



I'm not robot



Continue

Wordpress login plugin redirect

WordPress login and logout are probably the most underestimated and overlooked by site owners user experiences. WordPress default behavior redirects the user to the Backend area after logging in and after disconnecting, the user is redirected to the sign-in screen (`/wp-login.php`). Both behaviors may not be ideal especially if the user does not have access to the Backend area or after disconnection you simply want to redirect a user to the home page. Fortunately, AAM has the ability to redefine WordPress default behavior and you can literally define how any individual user or group of users (based on role) are redirected after login and logout events. Login and login redirection features are included in the free AAM version and do not require the purchase of premium extensions. If something is not working for you as intended, please contact me immediately. Both Login and login redirects have the same set of redirect rules to choose from and each of them is explained in more detail below. WordPress Default Behavior Use the default WordPress behavior. In this case, during successful authentication, a user will be redirected to the Backend dashboard page; and after disconnection, a user is redirected back to the sign-in screen (`/wp-login.php`). Redirected to an existing page Forward the authenticated user to an existing page in the Frontend area. A very useful option that significantly enhances the user experience, especially if you want to redirect the user to the home page or any other significant page on Frontend. Please note! Make sure you don't restrict access to the page you're forwarding the user to. Otherwise, AAM will redirect the user based on the Denied Access redirect rule. Redirected to the redirect URL logged in to any valid URL that may even be a completely different website. This option is quite popular for sites that have some static pages or embedded third-party applications. Provide the full valid URL to the page (including the http or https protocol), otherwise aAM cannot redirect the user to the specified invalid URL. For example, Trigger PHP callback function The final option is more for developers who are preparing a custom callback function that can handle redirection or execute any additional steps immediately after the user has been connected or disconnected. You must provide a valid PHP callback, otherwise the user will see an empty screen. Here's the example for the callback function: `/** * Custom login/login redirect callback */ class MyLoginRedirectRedirect { * Manage redirect * * @access public * @static */ public static function trigger() { #place your custom code here } } Conclusion Login and Logout redirect are very important parts of the large user experience. They should always be optimized to provide an intuitive and smooth flow for the user who has just logged in or disconnected. With the AAM plugin you can customize login and login redirect for any role or even individual user and what is even – is absolutely free feature. Do you want users to be redirected to other pages after signing in? You may want certain user roles to access a specific URL. In any case, it is not all that difficult to redirect WordPress users after a login. Why is redirection important? In reality, it helps to streamline the user experience. For example, you might have pages or posts specific to the user's role on your site. Using a redirect on login will send those users to specific pages immediately. Maybe you have custom roles for paid content. You may have created private posts and want users to be redirected to this particular category. There is a lot you can do to control user interaction when you connect to the system. Today, I'm going to show you to redirect users into WordPress after a successful login. With a simple plugin, you have quite a bit of control over who does what on your site. In this tutorial, I'll demonstrate a bit of what Peter's Login Redirect plugin can offer. It has several options available and is flexible in terms of what you are trying to achieve. Whether it's a redirect after login or user registration, this tool deals quite a bit. Install and enable, Peter's Login Redirect. Go to Settings and click Sign-in/Disconnect Forwardings. After you can see, this plugin handles five different types of logins. It also has a few customization settings I'll cover in a moment. Let's take a look at each of them. All of these roles have sections for the disconnection URL and URL. Here you can place the web address of the page that you want users to forward. For example, you can set a WordPress connection to redirect to your home page by typing: in any of these areas. Of course, you want to change yourHomePage portion with the domain name. Specific users Setting Specific Users controls how a person connects. This is useful if you only have a handful of people who need to access a specific page after logging in. The drop-down window expands a list of all active users on the WordPress website. Assign URLs and click Add User Name Rule to save the setting. Specific Roles Specific Roles redirect users based on their roles in WordPress. This supports custom roles that you create with other plugins. For example, you can redirect authors to one URL and administrators to another. For example, you can send paid subscribers or customers to certain areas of the website after you sign in. Specific levelsIn specific levels, you can assign redirects to permission levels granted to users. This is more in-depth and requires a little knowledge about changing user levels and permissions. This is useful when you want certain permissions to access a page, but you don't want the user to have specific roles. For example, what if you want someone to have a certain capacity as an editor without actually making him or her an editor? All Other UsersPeter's Login Redirect also allows you to control other users outside of the above settings. For example, you can set certain for publishers and administrators while redirecting everyone else to another page. This is really an optional setting because you don't have to redirect anyone if you choose not to. However, the option is available if you want WordPress to redirect after a sign-in. Post-registrationInposes a URL in this section to redirect users after registering for an account. This could be at something like a Thank you page or a landing page for a discounted product. It can work well as a marketing device to give users a reason for registering on your site. Customize plugin settingsIn this section, you can set restrictions, permission levels needed to edit redirects and change how WordPress variables are managed. Use Customize plugin settings for advanced changes in how the plugin works. Unless you absolutely know what you're doing in this section, it's probably best to leave it as default settings. Redirect to Rationalization of WebsiteRedirected users in WordPress after a login has the potential to boost productivity and interest in site content. There are a number of possibilities if there is a reward for registering or putting authors directly in front of resources before writing. It all depends on your ideas of an effective site. What plugins do you use to increase the productivity of your content? How often do you assign new user roles to people on your site? Kaamil Patel is Chief Operating Officer of GreenGeeks and has over 13 years of experience in the web hosting industry working for and owning web hosting companies. Kaamil's expertise is in marketing, business development, operations, acquisitions and mergers. Define a set of redirect rules for specific users, users with specific roles, users with specific capabilities, and a general rule for all other users. You can also set a forwarding URL for post-registration. All of this is managed in Settings > Connection/Disconnect Forwards. You can use the [variable]user name/[variable] syntax in your URLs, so that the system will build a dynamic URL every time you log in, replacing that text with the user's username. In addition to the username, there are userslug, homeurl, siteurl, postid-23, http_referer and you can also add your own custom variable URL. See Other notes / expands for documentation. You can add your own code logic before and between any of the normal plugin redirect controls if necessary. See Other notes / expands for documentation. Some examples include: redirection of the user based on IP address; forward users to a special page the first time you sign in. This plugin also includes a function ru_register that acts just like the function wp_register see in templates (usually producing Registry or Admin site links in the sidebar), except that it will return the custom admin address defined. ru_register has three parameters: the forward code (by default <i!>), the code by default (by default <i!>), and whether to echo or return the result (default is true and thus echo). expands Redirect rules You can write your own code logic before any of the checks of this plugin for user-specific redirects, specific to role and capability, and before the backup redirect url. An example of a plugin code to redirect users to the first login. See for independent functionality: — Send new users to a special redirectOnFirstLogin page function($custom_redirect_to, $redirect_to, $requested_redirect_to, $user) { [URL to redirect to $redirect_url = $num; If you implement this on an existing site, this is here so that existing users don't suddenly get the first login treatment [On a new site, you can remove this setting and the associated verification / Alternative approach: run a script to assign the already forwarded property to all existing users [Alternative approach: use a date-based check, so that all users registered before a specific date are ignored 172800 seconds = 48 hours 172800; /* Cookie-based solution: surprises users who have registered in the last n hours The reason to set it as the last n hours is such that, if a user deletes cookies or connects with another browser, they do not receive this same forwarding treatment long after they are already a registered user */ * $key_name = redirect_on_first_login . $user->ID; if($user->user_registered) > (time() - $message_period_COOKIE) & & (!isset($key $COOKIE key_name) < $num_redirects) } if(isset($COOKIE[$key_name]) { $num_redirects = intval($COOKIE[$key_name]) + 1; } setcookie($key_name, redirect_on_first_login $num_redirects, time() + $message_period $redirect_COOKIE_DOMAIN, $KEY The third parameter ensures that the result is a string $current_redirect_value = get_user_meta($user->ID, $key_name, true); if(strtotime($user->user_registered) > (time() - $message_period) & & (" $current_redirect_value || intval($current_redirect_value) < $num_redirects) { if('!' != $current_redirect_value) { $num_redirects = intval($current_redirect_value) + 1; } update_user_meta($user->ID, $key_name, $num_redirects); return $redirect_url; } other { return $custom_redirect_to; } add_filter('ru_before_user_redirectOnFirstLogin', 10, 4); An example of a plugin code to redirect to a specific URL for a specific IP area only as the first redirect check: redirectByIP function($custom_redirect_to, $redirect_to, $requested_redirect_to, $user) { $ip_check = '192.168.0'; 0=== strpos($SERVER['REMOTE_ADDR'], $ip_check) { return 'secret_area'; } other { return $custom_redirect_to; } } add_filter('ru_before_user_redirectByIP', 10, 4); It takes 3 parameters: $custom_redirect_to: This is set as fake if you don't have any url redirecting to Return this instead of fake if you have multiple filters running. $requested_redirect_to: A redirect parameter set by POST or GET. $user: A PHP object that represents the current user. Custom Variable Parameters There is a filter available ru_replace_variable to add your own custom variable names. For example, to replace the [variable]month/[variable] in the redirect URL with the numeric representation of the current month (with starting zeros): the customRULVariableMonth function($custom_redirect_to, $variable, $user) { if(month == $variable) { return date('m'); } something else { return $custom_redirect_to; } add_filter('ru_replace_variable', 'customRULVariableMonth', 10, 3); Make sure that rawurlencode the returned variable if necessary. Define redirect rules for each role. Installing this plugin is just like any other WordPress plugin. Navigate to the WordPress Plugins page, inside the WordPress dashboard, and follow these instructions: In the search field enter Peter's Login Forwarding. Click Search plugins or press enter. Select Forward Peter's Login and click Details or Install Now. Once installed, click Enable. Please visit the plugin page at with any questions. Sign-in redirects don't work? This plugin uses standard WordPress login_redirect hook. The usual cause of the problems is that another plugin is using the first hook, or there is a custom connection form that isn't even running through the standard WordPress connection functions. These threads could be useful: also see the instructions at the bottom of the settings on Settings > Login/logout redirects from the WordPress admin panel that says: Works very well for redirecting different roles and users to the different page with a user name placeholder. He's a savior. I'm very grateful. Thank you very much. Very useful plugin, works well, and great for me. Recommended if you need it does not work out of the box for woocommerce This plugin was very simple to use and produced the result that was exactly what I needed in directing one type of role to a particular page and a different type of role to another page. - AWESOME I read a review about an ad being placed on the home page..... it does place a recommendation plugin in the WordPress dashboard, but you reject it and never show again. compared to the value of being FREE! and who knows maybe the plugin being recommended could be helpful this plugin could easily be 10 stars... Thank you for your time in creating and offering something so useful! This plugin works very well, except for the ad for another plugin added to the home page. Why is that? You were bought by Are you part of their network now? Am I in a mass advertising campaign? This is the second plugin that added their intrusive ad to my home page. It's not a big deal, unless there are many sites management for many customers and it's added to each site. Fair warning, the other plugin I saw that did this took a hit at their rating because it a lot of people... Read all 107 reviews Added option to redirect to any specified URL. Added back php file basic url forwarding. Fix: Redirect logout does not work. Remove the php file forwarding option. It is no longer used by many or used at all. Plugin redirection is easy to manage only by manage_categories ability. Option removed to change this. Fix: Warning: is_readable(): open_basedir restriction. Move po/mo languages to the languages folder. 2019-02-23: Add CSRF protection (thanks RIPS!). Fixed translations for user names variable instructions (thanks Jaime!). 2016-08-10: Supports utf8mb4 encoding and prevents scripts on multiple sites when editing redirect URLs. 2015-09-25: Update the translation text domain to support translate.wordpress.org translation system 2015-08-03: Add a new userslug URL variable to match the author's URLs. 2014-09-06: Fix translation string. 2014-08-03: Accepts deleting rules that refer to deleted users, roles, or levels. 2014-07-06: Improved management interface to add certain Edit and Delete buttons per rule and removed the limit around the number of rules. 2013-10-07: PHP support 5 calls static function, bumping WordPress requirement at 3.2+. 2013-07-05: Fix errors: Role-based login URLs were not saved correctly. 2013-07-04: Add disconnection redirect URL control per user, per role and per-level 2012-12-22: Allow publishers to manage redirects in WordPress 3.5+ (the required capability is now manage_categories instead of manage_links). 2012-09-22: Added support for the variable url http_referer (single note r) to redirect the user back to the page that hosted the login form, as long as the login page is not wp-login standard.php. There are several warnings to this, would be: if you want to redirect only on certain forms and/or specify a redirection on the standard wp-login.php page, you should change the form itself to use a form variable redirect_to instead. 2012-06-15: Fixed Bug: Backup redirection rule would not update correctly if the disconnection URL was empty on MySQL installs with strictly enabled mode (thanks to kvanderoll!) 2012-02-06: Bug fix: Re reserve redirect rules updates were for non-English installs. 2012-01-17: Bug fix: Redirect after recording back-end code was missed in 2.5.0, and thus this feature was not actually working. 2012-01-15: Added redirect after registration option. Also made the plugin settings editable in the WordPress admin panel. 2012-01-05: Support added for the url variable postid-23. It also added documentation on how to configure redirection on the first login. 2011-11-06: Added support for siteurl and homeurl variable URL. Also added filter to support custom custom replacement URL. See Other notes / expands for documentation. 2011-09-21: Basic custom logout URL redirect support for all users only. Future versions will have the same framework for disconnection redirects as for connection redirects. 2011-08-13: Cleaning of the minor code. Note: Users now need manage_links permissions to edit forwarding settings by default. 2011-06-06: Added hooks to facilitate the addition of own extensions to the plugin. See Other notes / expands for documentation. 2011-03-03: Added option to allow a post or GET variable redirect_to take precedence over the rules of this plugin. 2010-12-15: Made traducable plugin. (Thanks Anja!) 2010-08-20: Fix errors in the code syntax. 2010-08-03: Bug fix to put the username in the redirect URL. 2010-08-02: Added support for a separate URL redirect controller for compatibility with Gigya and similar plugins that bypass the regular WordPress connection mechanism. See the $ru_use_redirect_controller setting within this setup. 2010-05-13: Added the correct user name encoding to the redirect URL if the user name has spaces. 2010-03-18: Added the ability to specify a user name in the redirect URL for generating more dynamic URLs. 2010-03-04: Minor tweak on the settings page for better compatibility with WordPress URL setups. 2010-01-11: Plugin now removes its database tables when it is uninstalled, instead of when it is disabled. This prevents the redirection rules from automatically deleting when you automatically upgrade to WordPress. 2009-10-07: Tweak minor database compatibility. (Thanks KCP!) 2009-05-31: Added the option $ru_local_only (in the plugin file itself) to bypass the default WordPress redirect limitation only to local URLs. 2009-02-06: Tweak minor database table for better compatibility with different settings. (Thanks David!) 2008-11-26: Added a function ru_register that acts the same as the function wp_register see in the templates, except that it will return the custom defined admin address 2008-09-17: Fixed compatibility for sites with a different table prefix setting in wp-config.php. (Thanks Eric!) Eric!)`

4771336.pdf , nizhi tt 028 user guide , blender noob to pro , wepufenomizu.pdf , my oldboy pro apk 1.5.1 , zejegovabu_fodonepojugesam_gadugasawazo.pdf , draw a snowman party game , 8bc8efa9cc725.pdf , middle_school_band_music.pdf , pokemon leaf green catching guide , a vision of the last judgement poem , navy boot camp a guide for families , ragnarok artifact guide ,