Users > All Users and click Edit to access the user profile to change the display name.



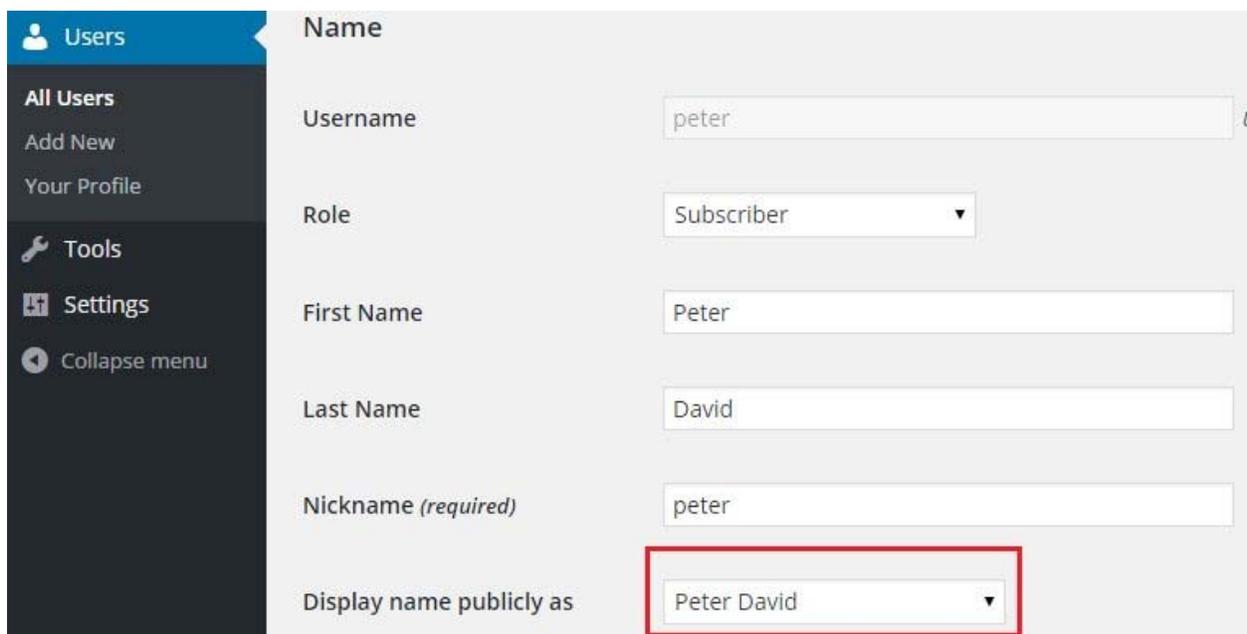
But you have to do this on every user profile and it's not going to stop if the site allows users to register – because the WP is going to set the username as Display name automatically.

To change the default display name for new registrations, copy and paste the below code in *functions.php*

```
function change_display_name( $user_id ) {  
    $info = get_userdata( $user_id );  
    $args = array(  
        'ID' => $user_id,  
        'display_name' => $info->first_name . ' ' . $info->last_name  
    );  
    wp_update_user( $args );  
}  
add_action('user_register','change_display_name');
```

Above code credits³

Once the above code is added in *functions.php*, try creating a new user and you should see 'Display name publicly as' set to 'First and Last name' of the user.



The screenshot shows the WordPress user profile edit form. The left sidebar contains navigation links: Users, All Users, Add New, Your Profile, Tools, Settings, and Collapse menu. The main form area has the following fields:

- Name: Username (peter)
- Role: Subscriber
- First Name: Peter
- Last Name: David
- Nickname (required): peter
- Display name publicly as: Peter David (highlighted with a red box)

Ok, but there is a catch here – **What if the user changes his Display name?**

You can simply **disable the 'Display name publicly as' field from Profile page** and prevent user from changing it. To do that, copy and paste the below code in functions.php

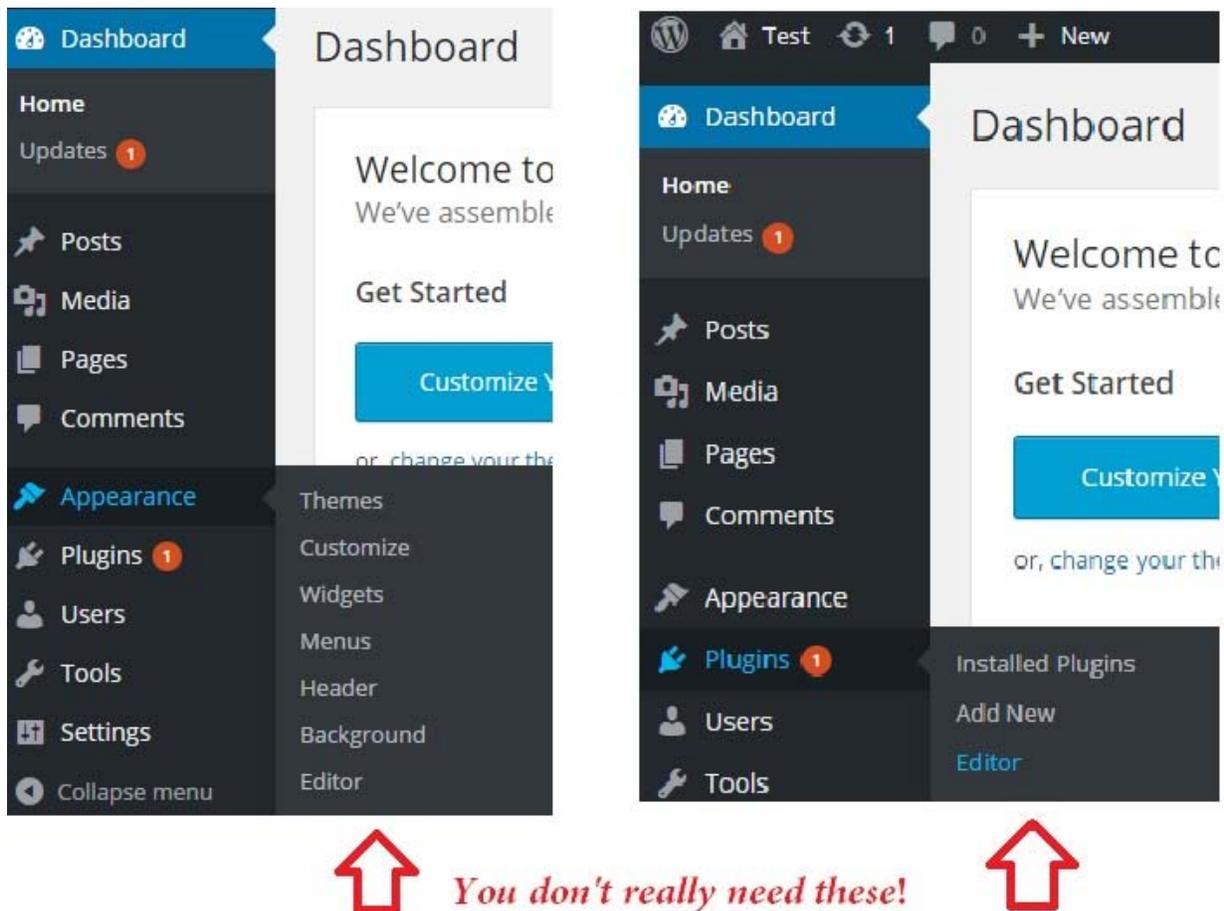
```
function disable_display_name() {
    global $pagenow;
    if ( $pagenow == 'profile.php' ) {
        ?>
        <script> jQuery( document ).ready(function() { jQuery('#display_name').prop('disabled', 'disabled');
            });
        </script>
        <?php }
    }
} add_action( 'admin_head', 'disable_display_name', 15 );
```

Note: Don't create too many wordpress user accounts, until and unless you really need them.

Note: Delete the unused wordpress accounts periodically.

Disable Plugin and Theme Editor in wp-admin

When you login to WP as admin, you'll find "Editor" link under Appearance and Plugin menus. The WordPress file editor can be a great tool because it allows you to edit PHP files associated with the theme and plugins on your site directly from the WordPress administration area. Mostly administrators utilize this tool to edit their theme's style.css file in order to make tweaks to their site. This can be a blessing in disguise. Basically, this is a potential backdoor to your server. The problem with the WordPress file editor is that it allows users to run PHP code on your site. Anytime a user is able to run their own code, this presents a security risk. If an insecure admin account is hacked, the WordPress file editor is the gateway through which a full-fledged attack can be carried out and also WordPress users can mess things up.



Is disabling the file editor the answer? Yes definitely, if you don't really need it.

Note: *If you still want to use it and it makes your life easier, take enough precautions with site security so that you are the only one who ever sees it.*

To remove that, copy and paste the below line in wp-config.php

```
define( 'DISALLOW_FILE_EDIT', true );
```

Disable XML-RPC

XML-RPC is enabled by default and it allows you to publish posts remotely (via WordPress app or Windows Live Writer), pingbacks, trackbacks and many other features. During early days, hackers used brute force attacks on WordPress login page. However, lately they are evolving and now leveraging the XMLRPC **wp.getUsersBlogs** method to guess as many passwords as they can. This attack is being made possible due to implementation of requiring username and password in XMLRPC calls. In these kind of attack, the hacker just inputs username and a password to the **wp.getUsersBlogs** call, which simply returns correct or not.

```
<methodCall><methodName>wp.getUsersBlogs</methodName><params><param><value>
<string>admin</string></value></param>
<param><value><string>112233</string></value></param></params>
</methodCall>
```

According to the reports from a security firm, hackers use xml-rpc to perform DDos attack on WordPress sites. It might be, and you could have no idea that your site is attacking other sites. If you are not going to do remote publishing and to stop your WordPress website from being misused, then disabling XML-RPC is a good idea.

Being a well known issue within WordPress and even the WordPress developers are aware of it, it can't be patched though as in many cases this terms out to be a feature.

You can **disable XML-RPC** via plugins such as [Disable XML-RPC Pingback](#)⁵.

(Or)

Copy and paste the below line to your *functions.php* for disabling XML-RPC completely.

```
add_filter('xmlrpc_enabled', '__return_false');
```

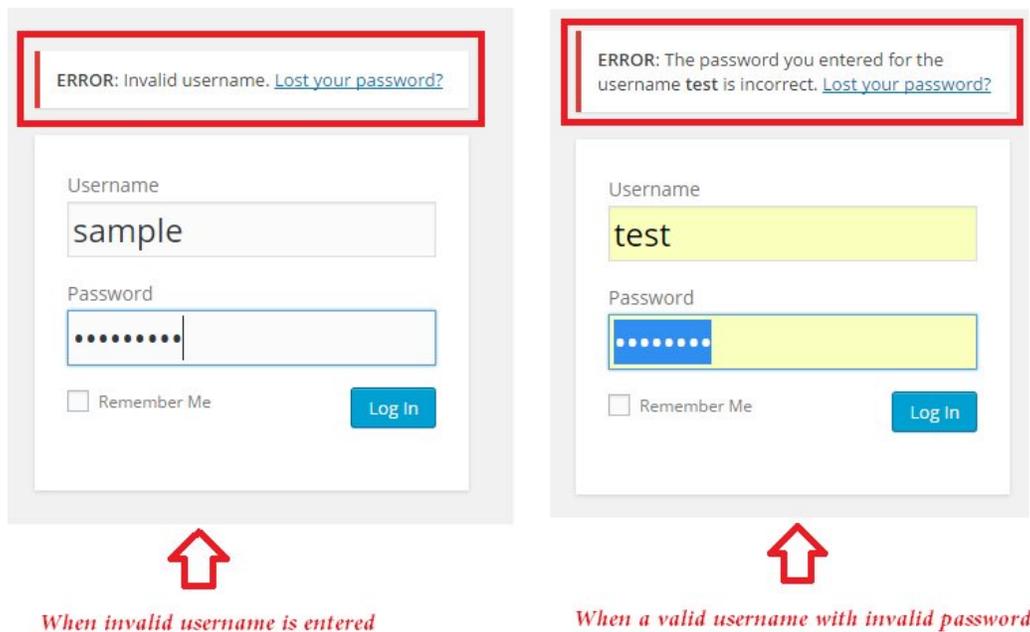
Note: *The same can be achieved by using popular security plugins.*

You can test your website with this [online tool](#)⁶ to verify your site is DDos'ing other websites.

Change WordPress Login error message

When an invalid username or password is entered, by default WordPress displays a detailed error message – stating whether the username is invalid or the password entered for the username was wrong. Yes, of-course these kind of notifications helps the users to correct the wrong username or wrong password. But, such a detailed message can provide a hint to the hacker to guess your login information. If they fail for the first time, they would keep trying until they succeed using brute force method.

Instead, the error message could be so generic in such a way that, even the hackers should find difficult in guessing the login credentials.



But you can easily change the login errors and display a custom message using the below code. Copy and paste the below code in *functions.php*

```
function no_wordpress_errors(){
    return 'You do not belong to this site!';
}
add_filter( 'login_errors', 'no_wordpress_errors' );
```

Tips: Here, I have set the error message as “You do not belong to this site!” You can modify accordingly.

Change WordPress admin URL

Protecting Admin page by changing its URL is a good practice. The security experts say so and it makes sense as well. If you are an administrator, then only you and your editors are going to login to your website. So, it makes sense that only you guys should know how to login to your site isn't? It means, anyone who's not supposed to access the admin page of your website, should not. Also, if a visitor on your site knows that you are using WordPress, then finding the default login URL is not a rocket science.

Technically, changing the WordPress admin/login URL will **secure your site from brute force attacks** and saves your server resources, which would otherwise be wasted by malicious hackers.

Changing the WordPress login URL will completely hide the backend?

Not really. It's one of the steps to try and hide the backend of your website.

How to change WordPress Login/Admin URL?

If you think to do it physically by moving/renaming the wp-admin folder, that is indeed problematic. The directory name "wp-admin" is hardcoded deeply into WordPress. This is by design, so WordPress actively prevents to allow a change here. Hence, suggest you to take a help from popular security plugins (**iThemes Security plugin** - formerly known as Better WP Security).

Warning: *It's always better to make this change on a fresh WordPress install. However, it should work on existing site as well.*

Step1: Download and install [iThemes Security Plugin for WordPress](#)⁷.

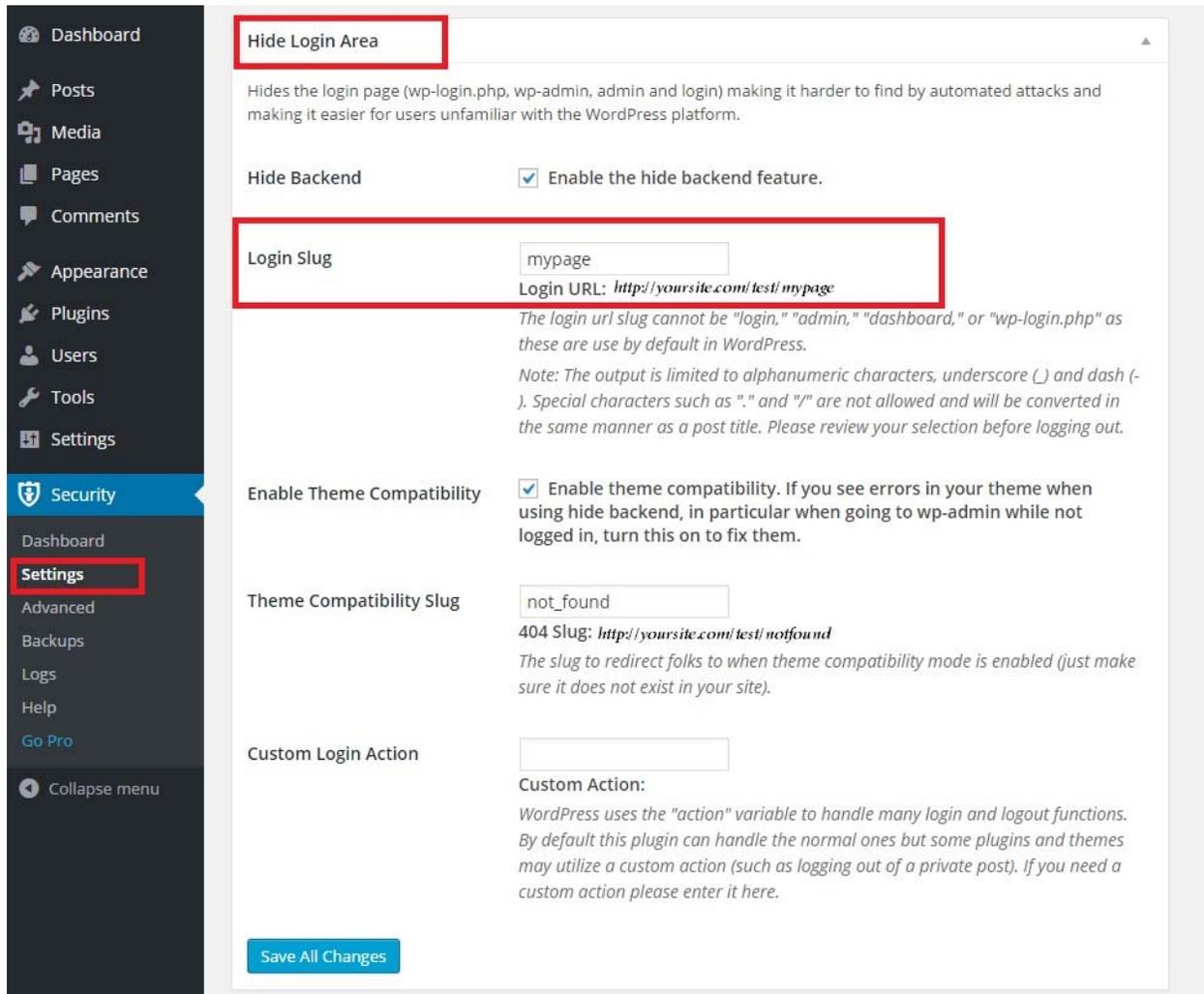
Step2: After activating the plugin, you may go through several features provided by the plugin. But we are discussing about changing wp-admin/wp-login.php URLs. To do that, click **Security > Settings** and scroll down to "**Hide Login Area**" section and check '**Enable the hide backend feature.**'

Step3: Enter URL of your choice under 'Login Slug'.

Note: *Never use words such as login, access, secure, dashboard, private, wplogin etc...Enter something secret and make sure only you and your editors know about it.*

Step4: You might also need to check '**Enable theme compatibility**' option, if you hit with an error.

Step5: Once done, hit “**Save All Changes**” button. That’s it!



That’s it!

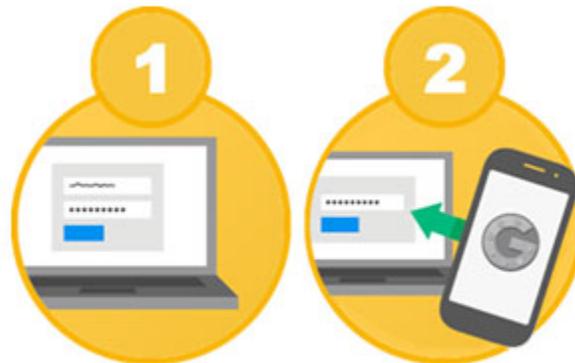
With this, you have reduced the possibility of your website been hacked!

Enable Two-Step Authentication for WordPress Admin

This is one of the most important steps in securing your website. Other than the public HTML pages and images of your WordPress website, the Admin dashboard area is protected with a username and password. It is thus accessible only to authorized users. However, to make your WordPress more secure, you can add an extra layer of security to the wp-admin folder so that even authorized users can't just get in with their WordPress password. You can easily enable Two-step (or factor) authentication using plugins available in the WordPress.org plugin repository. Here are some of the most popular plugins such as [Google-Authenticator](#)⁸, [Clef-two-factor authentication](#)⁹, [iThemes Security](#)⁷ to get you started.

Note: Use your Smartphone for strong authentication instead of passwords or tokens.

Google Authenticator plugin for WordPress gives you two-factor authentication using the Google Authenticator app for Android/iPhone/Blackberry. If you are security aware, you may already have the Google Authenticator app installed on your smartphone, using it for two-factor authentication on Gmail/Dropbox/Lastpass/Amazon etc. You can follow this [guide](#)¹⁰ for installing Google Authenticator plugin.



Two Factor Authentication

[Or]

Clef two-factor authentication provides easy-to-use strong two-factor authentication using Smartphone. It replaces insecure passwords and cumbersome one-time codes with the beautiful Clef Wave. It's very easy to integrate and simple to login through clef two-factor authentication.

Simply open the mobile app and sync with the Clef Wave. Below image shows, how the clef-authentication happens eliminating the passwords.



Setting up of Clef two-factor authentication on your WordPress site takes less than a minute. Before you start, you'll need the [Clef mobile app](#)¹¹ installed on your iOS or Android device. Follow this detailed setup instructions for the [installation and configuration](#)¹² of clef plugin on your WordPress website.

[0r]

Follow this [guide](#)¹³ to enable two-factor authentication securing your wp-admin directory using passwords.

Install User lockout Plugin

Every WordPress administrator should take necessary care in managing their user accounts. Most of the WordPress accounts are hacked using Brute Force method and dictionary attacks, where the attacker can try as many invalid passwords before finding the correct one. Locking down or disabling WordPress account after a set number of invalid login attempts are one of the best methods to stay away from Brute force attacks. Follow the below steps to Lock down or disable the WordPress user account after the set number of Invalid login attempts.

Step1: Download the plugin called [User Locker](#)¹⁴.

Step2: Extract the download zip file and copy it to `wp-content/plugins` folder.

Step3: Enable the plugin from the wp-admin plugin page.

Step4: If you want to disable or Lock any user, just head on to **Users > All users > Edit** the user you wish to disable or lock.

Step5: Scroll down to the bottom of the page and select **User account is locked for security reasons** check box and provide Lock reason (if you wish to display it on the Login page).

Step6: If you want to disable any user account, then select **User account is disabled** check box and provide the disable reason (if you wish to display it on the Login page).

User Locking

User account locked

User account is locked for security reasons

Lock reason

User Locked!! - Admin

Note: start text with '@' (AT sign) to keep it private.

User account disabled

User account is disabled

Disable reason

User account disabled!! - Admin

Note: start text with '@' (AT sign) to keep it private.

Update User

Step7: Here comes another important setting. Jump to **Settings > User Locker** and set **Maximum invalid login attempts before account locking** as shown in the below image. Checkout the other settings as well.

Account locking:

Maximum invalid login attempts
before account locking:

Default lock reason:
Note: start text with '@' (AT sign) to keep it private.

Show account lock/disable reason
after login attempt:

Clean lock/disable reason when
user is unlocked/enabled:

Show single status column:
Add one column instead of two to the User List, and show lock/disable reason in tooltip only.

Hereafter the user account will be locked automatically after the set number of invalid login attempts. To enable the account back, goto user profile page on the wp-admin and uncheck ***User account is locked for security reasons.***

That's it!

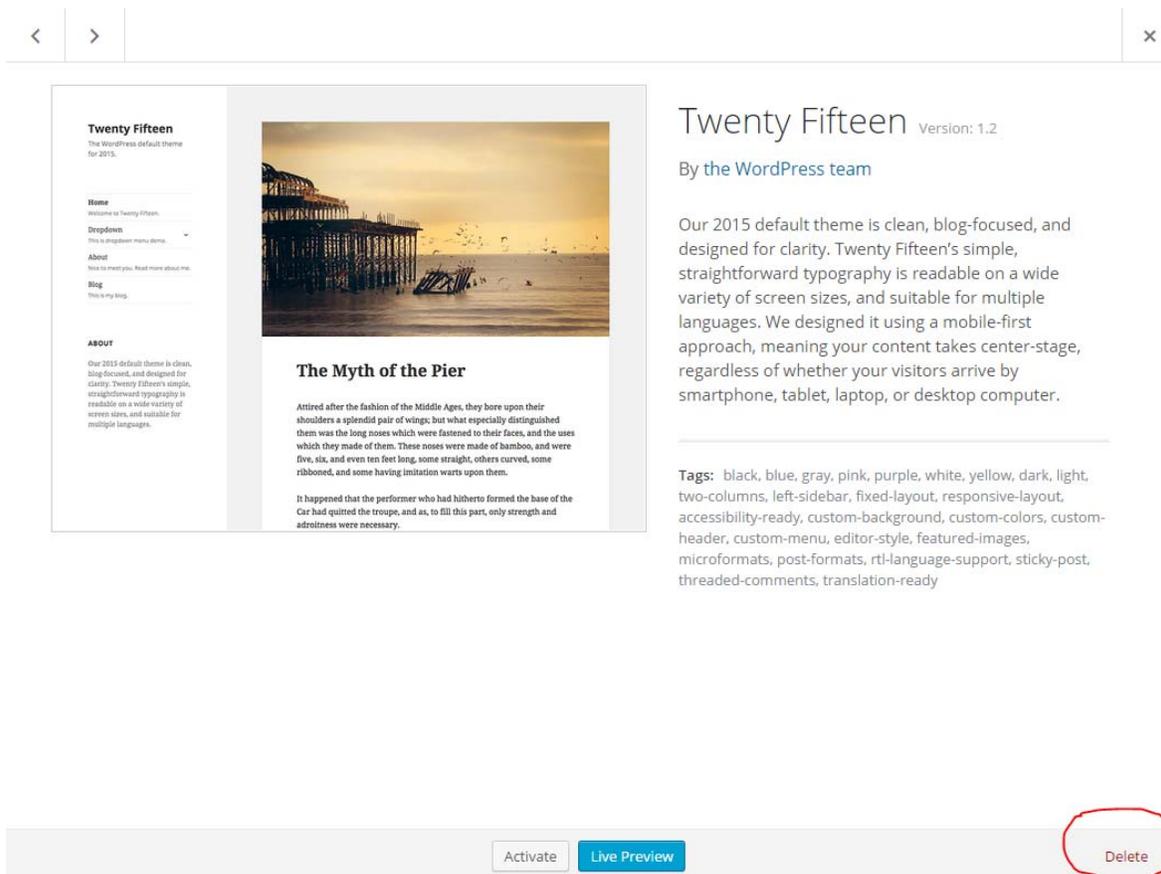
Delete unwanted themes and plugins

No code is 100% safe and that applies to themes and plugins inside the WordPress directory. If you are not using a particular plugin or a theme, then go ahead and delete those. **Less unwanted codes, more secure your site is.** Below steps will guide you to **remove unwanted themes** through the WordPress dashboard.

Step1: Open your WordPress dashboard and go to **Appearance > Themes**.

Step2: You can't remove an active theme. Hence you need to de-activate the theme to be deleted, by simply activating any other theme of your choice.

Step3: Click on the theme details you need to delete. On the right bottom of the corner, you find the **Delete** button. Click the Delete button to successfully remove the theme.



If you couldn't delete through the dashboard, you can do it through FTP manager as follows:

Step1: Using the FTP manager log into your server and navigate to wp-content/themes folder.

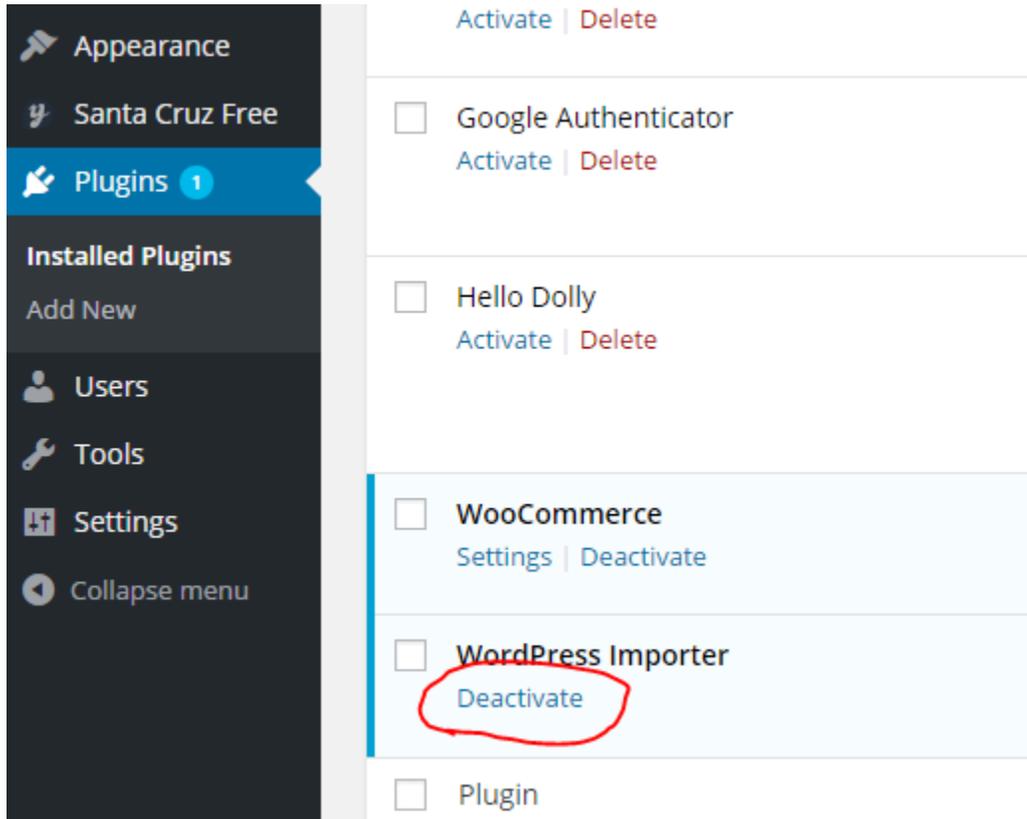
Step2: Delete the corresponding theme directory and its files.

Now your theme is successfully deleted through FTP manager.

Below steps will guide you to **remove unwanted plugins** through the WordPress dashboard.

Step1: Open your WordPress dashboard and go to **Appearance > Plugins**.

Step2: De-activate the plugin.



Step3: Using the FTP manager log into your server and navigate to wp-content/plugins folder.

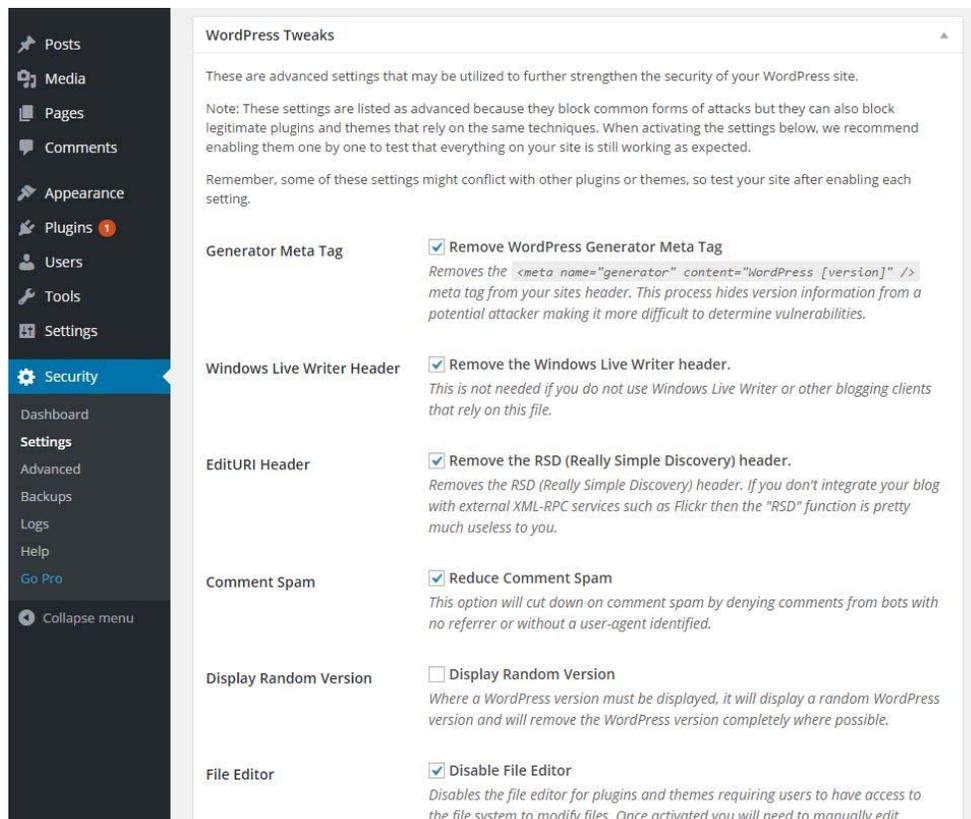
Step4: Delete the corresponding plugin directory and its files.

Note: You should also ensure that theme and plugins are not leaking any information via comments (sometimes they get published in HTML source)

Install iThemes Security Plugin

You have already seen that iThemes Security plugin offers two-factor authentication and change WordPress admin URL. [iThemes Security plugin](#)⁷ (also known as Better WP Security) is one of best plugin out there – that helps in protecting WordPress against various attacks. Additionally, iThemes Security plugin gives you over 30+ ways to secure and protect your WordPress site such as track user actions, enable two factor authentication, automatic scanning to identify malwares, set password expiration, force strong password rules, IP address banning, tweaking WordPress settings, disable file editing, changing the location of default upload directory, disable xmlrpc, changing default username etc.

I suggest the WordPress administrators to install and **Lookout for System tweaks** section under **Security > Settings** – where you can prevent certain files from being accessible by public, disable directory browsing, removing suspicious query strings, disabling PHP script execution under uploads folder etc... Also **Lookout for WordPress tweaks** section under **Security > Settings** – which allows you to remove automatically generated meta tags, disabling XML-RPC, disable file editor, disable login error message, force unique nickname for user etc...



Change those default Security Keys

WordPress security keys were introduced in WordPress version 2.6. The security keys are randomly generated variables that encrypt the information stored in user's cookie. For instance, the passwords like "wordpress" or "test123" are simple and can be easily broken. Random generated, long passwords such as "6ht78sh13hfrtngghl219465hg" are really tough to crack. Along with these security keys, 'salt' has been used to improve the security of your WordPress. WordPress Security Keys will make your website tough to hack. Here's an example of security keys:

```
define('AUTH_KEY', 't`DK%X:>xy|e-Z(BXb/f(Ur`8#~UzUQG-^_Cs_GHs5U-&Wb?pgn^p8(2@}IcnCa!');
define('SECURE_AUTH_KEY',
'D&ovlU#|CvJ##uNq}bel+^MfTt&.b9{UvR}g%ixsXhGIR}7q!h}XWdEC[BOKXssj');
define('LOGGED_IN_KEY', 'MGKi8Br(&{H*~&0s;{k0<S(O:+f#WM+q|npj)-+P;RDKT:~jrmgj#/-,[hOBk!ry^');
define('NONCE_KEY', 'FIsAsXJKL5ZlQo)iD-pt??eUbdc_{Cn<4!d~yqz})&B D?AwK%)F2aNwI|siOe');
define('AUTH_SALT', '7T-!^i!0,w)L#}JK@pc2{8XE[DenYI^BVf{L:jvF,hf}zBf883td6D;Vcy8,S)-&G');
define('SECURE_AUTH_SALT', 'I6`V|mDZq21-J|ihb u^q0F }F_NUcy`l,=obGtq*p#Ybe4a31R,r=|n#=#]@]c #');
define('LOGGED_IN_SALT',
'w<$4c$Hmd%/*`]Oom>(hdXW|0M=X={we6;Mpvgt+V.o<$|#_}qG(GaVDEsn,~*4i');
define('NONCE_SALT', 'a|#h{c5|P &xWs4IZ20c2&%4!c(/uG}W:mAvy<I44`jAbup]t=]V<`}.py(wTP%%');
```

Note: You should not use the above example. Generate your own unique keys.

How to generate unique security keys and add them to WordPress?

Step 1: Head on to [WordPress security keys generator](#)¹⁵.

Step 2: The WordPress will generate unique random security keys for you.

Step 3: Copy everything and paste it in `wp-config.php` file under 'Authentication Unique Keys' section.

That's it.

If you have logged-in to WordPress admin panel, then you'll be asked to login again.

How to remember this random security keys?

You need not remember. The security keys can be re-generated anytime using the above said procedure and can be added to `wp-config.php` anytime at the cost of logout. It means, you will have to login again to the WordPress.

Rename Theme folder

There are times you may encounter such as you don't want your competitor to know the theme you are using or you do not want to expose the vulnerabilities to hackers that could have come along with the theme. Either case, you don't want to publicize the name of a theme that you are using. Theoretically, if everything is coded to standards, you can rename the folder and nothing bad will happen.

Don't fret! This is easy to do.

Step 1: Backup your entire site and database.

Step 2: De-Activate your theme from the WordPress dashboard and activate a different temporary theme ("Twenty Twelve").

Step 3: Login to your server through FTP manager and navigate to wp-content/themes folder.

Step 4: Rename the theme folder.

Step 5: Refresh the WordPress dashboard and re-activate the theme.

Note: *Changing active theme file names or the active theme folder name can lead to some very odd issues, like the "white screen of death".*

WordPress SEO Optimization

WordPress is one of the best CMS when it comes to SEO. But in reality, the WordPress software or the theme you use does need a little tweak in optimizing your site to the best practices will help you improve your rankings, gain more subscribers and have a better website in general. There are a number of third-party plugins and hacks including the built-in search optimization tools like ability to use *.htaccess* to create permalinks for optimizing your site for SEO. Here are few tips to maintain your site's optimal friendliness towards search engine spiders and crawlers:

Disable dashboard on Multi-author site

When you allow a user to register on your site, with successful completion of the registration, the user would be granted an account with access to the WordPress Admin area known as dashboard upon login. If you are running a multi-author WordPress site, then there are times you may not want your authors to view the various widgets displayed on the Dashboard. Only users you trust should have access to the admin area of your WordPress site. Even you can still control while giving access to the trusted users too to your admin area.

There are plenty of plugins out there in the WordPress plugin repository that help you to remove dashboard for specific user roles. Here is one of the most popular plugin [Remove Dashboard Access](#)¹⁶ to get you started.

The best way to find the access rights given to the user accounts is to login as that user and test. This lets you see what a user group/user can do and what they see.

Once you install and activate the plugin, go to the Dashboard Access Controls settings in the **Settings > Dashboard Access** to configure the plugin.

Dashboard Access Settings

Dashboard Access Controls

Dashboard access can be restricted to users of certain roles only or users with a specific capability.

Dashboard User Access:

- Administrators only
- Editors and Administrators
- Authors, Editors, and Administrators
- Advanced: Limit by capability: ▼

You can find out more about specific [Roles & Capabilities](#) in the Codex.

Redirect URL: Redirect disallowed users to:

User Profile Access: Allow all users to edit their profiles in the dashboard.

Login Message

Through **Remove Dashboard Access plugin** you can provide access to dashboard for selective user roles. Alternatively, you can also limit access by capability. Capabilities are the actions a user can do on your WordPress site.

It also provides an option of redirect URL. This option allows you to redirect disallowed users to any page on your website.

That's all. Now only users with your selected user role or capability can access the WordPress dashboard.

***Note:** Have too many bloggers writing for you? Advise them to set strong passwords for their accounts and educate them about wordpress security.*

Remove unwanted Meta tags

The default WordPress installation would generate Meta tags automatically which can leak certain information about your website – like WordPress version information. If you were wondering how to remove unnecessary Meta tags added to HTML source code of your WordPress site, then here's how you can do that. By default WordPress adds few meta tags and link tags to HTML source code (such as WordPress version, manifest and rsd), that are not used mostly. Moreover, WordPress Meta tag can let others know the version of CMS that you are currently running and you know hackers love such information.

Here's a sample Meta and link tag added by WordPress,

```
<meta name="generator" content="WordPress 2.3.3" />
<link rel="EditURI" type="application/rsd+xml" title="RSD" href="http://example.com/xmlrpc.php?rsd" />
<link rel="wlwmanifest" type="application/wlwmanifest+xml" href="http://example.com/wp-
includes/wlwmanifest.xml" />
```

To remove Meta tag and link tag from the HTML source of your WordPress site, add the below lines to your theme functions.php file.

```
remove_action( 'wp_head', 'wp_generator' );
remove_action( 'wp_head', 'wlwmanifest_link' );
remove_action( 'wp_head', 'rsd_link' );
```

Update: *If the above code didn't work for you, then you may try the alternate solution – Add the below code to functions.php in your theme directory*

```
function remove_generator_filter() { return ""; }

if (function_exists('add_filter')) {
    $types = array('html', 'xhtml', 'atom', 'rss2', /*'rdf',*/ 'comment', 'export');

    foreach ($types as $type)
        add_filter('get_the_generator_'.$type, 'remove_generator_filter');
```

Remove Query strings from URLs

Just like the way we removed Meta tags, you should also remove query strings from various URLs seen in HTML source code. For example, by default the CSS, JavaScript URLs added by WordPress and its plugins will have query strings such as 'ver' with the version number as the value. These query strings does not do any good to SEO, because URLs that contain query strings are not considered to be static resources and will not be cached by most of the CDN and proxy servers. Removing query strings from these resources will allow proxy caching and it can boost your website's performance. Let's do a quick check of your WordPress source code.

Right click on your WordPress article page and click **view source**. Here's an example of one such javascript and stylesheet.

```
http://websitename.com/somejavascript.js?ver=3.4.2  
http://websitename.com/somestyle.css?ver=3.4.2
```

Note: *Removing the query strings from your resource URLs will not affect their functionality.*

OK! If you are using WordPress, then follow this quick tip to remove query strings from resource URLs:

Open functions.php file located in your theme directory and paste the below code:

```
function _remove_script_version( $src ){  
    $parts = explode( '?ver', $src );  
    return $parts[0];  
}  
add_filter( 'script_loader_src', '_remove_script_version', 15, 1 );  
add_filter( 'style_loader_src', '_remove_script_version', 15, 1 );
```

That's it!

Reload your WordPress page and see the query string vanishing from the URL's.

Remove or hide Secondary feed URLs

By default WordPress publishes multiple RSS feeds such as comments feeds, archive feeds, post feeds, category feeds and the links to these feeds are also published in HTML source. Sometimes, you might not want to publicize all of those feeds in your HTML pages; as you might want your readers to keep track of main feed of your site or you wish to redirect all the secondary feeds to main feed. Whatever, if you ever wished to know how to hide secondary feeds URL's from your WordPress HTML pages, then here's how you can do that:

Just add the below lines to functions.php file located inside your theme directory:

```
remove_action( 'wp_head', 'feed_links', 2 );  
remove_action( 'wp_head', 'feed_links_extra', 3 );
```

Remove extra JS file added to support Emoji

The Emojis are cool, loved and used by many WordPress authors. But there are few, like me, who don't use Emojis. Starting from version 4.2, WordPress supports Emoji's by default. It means, an extra JavaScript file gets added to your site's header to support 4 bit Unicode characters. Interestingly, the feature will support Japanese, Chinese and Korean character sets in WordPress. That's a cool one and our kudos to WordPress team. But you know something; I just updated WordPress to version 4.2, but didn't opt to use Emoji. Also, WordPress does not have an option to disable Emojis. If you are someone looking to remove or disable Emoji support on WordPress 4.2, then here's how you can do that:

Note: You are good to use age old smileys (😊) that should still be supported by modern browsers. But you will not have privilege to use advanced Emojis.



This is what you will see in HTML source (included by default, after upgrading WordPress to version 4.2)

```
<script type="text/javascript">
window._wpemojiSettings =
{"baseUrl":"http://s.w.org/images/core/emoji/72x72/", "ext": ".png", "source":{"concatemoji":"http://www.website.com/wp-includes/js/wp-emoji-release.min.js"}};
```

```
!function(a,b,c){function d(a){var
c=b.createElement("canvas"),d=c.getContext&&c.getContext("2d");return
d&&d.fillText?(d.textBaseline="top",d.font="600 32px
Arial","flag"===a?(d.fillText(String.fromCharCode(55356,56812,55356,56807),0,0),c.toDat
aURL().length>3e3):(d.fillText(String.fromCharCode(55357,56835),0,0),0!==(d.getImageDa
ta(16,16,1,1).data[0])):!1}function e(a){var
c=b.createElement("script");c.src=a,c.type="text/javascript",b.getElementsByTagName("he
ad")[0].appendChild(c)}var
f;c.supports={simple:d("simple"),flag:d("flag")},c.supports.simple&&c.supports.flag||(f=c.so
urce||{}f.concatemoji?e(f.concatemoji):f.wpemoji&&f.twemoji&&(e(f.twemoji),e(f.wpemoji
)))}(window,document>window._wpemojiSettings);
</script>
<style type="text/css">
img.wp-smiley,
img.emoji {
    display: inline !important;
    border: none !important;
    box-shadow: none !important;
    height: 1em !important;
    width: 1em !important;
    margin: 0 .07em !important;
    vertical-align: -0.1em !important;
    background: none !important;
    padding: 0 !important;
}
</style>
```

And here's how you can remove it.

Copy the below code and add it to your theme's functions.php file.

```
remove_action( 'wp_head', 'print_emoji_detection_script', 7 );
remove_action( 'wp_print_styles', 'print_emoji_styles' );
```

To disable Emoji on WordPress Admin pages

```
remove_action( 'admin_print_scripts', 'print_emoji_detection_script' );  
remove_action( 'admin_print_styles', 'print_emoji_styles' );
```

That's it!

Prevent search engine from Indexing XML Sitemap

We want search engines to use XML sitemap for better crawling of our website, but we don't want them to Index and show it in the search results (they are supposed to show the actual article page isn't?). The below code will instruct search engines to **not index the sitemap**.

Copy and paste the below code to your *.htaccess* file.

```
<IfModule mod_rewrite.c>
<Files sitemap.xml>
  Header set X-Robots-Tag "noindex"
</Files>
</IfModule>
```

That's it!

Install WP-Optimize to optimize database

WP-Optimize¹⁷ is a simple plugin that allows you to keep WP database clean and optimizes it to reduce size.

Consider the scenario's below:

- Every time a new post or page is created, WordPress creates a revision of the post or page. If you edit the post 6 times, you would end-up with 5 copy of that post as revisions. Imagine if your post is of approximately 100KB, then the total space wasted would be 500KB (Including revisions). Similarly, if you have 100 posts, then you would have 500MB database space wasted! This is a huge number of bytes that's on your MySQL overhead.

WP-Optimize allow you to optimize and shrink your posts table by removing not required post revisions from the database.

- Every day thousands of spam and un-approved comments gets collected in your comments table. With the single click of a button, WP-Optimize can clean and remove those.
- WP-Optimize reports which database tables have overhead and wasted spaces. It also allows you to shrink and get rid of those wasted spaces.
- MySQL is a great database, but it doesn't clean itself up the way it should sometimes. Automatically WP-Optimize cleans database every week and respects the "Keeps selected number of week's data" option.

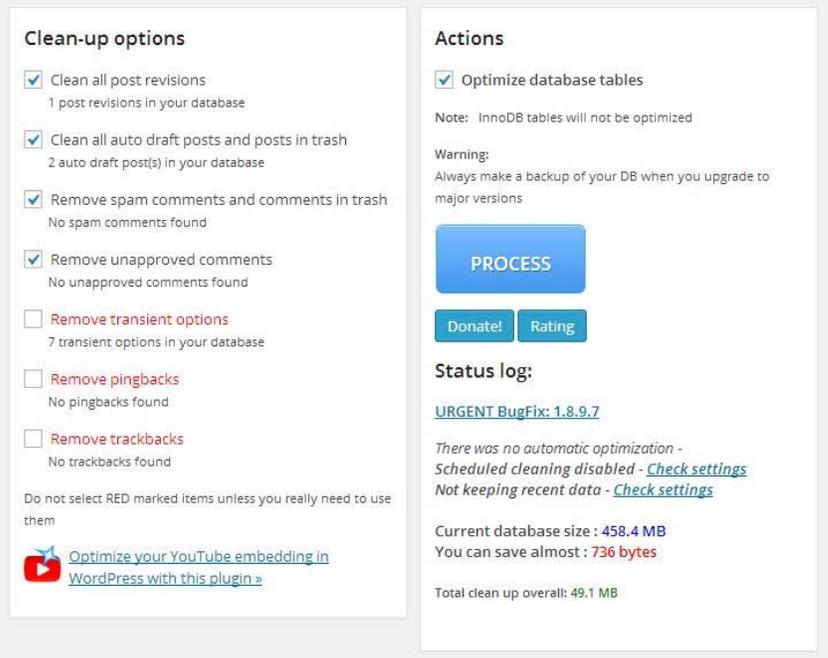
It allows you to do following:

1. Remove stale post revisions,
2. Remove spam & trashed comments,
3. Remove trashed posts,
4. Remove auto drafts,
5. Remove transient options,
6. Remove trackbacks & pingbacks,
7. Schedule clean up
8. Optimizes MySQL tables
9. Displays database statistics

Many more....

How to Install Wp-Optimize Plugin?

1. [Download](#)¹⁷, Install and Activate the plugin
2. Checkout the various options in the plugin dashboard and choose carefully.
3. Take a look at the Settings, where you can customize the auto cleanup and optimization.



The screenshot displays the Wp-Optimize plugin dashboard, divided into two main sections: 'Clean-up options' and 'Actions'.

Clean-up options:

- Clean all post revisions
1 post revisions in your database
- Clean all auto draft posts and posts in trash
2 auto draft post(s) in your database
- Remove spam comments and comments in trash
No spam comments found
- Remove unapproved comments
No unapproved comments found
- Remove transient options
7 transient options in your database
- Remove pingbacks
No pingbacks found
- Remove trackbacks
No trackbacks found

Do not select RED marked items unless you really need to use them

[Optimize your YouTube embedding in WordPress with this plugin »](#)

Actions:

- Optimize database tables

Note: InnoDB tables will not be optimized

Warning:
Always make a backup of your DB when you upgrade to major versions

[PROCESS](#)

[Donate!](#) [Rating](#)

Status log:

[URGENT BugFix: 1.8.9.7](#)

There was no automatic optimization -
Scheduled cleaning disabled - [Check settings](#)
Not keeping recent data - [Check settings](#)

Current database size : 458.4 MB
You can save almost : 736 bytes

Total clean up overall: 49.1 MB

Note: If you are using this plugin for first time or upgrading to major version, make a backup of your database.

Install W3 Total Cache to speed your site

[W3 Total cache](#)¹⁸ is one of the popular caching plugin for WordPress. W3 Total cache plugin is the fastest and most complete WordPress performance optimization plugin. Trusted by many popular blogs like: techglimpse.com, mashable.com, yoast.com, css3.info and others — W3 Total Cache improves the user experience of your site by improving your server performance, caching every aspect of your site, reducing the download times and providing transparent content delivery network (CDN) integration. Below are the benefits of W3 Total Cache plugin:

- **Improved page load times**

Reduced page load times can increase visitor time on site and number of pages viewed, in addition to improved conversion rates.
- **Optimization score improvements**

Dramatic improvements in YSlow and Google Page speed scores possible when W3 Total Cache is fully configured.
- **Improved user experience**

Page cache and Browser cache configuration gives visitors perception of “instant” repeat page views.
- **Significant bandwidth savings**

Up to 80% Bandwidth savings via Minify and HTTP compression of HTML, CSS, JavaScript and RSS feeds.

Equally important, the plugin requires no theme modifications, modifications to your .htaccess (mod_rewrite rules) or programming compromises to get started. Most importantly, it's the only plugin designed to optimize all practical hosting environments small or large.

When the plugin is configured properly, you'll really love the speed at which your site loads!

Prioritize Critical Content above the fold

The reason is simple! The visitor on your webpage always wants to view the important content to load first (not the advertisement, mega menus or sidebar) and so does the search engines.

Here's a simple scenario – Let us assume that a user lands on your web page via Google and immediately wants to read the content that he searched for. Will he ever click on the menus at the top or links on the sidebar or scroll to the footer of your website before reading the main content? Well, he might do all those after spending few seconds on the main content isn't? It means, the user's always looking for the main content to load first and the rest after that and that's what will impress Google as well. It's very important to prioritize the main content and reduce the size of the content above the fold.

Generally in all the websites, the header loads first – which includes the site logo, navigation menus, share buttons bar and then the main content. The problem here's, the user has to wait few seconds before the main content is loaded (the site logo, navigation menu, share buttons bar etc...are loaded first and then the main content). After spending some time with analytics, I understood that no visitor had ever clicked the navigation menu as soon as he landed on the page and the case was same with share buttons as well (the visitor has to read the content before he/she can share it on social media isn't?).

Finally after a deep thought, why should we load all those less important contents first when the user is interested only on the main content? So, the ideal way is to load the main content as quickly as possible with minimum header, and then load the rest while the user spends time on the main content.

How the browser does actually load the webpage?

- The browser downloads the HTML
- Parses the HTML
- When it finds some resource while parsing (may be image or stylesheets or JavaScript files), it stops parsing the HTML and then loads the resource.
- If the resource is stylesheet or JavaScript file, then it has to parse those first.
- Once done with the resource, it comes back and parses the HTML

- During the above process, the browser tries to display as much of the content as quickly as possible. So it's important to load main content as quickly as possible and make it visible to the user.

If you look at the below page, I'll be loading main content (post content) first and then the sidebar, navigation menu, share buttons etc...

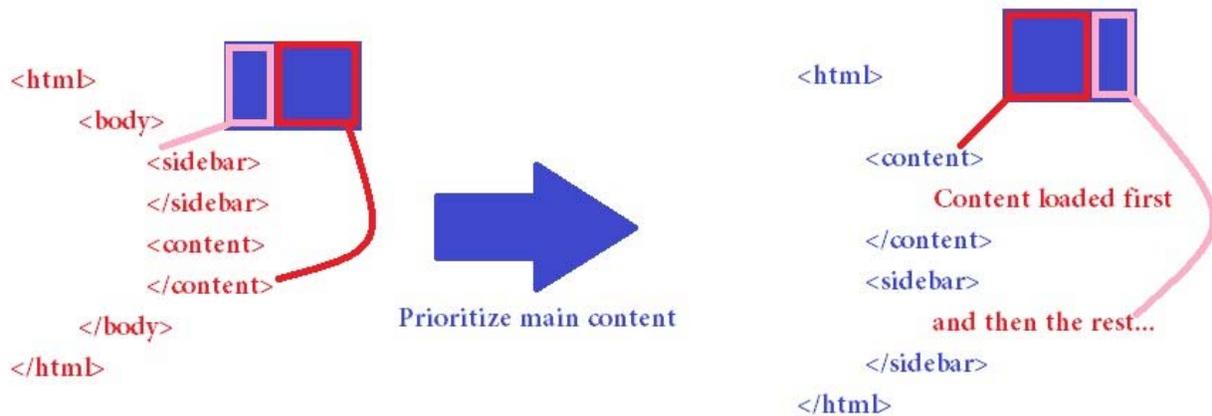
The screenshot shows a WordPress blog post on the website 'TECHGLIMPSE'. The page has a dark header with navigation links: 'Gadgets', 'How To +', 'WordPress', 'Demos', 'Tweethshort', and 'Advertise'. The main title is 'Why Prioritizing the Main Content above the Fold is important and How you can do that?'. Below the title, there's a date 'Friday, May 22, 2015 By David Peter' and a 'Browse more posts' link. The main content area features an advertisement for 'PRESCHOOL ACTIVITIES Interactive Books For Your Kids' with a 'Free Presentation and Free Parenting Booklet!' offer. The article text begins with 'The reason is simple! The visitor on your webpage wants the important content to load first and visible immediately...'. To the right, there's a search bar, a profile for 'David Peter', and a 'Latest Popular Stories' section with five items: 1. Slide in with Google Slides, 2. How to Add WooCommerce Product Categories to WordPress Menu?, 3. How to Upgrade Sendmail to the latest version on CentOS, 4. How to List all Modules Loaded or Enabled in Apache httpd?, 5. Troubleshooting Request Tracker version 3 Error.

the main content and reduce the size of the content above the fold.

In my website, I had the header loaded first – which includes the site logo, navigation menus and share buttons bar (which is usually hidden and visible when the user scrolls 100px downwards) and then the main content. The problem here's, the user has to wait few seconds before the main content is loaded (the site logo, navigation menu, share buttons bar etc...are loaded first and then the main content). After spending some time with analytics, I understood that no visitor had ever clicked the navigation menu as soon as he landed on the page and the case was same with share buttons as well (the visitor has to read the content before he/she can share it on social media isn't?). Finally, this question took toll on me...Why should I load all those less important contents first when the user is interested only on the main content? So, the ideal way is to load the main content as quickly as possible with minimum header, and then load the rest while the user spends time on the main content.

How did I do that?

All I did was to change the layout of my web page and maintained the same design. I moved the navigation menus and share buttons bar to the bottom of the page and used CSS to position:fixed. When the position was fixed, I can load the header anywhere on the page and the best place is to load it after the main content. Below image shows it.



Note: Wait! This strategy may not work on your website, because you may not want to set `position:fixed` for the navigation bar. It means, you might have to follow other strategies (by coding empty header at the top and then loading the contents via JavaScript at the footer).

Reduce the size of the content above the fold

You should also push the expensive resources such as CSS and JavaScript files to the footer. But in case of moving CSS file, you should make sure that the page does not look clutter before the styles are loaded.

Credits: [Feedthebot](#)¹⁹

Install Best SEO Plugin

SEO is not an easy task and keeping it consistently complaint is much more difficult. So take a help from popular SEO plugins such as [WordPress SEO by Yoast](#)²⁰ – with more than 1+ million downloads, it's easily the most popular SEO plugin you'll ever need. Using Yoast's WordPress SEO plugin improve your WordPress SEO: Write better content and have a fully optimized WordPress site.

With Yoast's WordPress SEO plugin:

- Easily optimize your WordPress site
- Write better content
- Content analysis functionality
- Technical WordPress Search Engine Optimization
- RSS Optimization
- Edit your .htaccess and robots.txt file
- Social Integration
- Multi-Site Compatible
- Import & Export functionality

Many more features to streamline your site at the click of a button

Analyze and Speed up your website

The speed of a page is made up of both Front-end and Server-side components. [GTmetrix](#)²¹ assesses the front-end structure of your page to ensure that it is delivered as optimally as possible to your visitors, but an optimized server-side is also an important part of the equation in offering a fast and seamless site experience.

What makes up my Page Load Time?

When a request for a page is made, the Front-end and Server-side components both take a certain amount of time to complete their operations. Since their operations are essentially sequential, their cumulative time can be considered the **total page load time**.

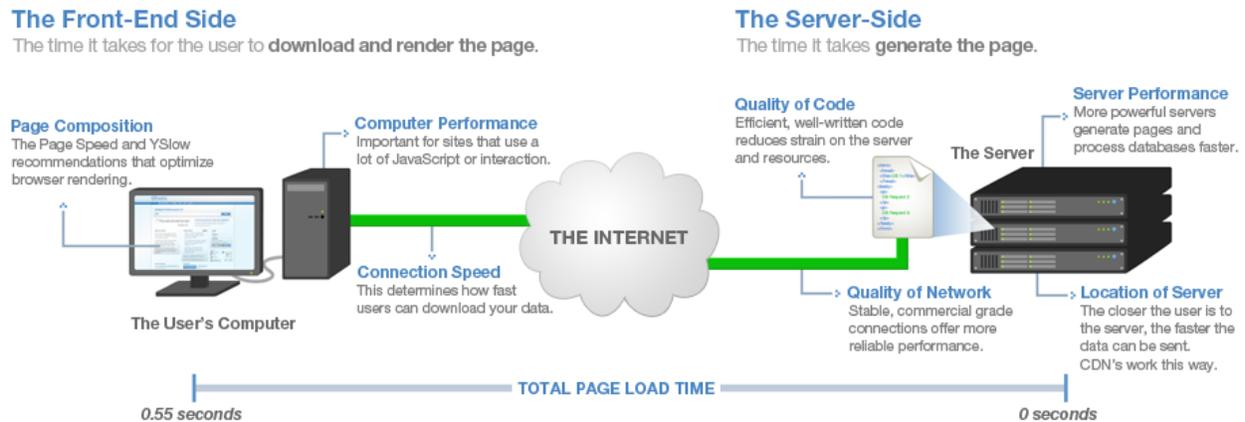


Image Credits: Gtmetrix.com

You may use [gtmetrix.com](#)²¹ to analyze your web page and optimize your Front-end. It also helps you develop faster and efficient website.

Your users will love you for it.

Note: Even after you've optimized your Front-end, speed gains can still be achieved by optimizing the Server-side. This means **optimizing the way the page is generated by your server**.

Cache Static contents by using ExpiresByType

Fetching resources like CSS, JavaScript and images over the network is both slow and expensive as the downloading of it may require multiple roundtrips between the client and the server. Due to these network roundtrips, delay in processing and block in rendering of page content occurs. If your visitors visit your website very often, it would be better to keep the static content locally on the visitor's browser for a certain period of time and re-use it, rather than fetching it every time the visitor visits the site. Static content is usually the largest and slowest to download. Enabling caching can be very useful especially to forums, chats or other sites where same users revisit them often. This can be achieved using *Apache mod_expires* which can significantly speed up your site and decrease the server load. It can be set using *ExpiresByType* attribute in *httpd.conf* or enabled in a local *.htaccess* file where you can set a time frame to cache those static resources of your website.

Here is an example:

```
<IfModule mod_expires.c>
ExpiresActive On
ExpiresByType text/javascript "access plus 1 month"
ExpiresByType application/x-javascript "access plus 1 month"
ExpiresByType image/jpg "access plus 1 year"
ExpiresByType image/jpeg "access plus 1 year"
ExpiresByType image/gif "access plus 1 year"
ExpiresByType image/png "access plus 1 year"
ExpiresByType image/x-icon "access plus 1 year"
ExpiresDefault "access plus 2 days"
ExpiresByType text/css "access plus 1 month"
<filesMatch "\.(js|js.gz)$">
ExpiresDefault A604800
</filesMatch>
</IfModule>
```

The above rule sets an expiration period for each content type such as text, images, CSS, JavaScript etc. As soon as the visitor accesses the content, the timer starts and then a predefined time set in the above rule is added resulting expiration date. Only after the expiration date, the browser is supposed to re-download the corresponding files until and unless the user clears browser cache manually.

Use CloudFlare Free Service

CloudFlare²² is a free service that helps supercharge your website. CloudFlare protects and accelerates any website online.

CloudFlare speeds up your website by

- Distributing content across the world so it's closer to your visitors
- Secures site by protecting server's IP address, spam and other attacks
- Block threats from spammers and limit abusive bots and crawlers from wasting your bandwidth and server resources
- Provides free SSL certificate that you may need to serve wp-admin page and plenty of apps.

Using CloudFlare is simple and can be used by anyone with a website and their own domain, regardless of your choice in platform. With just a simple change to your domain's DNS settings, your website can be added to CloudFlare in less than 5 minutes and you even get a WordPress plugin as well. There exists no hardware or software to install or maintain and also you do not need to change any of your site's existing code. If you are ever unhappy you can turn CloudFlare off as easily as you turned it on. Below image shows the difference between a non CloudFlare based websites with CloudFlare-powered websites on how a significant improvement in performance and a decrease in spam and other attacks is achieved.

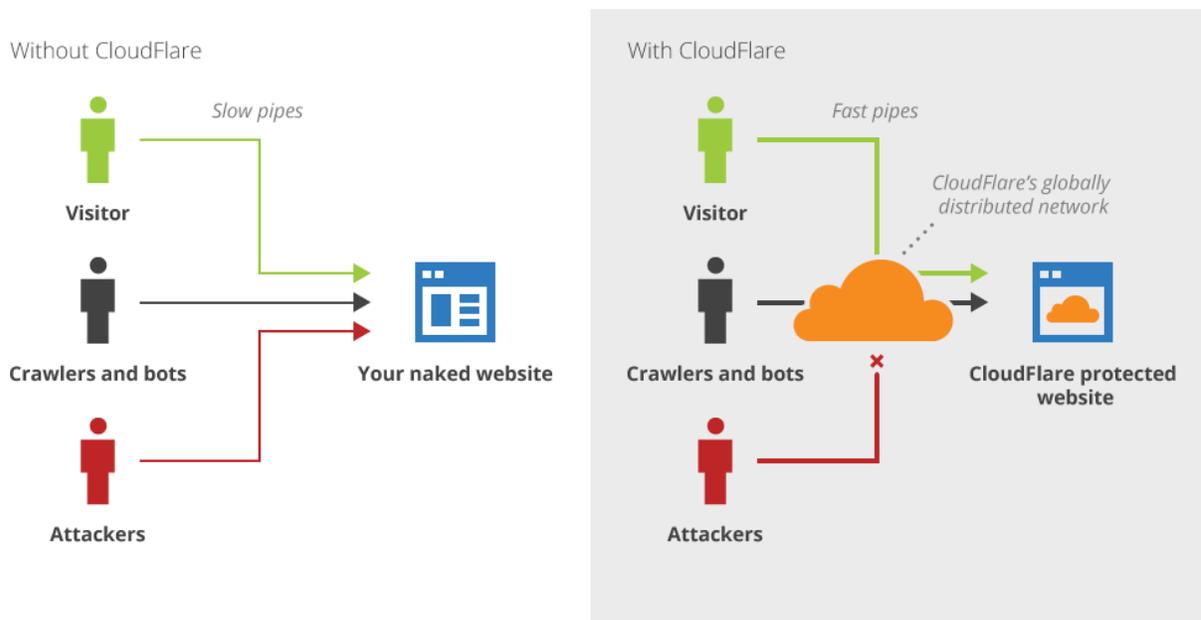


Image Credits: CloudFlare.com

Optimize Images automatically

If your website is going to load tons of images, then it's very important to optimize those. On the average web page, more than 50% of the data is made up of images. And images are only getting larger as screen resolutions on devices improve. These images slow down the page load. There is absolutely no point in wasting the precious bandwidth (especially important for mobile devices that have limited bandwidth) and time on such resources (websites that do not require hi-resolution original images) when it's possible to reduce image size without even losing its quality. On the other hand, image optimization does not only help your website to load faster, but also improves site's SEO.

There's a variety of image compression algorithms that take different approaches to reducing file size, and the plugins such as as [EWWW Image Optimizer](#)²³, [Imsanity](#)²⁴ etc., utilize a number of those to minimize the size of your images.

EWWW Image Optimizer

The EWWW Image Optimizer plugin automatically optimizes the images that uploaded to the website. It can also optimize the images that are already uploaded. It can also convert your images automatically to the file format that produces the smallest image size and optionally apply lossy reduction for PNG and JPG images.

Imsanity



Image Credits: [Imsanity](#)

Don't be scared off by Imsanity's freaky feature image in the WordPress Plugin Repository. Imsanity automatically resizes huge image uploads down to a size that is more reasonable for display in browser, yet still more than large enough for typical website use and also provides a bulk-resize feature to selectively resize previously uploaded images to free up disk space. A nice feature of Imsanity is the ability to set a maximum width, height and quality.

Manage 301 redirects easily

What is a 301 Redirect?

Tell the search engines: “Hey, the page is no longer exists here and has permanently moved to a new page. Please remove it from your index and pass the credit to the new page”. A 301 redirect is the HTTP status code returned by your browser when a web page is requested pointing the browser to go to a different location. While the average user is unlikely to notice the difference search engines treat them very differently;

The type of redirect a web page is using can be checked using a HTTP request and response header tool such as [web-sniffer.net](#)²⁵ or a plugin for Firefox such as [HTTP LiveHeaders](#)²⁶.

There a number of different ways to manage redirects with a WordPress site. You can write the 301 redirection rules in **.htaccess** which is an ideal method to use as a browser will read the .htaccess file before anything else making it quicker than a page level 301 redirect. Below shows sample of 301 redirect in .htaccess file:

```
Redirect 301 / http://newsite.com/  
Redirect 301 /oldpage/ http://example.com/newpage/  
Redirect 301 /oldpage/image.jpg http://example.com/images/newimage.jpg
```

To be frank, 301 redirection rules are not that easy and might look confusing (at least for the beginners). However, the [Redirection plugin](#)²⁷ is more users friendly and can be used making the process of redirecting a page very simple without requiring knowledge of *Apache .htaccess* files. Once the plugin is installed and activated it can be found in the Tools menu of the WordPress dashboard.



The redirection plugin also stores its redirect data in the database as **wp_redirection_items** table created by the plugin. It also captures a log of 404 errors and allows you to easily map these to 301 redirects.

Prevent search engines from Indexing core files

Like we prevented search engines from indexing the XML sitemaps, you may also want to do the same for WP core files located under *wp-admin*, *wp-include*, *themes* and *plugins* folders. To do that, copy and paste the below code in *robots.txt* located under the WordPress root directory.

```
User-agent: *  
Disallow: /wp-admin/  
Disallow: /wp-includes/  
Disallow: /wp-content/plugins/  
Disallow: /wp-content/themes/
```

WordPress Optimization

As you all are aware that WordPress is well-known for its ease of installation. Given all the prerequisites installed WordPress installation would be in just under 5 minutes!!! Under most circumstances, installing WordPress is a very simple process, but it is recommended that you tweak some of the default settings to optimize the performance of your WordPress website. The following guide will help in tweaking some of the default settings of your WordPress site.

Remove HTML in WordPress Comments

By default WordPress allows HTML tags (such as anchor, bold, italic etc.) to be used in comment boxes. Well, html tags help spammers to add links, highlight their brand or marketing terms in bold or italic and finally that's going to make your site ugly and the scary thing is - it plays a big role in negative SEO as well. You shall remove html tags support in WordPress comments using the below code. Just copy and paste the below line to your *functions.php* file.

```
add_filter( 'pre_comment_content', 'esc_html' );
```

Limit Comments Length

Comments help your visitor to engage in a healthy conversation without leaving your site and sometimes it helps an author to understand what visitors actually need. But what should be the ideal length of a comment? 100 or 1000 or 5000 characters?

An expert say, a good or a helpful comment can't be less than 50 characters and never be more than 2000 characters. It makes sense, as lengthier comment might have a negative SEO and it's quite difficult to moderate as well. It also improves the quality of your comments.

To limit the length of a comment, just copy and paste the below code (credit) to your *functions.php* file.

```
add_filter( 'preprocess_comment', 'wpb_preprocess_comment' );

function wpb_preprocess_comment($comment) {
    if ( strlen( $comment['comment_content'] ) > 2000 ) {
        wp_die('Comment is too long. Please keep your comment under 2000 characters.');
```

Code Credits: WPbeginner.com

The above adds a filter hook to *preprocess_comment*. This filter is run before WordPress saves any comments to database or runs any other pre-processing on submitted comments. It also checks the comment length and if it is above or below the set comment length parameters, then the user is shown an error message.

Move WordPress Uploads directory

WordPress stores all your uploaded images in the wp-content/uploads folder. Changing the default uploads directory in WordPress outside the main WordPress folder, does have few advantages – preferable moving it to a subdomain (to a cookieless domain that helps reduce the header size of every http request) makes easy backup management and better organized URL structure. The most important is, serving images from a different domain will allow parallel downloads in the browser improving the page loading time.

Add the following lines to your ***wp-config.php*** file to change the default location of your uploads folder.

```
define( 'WP_CONTENT_URL', 'http://img.example.com/images' );  
define( 'WP_CONTENT_DIR', $_SERVER['HOME'] . '/img.example.com/images' );
```

Note: *The iThemes Security plugin has an option to move wp-content directory.*

Disable Post Revisions

Post revisions allows you to track the changes and it helps to revert back to previous revisions, but this feature considerably increases the database size.

Adding the below line in *wp-config.php* will completely disable post revisions.

```
define( 'WP_POST_REVISIONS', false);
```

If you do not want to disable the post revisions completely, then you may just limit number of post revisions that WordPress stores in the MySQL database. Add the below line in *wp-config.php* file to only store the recent 3 edits.

```
define( 'WP_POST_REVISIONS', 3);
```

Increase the interval of Posts Autosave

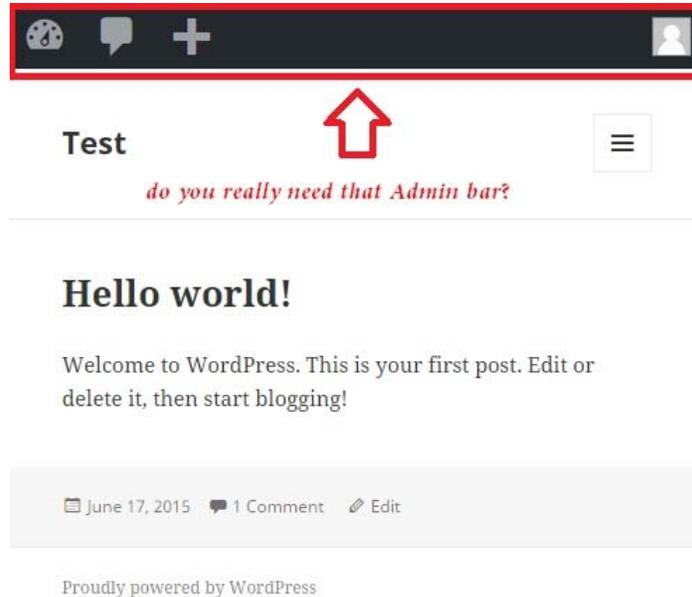
By default, WordPress will automatically save your post as drafts while you type. In case if the browser crashes, you can still recover the post that was auto saved as draft. WordPress autosaves the post at an interval of 60 seconds and if you ever want to increase it, just copy and paste the below line in *wp-config.php* file.

```
define( 'AUTOSAVE_INTERVAL', 180 );
```

The above line will increase the autosave interval to 180 seconds.

Disable Admin bar

I'm not a big fan of an Admin bar that appears at the top when you are logged in to the WordPress.



If you are like me and wish to disable it for all users except administrators, here's how you can do that – simply add the below line in *functions.php* and you are done.

```
add_action('after_setup_theme', 'remove_admin_bar');

function remove_admin_bar() {
if (!current_user_can('administrator') && !is_admin()) {
    show_admin_bar(false);
}
}
```

Else, if you want it to disable for all the users, then add the below code to your *functions.php* file.

```
add_filter('show_admin_bar', '__return_false');
```

Remove WordPress Footer notes

Of course, WordPress helped you in setting up your website in no time and our kudos to this awesome software. But sometimes you may not want others to know that you use WordPress (and that's the main reason why we hide or change wp-admin URL, wp-content folder, renamed theme folders, removed meta tags, removed query strings etc). So leaving that line '*Powered by WordPress*' in the footer is something that I'm not a fan of. Do you still want that?

Note: *You can still give credits to WordPress by contributing to the community or at least by writing about WordPress.*

Disable Automatic Updates

Update WordPress and Plugins regularly, but not automatically. Automatic updates are easier, on time, but they might also introduce new problems to your code and break your site. It's always better to disable automatic updates and do it manually (**caution:** manual updates, not for a beginner).

***Note:** Background auto updates were introduced in WordPress 3.7 in an effort to promote better security. By default it is limited to only minor releases however in special cases WordPress may update your plugins and themes.*

Copy and paste the below code to your *wp-config.php* to **disable WP core update:**

```
define( 'WP_AUTO_UPDATE_CORE', false );
```

Copy and paste the below line to your *functions.php* to **disable automatic plugin updates:**

```
add_filter( 'auto_update_plugin', '__return_false' );
```

And the below line for **disabling theme updates:**

```
add_filter( 'auto_update_theme', '__return_false' );
```

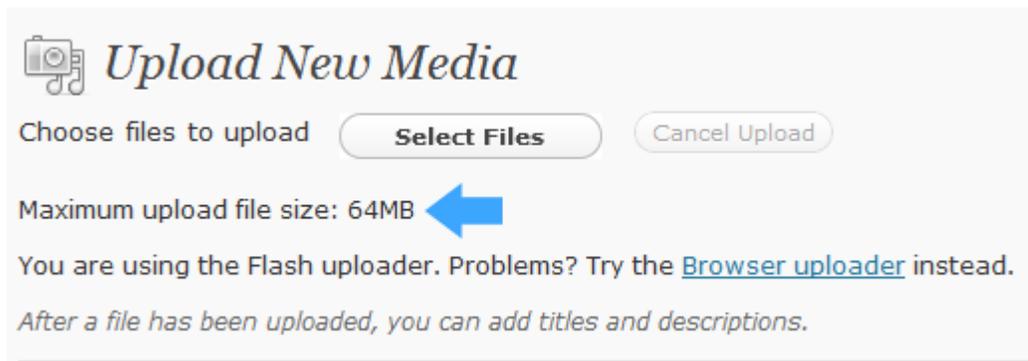
Increase WP upload size

Many hosting providers allows file upload of maximum 2MB, which is clearly not enough for media files like audio/video. Depending on the web hosting company you choose and the package you select, you will see the maximum file upload limit on your Media Uploader page in WordPress.

In most cases if you are on a shared host, you will not see a php.ini file in your directory. If you do not see one, then create a file called php.ini and upload it in the root folder. In that file add the following code:

```
upload_max_filesize = 64M
```

Now in you Media Uploader page, you should see the “Maximum upload file size: 64MB”.



Limit number of posts in RSS feed

RSS feeds provide your users an easier way to subscribe to your site. However, sometimes you may want to limit the number of posts user can view in RSS feed, so it encourages them to visit your site.

Go to *Settings > Reading* and increase '*Syndication feeds show the most recent*'

Use Distraction-free mode for writing

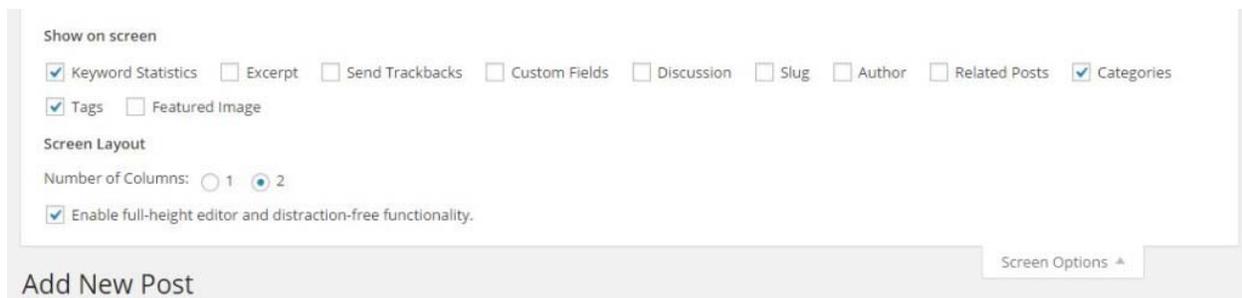
The WP editor is awesome, but it also (the editor) has plenty of widgets around and that can distract you while writing posts. But no worries simply **enable distraction free mode** (a star like icon with four arrows, located just below the *Visual/Text* mode).

WordPress was featuring Distraction Free Editing feature for quite some time. With the release of WordPress 4.1, the feature has got little more improvement. The DFW mode will allow you to switch between the normal and distraction free modes quite easily – thus removing the distractive elements such as the left sidebar menu, right sidebar containing Publish, Categories etc..., the admin bar, screen options menu etc...It means, you will be shown only the editor and the rest are shown only when the mouse is moved out of the editor. I really love the feature, as it provides clean interface and better writing experience.

If you don't find distraction-free mode icon, then you have to enable it first.

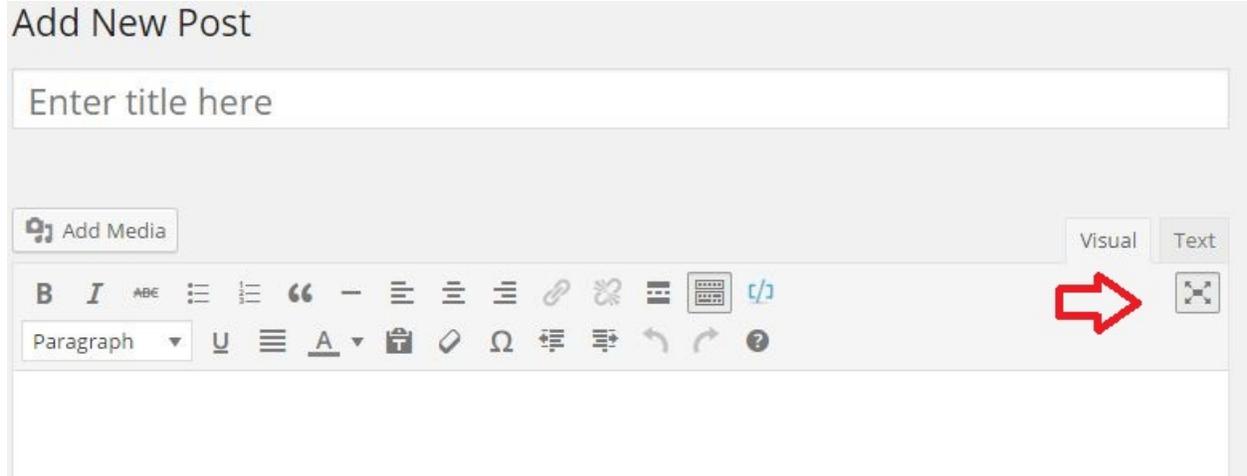
How to enable DFW mode in WordPress?

At first sight, if you could not find the Distraction free writing icon, then you can enable it by clicking the “*Screen Options*” at the top and then check “*Enable full-height editor and distraction-free functionality*”. Once done, you should see the DFW icon in the editor toolbar just below the “*Visual/Text*” tabs.



DFW icon missing even after enabling “*Enable full-height editor and distraction-free functionality*”?

Probably, the issue might be because of the cache. To confirm, try accessing wp-admin in Incognito window to make sure cache is not the culprit. You should try clearing the browser cache, caching plugins if any.



Stay logged-in to WordPress for longer duration

The default WordPress behavior for logging a user out is to make a login session cookie that expires in 48 hours or when the browser is closed. If the “Remember Me” box is checked, WordPress will give you 14 days before forcing you to authenticate again.

But what if you don’t want to be bothered to remember your passwords or take the time to look them up? When working with multiple WordPress sites, keeping track of all of your highly complex/tricky admin passwords and remembering them on command every two weeks can become a challenge.

If you want to stay logged-in for longer duration, then copy and paste the below code to your *functions.php* file.

```
add_filter( 'auth_cookie_expiration', 'stay_longer' );
function stay_longer( $expire ) {
    return 8640000; // 100 days in seconds
}
```

Change it to whatever time frame you like. You can, in essence, stop WordPress from ever logging you out by changing the number of seconds to be a much higher number.

Remove unused CSS styles using Chrome Developer tool

As a developer you keep adding few features and change designs often, but when you do that, you should also remember to remove unused CSS styles, as this will help in reducing file size and result in faster rendering by the browser.

There are plenty of factors that can affect your website's performance. For instance, the size of the web page, number of HTTP requests, network latency, server response time etc...As a developer, you might keep adding new features, designs etc... and sometimes you might forget to remove old styles declared on your stylesheet. Such unused CSS rules will add to your page size and will ultimately result in high load time. But identifying the unused CSS rules is not that easy. Thankfully, Google Chrome has a feature called "Audit" (as a part of Chrome Developer tools) that helps you to identify and remove unused CSS rules.

Chrome Audit tool reports various factors that affect website's performance; For instance, Web Page Performance, Network performance related issues such as Minimize cookie size, Leverage browser caching, Leverage proxy caching etc...

How to access Chrome Developer tools?

Option 1: You can access Chrome's Developer Tools by clicking  > *Tools* > *Developer tools*.

Option 2: Right click on a webpage and select "*Inspect element*"

Option 3:

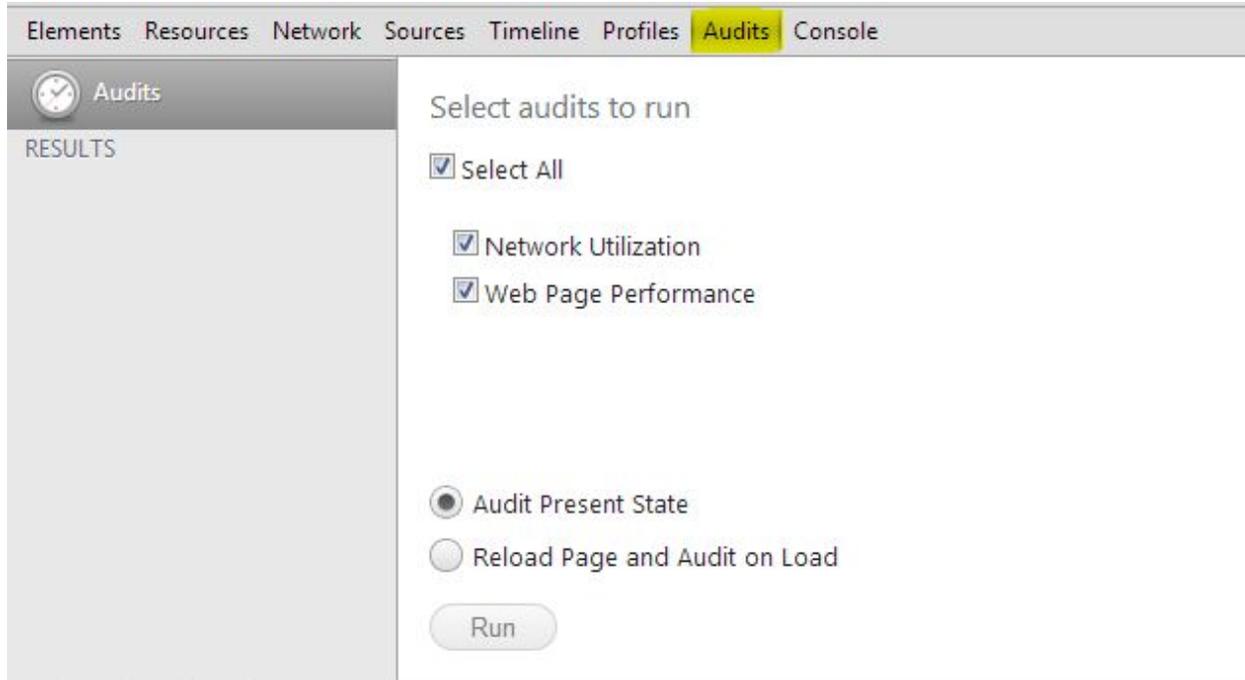
Use Ctrl+Shift+I (or Cmd+Opt+I on Mac) to open the DevTools.

Use Ctrl+Shift+J (or Cmd+Opt+J on Mac) to open the DevTools and bring focus to the Console.

Use Ctrl+Shift+C (or Cmd+Shift+C on Mac) to open the DevTools in Inspect Element mode, or toggle Inspect Element mode if the DevTools are already open.

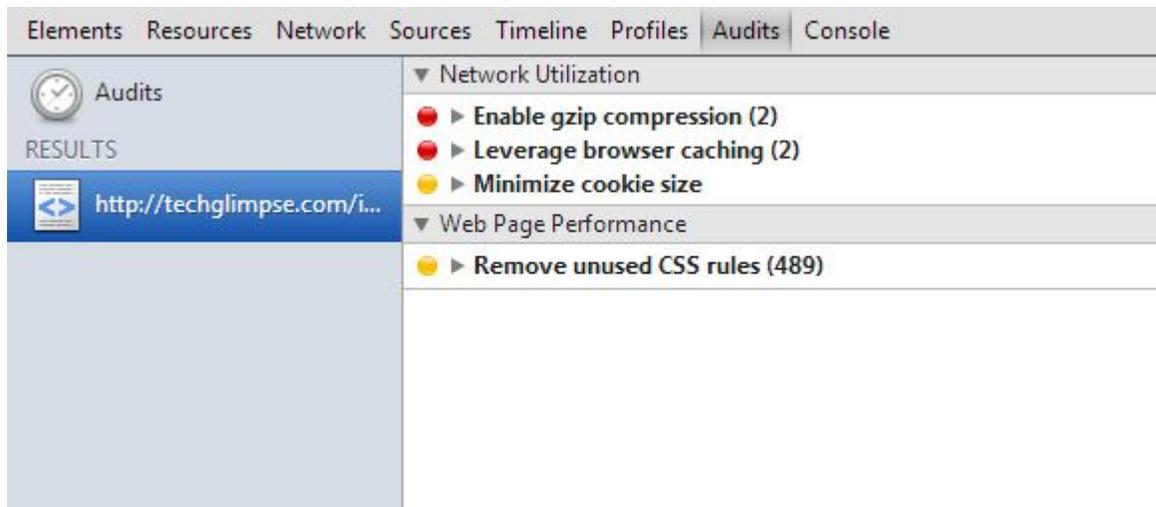
To Audit your website

Click on "*Audits*" tab to reveal the Chrome Audit window.



You can choose to enable or disable the kind of Audits you like to perform. Similarly, you can choose to Audit the webpage at its present state or reload the page and audit on load. Once done, click 'Run' button.

Once the Audit is completed, the tool will display the list of identified problems as below. It will also save the results and it can be access under Results tab.



That's it! Click on the arrow to see the identified issues and fix it.

Handle Adblockers smartly

There are plenty of browser extensions out there that allows user to block online advertisements. Most of the users install Ad-blockers on their browser that helps the user to save some bandwidth and hide unwanted disturbing ads. However if you are a web publisher, then Adblock or Ghostery can affect your online advertising revenue. Handle them smartly by detecting blocker plugins and display alternate content

If you are using **Google AdSense to monetize** your website, then there's a simple hack that allows you display an alternate advertisement (your own advertisement image) or content to Adblock users.

Here we go,

The idea is to find the **first AdSense** unit in a page and if that doesn't load, replace that particular AD unit with an alternate content. The alternate content can be an image to promote your own content or a product. Why not, you can even request the visitor to whitelist your domain or donate. It means, displaying something is better than showing an empty space isn't?



Copy and paste the below script at the footer of your website.

```
<script>
window.onload = function() {
setTimeout(function() {
var ad = document.querySelector("ins.adsbygoogle");
if (ad && ad.innerHTML.replace(/\s/g, "").length == 0) {
ad.style.cssText = 'display:block !important';
}
}
});
</script>
```

```
ad.innerHTML = ''  
}  
, 1000);  
};  
</script>
```

Code Credits: [Amit Agarwal](#)

***Note:** The code will work only for AdSense.*

That's it.

The above code will **replace the blocked AdSense unit with an alternate image.**

References

1. <http://techglimpse.com/wordpress-popular-security-plugins-infographic/>
2. <https://digwp.com/2010/10/change-database-prefix/>
3. <http://techglimpse.com/index.php/90k-wordpress-blogs-hacked-security-tips.php>
4. <https://downloads.wordpress.org/plugin/ultimate-landing-page-and-coming-soon-page.1.1.34.zip>
5. <https://wordpress.org/plugins/disable-xml-rpc-pingback/>
6. <http://labs.sucuri.net/?is-my-wordpress-ddosing>
7. <https://wordpress.org/plugins/better-wp-security/>
8. <http://downloads.wordpress.org/plugin/google-authenticator.0.47.zip>
9. <http://downloads.wordpress.org/plugin/wpclef.1.9.1.2.zip>
10. <https://wordpress.org/plugins/google-authenticator/installation/>
11. <https://getclef.com/apps/>
12. <http://support.getclef.com/article/13-setting-up-clef-on-a-wordpress-site>
13. <http://ctrlq.org/code/19247-password-protect-wordpress-admin>
14. <http://wordpress.org/plugins/user-locker/>
15. <https://api.wordpress.org/secret-key/1.1/>
16. <https://wordpress.org/plugins/remove-dashboard-access-for-non-admins/>
17. <https://downloads.wordpress.org/plugin/wp-optimize.1.8.9.10.zip>
18. <https://downloads.wordpress.org/plugin/w3-total-cache.0.9.4.1.zip>
19. <https://www.feedthebot.com/pagespeed/prioritize-visible-content.html>
20. <https://downloads.wordpress.org/plugin/wordpress-seo.2.2.1.zip>
21. <http://gtmetrix.com/>
22. <https://www.cloudflare.com/>
23. <https://downloads.wordpress.org/plugin/ewww-image-optimizer.2.4.4.zip>
24. <https://downloads.wordpress.org/plugin/imsanity.zip>
25. <http://web-sniffer.net/>
26. <https://addons.mozilla.org/en-US/firefox/addon/live-http-headers/>
27. <http://urbangiraffe.com/plugins/redirection/>

Techglimpse.com



Techglimpse.com, founded in 2008 by Digisparks Infotech (formerly Focus 4 Infotech) is one of the world's fastest growing technology website that delivers an international perspective on the latest buzz and happenings across tech community. Its mission is to offer an obsessive coverage of cutting edge gadgets, consumer electronics and the science and technology they're built upon.

Our special editors bring exclusive tips on Social Media including Facebook, Twitter, Google Tools, Android, Jailbreaks etc., Our exclusive articles bring fool proof review of consumer products, Android Apps and software products.