WordPress Sicherheit

WordPress gegen Hacker absichern.

Inhaltsverzeichnis

Die Absicherung von WordPress ist von entscheidender Bedeutung, um die Sicherheit Ihrer Website zu gewährleisten. Es gibt keine 100-prozentige Sicherheit. Dafür kann aber der Zugriff für Hacker so erschwert werden, sodass diese kein Interesse haben, eine Webseite hacken zu wollen.

Das Dokument erhebt keinen Anspruch auf Vollständigkeit. Die Auswahl der Aktionen zur Absicherung einer Webseite ist Abwägungssache je nach Budget, Zeit, Aufwand und anderen Gründen.

Warum ist WordPress-Sicherheit wichtig?

Eine gehackte WordPress-Website kann Ihren Geschäftseinnahmen und Ihrem Ruf ernsthaften Schaden zufügen. Hacker können Benutzerinformationen und Kennwörter stehlen, bösartige Software installieren und sogar Malware an Ihre Benutzer verteilen. Im schlimmsten Fall müssen Sie sogar Lösegeld an Hacker zahlen, nur um wieder Zugang zu Ihrer Website zu erhalten.

Wie sicher ist WordPress?

WordPress ist eines der beliebtesten Content-Management-Systeme (CMS) auf dem Markt. Allerdings ist WordPress nicht perfekt. Wie jedes andere CMS hat auch WordPress seine Sicherheitslücken. Täglich programmieren viele Entwickler an der Sicherheit von WordPress und versuchen die Sicherheitslücken so schnell wie möglich zu schließen. Das bedeutet aber nicht, dass WordPress unsicher ist. WordPress ist immer noch eines der sichersten CMS auf dem Markt. Es gibt jedoch einige Dinge, die Sie tun können, um Ihre WordPress-Website noch sicherer zu machen.

Was sind häufige WordPress-Sicherheitsprobleme?

- **Brute-Force Login Attacken**
  Der Brute-Force-Anmeldeversuch ist eine der einfachsten Formen des Angriffs. Dabei nutzt ein Hacker die Automatisierung, um sehr schnell möglichst viele Kombinationen aus Benutzername und Kennwort einzugeben und schließlich die richtigen Anmeldedaten zu erraten.
- **Cross-Site Scripting (XSS)**
  Bei dieser Art von Angriff "injiziert" ein Angreifer bösartigen Code in das Backend der Ziel-Website, um Informationen zu extrahieren und die Funktionalität der Website zu beeinträchtigen. Der Code kann entweder auf komplexere Weise in das Backend eingeschleust werden oder einfach als Antwort in einem benutzerseitigen Formular übermittelt werden.
- **Database Injections**
  Diese Form des Angriffs wird auch als SQL-Injektion bezeichnet, wenn ein Angreifer über eine Benutzereingabe, z. B. ein Kontaktformular, eine Zeichenkette mit schädlichem Code in eine Website einspeist. Die Website speichert dann den schädlichen Code in ihrer Datenbank.
- **Backdoors**
  Eine Backdoor ist eine Datei, die Code enthält, der es einem Angreifer ermöglicht, den Standard-WordPress-Login zu umgehen und so jederzeit auf Ihre Website zuzugreifen.
- **Denial-of-Service (DoS) Attacks**
  DoS-Angriffe werden am häufigsten durchgeführt, indem ein Server mit Datenverkehr überlastet und zum Absturz gebracht wird.
- **Phishing**
  Phishing tritt auf, wenn ein Angreifer ein Ziel kontaktiert und sich als seriöses Unternehmen oder einen Dienst ausgibt. Bei Phishing-Versuchen wird die Zielperson in der Regel aufgefordert, persönliche Daten preiszugeben, Malware herunterzuladen oder sogar eine gefährliche Website zu besuchen, die ihrem Computer schaden könnte.

- **Böswillige Umleitungen**
  Böswillige Umleitungen erstellen Backdoors in WordPress-Installationen mit FTP, SFTP, wp-admin und anderen Protokollen und injizieren Umleitungscodes in die Website. Die Umleitungen werden oft in Ihrer .htaccess-Datei und anderen WordPress-Kerndateien in verschlüsselter Form platziert und leiten den Webverkehr zu bösartigen Websites.

Maßnahmen zur Absicherung von WordPress
Allgemeine Maßnahmen

1.Wordpress Installationspakete nur von offiziellen Quellen herunterlanden
WordPress-Installationspakete können von Dritten bezogen und mit integriertem Schadcode angeboten werden. Daher ist es sinnvoll, WordPress-Installationspakete nur von offiziellen Quellen zu beziehen.

2. Halten Sie WordPress auf dem neuesten Stand
Genauso wie Sie Ihre Plugins und Themes auf dem neuesten Stand halten sollten, sollten Sie auch darauf achten, dass die von Ihnen verwendete WordPress-Version auf dem neuesten Stand ist. Glücklicherweise ist dies jetzt viel einfacher als früher, da kritische Sicherheitsupdates automatisch installiert werden. Natürlich nur, wenn Sie diese Funktion nicht ausdrücklich deaktivieren.
Neben neuen Funktionen, Verbesserungen und Fehlerkorrekturen enthalten WordPress-Core-Updates auch Sicherheitskorrekturen, die Sie vor Angreifern schützen können, die Ihre WordPress-Website ausnutzen.

3. Backup der Webseite
Prüfen Sie, ob Ihr Hoster wöchentliche, monatliche oder tägliche automatische Backups anbietet. Dieser Dienst ist in der Regel kostenpflichtig, gelegentlich aber auch kostenlos. Wenn dies der Fall ist und Ihr Hoster sowohl Ihre Dateien als auch Ihre Datenbank sichert, brauchen Sie nichts weiter zu tun. Eine der einfachsten Möglichkeiten, Ihre Daten auf WordPress zu sichern, ist die Installation eines Backup-Plugins, wie BlogVault oder Updraft Plus.
Wenn Sie es vorziehen, ein manuelles Backup durchzuführen, greifen Sie über einen FTP-Client auf die Dateien Ihrer Website zu und kopieren Sie alle Dateien an einen sicheren Ort. Sie können auch phpMyAdmin verwenden, um Ihre Datenbank zu exportieren.
Es kann jedoch eine gute Idee sein, für den Fall der Fälle regelmäßig manuelle Sicherungen vorzunehmen.
Selbst wenn eine Website gehackt wird und der Schaden irreparabel ist, müssen Sie sie nicht noch einmal von Grund auf neu aufbauen.

Maßnahmen zur Absicherung von WordPress
Maßnahmen auf Serverebene

Vergewissern Sie sich, dass Sie eine sichere, stabile Version Ihres Webservers und der darauf installierten Software verwenden. Stellen Sie sicher, dass Sie einen vertrauenswürdigen Hoster nutzen, der sich um diese Dinge für Sie kümmert. Die Nutzung eines managed (WordPress-) Hostingdienstes bietet eine sicherere Plattform für Ihre Website. Managed(-WordPress-)Hosting-Unternehmen sollten automatische Backups, automatische WordPress-Updates und erweiterte Sicherheitskonfigurationen zum Schutz Ihrer Website bieten.

1. aktuelle Serversoftware
Zur Sicherheit sollten Server, auf denen WordPress läuft, mit dem neuesten Betriebssystem und der neuesten (Sicherheits-)Software aktualisiert sowie gründlich getestet sein.

2. Firewalls auf Serverebene
Firewalls auf Serverebene sollten vor der Installation von WordPress auf dem Server vorhanden sein, um es auch während der Installations- und Erstellungsphase von WordPress gut geschützt zu halten.

3. dediziertes Webhosting
Wenn Sie einen gemeinsam genutzten Server verwenden (einen, auf dem neben Ihrer eigenen auch andere Websites gehostet werden) und eine Website auf demselben Server kompromittiert wird, kann auch Ihre Website kompromittiert werden, selbst wenn Sie alles in diesem Leitfaden befolgen. Deshalb ist es sicherer, ein dediziertes Webhosting, bzw. einen dedizierten Webserver zu nutzen. Hierbei wird nur die eigene Webseite gehostet.

4. Datenübertragung per SFTP-Verschlüsselung
Wenn Sie eine Verbindung zu Ihrem Server herstellen, sollten Sie die SFTP-Verschlüsselung verwenden. Dabei werden Ihr Passwort und andere Daten verschlüsselt, während sie zwischen Ihrem Computer und Ihrer Website übertragen werden. Das bedeutet, dass Ihr Kennwort verschlüsselt gesendet wird und von einem Angreifer nicht abgefangen werden kann.

5. Benutzen Sie neuere PHP-Version
PHP ist das Rückgrat Deiner WordPress-Seite und deshalb ist es sehr wichtig, eine aktuelle Version auf Ihrem Server zu verwenden. Wenn der Server mit veralteten PHP-Versionen läuft, hat er keinen Sicherheits-Support mehr und ist Sicherheitslücken ausgesetzt.

6. HTTPS für verschlüsselte Verbindungen verwenden – SSL-Zertifikat
Eine der am häufigsten übersehenen Möglichkeiten, die WordPress-Sicherheit zu verbessern, ist die Installation eines SSL-Zertifikats und der Betrieb der Website über HTTPS. SSL (Secure Sockets Layer) ist ein Protokoll, das die Datenübertragung zwischen Ihrer Website und dem Browser des Benutzers verschlüsselt. Diese Verschlüsselung erschwert das Ausspähen und Stehlen von Informationen. Viele Hosting-Unternehmen bieten jetzt ein kostenloses SSL-Zertifikat für Ihre WordPress-Website an. Sobald Sie SSL aktiviert haben, verwendet Ihre Website HTTPS statt HTTP, und Sie sehen ein Vorhängeschloss neben der Adresse Ihrer Website im Browser.
Jetzt ist es einfacher denn je, SSL für alle Ihre WordPress-Websites zu verwenden.
Erzwingen Sie SSL, um der Möglichkeit einer serverseitigen SSL-Fehlkonfiguration vorzubeugen.
In der wp-config.php:

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

6.1 Garantierte HTTP zu HTTPS Umleitung

Dieser Code sorgt für eine garantierte Weiterleitung aller HTTP-Anfragen auf die HTTPS-Version. Tragen Sie den Code in die .htaccess Datei im Stammverzeichnis ein.

```
<IfModule mod_rewrite.c>
  RewriteEngine On
  RewriteCond %{HTTPS} !=on
  RewriteRule ^ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
</IfModule>
```

7. Hinzufügen der neuesten HTTP-Sicherheits-Header

Ein weiterer Schritt, um die WordPress-Sicherheit zu verbessern, ist die Nutzung von HTTP-Sicherheits-Headern. Diese werden in der Regel auf der Ebene des Webservers konfiguriert und sagen dem Browser, wie er sich beim Umgang mit den Inhalten einer Website verhalten soll. Es gibt viele verschiedene HTTP-Sicherheits-Header, aber nachfolgend sind die typischerweise wichtigsten.

- Content-Security Policy
- X-XSS-Protection
- Strict-Transport-Security
- X-Frame-Options
- Public-Key-Pins
- X-Content-Type

Den folgenden Code können Sie verwenden, um die HTTP-Sicherheits-Header zu aktivieren. Fügen Sie den Code in die .htaccess Datei im Stammverzeichnis ein.

```
## X-FRAME-OPTIONS-Header
<IfModule mod_headers.c>
   Header set X-Frame-Options "sameorigin"
</IfModule>

## Strict Origin when cross origin Header
#@see https://scotthelme.co.uk/a-new-security-header-referrer-policy/
<IfModule mod_headers.c>
   Header set Referrer-Policy "strict-origin-when-cross-origin"
</IfModule>

## X-XSS-PROTECTION-Header
<IfModule mod_headers.c>
   Header set X-XSS-Protection "1; mode=block"
</IfModule>

## X-Content-Type-Options-Header
<IfModule mod_headers.c>
   Header set X-Content-Type-Options "nosniff"
</IfModule>

## Strict-Transport-Security-Header - für HTTPS
<IfModule mod_headers.c>
  Header set Strict-Transport-Security "max-age=31536000; includeSubDomains; preload"
</IfModule>
```

```
# Upgrade Insecure Requests to prevent mixed content
<ifModule mod_headers.c>
  Header always set Content-Security-Policy "upgrade-insecure-requests"
</IfModule>

# Permissions Policy is a new header that allows a site to control which features and APIs can be used in the browser.
<IfModule mod_headers.c>
Header always set Permissions-Policy "geolocation=(), midi=(),sync-xhr=(),accelerometer=(), gyroscope=(),
magnetometer=(), camera=(), fullscreen=(self)"
</IfModule>
```

Mit dem kostenlosen securityheaders.io Tool können Sie Ihre Webseite scannen. Dies zeigt Ihnen, welche HTTP-Sicherheitsheader derzeit für Ihre Website aktiviert sind.

8. Passwortschutz

Die häufigsten WordPress-Hacking-Versuche verwenden gestohlene Passwörter. Sie können dies erschweren, indem Sie sichere Passwörter verwenden, die für Ihre Website einzigartig sind. Nicht nur für den WordPress-Administrationsbereich, sondern auch für SFTP-Konten, die Datenbank, das WordPress-Hosting-Konto und Ihre benutzerdefinierten E-Mail-Adressen, die den Domainnamen Ihrer Website verwenden.

So erstellen Sie ein sicheres Passwort:

- Verwenden Sie Groß- und Kleinbuchstaben, Zahlen und Symbole.
- Vermeiden Sie persönliche Informationen wie Ihren Namen, Ihre Adresse oder Ihr Geburtsdatum.
- Verwenden Sie ein Passwort, das mindestens 8 Zeichen lang ist.

9. Verzeichnisindizierung und Verzeichnisdurchsuchung deaktivieren

Das Durchsuchen von Verzeichnissen kann von Hackern verwendet werden, um herauszufinden, ob Sie Dateien mit bekannten Sicherheitslücken haben, damit sie diese Dateien ausnutzen können, um sich Zugang zu verschaffen.

Das Durchsuchen von Verzeichnissen kann auch von anderen Personen genutzt werden, um Ihre Dateien einzusehen, Bilder zu kopieren, Ihre Verzeichnisstruktur herauszufinden und andere Informationen zu erhalten. Aus diesem Grund sollten Sie die Verzeichnisindizierung und das Durchsuchen von Verzeichnissen unbedingt deaktivieren.

Verbinden Sie sich mit Ihrer Website über SFTP oder den Dateimanager von cPanel. Suchen Sie dann die .htaccess-Datei im Stammverzeichnis Ihrer Website.

Danach müssen Sie die folgende Zeile am Ende der .htaccess-Datei einfügen:

```
Options -Indexes
```

10. IP-Zugriffsbeschränkungen des Adminlogins einrichten

Die Einschränkung des IP-Zugriffs ist wahrscheinlich die robusteste Maßnahme, die Sie ergreifen können, um Ihr Admin-Panel zu sperren, da jede Anfrage an wp-admin, die nicht von einer zulässigen Adresse stammt, zu einem 403 Forbidden-Antwortfehler führt.

10.1 IP-Zugangsbeschränkung über .htaccess

In Apache-Umgebungen können Sie eine sogenannte .htaccess-Datei verwenden, um unbefugten Zugriff auf Ihre Website zu verhindern. Wir erstellen und platzieren oder bearbeiten eine Datei mit dem Namen .htaccess im wp-admin-Verzeichnis.

Stellen Sie sicher, dass der Inhalt wie folgt lautet (wobei 1.2.3.4 durch Ihre IP-Adresse ersetzt

wird):

```
order deny,allow
deny from all
allow from 1.2.3.4
```

Alle Anfragen an wp-admin, die nicht von der von Ihnen angegebenen IP-Adresse stammen, führen zu einer 403 Forbidden-Antwort.

10.2 IP-Zugangsbeschränkung über die Konfigurationsdatei

Wenn Ihr Webserver auf NGINX (und nicht auf Apache) basiert, müssen Sie stattdessen nur die Konfigurationsdatei für die Website bearbeiten:

```
location ~ ^/(wp-admin|wp-login\.php) {
        allow 1.2.3.4;
        deny all;
}
```

11. WordPress XML-RPC blockieren

Die Schnittstelle stellt nicht nur nützliche Funktionen bereit, sondern dient auch als ein wichtiges Angriffsziel für Hacker. Die Angreifer nutzen immer mehr die xmlrpc.php für ihre Brute-Force-Angriffe gegen WordPress, weil ein Angriff gegen diese Schnittstelle wesentlich effizienter und mit weniger Aufwand als andere Methoden verbunden ist. Wenn Ihre Website die xml-rpc-Funktionalität von WordPress nicht nutzen muss, blockieren Sie sie entweder mit einer .htaccess-Datei Ihrer WordPress-Installation oder durch Installation des Plugins Disable XML-RPC.

```
# disable access to xmlrpc
<Files "xmlrpc.php">
Order Allow,Deny
deny from all
</Files>
```

12. Zugriff auf wp-config.php blockieren

Den Zugriff zu verweigern, ist eine viel konkretere Maßnahme. Wenn Sie dies tun, müssen Sie die Datei überhaupt nicht verschieben. Gehen Sie zu Ihrer .htaccess-Datei und fügen Sie den folgenden Code ein, ganz oben ein:

```
# disable access to wp-config
<files wp-config.php>
order allow,deny
deny from all
</files>
```

13. HTTP-Authentifizierung des Adminbereiches

Eine weitere Möglichkeit, den Admin zu sperren, besteht darin, eine HTTP-Authentifizierung hinzuzufügen. Dies erfordert einen Benutzernamen und ein Passwort, bevor man überhaupt auf die WordPress Login-Seite zugreifen kann.

Die folgende Variante ist für einen Apache-Webserver geeignet.
Um es manuell einzurichten, müssen Sie zuerst eine .htpasswd-Datei erstellen. Sie können dazu dieses praktische Generator-Tool verwenden. Dann laden Sie die Datei in Ihrem wp-admin-Ordner

hoch.
Anschließend erstellen oder bearbeiten Sie die .htaccess-Datei mit dem folgenden Code und speichern Sie ihn im /wp-admin-Verzeichnis. Achten Sie darauf, dass Sie den Verzeichnispfad und den Benutzernamen aktualisieren.

```
AuthName "Admins Only"
AuthUserFile /yourdirectory/wp-admin/.htpasswd
AuthType basic
require user yourusername
```

Der einzige Vorbehalt, eine HTTP-Authentifizierung auf diese Weise zu tun, ist, dass es AJAX (admin-ajax) auf dem Frontend Deiner Website zerstört. Dies wird von einigen Plugins von Drittanbietern benötigt. Daher sollten Sie in die Datei .htaccess auch den folgenden Code einfügen. Der AJAX Zugriff wird somit freigegeben.

```
<Files admin-ajax.php>
Order allow,deny
Allow from all
Satisfy any
</Files>
```

14. Dateiberechtigungen

Datei- und Verzeichnisberechtigungen sind entscheidend, um die WordPress-Sicherheit zu verbessern. Wenn die Berechtigungen zu locker sind, könnte jemand leicht Zugang zu einer Website erhalten und Chaos anrichten. Andererseits, wenn die Berechtigungen zu streng sind, könnte dies die Funktionalität Ihrer Website beeinträchtigen. Daher ist es wichtig, dass die richtigen Berechtigungen eingestellt sind.

- Alle Dateien sollten 644 sein.
- Ausnahme: wp-config.php sollte 440 sein, um zu verhindern, dass andere Benutzer auf dem Server es lesen.
- Alle Verzeichnisse sollten 755 sein.

Ändern von Dateiberechtigungen.

Wenn Sie Shell-Zugriff auf Ihren Server haben, können Sie die Dateiberechtigungen mit dem folgenden Befehl rekursiv ändern:
für Verzeichnisse:

```
find /path/to/your/wordpress/install/ -type d -exec chmod 755 {} \;
```

für Dateien:

```
find /path/to/your/wordpress/install/ -type f -exec chmod 644 {} \;
```

15. Autoren-Scans blockieren.

Hier werden Bots blockiert, die alle WordPress-Websites nach den Usernamen der Autoren scannen, um im Anschluss diese Websites anzugreifen. Fügen Sie in die Datei .htaccess, im Stammverzeichnis, den folgenden Code ein.

```
#Block WordPress Author Scans
<IfModule mod_rewrite.c>
```

```
RewriteEngine On
RewriteBase /
RewriteCond %{REQUEST_URI} ^/wp-json/wp/v2/users [OR]
RewriteCond %{QUERY_STRING} (author=\d+) [NC]
RewriteRule ^ – [NC,F,L]
</ifModule>
```

16. Zugriff auf wp-includes Ordner sperren

Folgender Code sperrt den Zugriff von außen auf den gefährdeten wp-includes Ordner. Fügen Sie in die Datei .htaccess, im Stammverzeichnis, den folgenden Code ein.

```
# Block access to includes folder
<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>
```

Maßnahmen zur Absicherung von WordPress

Maßnahmen auf Softwareebene

1. Dateiversion verbergen

Das Anzeigen der WordPress-Version im Webseitenbereich (Frontend) kann möglichen Angreifern Aufschluss über mögliche Schwachstellen geben. Gerade ältere WordPress-Versionen können hierbei betroffen sein. Bei neueren WordPress-Versionen wird die Version im Webseitenbereich (Frontend) nicht mehr angezeigt.

1.1 Anzeige im Template

Sie können den folgenden Code verwenden, um die Dateiversion zu verbergen. Fügen Sie den Code einfach zur Datei `functions.php` Ihres aktivierten WordPress-Themes hinzu.

```
function wp_version_remove_version() {
    return '';
}
add_filter('the_generator', 'wp_version_remove_version');
```

1.2 Dateien in WordPress

Ein weiterer Ort, an dem die WordPress-Version angezeigt wird, ist in der standardmäßigen readme.html-Datei, die in jeder WordPress-Version enthalten ist. Sie befindet sich im Stammverzeichnis Ihrer Installation, domain.com/readme.html. Sie können diese Datei auf sichere Weise per SFTP löschen.

2. Url der Loginseite ändern

Der häufigste und wahrscheinlich einfachste Weg, Ihre WordPress Login-URL-Seite zu ändern, ist die Verwendung eines kostenlosen Plugins wie WPS Hide Login (https://de.wordpress.org/plugins/wps-hide-login/). Das Plugin fängt einfach Seitenanfragen ab und funktioniert auf jeder WordPress-Website. Ändern Sie die URL des Anmeldeformulars einfach und sicher in eine beliebige Url. Das wp-admin-Verzeichnis und die Seite wp-login.php werden unzugänglich. Dies erschwert Dritten den Zugang zu Ihrem Adminbereich. Die Deaktivierung dieses Plugins bringt Ihre Website genau in den vorherigen Zustand zurück.

3. Deaktivieren der Dateibearbeitung in WordPress-Dashboard

Hacker könnten versuchen, über den Editor zur Dateibearbeitung schadhafte Programmierungen einzufügen. Dies ist ein schneller Weg für Sie, um bösartigen Code auf Ihrer Website auszuführen. Wenn Sie keinen Zugriff darauf über das Dashboard haben, kann dies zunächst dazu beitragen, Angriffe zu verhindern. Fügen Sie den folgenden Code in die wp-config.php-Datei ein, um die Möglichkeiten von ‚edit_themes', ‚edit_plugins' und ‚edit_files' aller Benutzer zu entfernen.

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

Es ist besser, Dateien lokal zu bearbeiten und per SFTP hochzuladen.

4.Sicherheitsfragen zum WordPress-Anmeldebildschirm hinzufügen

Wenn Sie Ihrem WordPress-Anmeldebildschirm eine Sicherheitsfrage hinzufügen, wird es noch schwieriger für jemanden, sich unbefugt Zugang zu verschaffen.
Sie können Sicherheitsfragen hinzufügen, indem Sie das WP Security Questions Plugin installieren. Nach der Aktivierung müssen Sie nur noch die Einstellungen des Plugins konfigurieren.

5. Anmeldeversuche begrenzen

Die Begrenzung von Anmeldeversuchen kann Brute-Force-Angriffe verhindern, bei denen Hacker versuchen, Ihr Passwort durch wiederholte Anmeldeversuche zu erraten.

Verwenden Sie ein Plugin wie Limit Login Attempts Reloaded, das die Anzahl der Anmeldeversuche begrenzt, oder eine Web Application Firewall, die verdächtige Anmeldeversuche blockieren kann.

Maßnahmen zur Absicherung von WordPress

Maßnahmen auf Datenbankebene

Es gibt eine leider häufig übersehene WordPress-Sicherheitsrichtlinie:
Weisen Sie dem von WordPress verwendeten MySQL-Benutzer nur die minimal erforderlichen WordPress-Datenbankberechtigungen zu.
Der MySQL-Benutzer hat in den meisten Fällen auf die WordPress-Datenbank vollen Zugriff. Oder, noch schlimmer, hat er Zugriff auf den gesamten MySQL-Datenbankserver. Die WordPress-Datenbankprivilegien des MySQL-Benutzers sollten eingeschränkt werden. Dem MySQL-Benutzer, der auf die WordPress-MySQL-Datenbank zugreift, sollten nur die minimal erforderlichen Datenbankprivilegien (Datenbankberechtigungen) zugewiesen werden.

Sicherheitsrichtlinien für WordPress-MySQL-Datenbanken

- Verwenden Sie niemals den MySQL-Root (MySQL-Superuser) in WordPress
- Verwenden Sie für jede installierte Webanwendung einen anderen MySQL-Benutzer
- Weisen Sie dem MySQL-Benutzer nur die erforderlichen Mindestrechte für die Datenbank zu.
- Wenn Sie einen neuen MySQL-Benutzer anlegen, verwenden Sie einen nicht vorhersehbaren Benutzernamen
- Konfigurieren Sie ein sehr starkes MySQL-Benutzerpasswort.
- So erstellen Sie ein sicheres Passwort:
    - Verwenden Sie Groß- und Kleinbuchstaben, Zahlen und Symbole
    - Vermeiden Sie persönliche Informationen wie Ihren Namen, Ihre Adresse oder Ihr Geburtsdatum
    - Verwenden Sie ein Passwort, das mindestens 8 Zeichen lang ist.

Sicherheitstipp: WordPress-Backups sind ein Muss. Auch von der Datenbank sollten regelmäßig Backups erstellt werden.

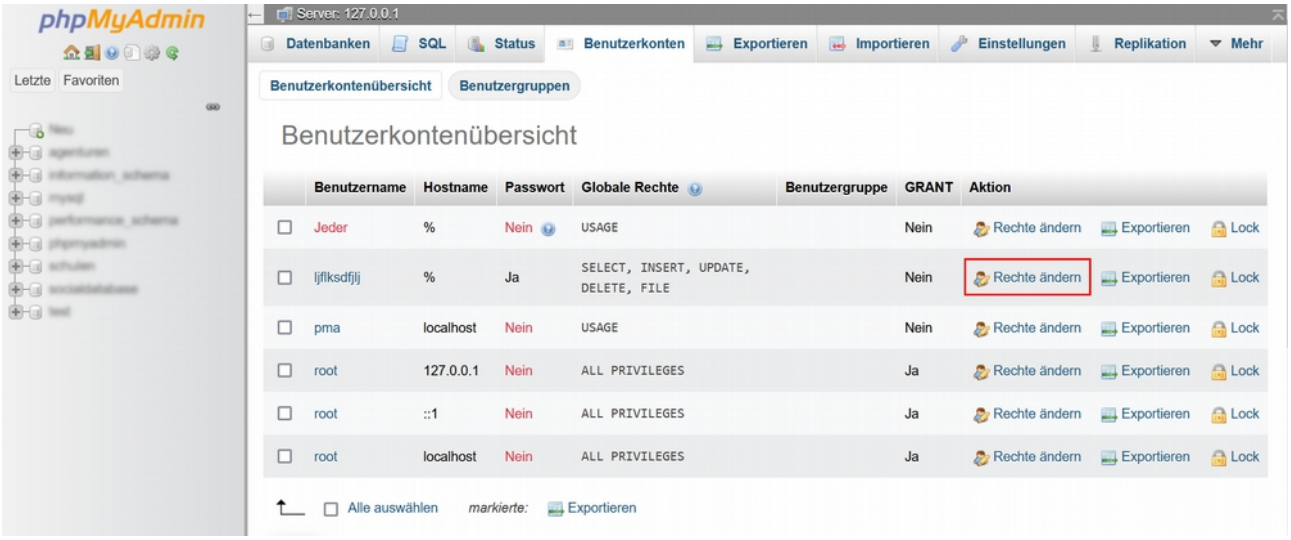
Warum sollten die MySQL-Benutzerrechte eingeschränkt werden?
Es kann passieren, Sie installieren versehentlich ein bösartiges Plugin, das eine Hintertür oder einen Trojaner enthält. Oder ein bösartiger Benutzer schafft es, die Anmeldedaten des von WordPress verwendeten MySQL-Benutzers in die Hände zu bekommen. In beiden Fällen kann der Schaden begrenzt und das Problem leicht behoben werden, wenn die Rechte des MySQL-Benutzers nur auf die Daten in der MySQL-WordPress-Datenbank beschränkt werden.

Wann man alle WordPress-MySQL-Datenbankprivilegien anwendet
Es ist akzeptabel, dem MySQL-Benutzer, den Sie während der Installation in der WordPress-Datei wp-config.php verwenden werden, ALLE Datenbankprivilegien zu gewähren. Dies sollte jedoch nur eine vorübergehende Maßnahme sein, bis die WordPress-Installation alle notwendigen Tabellen und andere Datenbankobjekte in der MySQL-Datenbank erstellt hat, damit WordPress funktioniert.

1. Wie man sichere WordPress-MySQL-Datenbankprivilegien anwendet

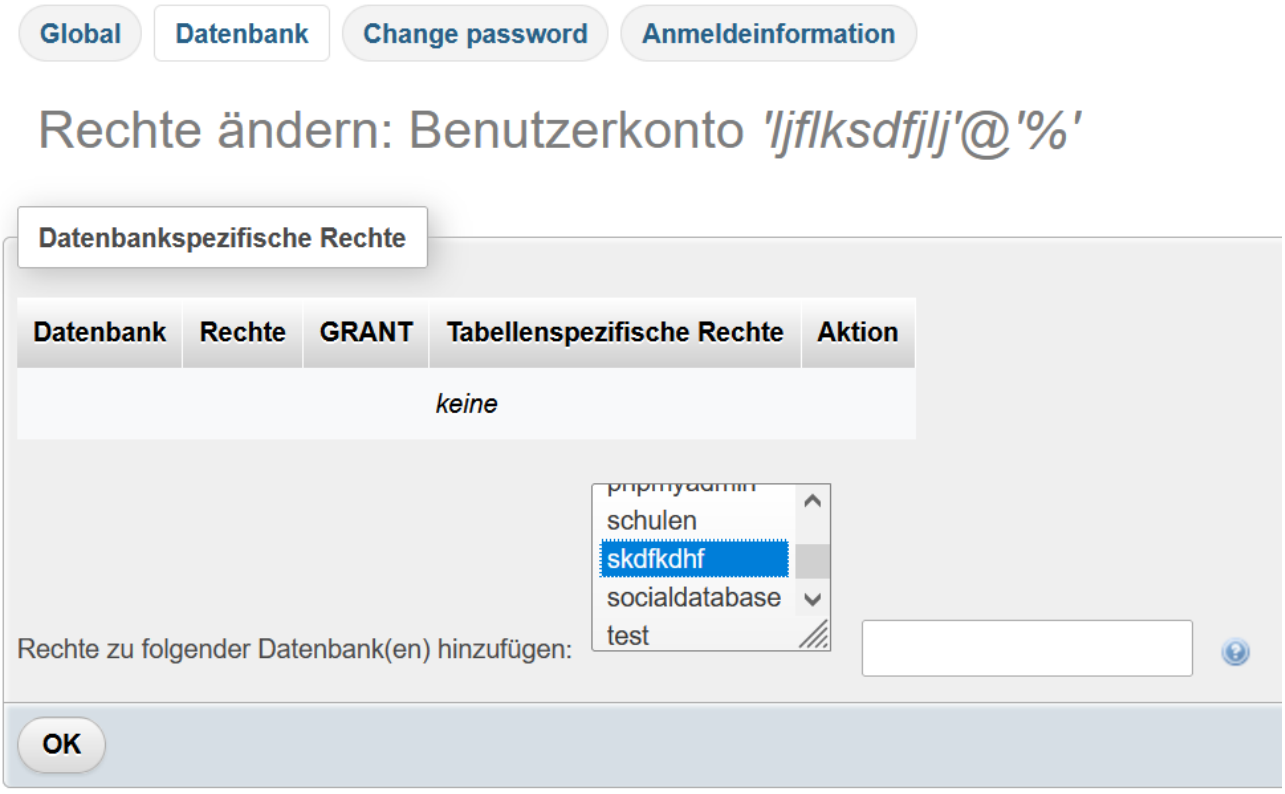
1.1 Bearbeiten der Datenbankberechtigungen des MySQL-Benutzers mit phpMyAdmin

Sobald Sie in der MySQL phpMyAdmin Web-Oberfläche eingeloggt sind, klicken Sie auf die Registerkarte Benutzer(konten) und dann auf die Berechtigungen des MySQL-Benutzers bearbeiten, den Sie für den Zugriff auf die WordPress-MySQL-Datenbank verwenden, wie im folgenden Screenshot dargestellt.



1.2 Wählen Sie die WordPress-MySQL-Datenbank aus, um dem Benutzer Berechtigungen zuzuweisen

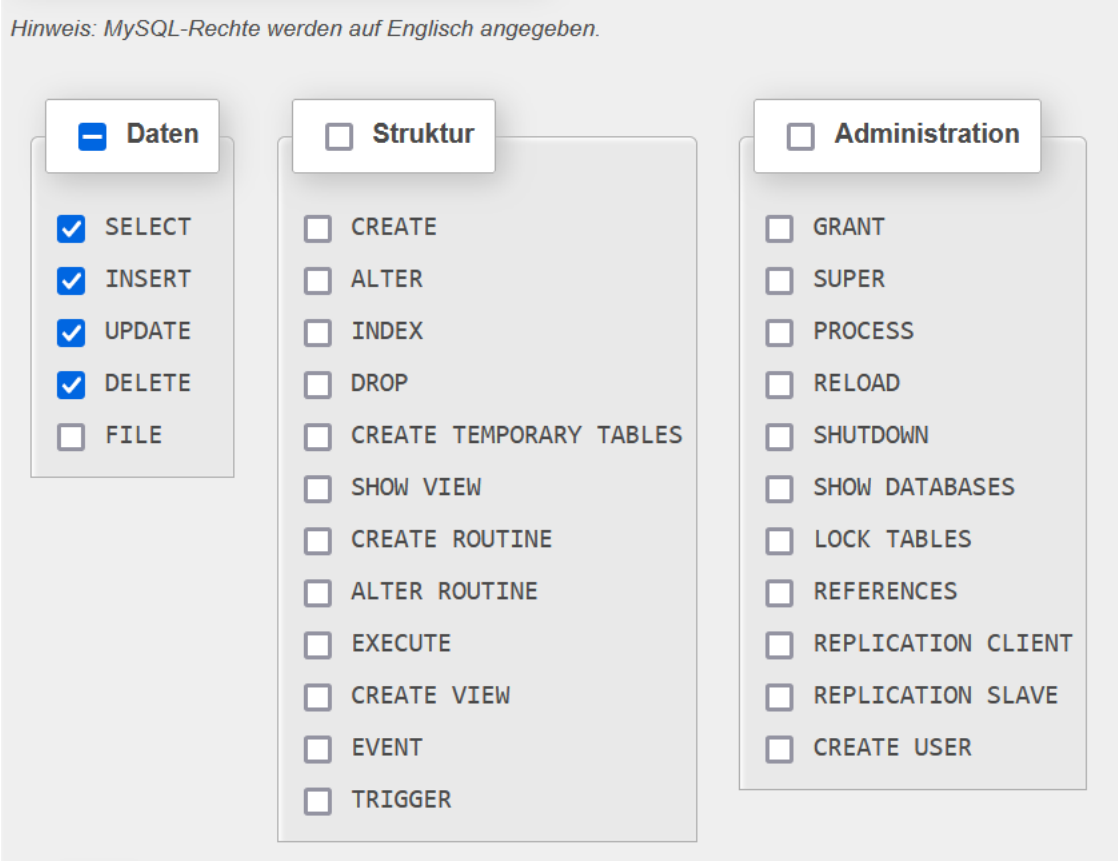
Scrollen Sie im Fenster mit den Benutzerrechten nach unten zum Abschnitt Datenbankspezifische Rechte und wählen Sie im Dropdown-Menü die WordPress-MySQL-Datenbank aus. Klicken Sie auf OK. Im folgenden Beispiel ist die WordPress-MySQL-Datenbank "skdfkdhf".



1.3 Zuweisung der Benutzerrechte für die WordPress-MySQL-Datenbank

Für den normalen Betrieb von WordPress muss der MySQL-Benutzer nur Daten aus der Datenbank lesen und in die Datenbank schreiben können. Erlauben Sie also in diesem Fall nur die unten aufgeführte Liste von Berechtigungen unter der Datenspalte, wie im folgenden Screenshot zu sehen:

- Select
- Insert
- Update
- Delete



1.4 Zuweisung von MySQL-Benutzerrechten über die MySQL-Befehlszeile

Wenn Sie die MySQL-Befehlszeile verwenden, können Sie die folgende SQL-Syntax verwenden, um dem Benutzer, der von WordPress für den Zugriff auf die MySQL-Datenbank verwendet wird, d. h. dem in der WordPress-Datei wp-config.php angegebenen Benutzer, ausschließlich Lese- und Schreibrechte zu erteilen.

```
GRANT SELECT , INSERT , UPDATE , DELETE ON  `[DATABASE]` . * TO
‘[USER]’@'[HOST]’;
```

Ersetzen Sie [DATABASE] durch den Datenbanknamen, [USER] durch den MySQL-Benutzernamen und [HOST] durch den Host (z. Bsp.: localhost.)

MySQL-Benutzerdatenbankberechtigungen für die Aktualisierung von WordPress
Beim Upgrade einer WordPress-Installation kann es vorkommen, dass der Upgrade-Prozess die Struktur der WordPress-MySQL-Datenbank ändern muss. In diesem Fall ist es sicher, dem MySQL-Benutzer, alle strukturbezogenen Rechte nur während des WordPress-Upgrade-Prozesses zu gewähren. Sobald das Upgrade erfolgreich abgeschlossen ist, können Sie die Änderungen wieder rückgängig machen.
Spezielle MySQL-Benutzerdatenbankprivilegien zur Installation von Plugins und Themes
Obwohl es nicht so häufig vorkommt, kann es sein, dass einige WordPress-Plugins und WordPress-Themes die WordPress-MySQL-Datenbankstruktur und -Objekte ändern müssen, z. B. durch Hinzufügen von Tabellen, Routinen oder Ansichten. In diesem Fall müssen die oben konfigurierten Datenbankberechtigungen (nur Lese- und Schreibrechte) geändert werden, damit solche WordPress-Plugins und -Themes installiert werden können.
Sobald das WordPress-Plugin oder WordPress-Theme installiert ist und die erforderlichen Änderungen an der MySQL-Datenbank vorgenommen wurden, können Sie zu den reinen Lese- und Schreibrechten für Daten zurückkehren.

2. Tabellenpräfix für WordPress ändern

Das Tabellenpräfix ist ein Bestandteil des Tabellennamens und wird in den allermeisten Fällen von dem Webseitenbetreiber nicht geändert. Als Tabellenpräfix gibt WordPress standardmäßig „wp_“ vor. Es ist jedoch empfehlenswert, dieses zu ändern. WordPress sichert Ihre Beiträge und Einstellungen in der Datenbank, weswegen Hacker diese oft angreifen. Indem Sie das Tabellenpräfix ändern, erschweren Sie Ihnen den Zugriff.
Haben Sie den Präfix bei der Installation nicht ändern können, können Sie dies nachträglich tun:

2.1 - Tabellenpräfix in wp-config.php ändern

Suchen Sie die Datei wp-config.php, um diese im Editor Ihrer Wahl zu bearbeiten.
Suchen Sie nach der folgenden Zeile:

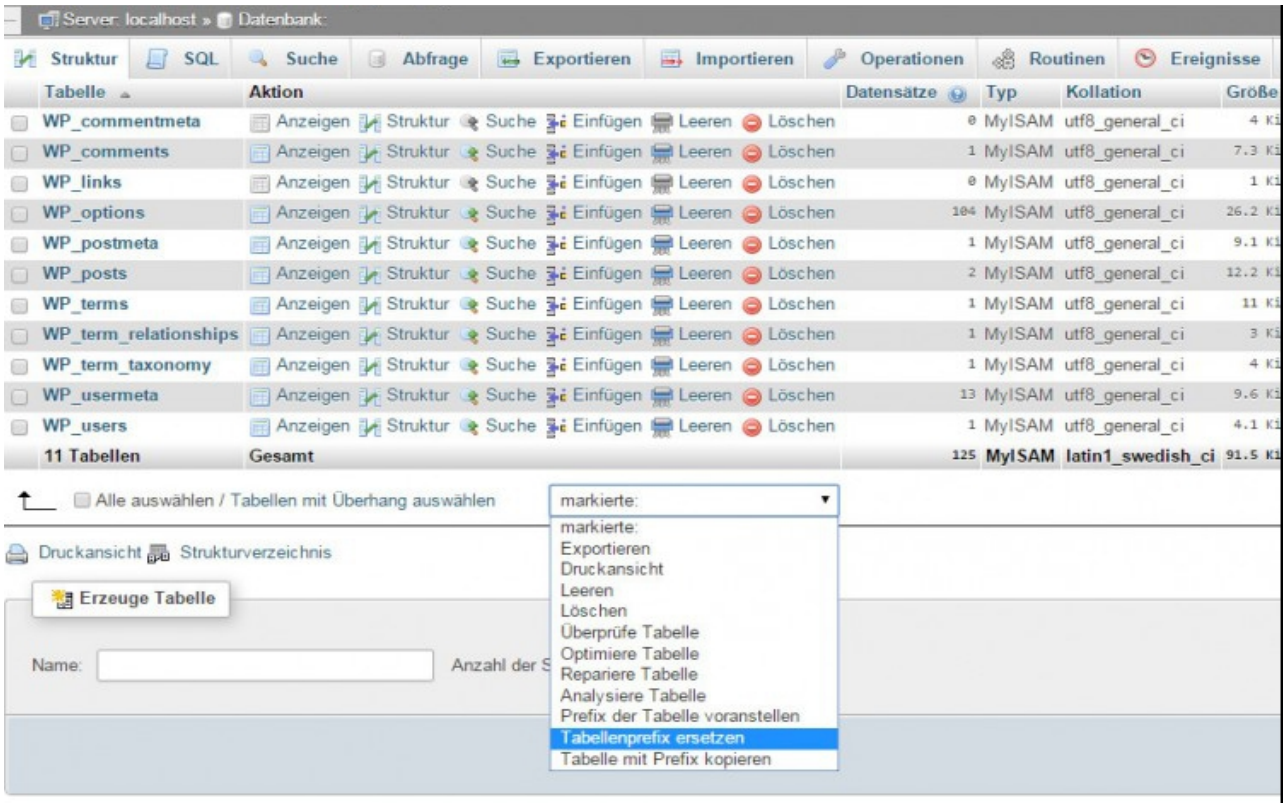
```
$table_prefix = 'wp_';
```

Ersetzen Sie 'wp_' durch ein anderes Präfix, das auf einen Unterstrich endet. In diesem Beispiel nehmen wir 'bsp_':

```
$table_prefix = 'bsp_';
```

2.2 - Tabellenpräfix in der Datenbank ändern

Neue PhpMyAdmin Versionen bieten noch eine komfortablere Möglichkeit den Präfix zu ändern. Anstatt wie oben über den „RENAME“ Befehl, markieren Sie alle Tabellen und wählen im Dropdown Menü den Punkt „Tabellenpräfix ersetzen“.



Hier geben Sie den alten und neuen Präfix ein.

Per SQL Anweisung lassen sich die Präfixe auch anpassen. Ändern Sie mittels der Rename Funktion den Präfix aller vorhandenen Tabellen. Folgende SQL Anweisungen gelten für eine Basisinstallation von Wordpress. Es können jedoch bei Ihnen noch andere Tabellen existieren.

```
RENAME table `wp_commentmeta` to `bsp_commentmeta`;
RENAME table `wp_comments` to `bsp_comments`;
RENAME table `wp_links` to `bsp_links`;
RENAME table `wp_options` to `bsp_options`;
RENAME table `wp_postmeta` to `bsp_postmeta`;
RENAME table `wp_posts` to `bsp_posts`;
RENAME table `wp_terms` to `bsp_terms`;
RENAME table `wp_term_relationships` to `bsp_term_relationships`;
RENAME table `wp_term_taxonomy` to `bsp_term_taxonomy`;
RENAME table `wp_usermeta` to `bsp_usermeta`;
RENAME table `wp_users` to `bsp_users`;
```

2.3 - Alle Verweise zum alten Präfix ersetzen

In manchen Tabellen verweist WordPress noch auf das alte Tabellenpräfix. Um Ihre Änderung abzuschließen, müssen die Verweise entsprechend korrigiert werden.

```
UPDATE bsp_options SET option_name = REPLACE(option_name, 'wp_', 'bsp_');

UPDATE bsp_usermeta SET meta_key = REPLACE(meta_key, 'wp_', 'bsp_');
```

3. Tabellenpräfix mit dem Plug-in Change DB Prefix ändern

Das Plugin Change DB Prefix erledigt alle oben beschriebenen Aufgaben vollautomatisch. Nach der Installation geben Sie den neuen Präfixnamen ein und klicken auf Speichern. Den Rest erledigt das Plugin.

Tools für WordPress-Sicherheit

WPSec

wpsec automatisierte WordPress-Scans

WPSec ist ein WordPress-Schwachstellen-Scanner. Er wird über ein Dashboard verwaltet, über das Sie Scans durchführen, Benachrichtigungen einrichten und erweiterte Berichte erstellen können. Für die Scans verwendet WPSec die so genannte Advanced Scan Technology, die WPScanner und eine eigene Technologie nutzt.
WPSec scannt unter anderem nach:

- Bekannte WordPress-Fehler
- Sicherheitsprobleme

WPScan

Wie WPScan Ihnen hilft, Ihre Website zu schützen

WPScan's WordPress plugin hilft Ihnen, indem es den Prozess der Identifizierung von gefährdeter Software auf Ihrer Website automatisiert. Sie können das Plugin so konfigurieren, dass es täglich oder sogar stündlich Scans durchführt und Ihnen eine E-Mail-Benachrichtigung mit den Scan-Ergebnissen sendet, sobald es Probleme identifiziert hat.

Inspectwp

InspectWP nimmt dir so viel wie möglich Arbeit eines WordPress Audits ab: Es liest viele Informationen einer WordPress Website, aus und analysiert & bewertet sie.
https://inspectwp.com/de