Restrict access to WordPress admin areas (wp-login.php)

The Problem:
You cannot access your WordPress admin. This is due to an error which says something like:

> "Not acceptable. Access to wp-login.php has been limited due to Brute Force Attack."

The Reason:
Recently GlowHost and other hosts across the globe have experienced (and have been experiencing for a long time now) a massive attack targeted at WordPress sites. Specifically, the attack was directed on WordPress sites admin login area. You can read more about this epidemic, here:
Brute Force Attacks « WordPress Codex

We've been fairly lucky to have been able to avoid these problems for so long, but the number of attack attempts has become so large that it has started to affect server performance, and we have had to take some measures to improve server performance as below...

GlowHost Action:
In order to reduce the load on the servers and stop the attack, we were required to block access to the wp-login.php file for all sites which use WordPress. This is the file which normally allows you to login to your WordPress admin area.

What to do Next:
The good news is that you can easily enable wp-login.php for your local IP. This will allow you to gain access to your WP Admin, without allowing the bad guys in. The ideas in this thread are also good for any other scripts that you may have.

The following recommendations will work best if you have a **static IP address** from your Internet Service Provider, but if you have a **dynamic IP**, it will work just as well. You may also accomplish the same end-goal by signing up for a **dynamic DNS service**.

Finally, there is the option of using password protection if whitelisting dynamic IPs and static IPs are not something that you want to do. That option is called **htpasswd** (which is essentially just using cPanel's built-in password protected directories option) and you can do it simply by using your control panel to create the password protection for you. However, you may need to modify .htaccess manually to remove the deny and allow rules from the template that you see below which may already be installed on your sites.

Novice users should choose 1 of the following 4 options. If you are an advanced user, you may wish to use a combination of password protection along with allowing only your local IPs.

Option #1: Get a **static IP** if you do not have one already, then modify your .htaccess file as needed.

GlowHost recommends usages of a static IP from your ISP (when possible) for all web site owners, due to the growing number of security risks that emerge every day. Static IP addresses can usually be obtained by your ISP for a nominal fee. A static IP can save you a lot of time, and allows you to block the entire Internet from gaining access to (or even viewing) your sensitive admin areas, or other private areas of your site. When you have a static IP, you are able to configure your web site to allow "only you" access, or allow access to only "your team." You should never have to change the settings on your server, once you have configured a basic set of rules assuming everyone who needs access has a static IP.

Option #2: Use your current **dynamic IP** address and modify your .htaccess file as needed every time your IP address changes.

If you have a **dynamic IP** address, like most Internet users, it just means that every time you login to WordPress admin, you will need to modify the allowed IP in your .htaccess file. The benefit of a static IP is that you do not have to change the allowed IP in your .htaccess file, since your local IP never changes.

We'll get into this more as you read along, but you will notice that not having to change your allowed IP every time you wish to make a change on your WordPress site or other scripts, might be worth the price of a static IP.

Please note, a static IP (from your ISP) is not the same as a **dedicated IP** (on your server from GlowHost). If you have questions about the differences, please feel free to post in this thread, or make a new thread. New posts and threads are great, and much appreciated!

If a static IP from your ISP is cost prohibitive, as the sometimes are with USA based mobile/cellular carriers, you may also consider a Dynamic DNS service. Static IPs from a DSL service may only run a couple of dollars per month.

Option #3: Use a Dynamic DNS Service.

Dynamic DNS services map a domain name to your local IP address, even if it is a dynamic IP. The means even if your local IP changes, the domain name that is mapped to your computer will always resolve to your computer, no matter what the current IP address is. It works the same way as a static IP address in most cases, but is not always as convenient. This is a good option if static IP addresses are too expensive for your needs.

As an example of such service we can suggest you NoIP.

In case Web Server doesn't support hostname lookup, you can use the following php script which updates .htaccess file

```
<?php
//Don't forget to update the path to htaccess and hostname and username
$htaccessFile = "/home/usernam/public_html/.htaccess";
$handle = fopen($htaccessFile, "r");
if ($handle) {
    $previous_line = $content = '';
    while (!feof($handle)) {
        $current_line = fgets($handle);
        if(stripos($previous_line,'# Allow from person.getmyip.com') !== FALSE)
        {
            $output = shell_exec('host MY-HOSTNAME.dynamic-dns.com');
            if(preg_match('#([0-9]{1,3}\.){3}[0-9]{1,3}#',$output,$matches))
            {
                $content .= 'Allow from '.$matches[0]."\n";
            }
        }else{
            $content .= $current_line;
        }
        $previous_line = $current_line;
    }
    fclose($handle);
    $tempFile = tempnam('/tmp','allow_');
    $fp = fopen($tempFile, 'w');
    fwrite($fp, $content);
    fclose($fp);
    rename($tempFile,$htaccessFile);
    chown($htaccessFile,'username');
    chmod($htaccessFile,'0644');
}
?>
```

And add this script to cron:

```
*/5 * * * * /usr/local/bin/php /home/user/public_html/allow_person.php >/dev/null 2>&1
```

Option #4: Use password protection.

To use password protection, you'll want to first add the password protection to your wp-admin area using the Password Protected Directories option in your cPanel. Once you have enabled password protection, if you have already set any rules in .htaccess to deny or allow IP addresses or dynamic DNS services. you can now consider if you want to additional protection from these deny rules, or if you want to solely rely on password protection.

Once you have categorized yourself, here is what to do:

Options 1-3 work the same way with .htaccess.

Assuming you currently have a static IP or dynamic IP, all that you need is to find your IP (What is My IP?) and edit the .htaccess file in the folder where you have installed WordPress. If this file does not exist, then you may need to create it.

If you do not know how to modify .htaccess, there is a link at the bottom of this post which will explain more.

Here is the code which you need to insert into your .htaccess file:

```
<Files wp-login.php>
Order Deny,Allow
Deny from All
Allow from xx.xx.xx.xx
</Files>
```

xx.xx.xx.xx is your IP address that you found at the link above.

Please, note that .htaccess file is hidden, and isn't shown by cPanel file manager by default.
Just go to **Setting** in the right corner on the top of the main page of cPanel File Manager and check ***Show Hidden Files***.

If you use dynamic DNS, please put your domain name that was assigned to you from your Dynamic DNS service provider here, instead of any IP addresses.

Please create a ticket at the helpdesk if you have any difficulties.

More WordPress

- Wordpress
- WordPress Gutenberg Editor (15 Video Tutorials)
- Increase memory limit for WordPress
- How to Stage a WordPress Website Using Softaculous
- How to reset WordPress admin password
- How to disable wp-cron.php