WordPress **Hardening Manuál**

Praktický postup zabezpečenia inštancie WordPress.

OBSAH

Úvod

Tento dokument opisuje základné kroky potrebné pre zvýšenie bezpečnosti vlastnej inštancie redakčného systému WordPress.

Tento návod je vhodný pre nové aj existujúce inštancie.

Typ hostingu

Kapitoly 1, 2 a 4 sú použiteľné bez ohľadu na typ hostingu. **Kapitola 3** (Konfigurácia servera) vyžaduje priamy prístup ku konfiguračným súborom servera, ktoré sú dostupné na VPS alebo dedikovanom hostingu, spravidla nie na zdieľanom hostingu. Kroky vyžadujúce prístup k serveru sú v texte zreteľne označené. Ak používate zdieľaný hosting, obráťte sa na svojho poskytovateľa. Mnohí dokážu nastavenia na úrovni servera aplikovať na požiadanie, alebo ich sprístupňujú cez ovládací panel.

Pred začatím

> ⚠ **PRED AKÝMIKOĽVEK ZMENAMI SI VYTVORTE ZÁLOHU**
>
> Vždy vytvorte úplnú zálohu súborov aj databázy pred vykonaním akýchkoľvek zmien na existujúcej inštalácii. Ak dôjde k problému, záloha je vaša jediná možnosť obnovy.

Čo budete potrebovať

- Prihlasovacie údaje administrátora WordPress
- SFTP klient (napr. FileZilla, WinSCP)
- Prístup k ovládaciemu panelu hostingu alebo k serveru (Kapitola 3)
- Nástroj na správu databázy (phpMyAdmin, DataGrip alebo podobný)

01

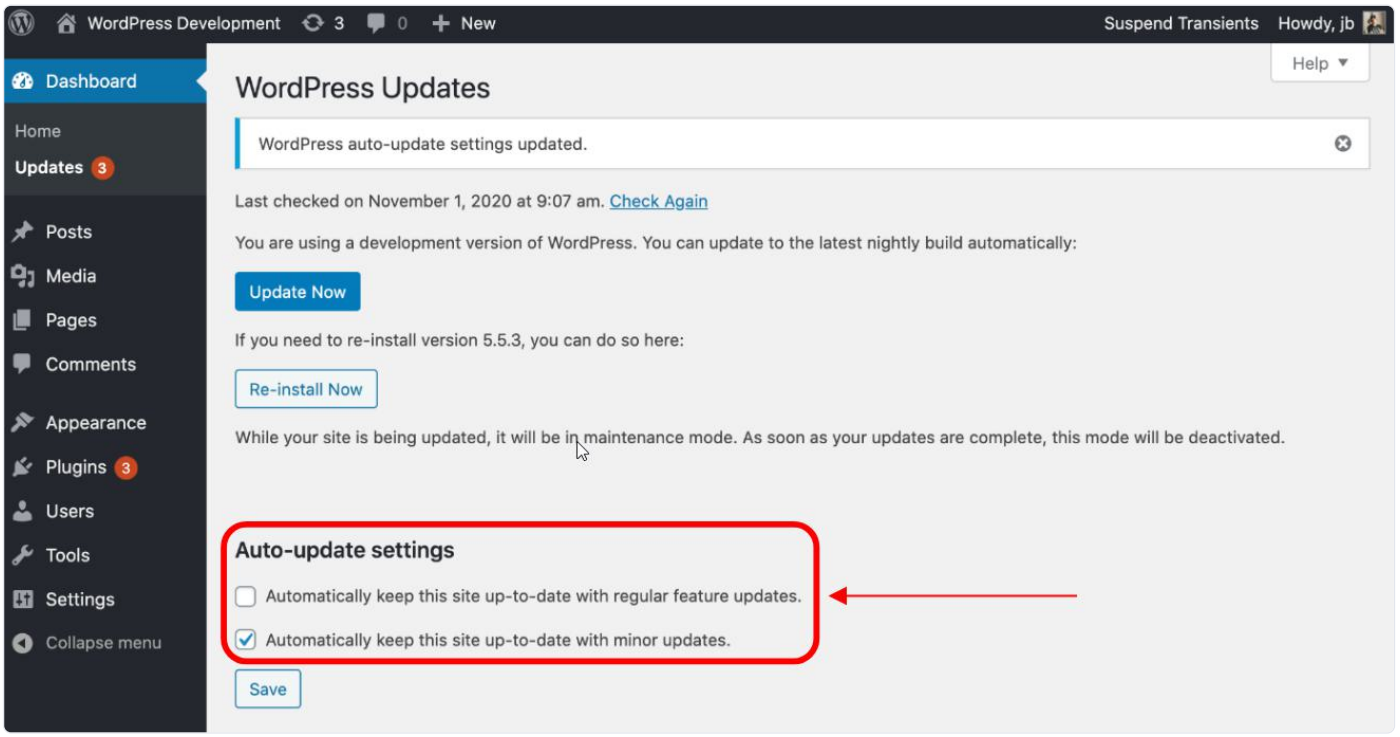
KAPITOLA PRVÁ

Základné zabezpečenie

Kroky s najvyšším dopadom. Vykonajte ich ako prvé. Väčšinou to sú jednorázové zmeny, ktoré výrazne znižujú plochu útoku.

1 Pravidelná aktualizácia

Aktualizácie plnia veľmi dôležitú úlohu. Opravujú chyby v operačných systémoch, softvéroch a aplikáciách, ktoré by mohli byť zneužité. Zastaraný softvér je najčastejším vstupným bodom útočníkov. Je preto nevyhnutné dbať na to, aby bol softvér pravidelne aktualizovaný a mal implementované najnovšie bezpečnostné záplaty.



Od verzie 3.7 je možné zároveň zapnúť možnosť automatickej aktualizácie.

- Aktivujte automatické aktualizácie jadra WordPress (dostupné od verzie 3.7)
- Pravidelne aktualizujte pluginy a témy cez administračné rozhranie
- Ak automatické aktualizácie nie sú možné, aktualizujte manuálne cez SFTP
- Zvážte nahradenie pluginov alebo tém, ktoré neboli aktualizované viac ako dva roky, lebo už nemusia byť naďalej aktívne udržiavané

2 Zabezpečenie súboru wp-config.php

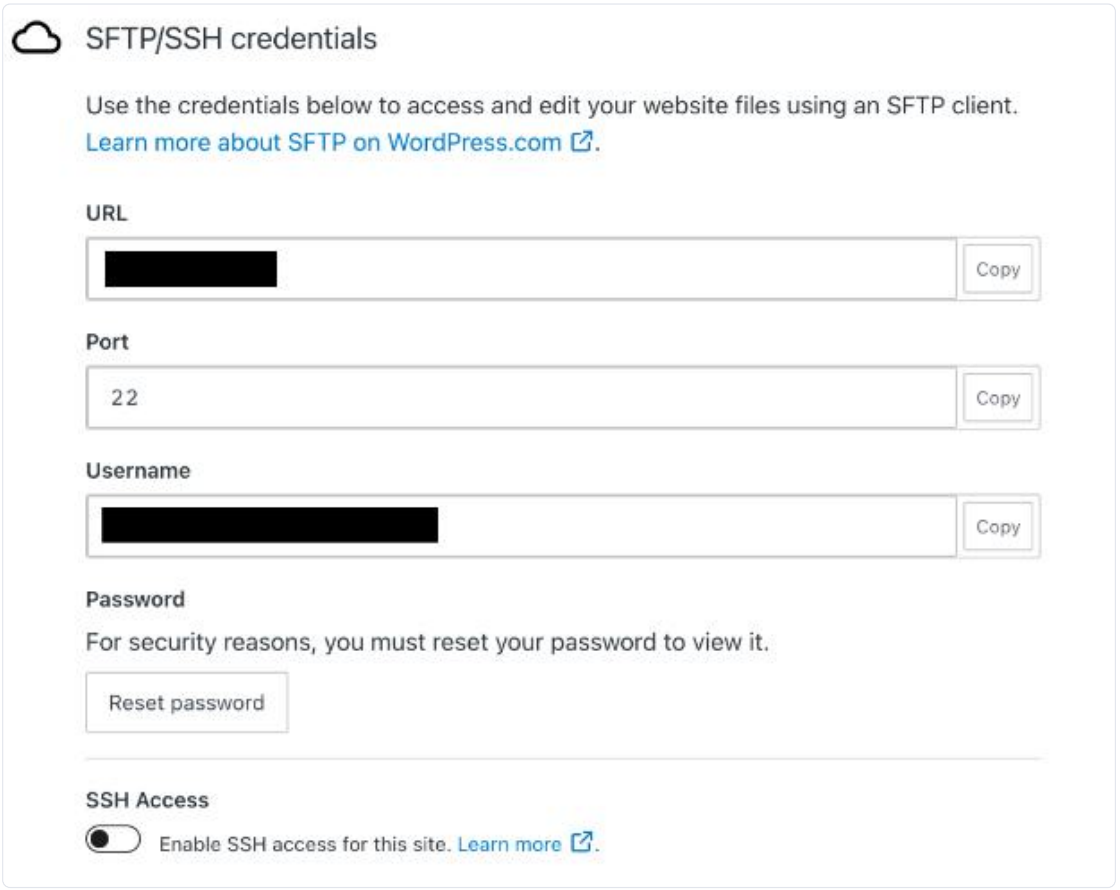
Súbor `wp-config.php` obsahuje prihlasovacie údaje k databáze, bezpečnostné kľúče a základnú konfiguráciu inštalácie. Je to jeden z najcitlivejších súborov vašej inštalácie.

Pripojenie cez SFTP

Pripojenie pomocou SFTP/SSH namiesto FTP zaručuje šifrovanie prenášaných dát.

Prihlasovacie údaje nájdete v ovládacom paneli hostingu:

`Dashboard → Settings → Hosting Configuration → SFTP/SSH credentials`



Nastavenia SFTP pripojenia.

Po úspešnom vygenerovaní príkazu pre vytvorenie SSH spojenia skopírujte príkaz do príkazového riadku, čím vytvoríte aktívne spojenie:

```bash
ssh uzivatel@vasa-stranka.sk -p 22
```

Presunutie súboru nad koreňový adresár

Predvolene sa `wp-config.php` nachádza v koreňovom adresári WordPress, ktorý je verejne prístupný. Presunutím o jednu úroveň vyššie ho umiestnite mimo webovo dostupného priečinka. WordPress ho na novom mieste automaticky rozpozná, nie je potrebná žiadna dodatočná konfigurácia.

Nastavenie oprávnení súboru

```bash
chmod 400 /cesta/k/wp-config.php
```

Použite `400` (čítanie len pre vlastníka) alebo `440` (čítanie pre vlastníka a skupinu). Oprávnenia uvoľnite len na čas nevyhnutný pre úpravy a ihneď ich obnovte.

> ⚠️ **NIKDY NEVKLADAJTE WP-CONFIG.PHP DO VERZIONOVACIEHO SYSTÉMU**
>
> Ak sa vaša inštalácia WordPress nachádza v Git repozitári, uistite sa, že `wp-config.php` je uvedený v súbore `.gitignore`. Súbor obsahuje prihlasovacie údaje k databáze a bezpečnostné kľúče, ktoré nesmú byť verejne prístupné.

Obsah súboru wp-config.php

Pridajte nasledujúci kód do súboru `wp-config.php` pred riadok *„That's all, stop editing!"*:

```php
// Vynútenie HTTPS pre administračné rozhranie
define('FORCE_SSL_ADMIN', true);

// Deaktivácia vstavaného editora súborov tém a pluginov
define('DISALLOW_FILE_EDIT', true);

// Na produkčnom prostredí nikdy nezobrazovať chyby
define('WP_DEBUG', false);
define('WP_DEBUG_DISPLAY', false);
define('WP_DEBUG_LOG', false);

// Obmedzenie počtu uložených revízií príspevkov
define('WP_POST_REVISIONS', 5);

// Skrátenie doby uchovávania odpadkového koša
define('EMPTY_TRASH_DAYS', 7);
```

> 📋 **DISALLOW_FILE_MODS**
>
> Nastavenie `define('DISALLOW_FILE_MODS', true)` zablokuje aj inštaláciu a aktualizáciu pluginov a tém, vrátane automatických aktualizácií. Použite ho len v prípade, že máte spoľahlivý proces manuálnych aktualizácií cez SFTP.

3 Zmena databázového prefixu

WordPress predvolene používa prefix `wp_` pre všetky databázové tabuľky. Predvolený prefix uľahčuje útočníkom realizovať SQL injection útoky. Zmenou na nepredvídateľný reťazec môžete eliminovať aspoň

časť plošných, automatizovaných útokov.

Pri novej inštalácii

V súbore `wp-config.php` zmeňte:

```php
$table_prefix = 'wp_';
```

Napríklad na:

```php
$table_prefix = 'wpsk_x9k2_';
```

Pri existujúcej inštalácii

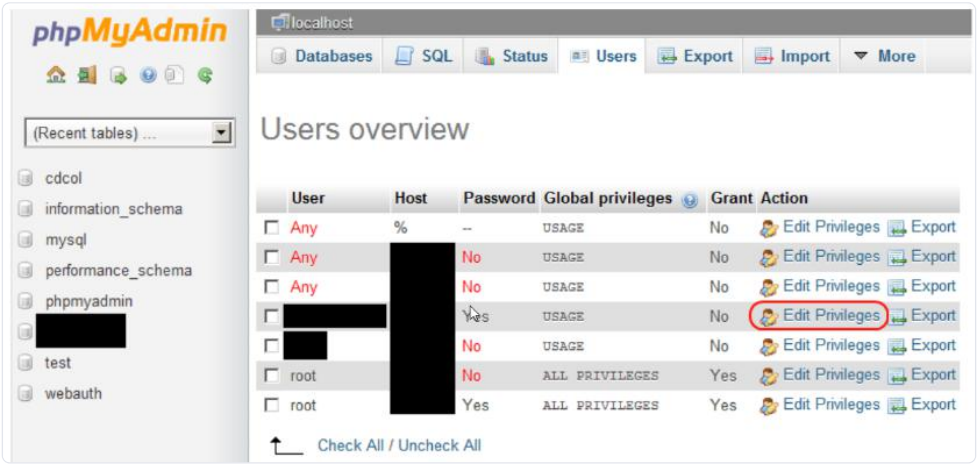
> ⚠️ **PRED ZMENOU SI VYTVORTE ZÁLOHU DATABÁZY**
>
> Zmena prefixu na existujúcej inštalácii vyžaduje premenovanie tabuliek a aktualizáciu záznamov. Chyba môže spôsobiť nefunkčnosť stránky. Pred začatím vytvorte úplnú zálohu databázy.

Použite plugin **Change Table Prefix**, alebo postupujte manuálne:

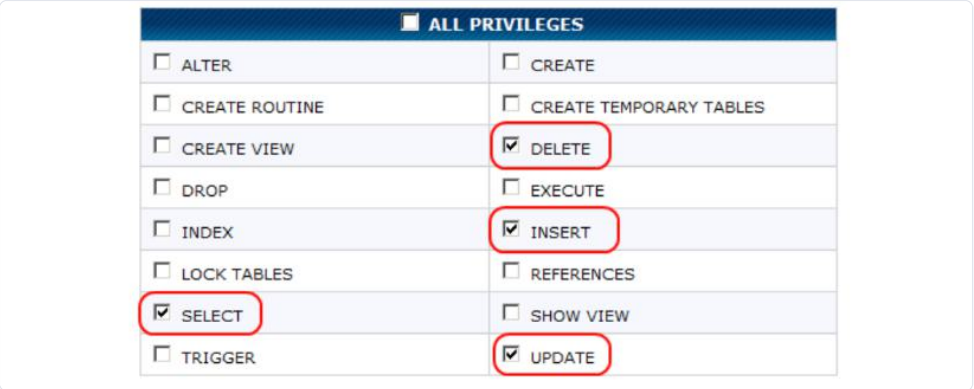
1. Premenujte všetky tabuľky v databáze na nový prefix
2. Aktualizujte hodnoty `option_name` v tabuľke `options`, kde hodnota začína reťazcom `wp_`
3. Rovnako aktualizujte hodnoty `meta_key` v tabuľke `usermeta`
4. Zmeňte hodnotu `$table_prefix` v súbore `wp-config.php`

4 Obmedzenie práv databázového používateľa

Databázový používateľ, cez ktorého sa WordPress pripája, potrebuje pre bežnú prevádzku len štyri oprávnenia: `SELECT`, `INSERT`, `UPDATE` a `DELETE`. Všetky ostatné vrátane `DROP`, `ALTER`, `CREATE` a `FILE` by mali byť odobrané. Vhodným nástrojom je phpMyAdmin, ovládací panel hostingu alebo databázový klient ako DataGrip.



Nastavenia databázy v nástroji phpMyAdmin



Nastavenia databázových privilégií v nástroji phpMyAdmin

Viac o nástroji PhpMyAdmin na https://docs.phpmyadmin.net/en/latest/privileges.html

> **TIP**
>
> Niektoré pluginy vyžadujú oprávnenie `CREATE` pri prvotnom nastavení. Udeľte ho dočasne a po dokončení inštalácie ho opäť odoberte.

5 Šifrovanie databázového spojenia

Väčšina moderných spravovaných hostingov (Azure, AWS RDS, DigitalOcean, a iné) dnes vynucuje TLS šifrovanie databázového spojenia automaticky na úrovni infraštruktúry. Pred akýmikoľvek zmenami overte u svojho poskytovateľa, či je TLS už aktívne. V mnohých prípadoch nie je potrebné nič konfigurovať.

> **NEPRIDÁVAJTE MYSQLI_CLIENT_SSL BEZ OVERENIA**
>
> Jednoduchý konštantný zápis `MYSQLI_CLIENT_SSL` bez verifikácie certifikátu môže na mnohých moderných hostingoch spôsobiť zlyhanie pripojenia k databáze. Ak váš poskytovateľ TLS nevynucuje automaticky a chcete ho aktivovať manuálne, použite verziu s overením certifikátu a cestu k CA bundle získajte od poskytovateľa:

```php
define('MYSQL_CLIENT_FLAGS', MYSQLI_CLIENT_SSL | MYSQLI_CLIENT_SSL_VERIFY_SERVER_CERT);
define('MYSQL_SSL_CA', '/cesta/k/ca.pem');
```

> **NAJLEPŠÍ POSTUP**
>
> Kontaktujte svojho poskytovateľa hostingu a opýtajte sa, či je TLS na databázovom spojení už aktívne. Väčšina kvalitných poskytovateľov to potvrdí, a ak áno, žiadna ďalšia konfigurácia nie je potrebná.

6 Implementácia súborových práv

V rámci používania WordPress je vhodné čo najviac obmedziť súborové práva na čítanie, zápis a spustenie. Príliš voľné súborové oprávnenia sú častou chybou v konfigurácii. Použite nasledujúce hodnoty ako základ:

Cieľ	Oprávnenie	Poznámka
Adresáre	`755`	Vlastník môže zapisovať, ostatní čítať a spúšťať
Súbory	`644`	Vlastník môže zapisovať, ostatní len čítať
wp-config.php	`440`	Čo najprísnejšie obmedzenie
.htaccess	`644`	WordPress príležitostne potrebuje zápis

```bash
find /cesta/k/wordpress/ -type d -exec chmod 755 {} \;
find /cesta/k/wordpress/ -type f -exec chmod 644 {} \;
```

> ⚠️ **NIKDY NEPOUŽÍVAJTE OPRÁVNENIE 777**
>
> Oprávnenie `777` udeľuje plný prístup všetkým používateľom. V produkčnej inštalácii WordPress ho nikdy nepoužívajte na žiadnom súbore ani adresári.

7 Deaktivácia editora súborov v administrácii

WordPress umožňuje editovať súbory pluginov a tém priamo z administračného panela. Táto funkcia predstavuje bezpečnostné riziko. Ak útočník získa prístup administrátora, editor sa stáva priamou cestou k spusteniu ľubovoľného kódu na serveri. Príslušné konštanty sú súčasťou zoznamu v kapitole 2.

```php
// Zakáže prístup k Appearance > Theme Editor a Plugins > Plugin Editor
define('DISALLOW_FILE_EDIT', true);
```

8 Deaktivácia funkcie XML-RPC

XML-RPC je zastaraná funkcia, ktorá umožňuje prenos údajov. Protokol HTTP funguje ako prenosový mechanizmus a XML ako kódovací mechanizmus. Táto funkcia je zastaralá, plne ju nahradilo REST API, avšak zostáva vo WordPress predvolene zapnutá. Bola opakovane zneužívaná pri brute-force a DDoS útokoch. Deaktivujte ju, pokiaľ ju nevyžaduje konkrétna aplikácia.

Apache

```apache
<files xmlrpc.php>
  order deny,allow
  deny from all
</files>
```

NGINX

```nginx
location = /xmlrpc.php {
    deny all;
}
```

> ⚠️ **NIEKTORÉ PLUGINY VYUŽÍVAJÚ XML-RPC**
>
> Po zablokovaní XML-RPC overte funkčnosť stránky. Pluginy ako **Jetpack** a staršie verzie mobilnej aplikácie WordPress komunikujú práve cez XML-RPC a po zablokovaní prestanú fungovať. Ak tieto nástroje aktívne využívate, zvážte selektívne blokovanie konkrétnych metód namiesto úplného zablokovania súboru.

9 Zabezpečenie úložiska nahraných súborov

Priečinok `wp-content/uploads/` môže predstavovať bezpečnostné riziko, ak nie je správne zabezpečený. Je zapisovateľný zo strany WordPress a zároveň verejne prístupný. Odporúčaná ochrana je zablokovanie spúšťania PHP súborov v tomto adresári.

Apache

Vytvorte súbor `.htaccess` v priečinku `wp-content/uploads/`:

```apache
<Files *.php>
    deny from all
</Files>
```

NGINX

```nginx
location ~* /wp-content/uploads/.*\.php$ {
    deny all;
}
```

Obmedzenie povolených typov súborov

Odporúčame povoliť iba niektoré typy súborov. Úpravou `functions.php` si tieto nastavenia môžete prispôsobiť podľa potrieb vašej inštancie WordPress:

```php
function restrict_upload_mimes($mimes) {
    return [
        'jpg|jpeg|jpe' => 'image/jpeg',
        'png'          => 'image/png',
        'gif'          => 'image/gif',
        'webp'         => 'image/webp',
        'pdf'          => 'application/pdf',
        'docx'         => 'application/vnd.openxmlformats-officedocument.wordprocessingml.document',
        'xlsx'         => 'application/vnd.openxmlformats-officedocument.spreadsheetml.sheet',
    ];
}
add_filter('upload_mimes', 'restrict_upload_mimes');
```

02

KAPITOLA DRUHÁ

Autentifikácia

Kontrola toho, kto môže vstúpiť na stránku a čo môže po prihlásení vykonávať. Väčšina útokov začína práve pri autentifikácii, preto tieto kroky uzatvárajú najčastejšie vstupné body.

10 Používanie silných hesiel

Tvorba prístupového hesla k WordPress by mala podliehať rovnakým pravidlám ako tvorba hesiel k iným citlivým systémom. Slabé heslá zostávajú jednou z najčastejšie zneužívaných zraniteľností. Silné heslá vyžadujte pre všetky účty, nielen pre administrátorov.

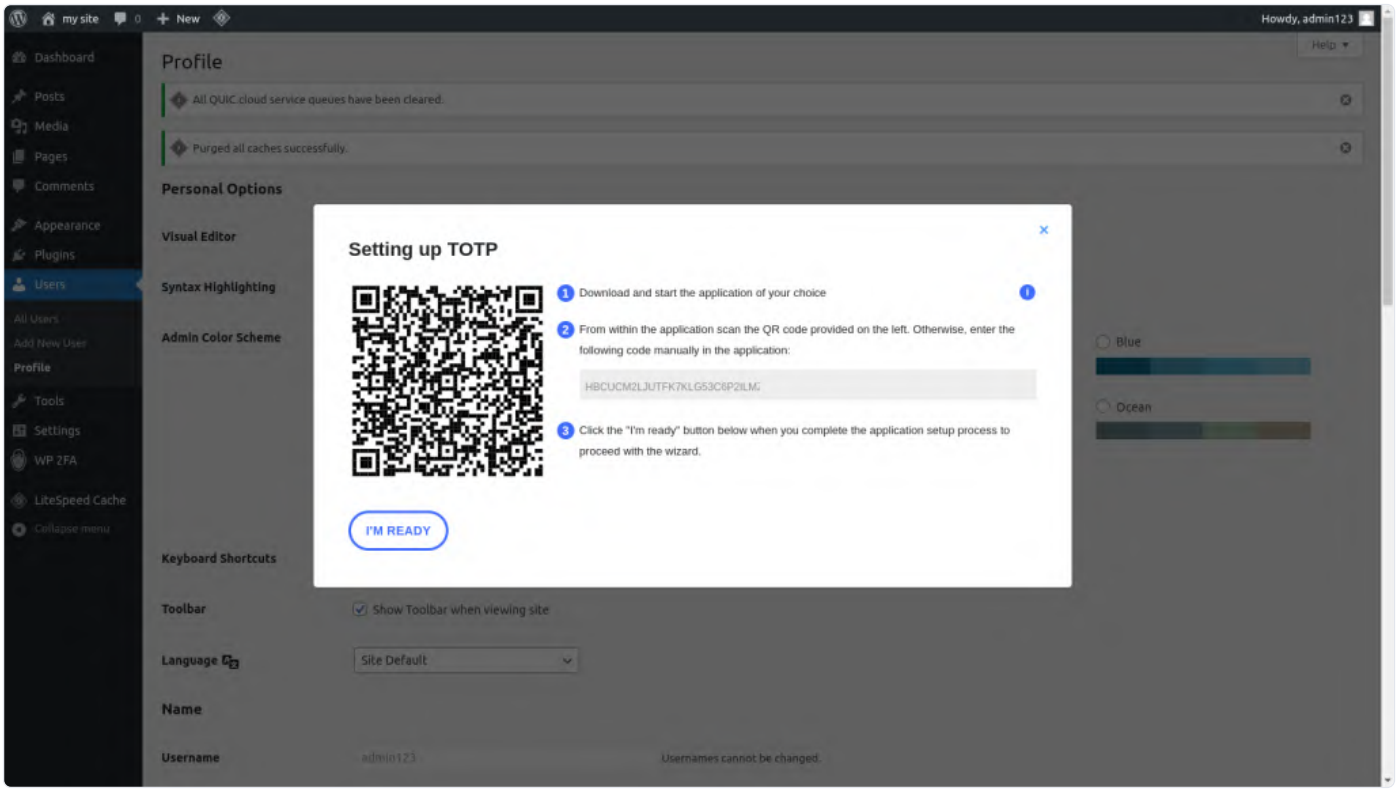
Mali by sme sa vyhýbať:

- Krátkym heslám
- Permutáciám osobných údajov
- Používaniu slovníkových slov
- Použitiu výhradne numerických alebo alfabetických hesiel
- Opakovanému použitiu hesiel z iných služieb

Na generovanie a uchovávanie jedinečných prihlasovacích údajov používajte správcu hesiel.

11 Viacfaktorová autentifikácia (MFA)

Viacfaktorová autentifikácia je veľmi dôležitý element zabezpečenia autentifikácie používateľov a predovšetkým administrátorov. Dnes je skôr nutnosťou ako „extra" opatrením. MFA vyžaduje od používateľov overenie identity druhým faktorom okrem hesla a pre všetky administrátorské a redaktorské účty by mala byť povinná.



Nastavenia MFA v rámci WordPress admin panelu.

Niektoré overené WordPress rozšírenia na implementáciu MFA:

- **Two Factor** - oficiálny plugin WordPress.org, pravidelne udržiavaný
- **Wordfence 2FA** - súčasť balíka Wordfence Security
- **miniOrange 2FA** - podporuje širokú škálu autentifikačných metód
- **Passkeys (WebAuthn)** - podpora hardvérových kľúčov pre prostredia s vyššími nárokmi na bezpečnosť

12 Obmedzenie počtu pokusov o prihlásenie

WordPress predvolene neobmedzuje počet neúspešných pokusov o prihlásenie, čo umožňuje automatizovaným brute-force útokom prebiehať bez obmedzenia. Inštaláciou pluginu na obmedzenie prihlásení pridáme blokovanie po konfigurovateľnom počte neúspešných pokusov.

Odporúčaný plugin: **Limit Login Attempts Reloaded**. Nastavte blokovanie po 5 pokusoch na dobu 20 minút s postupne sa predlžujúcou dobou blokovania pri opakovaných pokusoch.

13 Ochrana proti enumerácii používateľov

WordPress štandardne umožňuje zistiť existujúce používateľské mená, čo uľahčuje brute-force útoky. Odhalenie prebieha dvoma spôsobmi, cez URL parametre author archive a cez REST API.

Zablokovanie enumerácie cez URL

Pridajte do súboru `.htaccess`:

```apache
# Block User ID Phishing Requests
<IfModule mod_rewrite.c>
    RewriteCond %{QUERY_STRING} ^author=([0-9]*)
    RewriteRule .* - [F]
</IfModule>
```

Zablokovanie enumerácie cez REST API

Pridajte do súboru `functions.php`:

```php
// Disable user enumeration via REST API
add_filter('rest_endpoints', function($endpoints) {
    if (!is_user_logged_in()) {
        if (isset($endpoints['/wp/v2/users']))
            unset($endpoints['/wp/v2/users']);
        if (isset($endpoints['/wp/v2/users/(?P<id>[\d]+)']))
            unset($endpoints['/wp/v2/users/(?P<id>[\d]+)']);
    }
    return $endpoints;
});
```

14 Konfigurácia používateľských práv

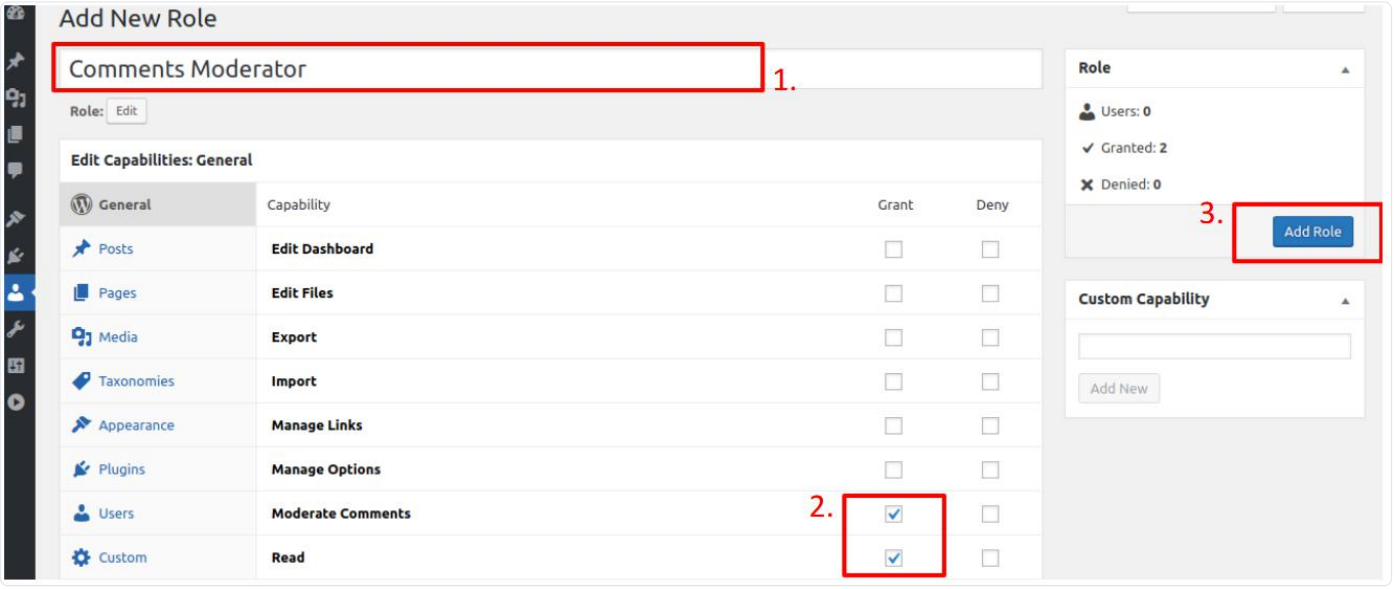
Vhodná konfigurácia používateľských rolí umožňuje jednoznačne definovať, čo môže, resp. nemôže daná skupina používateľov vykonávať. To môže v prípade incidentu výrazne znížiť škody. Každému používateľovi prideľte len minimálnu rolu potrebnú pre jeho prácu.

Rola	Oprávnenia
Super admin	Úplné administrátorské práva v rámci siete stránok
Administrátor	Úplné administrátorské práva v rámci jednej stránky
Editor	Uverejňovanie a úprava článkov všetkých používateľov
Autor	Uverejňovanie a úprava vlastných článkov
Prispievateľ	Úprava vlastných článkov bez možnosti uverejnenia
Odberateľ	Čítanie obsahu a úprava vlastného profilu

Pravidelne kontrolujte zoznam administrátorských účtov. Neaktívne účty odstráňte alebo preraďte na nižšiu rolu. Minimalizujte počet používateľov s rolou administrátora.

WordPress taktiež poskytuje možnosť vytvorenia vlastných rolí:

1. Názov role
2. Oprávnenia role
3. Tlačidlo na vytvorenie role

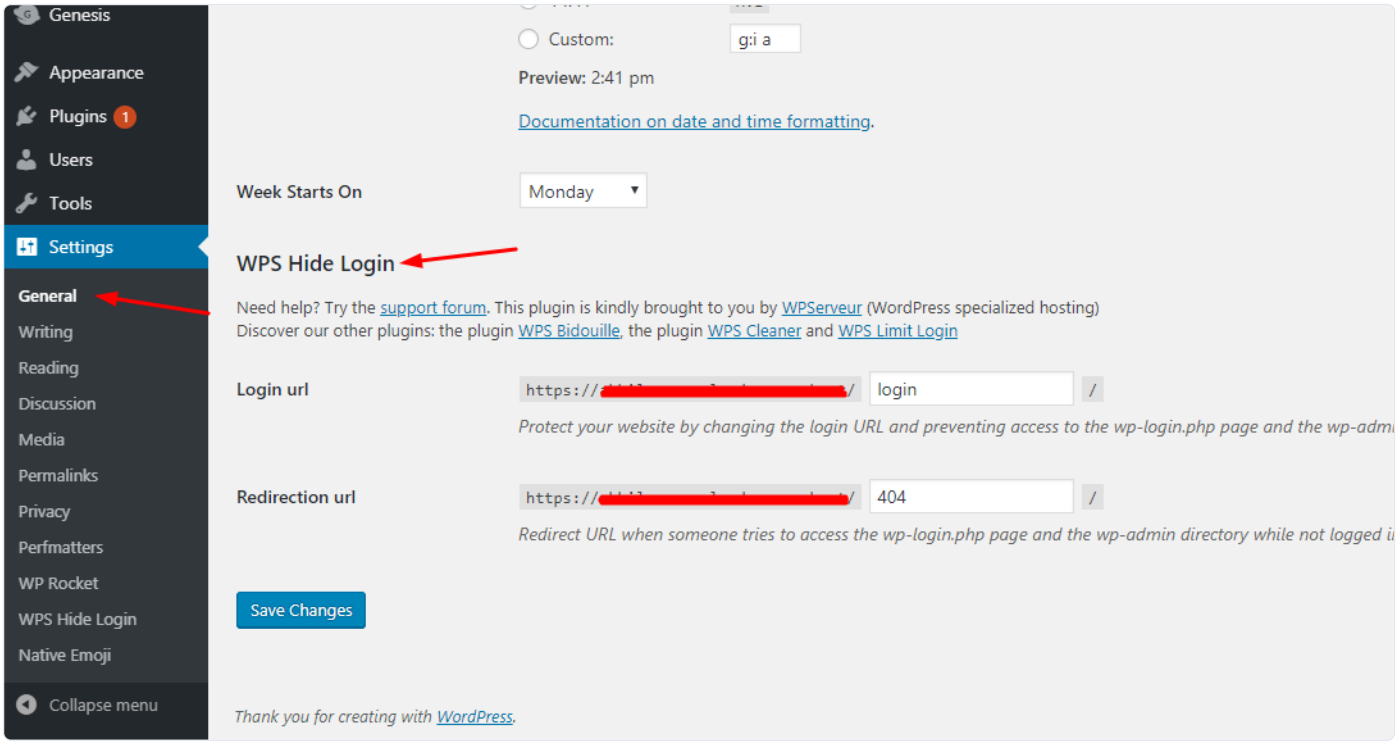


WordPress umožňuje vytvárať aj špeciálne role na mieru.

15 Ochrana prístupu k wp-admin

Pomocou nasledujúcich nastavení si vynútite dodatočné overenie pri prístupe do administrácie a znížite riziko automatizovaných útokov na systém.

- **Obmedzenie prístupu podľa IP adresy** - ak sa administrátori pripájajú z pevných IP adries, povolte prístup len z týchto adries na úrovni servera
- **Premenovanie wp-login.php** - použite plugin *WPS Hide Login* na presun prihlasovacej stránky na neštandardnú URL adresu, tým zabránite automatizovaným útokom na systém
- **Vynútenie HTTPS** - aktivujte administráciu prostredníctvom HTTPS a uistite sa, že konštanta `FORCE_SSL_ADMIN` je správne nastavená vo `wp-config.php` (Kapitola 2)
- **HTTP autentifikácia** - pridajte heslovú ochranu na úrovni servera ako druhú vrstvu pred samotným prihlasovacím formulárom WordPress



Nastavenia ochrany prístupu do administrácie pomocou WPS Hide Login rozšírenia.

16 Ochrana formulárov a komentárov pred botmi

Kontaktné formuláre a sekcie komentárov sú častým cieľom spam botov a injection útokov.

- Pridajte CAPTCHA ku všetkým verejne dostupným formulárom. Odporúčané možnosti: **Cloudflare Turnstile** (rešpektuje súkromie), **hCaptcha**, alebo **Google reCAPTCHA v3**
- Ak stránka komentáre nevyužíva, úplne ich deaktivujte: *Nastavenia > Diskusia > zrušte zaškrtnutie „Povoliť ľuďom pridávať komentáre"*
- Väčšina bezpečnostných pluginov (Wordfence, MalCare) obsahuje ochranu pred botmi. Po inštalácii overte, či je aktívna

03

KAPITOLA TRETIA

Konfigurácia servera

Vyžaduje prístup ku konfiguračným súborom servera. Nie je dostupné na všetkých zdieľaných hostingoch. Dostupnosť si overte u svojho poskytovateľa, alebo túto kapitolu preskočte a vráťte sa k nej, až keď budete mať potrebný prístup.

17 Konfigurácia SSL/TLS

Správna konfigurácia SSL/TLS je kritická. Používajte výhradne TLS 1.2 a TLS 1.3. Staršie verzie (TLS 1.0, 1.1, SSL 3.0) majú známe zraniteľnosti a musia byť explicitne zakázané.

Apache - ssl.conf

APACHE

```
SSLProtocol -all +TLSv1.2 +TLSv1.3
SSLCipherSuite HIGH:!aNULL:!MD5:!3DES
SSLHonorCipherOrder on
```

NGINX - nginx.conf

NGINX

```
ssl_protocols TLSv1.2 TLSv1.3;
ssl_ciphers 'ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256';
ssl_prefer_server_ciphers on;
```

✅ **OVERENIE KONFIGURÁCIE**

Použite SSL Labs test na adrese **ssllabs.com/ssltest**. Cieľom je hodnotenie A alebo A+. Na zdieľanom hostingu využite vstavaný nástroj Let's Encrypt od poskytovateľa namiesto manuálnej konfigurácie TLS.

18 Konfigurácia bezpečnostných HTTP hlavičiek

Bezpečnostné HTTP hlavičky poskytujú dodatočnú vrstvu ochrany pred bežnými webovými útokmi. Informujú prehliadač, ako má narábať s obsahom vašej stránky. Správnosť nastavenia môžete overiť pomocou online nástrojov **securityheaders.com** alebo **observatory.mozilla.org**.

Hlavička	Odporúčaná hodnota
`Strict-Transport-Security`	`max-age=63072000; includeSubDomains; preload`
`X-Frame-Options`	`DENY`
`X-Content-Type-Options`	`nosniff`
`Referrer-Policy`	`strict-origin-when-cross-origin`
`Permissions-Policy`	`geolocation=(), microphone=(), camera=(), payment=()`
`Cross-Origin-Opener-Policy`	`same-origin`
`Cross-Origin-Embedder-Policy`	`require-corp`
`Cross-Origin-Resource-Policy`	`same-origin`

> ⚠️ **HLAVIČKU X-XSS-PROTECTION NEPOUŽÍVAJTE**
>
> Táto hlavička je zastaraná, moderné prehliadače ju nepodporujú a v niektorých prípadoch môže sama o sebe zraniteľnosti spôsobiť. Vynechajte ju úplne, alebo jej priraďte hodnotu `0`.

Content-Security-Policy

V prípade hlavičky Content-Security-Policy začnite s režimom `Content-Security-Policy-Report-Only`. Ide o identickú hlavičku, ktorá porušenia len zaznamenáva do konzoly prehliadača bez toho, aby čokoľvek zablokovala. Používajte ju 2–4 týždne na produkčnom prostredí, sledujte výstupy v konzole a až potom prejdite na ostré `Content-Security-Policy`.

APACHE

```
# Fáza 1 - len sledovanie, nič sa neblokuje
Header set Content-Security-Policy-Report-Only "default-src 'self'; script-src 'self' 'unsafe-inline'; style-src 'self' 'unsafe-inline'; img-src 'self' data: https: blob:; connect-src 'self' https:;"

# Fáza 2 - po otestovaní nahraďte ostrou verziou
Header set Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline'; style-src 'self' 'unsafe-inline'; img-src 'self' data: https: blob:; connect-src 'self' https:;"
```

> ⚠️ **'UNSAFE-INLINE' A 'UNSAFE-EVAL' VÝRAZNE OSLABUJÚ CSP**
>
> Tieto parametre sú často nevyhnutné pre správne fungovanie WordPress pluginov, avšak značne znižujú ochranu pred XSS útokmi. Cieľom je ich postupné odstránenie. Hodnotu `'unsafe-eval'` skúste odstrániť ako prvú. Mnohé moderné WordPress stránky ju nevyžadujú. Pluginy ako **WP Content Security Policy** alebo **NinjaFirewall** dokážu generovať nonce-based CSP automaticky, čo je výrazne bezpečnejší prístup.

Apache - .htaccess

```apache
# Enable HSTS
Header always set Strict-Transport-Security "max-age=31536000; includeSubDomains"
Header always set X-Frame-Options "DENY"
Header always set X-Content-Type-Options "nosniff"
Header always set Referrer-Policy "strict-origin-when-cross-origin"
Header always set Permissions-Policy "geolocation=(), microphone=(), camera=()"
Header set Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline' 'unsafe-eval'; style-src 'self' 'unsafe-inline'; img-src 'self' data: https:; font-src 'self' data:; connect-src 'self';"
```

NGINX - nginx.conf

```nginx
add_header Strict-Transport-Security "max-age=31536000; includeSubDomains" always;
add_header X-Frame-Options "DENY" always;
add_header X-Content-Type-Options "nosniff" always;
add_header Referrer-Policy "strict-origin-when-cross-origin" always;
add_header Permissions-Policy "geolocation=(), microphone=(), camera=()" always;
add_header Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline' 'unsafe-eval'; style-src 'self' 'unsafe-inline'; img-src 'self' data: https:; connect-src 'self';" always;
```

19 Deaktivácia voľného listovania adresárov

Zabráňte zobrazeniu obsahu priečinkov pri neprítomnosti indexového súboru.

Apache

Pridajte do `.htaccess` v koreňovom priečinku:

```apache
Options -Indexes
```

NGINX

V konfiguračnom súbore overte, že je nastavené:

```nginx
autoindex off;
```

> ✅ **OVERENIE NASTAVENIA**
>
> Skúste pristúpiť priamo na adresu `https://vasa-stranka.sk/wp-content/uploads/` . Mali by ste vidieť chybu 403 Forbidden, nie zoznam súborov.

20 Zabezpečenie adresára wp-includes

Vložením tohto kódu do súboru `.htaccess` **mimo** bloku ohraničeného reťazcami `# BEGIN WordPress` a `# END WordPress` zablokujeme skripty pre používateľov, ktorí nemajú oprávnenia k ich použitiu:

```apache
<IfModule mod_rewrite.c>
    RewriteEngine On
    RewriteBase /
    RewriteRule ^wp-admin/includes/ - [F,L]
    RewriteRule !^wp-includes/ - [S=3]
    RewriteRule ^wp-includes/[^/]+\.php$ - [F,L]
    RewriteRule ^wp-includes/js/tinymce/langs/.+\.php - [F,L]
    RewriteRule ^wp-includes/theme-compat/ - [F,L]
</IfModule>
```

21 Vypnutie debug režimu v produkčnom prostredí

Debug režim by nemal byť nikdy aktívny na produkčnom prostredí. Odhaľuje citlivé informácie o štruktúre aplikácie vrátane ciest k súborom a databázových dotazov. Uistite sa, že v súbore `wp-config.php` sú nastavené tieto konštanty (sú súčasťou zoznamu v kapitole 2):

```php
define('WP_DEBUG', false);
define('WP_DEBUG_DISPLAY', false);
define('WP_DEBUG_LOG', false);
```

Navyše zakážte zobrazovanie PHP chýb na úrovni servera. Pridajte do `.htaccess`:

```apache
php_flag display_errors off
```

Ak je potrebné logovanie chýb pre účely ladenia, zabezpečte súbor denníka logov pred verejným prístupom:

```apache
# V súbore wp-content/.htaccess
<Files debug.log>
    Order allow,deny
    Deny from all
</Files>
```

04

KAPITOLA ŠTVRTÁ

Priebežná ochrana

Nastavíte raz, funguje nepretržite. Tieto nástroje a postupy zabezpečujú aktívnu obranu voči hrozbám, ktoré sa objavia po dokončení počiatočného zabezpečenia.

22 Inštalácia bezpečnostného pluginu

Bezpečnostné moduly dostupné pre WordPress poskytujú možnosti dodatočného zabezpečenia. Kvalitný bezpečnostný plugin sústreďuje viacero ochranných funkcií na jednom mieste: skenovanie malvéru, monitorovanie integrity súborov, ochranu prihlásenia a upozornenia v reálnom čase. Nainštalujte jeden plugin a overte, že jeho kľúčové funkcie sú aktívne. Neinštalujte viac ako jeden bezpečnostný plugin, môžu sa vzájomne blokovať.

- **Wordfence Security** - komplexné riešenie zahŕňajúce WAF a 2FA; bezplatná verzia postačuje pre väčšinu menších organizácií
- **Sucuri Security** - poskytuje bezpečnostný audit, kontrolu integrity súborov, skenovanie škodlivého kódu a hardening stránky
- **MalCare** - spravovaný prístup s minimálnou potrebnou konfiguráciou

23 Použitie nástroja Web Application Firewall (WAF)

Implementácia WAF umožňuje zablokovanie útočníka ešte pred navštívením stránky. WAF kontroluje IP adresy návštevníkov a prichádzajúcu komunikáciu, pričom blokuje škodlivé požiadavky skôr, ako sa dostanú k WordPressu.

Úroveň	Možnosti	Vhodné pre
Cloudová WAF	Cloudflare, Sucuri Cloud, AWS WAF	Blokovanie hrozieb pred kontaktom zo serverom. Odporúčané pre vládne stránky
Na úrovni servera	ModSecurity (Apache), NAXSI (NGINX)	Plná kontrola pravidiel; vyžaduje prístup k serveru
Plugin	Wordfence, MalCare	Najjednoduchšia inštalácia; funguje na zdieľanom hostingu

Pre väčšinu menších organizácií predstavuje kombinácia WAF cez Wordfence plugin a bezplatnej verzie Cloudflare silný základ s minimálnou zložitosťou.

24 Aktivácia logovania a jeho pravidelná kontrola

Pomocou logovania získame prehľad o dianí na stránke. Logovanie má zmysel len vtedy, keď ho niekto sleduje. Aktivujte ho a vytvorte si návyk pravidelnej kontroly. Aj krátka týždenná kontrola odhalí väčšinu problémov včas.

Odporúčaný plugin: **WP Security Audit Log**. Nakonfigurujte logovanie nasledujúcich udalostí:

- Neúspešné prihlásenia
- Zmeny administrátorských účtov
- Inštalácia alebo odstraňovanie pluginov a tém
- Zmeny súborov
- Prístup k citlivým súborom

25 Deaktivácia nepoužívaných pluginov a tém

Ak sa vo vašej inštancií WordPress nachádzajú rozšírenia (pluginy), ktorých využitie je zriedkavé, odporúčame ich deaktivovať a držať sa pravidla: povoliť len nutné. Každý nainštalovaný plugin a téma, aj neaktívne, predstavujú kód, ktorý môže obsahovať zraniteľnosti.

- Nepoužívané pluginy deaktivujte a **odstráňte**. Samotná deaktivácia nestačí, pretože súbory zostávajú na serveri
- Odstráňte predvolené témy, ktoré nepoužívate; jednu ponechajte ako zálohu pre prípad poruchy aktívnej témy, ostatné zmažte
- Pravidlo: *čo nepotrebujete, odstráňte*

05

KAPITOLA PIATA

Údržba

Tieto pravidelné návyky udržujú vašu stránku v bezpečnom stave a zabezpečujú rýchlu obnovu v prípade incidentu.

26 Stratégia zálohovania a obnovy

Pravidelné zálohy sú kľúčové pre rýchlu obnovu po bezpečnostnom incidente. Záloha, ktorú ste nikdy neotestovali, nie je záloha. Nastavte automatické zálohovanie, uložte zálohy bezpečne a overte, že proces obnovy funguje, skôr než ho budete potrebovať.

Čo zálohovať

- Súbory WordPress, najmä `wp-content/`, `wp-config.php` a `.htaccess`
- Kompletnú databázu
- Konfiguračné súbory servera, ak sú dostupné

Frekvencia zálohovania

- Denné automatické zálohy pre produkčné prostredie
- Zálohy pred každou väčšou zmenou
- Uchovávanie minimálne 30 dní histórie záloh

Umiestnenie záloh

- Mimo webového servera, nikdy neukladajte jedinú zálohu na rovnakom serveri ako stránku
- Testovanie obnovy minimálne raz štvrťročne

Odporúčané nástroje

- **UpdraftPlus** - automatické zálohy do cloudového úložiska (Google Drive, S3, Dropbox)
- **BackWPup** - flexibilné plánovanie a možnosti uloženia
- Manuálne: stiahnutie súborov cez SFTP + `mysqldump` pre databázu

> ✅ **TESTOVANIE OBNOVY MINIMÁLNE RAZ ŠTVRŤROČNE**
>
> Vykonajte úplnú obnovu do lokálneho alebo testovacieho prostredia aspoň raz za štvrťrok. Postup zdokumentujte, aby ho ktokoľvek z vášho tímu mohol vykonať aj pod tlakom.

27 Plán pravidelného auditu

V záujme bezpečnosti je vhodné vykonávať rôzne bezpečnostné audity na pravidelnej báze, ideálne v rámci systematického plánu.

Denné / týždenné úlohy

- Kontrola logov a prehľad neúspešných prihlásení
- Monitoring dostupnosti stránky
- Overenie zálohy
- Kontrola dostupných aktualizácií
- Kontrola zmien v súboroch

Mesačné úlohy

- Bezpečnostný scan a kontrola integrity súborov
- Kontrola dátumu platnosti SSL certifikátu
- Audit používateľských účtov - odstráňte alebo prerad'te neaktívne účty
- Test obnovy zo zálohy

Ročné úlohy

- Zmena bezpečnostných kľúčov a saltov (viď Kapitola 28)
- Audit prístupových práv
- Penetračný test, ak to odôvodňuje profil rizík
- Kontrola súladu s legislatívou
- Komplexný bezpečnostný audit a aktualizácia bezpečnostnej politiky
- Školenie administrátorov

28 Pravidelná zmena bezpečnostných kľúčov a saltov

Bezpečnostné kľúče (tzv. security keys & salts) sa nachádzajú v súbore `wp-config.php` a z času na čas sa odporúča ich výmena za nové. Ich výmena okamžite zneplatní všetky aktívne relácie. Ak chcete získať novú sadu bezpečnostných kľúčov, možete si ich vygenerovať na adrese **api.wordpress.org/secret-key/1.1/salt/**. Nahraďte existujúce definície v súbore `wp-config.php` novo vygenerovanými hodnotami. Výmenu vykonávajte minimálne raz ročne alebo bezprostredne po podozrení na kompromitáciu.

> 📋 **VÝMENA KĽÚČOV ODHLÁSI VŠETKÝCH POUŽÍVATEĽOV**
>
> Zmena bezpečnostných kľúčov okamžite zneplatní všetky aktívne prihlásenia na stránke. V prostredí s viacerými používateľmi vopred informujte administrátorov.

29 Postup pri bezpečnostnom incidente

V rámci prevencie je dôležité mať pripravený plán pre prípad bezpečnostného incidentu. Kvalitné plánovanie vám umožní udržať chladnú hlavu, dostať situáciu pod kontrolu a zabezpečiť čo najviac dôkazov a dokumentácie.

1 - Izolácia

Okamžite stránku vypnite alebo prepnite do režimu údržby. Zabránite útočníkovi v ďalšom využívaní kompromitovanej inštalácie a chránite návštevníkov stránky.

2 - Zachovanie dôkazov

Pred vykonaním akýchkoľvek zmien si skopírujte serverové logy, prístupové logy a chybové logy. Analýza logov je nevyhnutná pre pochopenie priebehu incidentu a záznamy môžu byť požadované pri plnení oznamovacích povinností.

3 - Identifikácia

Skontrolujte: neznáme administrátorské účty, nedávno modifikované súbory `.php`, cudzie vložené kódy v súboroch tém alebo pluginov a podozrivé záznamy v databáze. Na tento účel využite skener integrity súborov v bezpečnostnom plugine.

4 - Čistenie

Obnovte stránku z poslednej známej čistej zálohy. Ak si nie ste istí, kedy ku kompromitácii došlo, prechádzajte zálohy dozadu, kým nenájdete čistý stav. Nestačí jednoducho odstrániť škodlivé súbory z infikovanej inštalácie. Môžu totiž existovať ďalšie zadné vrátka.

5 - Posilnenie a dokumentácia

Po obnove: zmeňte všetky prístupové heslá a kľúče, vymeňte bezpečnostné kľúče (Kapitola 28), skontrolujte používateľské účty a identifikujte spôsob, akým bol útok realizovaný. Vykonajte post-incident analýzu, zdokumentujte čo sa stalo a aktualizujte postupy tak, aby bola identifikovaná medzera uzatvorená.