Cyberdream BBS

WordPress Security and Setup Guide

v0.4.0 - 2026

This guide covers the manual steps needed to fully secure a WordPress installation running the Cyberdream BBS theme. Some steps require server access via FTP, cPanel or SSH. Read through the whole guide before starting so you understand what each step does.

Contents

1. Identifying your web server
2. Protecting wp-login.php with Basic Auth
3. WordPress settings you need to configure manually
4. Automatic updates
5. Plugin philosophy
6. PHP version
7. Cloudflare setup and tips
8. fail2ban (VPS and dedicated server only)
9. General recommendations

1. Identifying Your Web Server

This guide has separate instructions for Apache and Nginx. Here is how to find out which one you are running.

Method 1 - Check your hosting control panel

Most shared hosting panels like cPanel, Plesk and DirectAdmin show the web server type in the overview or server information section.

Method 2 - Check response headers

If you use Cloudflare, temporarily set your DNS to DNS only (grey cloud instead of orange) so requests go directly to your server. Then open your site in a browser, press F12, go to the Network tab, click the first request and look for the Server header. It will say Apache or nginx. Switch Cloudflare back on afterwards.

Method 3 - Quick PHP file

Create a file called info.php in your WordPress root with this content:

```
<?php
echo $_SERVER['SERVER_SOFTWARE'] . '<br>';
echo phpversion() . '<br>';
echo __FILE__;
?>
```

Visit it in your browser. It shows the server software, PHP version and the absolute file path on disk. Delete the file immediately after - never leave it on a live server.

Method 4 - SSH

```
apache2 -v
nginx -v
```

One.com and most shared hosting providers run Apache. If you are not sure, follow the Apache instructions.

2. Protecting wp-login.php with HTTP Basic Auth

Adding HTTP Basic Auth in front of wp-login.php creates a second login layer. An attacker has to get past Basic Auth before they even reach WordPress. Combined with the brute force protection built into the Cyberdream BBS theme, this makes login attacks very difficult.

> This does not affect your visitors, comments, or any public part of the site.

Step 1 - Create a .htpasswd file

Choose a username that is not obvious - not admin, login, gate or anything similar. Use a strong random password of at least 16 characters.

If you have SSH access:

```
htpasswd -c /path/to/.htpasswd your_chosen_username
```

Place the file outside your public web root if possible. On shared hosting this may not be possible - placing it in the web root is acceptable as long as your server is configured to deny direct access to dotfiles, which Apache does by default.

If you only have FTP or cPanel access:

Use an online htpasswd generator - search for htpasswd generator and choose bcrypt as the hash algorithm. Save the output as a file named .htpasswd and upload it to your WordPress root.

Step 2 - Apache: edit .htaccess

Add the following to your .htaccess file in the WordPress root. Replace the path with the actual path to your .htpasswd file.

```
# Protect wp-login.php with Basic Auth
<Files wp-login.php>
    AuthType Basic
    AuthName "Restricted Access"
    AuthUserFile /absolute/path/to/.htpasswd
    Require valid-user
</Files>

# Protect wp-admin
<FilesMatch "^(.*)?wp-admin(.*)$">
    AuthType Basic
    AuthName "Restricted Access"
    AuthUserFile /absolute/path/to/.htpasswd
    Require valid-user
</FilesMatch>
```

Step 2 - Nginx: edit your server block

Add the following inside your server block in your Nginx config, typically in /etc/nginx/sites-available/yourdomain.conf.

```
location = /wp-login.php {
    auth_basic "Restricted Access";
    auth_basic_user_file /absolute/path/to/.htpasswd;
```

```
        fastcgi_pass unix:/run/php/php8.2-fpm.sock;
        include fastcgi_params;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    }

    location ^~ /wp-admin/ {
        auth_basic "Restricted Access";
        auth_basic_user_file /absolute/path/to/.htpasswd;
        try_files $uri $uri/ /index.php?$args;
    }
```

Reload Nginx after editing:

```
sudo nginx -t && sudo systemctl reload nginx
```

Step 3 - Test

Visit https://yourdomain.com/wp-login.php. Your browser should show a Basic Auth dialog before the WordPress login form appears. Your public site should be completely unaffected.

3. WordPress Settings to Configure Manually

3.1 Set a display name different from your username

Your WordPress username is one half of your login credentials. By default WordPress may show it publicly in author URLs and bylines. The Cyberdream BBS theme uses display names in author URLs instead of usernames, but you still need to set a display name that is different from your login name.

1. Go to Users - Your Profile in wp-admin.
2. Fill in First Name, Last Name or set a Nickname.
3. Change Display name publicly as to something that is NOT your username.
4. Click Update Profile.

> Never use your WordPress username as your display name.

3.2 Disable search engine indexing during setup

While you are setting up your site, go to Settings - Reading and check Discourage search engines from indexing this site. Remember to uncheck it when you are ready to go public.

3.3 Configure the Cyberdream BBS brute force protection

Go to Appearance - Customize - Cyberdream BBS - Security. Enable brute force protection and set your preferred limits. The defaults are 5 attempts and a 15 minute lockout.

3.4 Configure your color scheme

Go to Appearance - Customize - Cyberdream BBS - Color Scheme. Available schemes are Tales from the Loop, Amber Terminal, Neon Dusk, Deep Forest, NComm 3.05, Directory Opus Light, Directory Opus Dark and Custom.

3.5 Set up BBS stats widgets

Go to Appearance - Widgets and add a Custom HTML widget to the Sidebar area. Use these shortcodes to show live stats:

```
[cd_post_count]                 - number of published posts
[cd_comment_count]              - number of approved comments
[cd_uptime since="YYYY-MM-DD"]  - days since a given date
[cd_last_poll]                  - date of the most recent post
```

4. Automatic Updates

Keeping everything updated is the single most important security measure for WordPress. Most successful WordPress attacks exploit known vulnerabilities in outdated software.

WordPress core

Go to Dashboard - Updates. WordPress enables minor security updates automatically by default. For major updates, click Enable automatic updates for all new versions of WordPress.

Plugins

Go to Plugins - Installed Plugins. For each plugin, click Enable auto-updates. Check your site after updates run in case of compatibility issues.

Themes

Go to Appearance - Themes. Hover over each theme and enable auto-updates.

> Always keep a recent backup before major updates so you can roll back if something breaks.

5. Plugin Philosophy - Less is More

Every plugin you install is a potential attack surface. A single vulnerable plugin can compromise your entire WordPress installation no matter how well everything else is secured. Most WordPress sites that get hacked are compromised through plugins, not WordPress core.

- Install as few plugins as possible. If WordPress core or the theme can do it, use that instead.
- Only install plugins from the official WordPress.org repository or well-known trusted sources.
- Check when the plugin was last updated and how many active installations it has before installing.
- Remove plugins you no longer use. Deactivated plugins can still be exploited.
- Enable automatic updates for all installed plugins.
- Regularly audit your plugins and ask: do I still need this?
- Avoid page builder plugins like Elementor, WPBakery and Divi - they have large attack surfaces.
- Avoid plugins that load external scripts or have lots of dependencies.

6. PHP Version

Old PHP versions stop receiving security patches when they reach end of life. Running an outdated PHP version is a significant security risk.

Check your current PHP version at Tools - Site Health - Info - Server in wp-admin.

On cPanel and shared hosting: look for a PHP Version or MultiPHP Manager option in your control panel and select the latest stable PHP 8.x version available.

> Test your site after upgrading PHP. Some older plugins may not be compatible with newer versions.

7. Cloudflare Setup and Tips

Cloudflare sits in front of your server and provides DDoS protection, a CDN, a WAF (web application firewall) and hides your server's real IP address. Even the free tier provides significant security benefits.

Basic setup

1. Create a Cloudflare account and add your domain.
2. Update your domain's nameservers to the ones Cloudflare provides.
3. Set your DNS records. Make sure the records pointing to your server have the orange cloud icon enabled - this means traffic goes through Cloudflare.
4. Set SSL/TLS encryption mode to Full (strict) under SSL/TLS - Overview.

Recommended settings

- Security - Settings - Security Level: Medium or High

- Security - Settings - Bot Fight Mode: On (free tier)

- SSL/TLS - Edge Certificates - Always Use HTTPS: On

- SSL/TLS - Edge Certificates - Automatic HTTPS Rewrites: On

- Speed - Optimization - Auto Minify: check JS, CSS and HTML

Purging the cache

Cloudflare caches your CSS, JS and HTML aggressively. After updating your theme or any files, go to Caching - Configuration - Purge Cache - Purge Everything to force Cloudflare to fetch fresh files from your server. This is available on the free tier.

> If you update your theme and changes are not showing up, purge the Cloudflare cache first before troubleshooting anything else.

What Cloudflare does not protect against

Cloudflare is not a substitute for keeping WordPress and plugins updated. It also does not protect against attacks that come through your WordPress admin, compromised plugins or weak passwords.

Cloudflare and Basic Auth

Cloudflare passes Basic Auth headers through to your server correctly, so the htpasswd protection on wp-login.php works normally even with Cloudflare active.

8. fail2ban - Server Level Brute Force Protection

fail2ban monitors your server log files and automatically bans IP addresses that show signs of brute force attacks. It works at the firewall level, before requests even reach Apache, Nginx or WordPress.

> fail2ban requires root or sudo access. It is not available on most shared hosting plans. It is most useful on VPS or dedicated server setups.

Installation on Debian or Ubuntu

```
sudo apt install fail2ban
```

Basic WordPress jail - /etc/fail2ban/jail.local

```
[wordpress-auth]
enabled  = true
filter   = wordpress-auth
logpath  = /var/log/apache2/access.log
maxretry = 5
bantime  = 900
findtime = 300
```

Filter file - /etc/fail2ban/filter.d/wordpress-auth.conf

```
[Definition]
failregex = ^<HOST> .* "POST /wp-login.php
ignoreregex =
```

Reload fail2ban

```
sudo systemctl reload fail2ban
```

For Nginx change the logpath to /var/log/nginx/access.log. For Basic Auth protection, use the built-in apache-auth filter.

9. General Recommendations

Use strong unique passwords

Use a password manager and generate random passwords of at least 20 characters. Use different passwords for WordPress, your hosting account, your .htpasswd gate and Cloudflare.

Use HTTPS everywhere

Make sure your SSL certificate is valid and auto-renewing. Cloudflare provides free SSL. Enable Always Use HTTPS in Cloudflare settings.

Keep regular backups

Use your hosting provider's backup solution or export the database regularly. Test that you can actually restore from a backup before you need to.

Monitor WordPress Site Health

Go to Dashboard - Site Health regularly. WordPress will flag security and performance issues there.

Use a separate account for posting

Consider creating a separate WordPress user with the Editor role for writing posts and only using the admin account for administration. This limits the damage if that account is ever compromised.

Limit login access by IP

If you always log in from the same IP address, you can restrict wp-login.php to that IP only in .htaccess. Very effective but not practical if your IP changes frequently.

Do not install themes or plugins from unofficial sources

Only use themes and plugins from wordpress.org or directly from well-known developers. Nulled or pirated plugins and themes almost always contain malware.

Review your site regularly

Log in to your WordPress admin at least once a week. Check for updates, review comments and look at Site Health.