# Bilder Upload Script Version 2.0

Installationsanleitung

Datum: 24.09.2026
Version des Scripts: 2.0
Autor des Scripts: Nico Schubert / SchubertMedia
Info zum Script: www.php-space.info/bilder-upload-script/

**SchubertMedia**
Inhaber: Nico Schubert
Stauffenbergallee 57
D-99085 Erfurt

Telefon: +49 (0) 3612 3002247
Mo. bis Fr. 9 bis 12 und 13 bis 18 Uhr
Telefax: +49 (0) 3612 3004636
USt-IdNr.: DE 227763975
Web: www.schubertmedia.de
E-Mail: info@schubertmedia.de

## Inhalt

1. Einsatzgebiete
2. Anforderungen
3. Installation Schritt für Schritt
4. Bedienung
5. Konfiguration (config.php)
6. Sicherheit
7. Passwort vergessen
8. Update von Version 1.x auf 2.0
9. Neu in Version 2.0
10. Häufige Fragen und bekannte Probleme
11. Nutzungshinweis und Copyright
12. Support, Vorschläge und Erweiterungen
13. Sprachdateien

## 1. Einsatzgebiete

Mit dem Bilder Upload Script laden Sie Bilder über den Browser auf Ihren eigenen Webspace und bekommen sofort die fertigen Codes zum Einbinden: den Direktlink, ein HTML-Tag für Ihre Website, BBCode für Forenbeiträge und Markdown. Ein FTP-Programm brauchen Sie dafür nicht mehr, auch nicht am Handy.

Typische Einsätze sind Bilder für Forenbeiträge, Blogartikel oder Produktbeschreibungen, Bildschirmfotos für eine Support-Anfrage oder ein kleines Bildarchiv für die eigene Website. Auf Wunsch nimmt das Script auch andere Dateien an, in der Grundeinstellung des Dateimodus Textdateien, PDF-Dokumente sowie ZIP- und RAR-Archive.

Das Script kommt ohne Datenbank aus. Alle Dateien liegen im Ordner `img`, zu jedem Bild legt das Script ein Vorschaubild an. Seit Version 2.0 schützt ein Passwort den Zugang. Hochladen und löschen kann also nur, wer das Passwort kennt. Die hochgeladenen Bilder selbst sind über ihre Adresse für jeden abrufbar, sonst ließen sie sich nicht einbinden.

## 2. Anforderungen

- **PHP ab Version 8.2.** Entwickelt und geprüft ist das Script mit PHP 8.5. Läuft auf dem Server eine ältere Version, zeigt das Script nur den Hinweis *Das Script braucht mindestens PHP 8.2, auf diesem Server läuft PHP …* mit der gefundenen Version. Die PHP-Version stellen Sie im Kundenmenü Ihres Webhosters um.
- **GD-Erweiterung** mit Unterstützung für JPEG, PNG und GIF. Mit ihr speichert das Script hochgeladene Bilder neu und erzeugt die Vorschaubilder. Kann GD auch WebP, nimmt das Script zusätzlich WebP-Bilder an, sonst weist es sie ab.
- **fileinfo-Erweiterung.** Mit ihr prüft das Script den tatsächlichen Inhalt jeder Datei. Sie ist in PHP standardmäßig eingeschaltet.

- **exif-Erweiterung**, nicht nötig. Wie ein Handyfoto gehalten wurde, liest das Script bei JPEG-Bildern auch ohne sie aus der Datei und dreht Hochformatfotos richtig herum.
- **Datei-Uploads** müssen in PHP erlaubt sein (Einstellung `file_uploads`). Das ist bei Webhostern der Normalfall.
- **Apache mit `.htaccess`**, empfohlen. Die mitgelieferten `.htaccess`-Dateien sperren die Ausführung von Scripts im Bilderordner und den Zugriff auf den Ordner `daten`. Läuft Ihr Webspace mit nginx, lesen Sie den Abschnitt nginx ohne .htaccess. Die Dateien im Ordner `daten` geben aber auch ohne `.htaccess` nichts preis.
- **HTTPS**, dringend empfohlen, damit das Passwort verschlüsselt übertragen wird. Viele Hoster bieten kostenlose Zertifikate an.
- **Arbeitsspeicher:** Fotos bis etwa 24 Megapixel (zum Beispiel 5712 × 4284 Pixel, typisch für Systemkameras) verarbeitet das Script mit dem verbreiteten PHP-Wert `memory_limit = 128M`. Für größere Bilder hebt es die Grenze selbst an, wenn der Server das erlaubt, höchstens auf 512 MB.
- **Speicherplatz:** Die Programmdateien belegen weniger als 1 MB, dazu kommt der Platz für Ihre Bilder. Eine Datenbank ist nicht nötig.

Fehlt eine dieser Voraussetzungen, zeigt das Script beim Aufruf die Seite *Bitte prüfen Sie die Einrichtung* und nennt, was zu tun ist. Bei dem Anbieter www.hosterplus.de erhalten Sie PHP-Webspace, auf dem das Script läuft.

## 3. Installation Schritt für Schritt

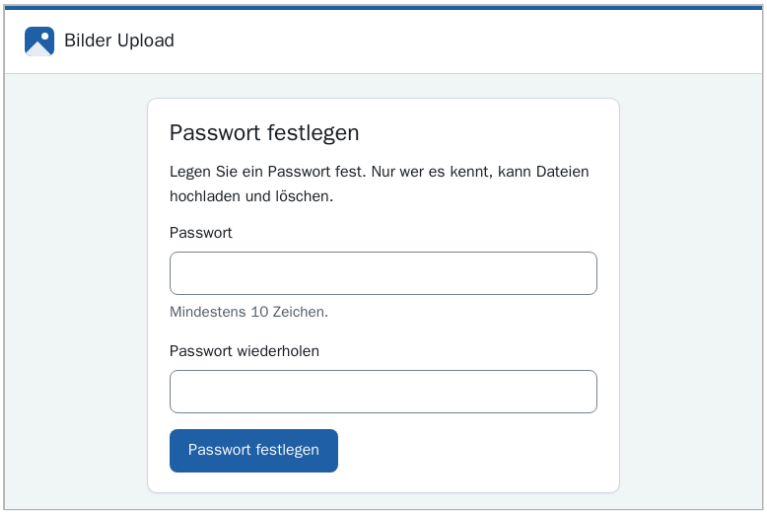
1. **Archiv entpacken.** Entpacken Sie das ZIP-Archiv auf Ihrem Rechner. Sie erhalten den Ordner `bilder-upload-script_2.0`.
2. **Einstellungen prüfen.** Die Datei `config.php` enthält sinnvolle Voreinstellungen, Sie müssen sie nicht ändern. Möchten Sie eine andere Sprache, eine andere Höchstgröße oder auch andere Dateitypen als Bilder zulassen, passen Sie die Datei vorher mit einem Texteditor an (siehe Konfiguration).
3. **Dateien per FTP hochladen.** Laden Sie den **Inhalt** des Ordners `bilder-upload-script_2.0` in ein Verzeichnis Ihres Webspace, zum Beispiel nach `/bilder/`. Übertragen Sie dabei unbedingt auch die drei versteckten Dateien `img/.htaccess`, `img/thumbnail/.htaccess` und `daten/.htaccess`. Manche FTP-Programme blenden Dateien mit einem Punkt am Anfang aus. In FileZilla schalten Sie die Anzeige unter *Server → Anzeige versteckter Dateien erzwingen* ein. Prüfen Sie nach dem Hochladen auf dem Server, ob die drei Dateien dort liegen.
4. **Ordnerrechte setzen.** In drei Ordner schreibt das Script: `img` für Ihre Dateien, `img/thumbnail` für die Vorschaubilder und `daten` für das Passwort, einen geheimen Schlüssel, die Anmeldesperre und das Fehlerprotokoll. Setzen Sie die Rechte dieser drei Ordner zuerst auf `755`. Bei den meisten Hostern genügt das, oft ist es schon voreingestellt. Nur wenn das Script die Ordner danach immer noch als nicht beschreibbar meldet und Ihr Hoster es ausdrücklich verlangt, setzen Sie `777`. Alle übrigen Dateien bleiben bei `644`, alle übrigen Ordner bei `755`.

   So setzen Sie die Rechte: In FileZilla klicken Sie mit der rechten Maustaste auf den Ordner, wählen *Dateiberechtigungen* und tragen den numerischen Wert `755` ein. Auf der FTP-Kommandozeile lautet der Befehl `quote site chmod 755 /bilder/img`, mit SSH-Zugang `chmod 755 img img/thumbnail daten` im Verzeichnis des Scripts. Die Ziffern bedeuten 7 = lesen, schreiben, ausführen, 5 = lesen und ausführen, 4 = nur lesen. Bei `777` darf jeder Benutzer des Servers in den Ordner schreiben, deshalb nur, wenn es nicht anders geht.
5. **Script aufrufen.** Öffnen Sie im Browser `https://www.ihre-domain.de/bilder/`. Ersetzen Sie *ihre-domain.de* und *bilder* durch Ihre Domain und Ihr Verzeichnis. Die `index.php` leitet auf die `upload.php` weiter. Das Script prüft bei jedem Aufruf, ob die Ordner vorhanden und beschreibbar sind, ob Uploads erlaubt sind und ob GD und fileinfo bereitstehen. Fehlt etwas, erscheint statt der Hauptseite *Bitte*

*prüfen Sie die Einrichtung* mit allen Befunden auf einmal, zum Beispiel *Diese Ordner fehlen oder sind nicht beschreibbar: daten/.* Beheben Sie das und klicken Sie auf **Erneut prüfen**.

Der Ordner `daten` fehlt oder ist nicht beschreibbar.

6. **Passwort sofort festlegen.** Beim ersten Aufruf zeigt das Script nur die Seite *Passwort festlegen*. Tragen Sie Ihr Passwort unter *Passwort* und *Passwort wiederholen* ein und klicken Sie auf **Passwort festlegen**. Es muss mindestens 10 Zeichen lang sein, und beide Eingaben müssen übereinstimmen. Danach meldet die Seite *Das Passwort ist gespeichert. Sie sind jetzt angemeldet.* und Sie sehen die Hauptseite. Das Script speichert das Passwort nicht im Klartext, sondern nur als Hash in der Datei `daten/zugang.php`. Aus dem Hash lässt sich das Passwort nicht zurückrechnen.




Beim ersten Aufruf legen Sie das Passwort fest.

**Legen Sie das Passwort gleich nach dem Hochladen fest.** Bis dahin kann jeder, der die Adresse des Scripts als Erster aufruft, selbst ein Passwort setzen. Rufen Sie das Script deshalb direkt nach dem FTP-Upload auf. Nehmen Sie ein Passwort, das Sie nirgends sonst verwenden. Ein Passwort-Manager erzeugt und speichert es für Sie.

Ist Ihnen die Installation zu aufwendig? Unser Installationsservice richtet das Script für Sie ein: www.php-space.info/script-installation-service/

## 4. Bedienung

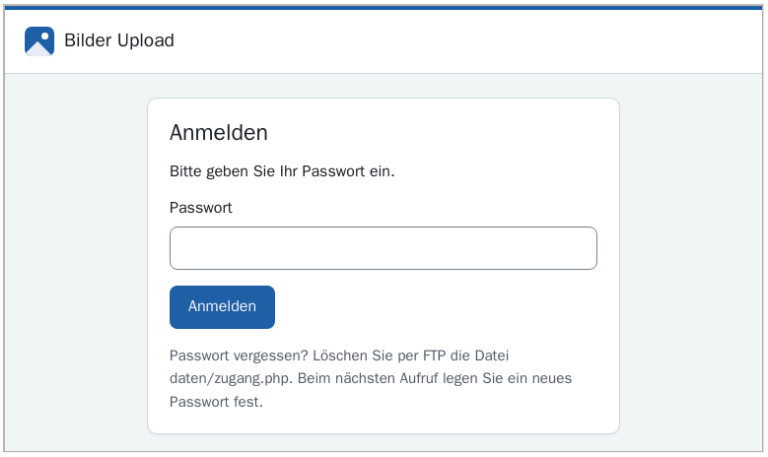
### Anmelden und abmelden

Rufen Sie das Script auf, geben Sie Ihr Passwort ein und klicken Sie auf **Anmelden**. Unter dem Formular steht der Hinweis, was bei einem vergessenen Passwort zu tun ist. Bei einem falschen Passwort meldet die Seite *Das Passwort ist falsch.* Nach fünf Fehlversuchen innerhalb von 15 Minuten sperrt das Script die

Anmeldung von dieser Internetverbindung aus für 15 Minuten und zeigt *Zu viele Fehlversuche. Die Anmeldung ist noch … Min. gesperrt.*

Probieren viele verschiedene Adressen gleichzeitig Passwörter aus (mindestens 20 Fehlversuche in 15 Minuten), bremst das Script die Anmeldung für alle: Es nimmt dann nur noch einen Versuch alle 10 Sekunden an und meldet *Gerade gab es sehr viele falsche Anmeldungen. Zum Schutz nimmt das Script nur alle 10 Sekunden einen Versuch an. …* Sie selbst kommen mit dem richtigen Passwort also weiter hinein, müssen höchstens kurz warten.

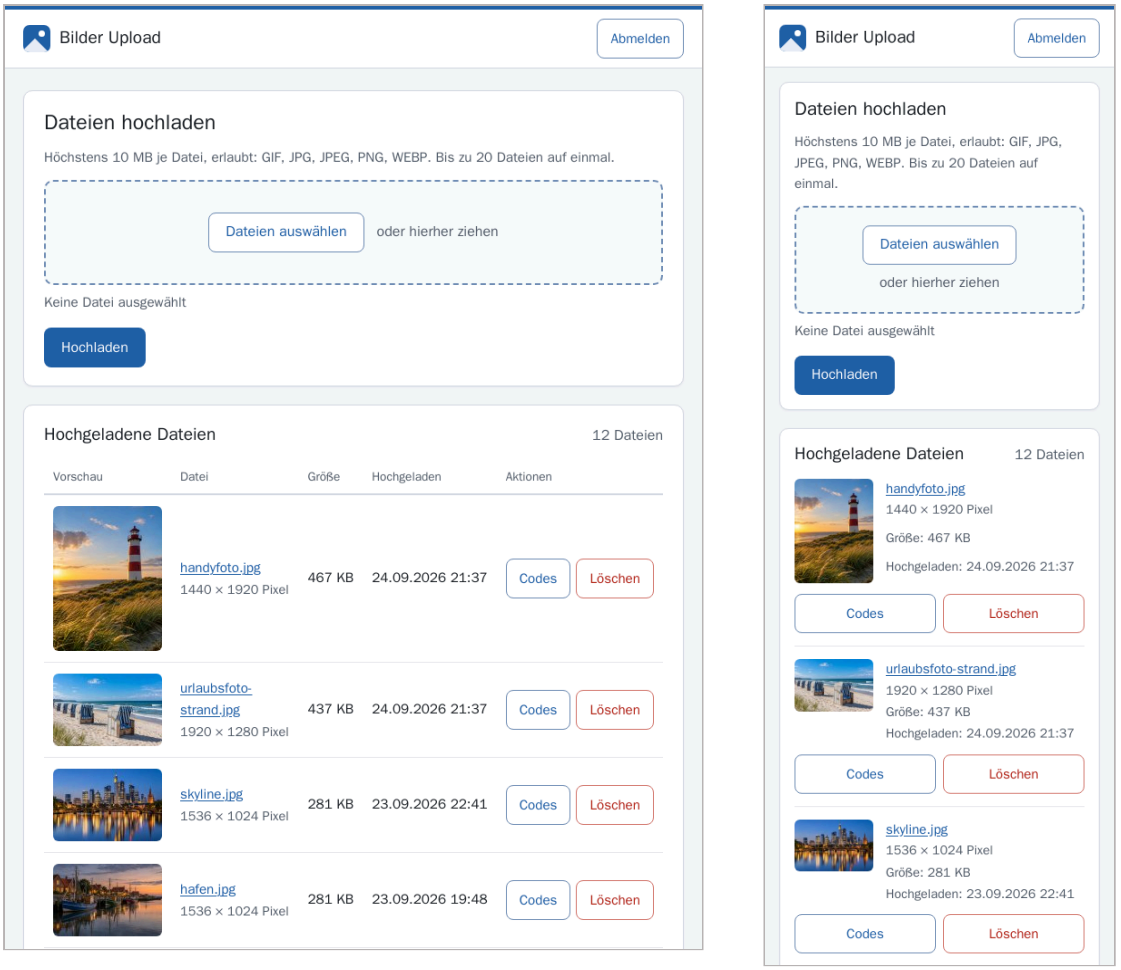
Oben rechts im Kopfbereich steht, solange Sie angemeldet sind, der Knopf **Abmelden**. Danach meldet die Seite *Sie sind abgemeldet.* Melden Sie sich immer ab, wenn Sie an einem fremden oder gemeinsam genutzten Rechner arbeiten.



Die Anmeldung mit dem Hinweis für ein vergessenes Passwort.

### Dateien hochladen

Die Hauptseite zeigt oben den Kasten *Dateien hochladen* und darunter die Liste der hochgeladenen Dateien. Über dem Auswahlfeld nennt das Script, was gilt, zum Beispiel *Höchstens 10 MB je Datei, erlaubt: GIF, JPG, JPEG, PNG, WEBP. Bis zu 20 Dateien auf einmal.* Die Größe ist der kleinste Wert aus `$groessemax` in der `config.php` und den PHP-Einstellungen `upload_max_filesize` und `post_max_size` Ihres Webspace, die Anzahl kommt aus der PHP-Einstellung `max_file_uploads`. Was dort steht, gilt also wirklich.




Die Hauptseite am Rechner (Ausschnitt) und auf dem Handy, wo jede Datei eine eigene Karte bekommt.

Klicken Sie auf **Dateien auswählen** und wählen Sie eine oder mehrere Dateien aus. Am Rechner markieren Sie mehrere Dateien mit gedrückter Strg-Taste (am Mac mit der Befehlstaste), am Handy wählen Sie mehrere Fotos aus der Galerie. Sie können die Dateien auch mit der Maus in das gestrichelte Feld ziehen. Unter dem Feld steht dann zum Beispiel *2 Dateien ausgewählt: hafen.jpg, markt.jpg*. Klicken Sie danach auf **Hochladen**. Ist JavaScript im Browser abgeschaltet, erscheint an dieser Stelle das gewohnte Dateifeld des Browsers.

Das Script prüft jede Datei einzeln. Für die gelungenen meldet es *Die Datei wurde hochgeladen.* bzw. *2 Dateien wurden hochgeladen.*, für jede abgewiesene Datei eine eigene Zeile mit Namen und Grund, etwa *foto.bmp: Dieser Dateityp ist nicht erlaubt. Erlaubt sind nur Bilder: GIF, JPG, JPEG, PNG, WEBP.* Eine abgewiesene Datei hält die übrigen nicht auf. Kommt keine Datei durch, bleiben Sie mit den Meldungen auf der Hauptseite.

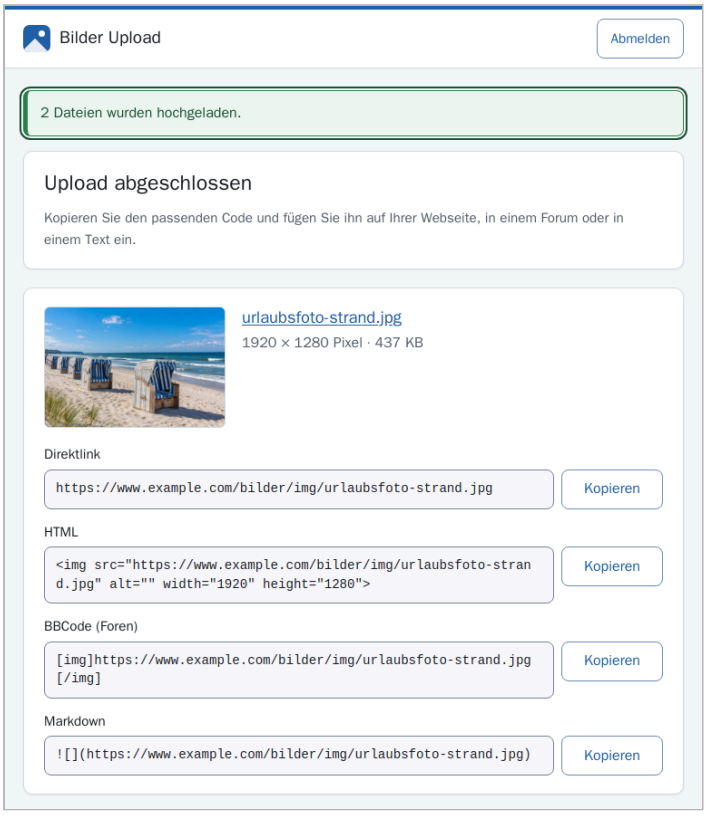
### Wie das Script die Dateien speichert

- **Dateinamen.** Das Script schreibt den Namen klein, ersetzt Umlaute (ä wird ae, ß wird ss) und Akzente vieler Sprachen (é wird e, ł wird l, ñ wird n) und macht aus Leerzeichen, Klammern und anderen Sonderzeichen einen Bindestrich. Nur der Punkt vor der Endung bleibt. Aus `Urlaub (1).JPG` wird `urlaub-1.jpg`, aus `Sommerfest Übersicht.JPG` wird `sommerfest-uebersicht.jpg`.
- **Nichts wird überschrieben.** Gibt es den Namen schon, hängt das Script eine Zahl an: aus `hafen.jpg` wird `hafen-1.jpg`, dann `hafen-2.jpg`.
- **Die Endung passt zum Inhalt.** Ein PNG-Bild mit dem Namen `foto.jpg` speichert das Script als `foto.png`, sofern PNG erlaubt ist. Eine PDF-Datei muss auch ein PDF sein, ein ZIP ein ZIP; eine Textdatei (`.txt`) darf jeden Text enthalten oder leer sein.

- **Erst fertig, dann sichtbar.** Während das Script eine Datei bearbeitet, liegt sie unter einem versteckten Namen. In der Liste erscheint sie erst, wenn sie fertig ist. Bricht der Upload ab, etwa weil der Server die Laufzeit begrenzt, räumt das Script die halbe Datei selbst weg.
- **Bilder werden neu gespeichert** (Grundeinstellung). Das entfernt eingeschmuggelten Code und alle Metadaten, auch die GPS-Position eines Handyfotos. Hochformatfotos dreht das Script richtig herum, und Bilder, deren längste Kante mehr als 1920 Pixel misst, verkleinert es auf 1920 Pixel. PNG-, GIF- und WebP-Bilder behalten ihre Transparenz. Animierte GIFs werden dabei zu Standbildern.
- **Sehr große Bilder.** Bilder mit mehr als 50 Megapixeln weist das Script ab (Schalter `$bild_max_megapixel`), zum Beispiel mit *Das Bild hat 100,0 Megapixel. Erlaubt sind höchstens 50 Megapixel.* Das schützt vor präparierten Dateien, die klein aussehen, beim Öffnen aber riesige Mengen Arbeitsspeicher belegen.
- **Zu wenig Arbeitsspeicher.** Vor dem Bearbeiten schätzt das Script, wie viel Arbeitsspeicher ein Bild braucht. Reicht er nicht, weist es das Bild mit einer Meldung ab, zum Beispiel *Das Bild ist mit 8000 × 6000 Pixeln zu groß für den Arbeitsspeicher Ihres Webspace.* Der Rest der Meldung empfiehlt, das Bild vorher zu verkleinern oder den Hoster um ein höheres `memory_limit` zu bitten. Fehlt der Speicher nur für das Vorschaubild, speichert das Script das Bild trotzdem, nur ohne Vorschau. Lässt sich ein Bild gar nicht lesen, meldet das Script *Das Bild konnte nicht verarbeitet werden und wurde nicht gespeichert.*

### Einbindecodes kopieren

Nach dem Hochladen zeigt die Seite *Upload abgeschlossen* für jede Datei das Bild, den Namen, bei Bildern die Maße und die Dateigröße (zum Beispiel *1920 × 1280 Pixel · 437 KB*) und vier Felder mit fertigen Codes. Die Felder sind so hoch, dass der ganze Code zu sehen ist. Neben jedem Feld steht der Knopf **Kopieren**. Ein Klick legt den Code in die Zwischenablage, der Knopf zeigt dann zwei Sekunden lang *Kopiert*. Ein Klick in ein Feld markiert den ganzen Code, sodass Sie ihn auch von Hand kopieren können. Mit **Weitere Dateien hochladen** kommen Sie zurück zur Hauptseite.



Nach dem Hochladen: Maße, Größe und die vier Einbindecodes, hier für die Beispieladresse example.com.

| Feld | Beispiel für ein Bild | Wofür |
|---|---|---|
| Direktlink | `https://www.ihre-domain.de/bilder/img/hafen.jpg` | zum Weitergeben per E-Mail oder Messenger |
| HTML | `<img src="https://www.ihre-domain.de/bilder/img/hafen.jpg" alt="" width="1536" height="1024">` | für Ihre Website. Tragen Sie bei `alt` eine kurze Beschreibung des Bildes ein. |
| BBCode (Foren) | `[img]https://www.ihre-domain.de/bilder/img/hafen.jpg[/img]` | für Forenbeiträge |
| Markdown | `![](https://www.ihre-domain.de/bilder/img/hafen.jpg)` | für Wikis, Notizprogramme und Chats, die Markdown verstehen |

Bei Dateien, die keine Bilder sind, enthalten die Felder stattdessen einen Link auf die Datei: `<a href="…">name.pdf</a>`, `[url]…[/url]` und `[name.pdf](…)`. Die Adresse ermittelt das Script selbst. Beginnen die Codes mit `http://`, obwohl Ihre Seite über HTTPS läuft, tragen Sie die Adresse in `$basis_adresse` ein (siehe Konfiguration).

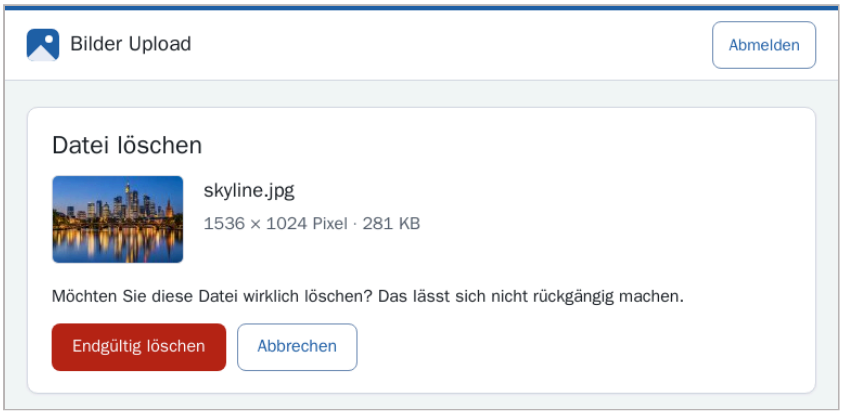
### Die Dateiliste

Unter dem Formular steht die Liste *Hochgeladene Dateien* mit der Anzahl aller Dateien, die neueste oben. Jede Zeile zeigt das Vorschaubild, den Namen mit den Maßen, die Größe, den Zeitpunkt des Hochladens und zwei Knöpfe. **Codes** öffnet eine Seite mit den Einbindecodes dieser Datei, **Löschen** führt zur Rückfrage. Ein Klick auf den Namen öffnet die Datei in einem neuen Fenster. Dateien ohne Vorschau, etwa PDF-Dokumente, tragen ein kleines Schild mit ihrer Endung. Auf dem Handy wird jede Zeile zu einer Karte, seitliches Verschieben ist nicht nötig.

Sind es mehr Dateien, als auf eine Seite passen (in der Grundeinstellung 10), blättern Sie unter der Liste mit **« Zurück** und **Weiter »**. Dazwischen steht, wo Sie sich befinden, zum Beispiel *Seite 1 von 2*.

### Dateien löschen

Ein Klick auf **Löschen** löscht noch nichts. Er öffnet die Seite *Datei löschen* mit Vorschau, Name, Maßen und Größe und fragt *Möchten Sie diese Datei wirklich löschen? Das lässt sich nicht rückgängig machen.* Erst der Knopf **Endgültig löschen** entfernt die Datei samt Vorschaubild, die Seite meldet dann *Die Datei wurde gelöscht.* Mit **Abbrechen** kommen Sie ohne Änderung zurück. Einen Papierkorb gibt es nicht.



Vor dem Löschen fragt das Script nach.

Steht `$delete_link` in der `config.php` auf `0`, zeigt das Script keine Löschen-Knöpfe und lehnt jeden Löschversuch ab. Dateien lassen sich dann nur noch per FTP entfernen.

## 5. Konfiguration (config.php)

Alle Einstellungen stehen in der Datei `config.php` im Hauptverzeichnis des Scripts, jede mit einer kurzen Erklärung. Öffnen Sie sie mit einem Texteditor, zum Beispiel Notepad++ oder dem Editor Ihres FTP-Programms, nicht mit Word. Speichern Sie die Datei in UTF-8 und laden Sie sie wieder hoch. Jede Einstellung ist eine Zeile wie diese:

```
$groessemax = 10240;
$erlaubte_endungen_img = ['gif', 'jpg', 'jpeg', 'png', 'webp'];
```

Ändern Sie nur den Wert zwischen `=` und `;`. Texte stehen in einfachen Anführungszeichen, Zahlen ohne. Bei Schaltern bedeutet `1` ja und `0` nein, `true` und `false` versteht das Script ebenso. Fehlt eine Einstellung, etwa in einer `config.php` aus Version 1.12 oder 1.13, oder ist ihr Wert unbrauchbar, verwendet das Script den Standardwert aus der Tabelle.

| Schalter | Standard | Bedeutung und Empfehlung |
|---|---|---|
| **Zugriffsschutz und Adresse** | | |
| `$zugriffsschutz` | `1` | 1 = Hochladen und Löschen nur nach Anmeldung mit Passwort (empfohlen). 0 = jeder Besucher der Seite kann hochladen und löschen. Stellen Sie 0 nur ein, wenn der Ordner schon anders geschützt ist, zum Beispiel durch einen Verzeichnisschutz im Kundenmenü Ihres Hosters. |
| `$basis_adresse` | `''` (automatisch) | Adresse des Scripts für die Einbindecodes, zum Beispiel `'https://www.ihre-domain.de/bilder'`. Leer lassen, dann erkennt das Script die Adresse selbst (empfohlen). Nötig ist die Angabe etwa, wenn Ihr Hoster HTTPS über einen Proxy bereitstellt und die Codes sonst mit `http://` beginnen. |
| **Ordner** | | |
| `$img_ordner` | `'/img/'` | Ordner für die hochgeladenen Dateien, vom Ordner des Scripts aus gesehen. Empfehlung: nicht ändern, denn nur dieser Ordner bringt die schützende `.htaccess` mit. |
| `$thumbnail_ordner` | `'/img/thumbnail/'` | Ordner für die Vorschaubilder. Empfehlung: nicht ändern. |
| **Hochladen** | | |
| `$groessemax` | `10240` | Größte erlaubte Datei in KB, 10240 KB sind 10 MB. Gibt Ihr Server eine kleinere Grenze vor, gilt diese, und das Script zeigt sie über dem Formular an. |
| `$upload_erlaubnis` | `1` | 1 = nur Bilder (empfohlen). 0 = auch die Dateitypen aus `$erlaubte_endungen`. |
| `$erlaubte_endungen_img` | `['gif', 'jpg', 'jpeg', 'png', 'webp']` | Erlaubte Endungen bei `$upload_erlaubnis = 1`. Groß- und Kleinschreibung spielt keine Rolle. WebP nimmt das Script nur an, wenn der Server WebP-Bilder verarbeiten kann. |

| Schalter | Standard | Bedeutung und Empfehlung |
|---|---|---|
| `$erlaubte_endungen` | `['gif', 'jpg', 'jpeg', 'png', 'webp', 'txt', 'pdf', 'zip', 'rar']` | Erlaubte Endungen bei `$upload_erlaubnis = 0`. Tragen Sie nur Endungen ein, die Sie wirklich brauchen. |
| `$dateityp_einschraenkung_img` | `['image/gif', 'image/jpeg', 'image/png', 'image/webp']` | Erlaubte Bildinhalte (MIME-Typen). Das Script bestimmt den Typ aus dem Inhalt der Datei, nicht aus der Endung. Möglich sind nur diese vier Typen. |
| `$dateityp_einschraenkung` | `text/plain`, `application/pdf`, `application/zip`, `application/x-zip-compressed`, `application/vnd.rar`, `application/x-rar`, `application/x-rar-compressed` | Weitere erlaubte Inhalte bei `$upload_erlaubnis = 0`. Eine Datei kommt nur durch, wenn Endung und Inhalt beide erlaubt sind. Für RAR stehen drei Schreibweisen in der Liste, weil Server den Typ unterschiedlich benennen. |
| `$dateiname_dynamisch` | `0` | 0 = der bereinigte Name der Datei bleibt erhalten. 1 = das Script vergibt Namen wie `datei_1758736800_3fa94c07b2e1.jpg`. Sinnvoll, wenn die Originalnamen nichts verraten sollen. |
| **Bildverarbeitung** | | |
| `$bild_neu_enkodieren` | `1` | 1 = jedes Bild wird neu gespeichert (dringend empfohlen). Das entfernt versteckten Programmcode und alle Metadaten samt GPS-Position und dreht Hochformatfotos richtig. Animierte GIFs werden dabei zu Standbildern. 0 = Bilder bleiben genau so, wie sie hochgeladen wurden. |
| `$bild_max_kante` | `1920` | Längste Kante in Pixel, größere Bilder verkleinert das Script. 0 = nie verkleinern. Wirkt nur mit `$bild_neu_enkodieren = 1`. 1920 passt für Webseiten und Foren und spart viel Speicherplatz. |
| `$bild_max_megapixel` | `50` | Größte erlaubte Bildgröße in Megapixel (Breite × Höhe in Millionen Pixel). Schützt vor präparierten Dateien, die beim Öffnen riesige Mengen Arbeitsspeicher belegen. 50 genügt für Fotos aller üblichen Handys und Kameras. 0 = keine Grenze. |
| `$bildqualitaet` | `85` | Qualität für JPEG- und WebP-Bilder von 1 (klein, unscharf) bis 100 (groß). 85 ist ein guter Kompromiss aus Qualität und Dateigröße. |
| `$thumbnail_create` | `1` | 1 = Vorschaubilder für die Dateiliste erzeugen. 0 = keine Vorschau. |

| Schalter | Standard | Bedeutung und Empfehlung |
|---|---|---|
| `$thumbnail_neuebreite` | `120` | Breite der Vorschaubilder in Pixel, höchstens doppelt so hoch wie breit. Nach einer Änderung erzeugt das Script die Vorschaubilder beim nächsten Aufruf der Liste neu. |
| **Dateiliste** | | |
| `$length` | `10` | So viele Dateien stehen auf einer Seite der Liste. |
| `$delete_link` | `1` | 1 = jede Datei hat einen Knopf *Löschen* mit Rückfrage. 0 = Dateien lassen sich nur per FTP löschen. |
| **Sprache, Zeit und Protokoll** | | |
| `$language_file` | `'language_de_sie.php'` | Sprachdatei aus dem Ordner `language`, siehe Sprachdateien. |
| `$timezone` | `'Europe/Berlin'` | Zeitzone für die Uhrzeiten in der Dateiliste. |
| `$log_datei` | `'daten/fehler.log.php'` | Fehlerprotokoll. Hier landen Fehlermeldungen von PHP, statt im Browser sichtbar zu werden. Laden Sie die Datei per FTP herunter und öffnen Sie sie mit einem Texteditor. Die erste Zeile `<?php exit; ?>` sorgt dafür, dass ein Aufruf im Browser nichts anzeigt. Ist der Ordner `daten` nicht beschreibbar, protokolliert das Script nichts. Über 1 MB benennt es die Datei in `fehler.alt.log.php` um und beginnt neu. |
| `$index_bot` | `0` | 0 = Suchmaschinen sollen die Seiten des Scripts nicht aufnehmen (empfohlen, die Seite ist ein Werkzeug für Sie). 1 = aufnehmen erlaubt. |

Entfallen sind `$charset` (das Script arbeitet immer mit UTF-8) und `$log_file`. Stehen sie noch in einer alten `config.php`, beachtet das Script sie nicht.

> **Gesperrte Endungen.** Endungen, die ein Server ausführen oder als Webseite anzeigen könnte, nimmt das Script nie an, auch wenn sie in der `config.php` stehen. Dazu gehören `php` in allen Varianten, `phtml`, `phar`, `cgi`, `pl`, `py`, `asp`, `jsp`, `sh`, `html`, `htm`, `shtml`, `xhtml`, `xml`, `xsl`, `svg`, `js`, `exe`, `bat` und `.htaccess`.

## 6. Sicherheit

Wer Dateien auf Ihren Webspace laden kann, kann dort auch Schaden anrichten. Version 2.0 schützt Sie an den folgenden Stellen. Drei davon hängen von Ihnen ab: ein gutes Passwort, HTTPS und die hochgeladenen `.htaccess`-Dateien.

- **Zugriffsschutz.** Ohne Anmeldung kann niemand hochladen oder löschen. Das Passwort legt das Script nur als Hash ab, im Klartext steht es nirgends. Legen Sie ein neues Passwort fest, sind alle bisherigen Anmeldungen ungültig.
- **Anmeldesperre und Bremse.** Nach fünf falschen Passwörtern innerhalb von 15 Minuten sperrt das Script die Anmeldung von dieser IP-Adresse für 15 Minuten (bei IPv6 für den ganzen Anschluss). Gibt

es sehr viele Fehlversuche von verschiedenen Adressen, nimmt es nur noch einen Versuch alle 10 Sekunden an. Die Adressen speichert es nur als Prüfsumme mit einem geheimen Schlüssel in `daten/anmeldesperre.php`; den Schlüssel legt es beim ersten Anmelden in `daten/schluessel.php` an. Kann das Script diese Dateien nicht schreiben, lehnt es jede Anmeldung ab, statt ohne Schutz vor Durchprobieren weiterzumachen.

- **Geschützte Formulare.** Jedes Formular trägt einen geheimen Schlüssel der Sitzung. Ein Link oder Formular von einer fremden Seite kann deshalb weder hochladen noch löschen, und Löschen geht nur über die Rückfrage-Seite. Ist der Schlüssel abgelaufen, meldet die Seite *Die Sitzung ist abgelaufen. Bitte versuchen Sie es noch einmal.*
- **HTTPS.** Rufen Sie das Script nur über `https://` auf. Ohne Verschlüsselung gehen Passwort und Sitzung lesbar über das Netz. Über HTTPS schickt das Script das Sitzungs-Cookie nur verschlüsselt.
- **`.htaccess` in `img` und `daten`.** Die Dateien in `img` und `img/thumbnail` verhindern, dass dort abgelegte Dateien als Programm laufen, und lassen eine Datei, die doch als Webseite geöffnet wird, keine Scripte ausführen. Die Datei in `daten` sperrt den Ordner für jeden Aufruf aus dem Browser. Zusätzlich sind alle Dateien in `daten` PHP-Dateien, die beim Aufruf sofort enden, sodass auch ohne Sperre nichts zu sehen ist. Ob die Sperre wirkt, prüfen Sie so: Rufen Sie `https://www.ihre-domain.de/bilder/daten/` auf. Erscheint *Forbidden* oder *Zugriff verweigert* (Fehler 403), ist alles in Ordnung. Sehen Sie dagegen eine leere Seite, fehlt die `.htaccess` oder Ihr Server beachtet sie nicht.
- **Prüfung jeder Datei.** Das Script nimmt nur Endungen aus der Liste in der `config.php` an, gleich in welcher Schreibweise, und nie eine der gesperrten Endungen. Zusätzlich prüft es den tatsächlichen Inhalt der Datei und bei Bildern das Bildformat. Eine Datei `bild.php` kommt nicht durch, und eine Textdatei mit der Endung `.jpg` ebenfalls nicht.
- **Neu speichern entfernt Schadcode und GPS-Position.** Mit `$bild_neu_enkodieren = 1` baut das Script jedes Bild aus den reinen Bilddaten neu auf. Angehängter Code und alle Metadaten fallen dabei weg, auch Kameramodell, Aufnahmezeit und der Ort, an dem ein Handyfoto entstand. Schalten Sie das nur ab, wenn Sie genau wissen, warum.
- **Dateirechte.** Hochgeladene Dateien speichert das Script mit den Rechten `644`, also ohne Ausführungsrecht, die Dateien mit Passwort-Hash und Schlüssel mit `600`.
- **Sicherheits-Kopfzeilen.** Die Seiten des Scripts laden nur eigene Scripte und Stile, und fremde Seiten können sie nicht in einem Rahmen einbetten.

## 7. Passwort vergessen

Ein vergessenes Passwort lässt sich nicht wiederherstellen, weil das Script nur den Hash kennt. Sie setzen es stattdessen zurück. Denselben Hinweis zeigt auch die Anmeldeseite unter dem Formular.

1. Verbinden Sie sich per FTP mit Ihrem Webspace und öffnen Sie im Verzeichnis des Scripts den Ordner `daten`.
2. Löschen Sie dort die Datei `zugang.php`. Ihre hochgeladenen Dateien bleiben unberührt.
3. Rufen Sie das Script im Browser auf. Es zeigt wieder die Seite *Passwort festlegen*. Legen Sie Ihr neues Passwort sofort fest.

Auf demselben Weg ändern Sie ein Passwort, das Sie noch kennen. Mit dem neuen Passwort sind alle bisherigen Anmeldungen ungültig, auch die von Personen, die das alte Passwort kannten. Eine Anmeldesperre nach zu vielen Fehlversuchen endet nach 15 Minuten von selbst. Die übrigen Dateien im Ordner `daten` (`schluessel.php`, `anmeldesperre.php`, `fehler.log.php`) lassen Sie liegen.

## 8. Update von Version 1.x auf 2.0

Ihre hochgeladenen Bilder bleiben beim Update erhalten. Gehen Sie so vor:

1. **Sicherung anlegen.** Laden Sie das Verzeichnis des Scripts per FTP komplett auf Ihren Rechner, mit dem Ordner `img` und Ihrer `config.php`.
2. **Neue Dateien hochladen.** Entpacken Sie das ZIP-Archiv und laden Sie den Inhalt des Ordners `bilder-upload-script_2.0` über die bestehende Installation. Überschreiben Sie dabei alle vorhandenen Dateien, nur Ihre `config.php` nicht, wenn Sie sie behalten möchten (siehe Schritt 4). Alle Programmdateien aus Version 1.x gibt es weiter, sie sind aber neu geschrieben. Dazu kommen `anmeldung.php`, `bildverarbeitung.php`, `codes.php`, `version.php`, der Ordner `js` (mit `kopieren.js`, `dateiwahl.js` und `meldungen.js`) und `css/ohne-js.css`. Im Ordner `img` bringt das Paket nur die Schutzdateien `.htaccess` und `index.html` mit, Ihre Bilder werden also nicht angetastet. Die beiden `.htaccess`-Dateien für `img` und `img/thumbnail` müssen aber hoch, in Version 1.12 gab es sie noch nicht.
3. **Ordner `daten` einrichten.** Der Ordner ist neu. Achten Sie darauf, dass er samt `.htaccess` und `index.html` auf dem Server liegt, und setzen Sie seine Rechte wie bei der Neuinstallation zuerst auf `755`.
4. **Einstellungen übernehmen.** Ihre alte `config.php` läuft weiter, denn für jede neue Einstellung hat das Script einen Standardwert. Wir empfehlen trotzdem die neue `config.php` aus dem Paket: Tragen Sie dort Ihre eigenen Werte ein, zum Beispiel Sprache, Höchstgröße und erlaubte Endungen, und laden Sie die Datei hoch. Nur so sehen Sie alle neuen Schalter mit ihrer Erklärung. Behalten Sie die alte Datei, bleiben Ihre bisherigen Werte gültig, also etwa 3 Dateien je Seite und 70 Pixel breite Vorschaubilder. Mit der `config.php` aus Version 1.13 weist das Script außerdem WebP-Bilder ab, weil deren Listen WebP noch nicht enthalten.
5. **Alte `web.log` löschen.** Das ist die einzige Datei, die Sie entfernen müssen. Das Fehlerprotokoll lag bis Version 1.13 als `web.log` im Hauptverzeichnis und war über den Browser lesbar, mit Serverpfaden. Version 2.0 schreibt nach `daten/fehler.log.php`. Liegen im Ordner `language` noch Archive `language_*.rar`, dürfen sie bleiben, das Script beachtet sie nicht.
6. **Passwort festlegen.** Rufen Sie das Script sofort auf. Weil es noch kein Passwort gibt, zeigt es die Seite *Passwort festlegen*.

Die Vorschaubilder aus Version 1.x ersetzt das Script beim ersten Aufruf der Dateiliste durch neue in der eingestellten Breite, wenn die alte Breite nicht mehr passt. Mit der neuen `config.php` sind das 120 Pixel.

## 9. Neu in Version 2.0

Version 2.0 enthält alle Verbesserungen der Version 1.13, die nicht einzeln auf php-space.info erschienen ist. Gegenüber der bisher angebotenen Version 1.12 ändert sich Folgendes:

- **Passwortschutz.** Beim ersten Aufruf legen Sie ein Passwort fest. Nach fünf Fehlversuchen sperrt das Script die Anmeldung für 15 Minuten, bei sehr vielen Fehlversuchen bremst es alle Versuche. Bisher konnte jeder Besucher hochladen und löschen.
- **Löschen mit Rückfrage.** Gelöscht wird erst nach Bestätigung. Bisher genügte ein einfacher Link, den auch eine Suchmaschine beim Durchsuchen der Seite oder ein fremder Link auslösen konnte.
- **Mehrere Dateien auf einmal**, auch per Hineinziehen, mit einer Rückmeldung für jede einzelne Datei.
- **Einbindecodes** für Direktlink, HTML, BBCode und Markdown mit Kopieren-Knopf, in Feldern, die den ganzen Code zeigen, nach dem Hochladen und für jede Datei über den Knopf *Codes*.

- **Nichts wird überschrieben.** Bei gleichem Namen hängt das Script `-1`, `-2` an. Bisher ersetzte eine neue Datei still die alte.
- **Dateinamen werden bereinigt statt abgewiesen.** Klammern und andere Sonderzeichen werden zu Bindestrichen, und ein Bild mit falscher Endung bekommt die passende.
- **Handyfotos stehen richtig herum**, große Fotos werden auf 1920 Pixel verkleinert, die Qualität ist einstellbar. Ist ein Bild zu groß für den Arbeitsspeicher, erscheint eine Meldung statt eines Abbruchs. Präparierte Riesenbilder hält die Grenze von 50 Megapixeln ab.
- **Datenschutz.** Das Neu-Speichern entfernt die GPS-Position und alle übrigen Metadaten.
- **WebP**, wenn der Server es unterstützt.
- **Bessere Vorschaubilder.** 120 statt 70 Pixel breit, scharf verkleinert, mit Transparenz bei PNG, GIF und WebP statt schwarzem Hintergrund. Das Script erzeugt sie nur, wenn sie fehlen oder veraltet sind, und nicht mehr bei jedem Seitenaufruf.
- **Neue Dateiliste.** Neueste Dateien zuerst, mit Maßen, Größe und Datum, 10 Dateien je Seite und einer Seitenangabe beim Blättern. Auf dem Handy erscheint jede Datei als Karte.
- **Neue Oberfläche.** Gut lesbare Schrift, große Knöpfe, sichtbare Markierung bei Bedienung mit der Tastatur, auf dem Handy ohne seitliches Verschieben. Bildschirmleser erfahren, zu welchem Feld eine Fehlermeldung gehört, und landen nach dem Absenden direkt auf der Meldung.
- **Einrichtung prüfen.** Fehlende oder nicht beschreibbare Ordner, eine fehlende GD- oder fileinfo-Erweiterung und abgeschaltete Uploads nennt das Script auf einer Seite, mit dem Knopf *Erneut prüfen*. Rechte 755 genügen, bisher verlangte das Script zwingend 777.
- **Verständliche Meldungen.** Eine zu große Datei heißt jetzt auch so. Bisher erschien in diesem Fall *Bitte wählen Sie eine Datei aus*. Alle deutschen Texte sind überarbeitet.
- **Geschütztes Fehlerprotokoll** im Ordner `daten`, auch auf Servern ohne `.htaccess`. Fehlt die Datei, legt das Script sie an, statt mit *No Log File existing.* abzubrechen.
- **Richtige Adressen** auch bei einer Installation direkt im Hauptverzeichnis der Domain (bisher entstanden Adressen mit `//img/`). Für Sonderfälle lässt sich die Adresse mit `$basis_adresse` fest einstellen.
- **Sprachdateien.** Fehlt in einer Sprachdatei ein Text, zeigt das Script den deutschen Text statt einer Fehlermeldung. Alle neuen Texte gibt es in allen sieben Sprachen.
- **Sicherheit.** Gesperrte Endungen, Prüfung des Dateiinhalts, Neu-Speichern gegen eingeschmuggelten Code, `.htaccess` im Bilderordner, geschützte Formulare und Sicherheits-Kopfzeilen.
- **PHP 8.2 bis 8.5** ohne Warnungen, durchgängig UTF-8.

## 10. Häufige Fragen und bekannte Probleme

**Die Seite zeigt Bitte prüfen Sie die Einrichtung.**

Das Script hat beim Aufruf etwas gefunden, das fehlt. Die Meldung sagt, was zu tun ist. Meist fehlen Ordner oder Schreibrechte: Setzen Sie die Rechte der genannten Ordner auf `755` (siehe Installation, Schritt 4). Hilft das nicht, läuft PHP bei Ihrem Hoster unter einem anderen Benutzer als Ihr FTP-Zugang. Dann verlangt der Hoster `777` für diese Ordner, für keine anderen. Fehlt der Ordner `daten` ganz, hat das FTP-Programm ihn nicht übertragen. Legen Sie ihn an und laden Sie `.htaccess` und `index.html` aus dem Paket hinein. Fehlen GD oder fileinfo oder sind Uploads abgeschaltet, kann das nur Ihr Hoster ändern. Danach klicken Sie auf
**Erneut prüfen**
.

**Die Seite zeigt nur Datei fehlt / file missing: … .**

Eine Programmdatei ist nicht auf dem Server angekommen. Laden Sie die genannte Datei aus dem Paket per FTP nach.

**Die Datei ist angeblich zu groß, obwohl `$groessemax` höher steht.**

Neben der `config.php` begrenzt PHP selbst die Größe, mit `upload_max_filesize` für eine Datei und `post_max_size` für alle Dateien eines Durchgangs zusammen. Viele Hoster stellen nur 2 oder 8 MB ein. Die Grenze über dem Formular zeigt immer den kleinsten der Werte. Erhöhen lassen sich die PHP-Werte meist im Kundenmenü unter den PHP-Einstellungen, bei manchen Hostern über eine Datei `.user.ini` im Verzeichnis des Scripts:

```
upload_max_filesize = 20M
post_max_size = 40M
```

**Von mehreren Dateien kommt nur ein Teil an.**

PHP nimmt je Durchgang höchstens `max_file_uploads` Dateien an, die Zahl steht über dem Formular. Außerdem müssen alle Dateien zusammen unter `post_max_size` bleiben. Laden Sie große Mengen in mehreren Durchgängen hoch.

**Ein Bild ist zu groß für den Arbeitsspeicher.**

Sehr große Bilder braucht das Script beim Neu-Speichern kurz vollständig im Arbeitsspeicher. Verkleinern Sie das Bild vor dem Hochladen oder bitten Sie Ihren Hoster um ein höheres `memory_limit`. Ein Foto mit 24 Megapixeln läuft mit 128 MB. Heißt die Meldung *Das Bild hat … Megapixel*, liegt das Bild über der Grenze aus `$bild_max_megapixel`.

**Meine Datei heißt nach dem Hochladen anders.**

Das ist gewollt. Das Script bereinigt den Namen, hängt bei einem vorhandenen Namen `-1` an und berichtigt eine Endung, die nicht zum Bildinhalt passt (siehe Bedienung). Mit `$dateiname_dynamisch = 1` vergibt es die Namen ganz selbst.

**Animierte GIFs bewegen sich nicht mehr.**

Das Neu-Speichern behält nur das erste Bild einer Animation. Das ist der Preis für den Schutz vor eingeschmuggeltem Code. Mit `$bild_neu_enkodieren = 0` bleiben Animationen erhalten, dann entfallen aber dieser Schutz, das Entfernen der GPS-Position, das Drehen von Handyfotos und das Verkleinern großer Bilder. Wir raten davon ab.

**Handyfotos liegen quer.**

Das Script dreht Fotos beim Hochladen nach der Angabe der Kamera, wenn `$bild_neu_enkodieren = 1` steht. Bereits vorher hochgeladene Fotos dreht es nicht nachträglich, laden Sie sie in dem Fall neu hoch.

**WebP-Bilder werden abgewiesen.**

Entweder kann die GD-Erweiterung Ihres Servers kein WebP, dann fehlt WEBP auch in der Liste über dem Formular. Ohne diese Fähigkeit könnte das Script WebP-Bilder nicht neu speichern und damit nicht von eingeschmuggeltem Code befreien. Oder Sie verwenden noch die `config.php` aus Version 1.13, in deren Listen `webp` und `image/webp` fehlen.

**Die Codes beginnen mit http:// statt https://.**

Manche Hoster leiten HTTPS über einen vorgeschalteten Server weiter, und das Script sieht davon nichts. Tragen Sie die Adresse dann fest ein, zum Beispiel `$basis_adresse = 'https://www.ihre-domain.de/bilder';`

**Mein Webspace läuft mit nginx, die `.htaccess`-Dateien wirken nicht.**

nginx liest keine `.htaccess`. Die Dateien im Ordner `daten` sind trotzdem geschützt, weil sie beim Aufruf sofort enden. Die Ausführung von Scripts im Bilderordner ist aber nicht abgeschaltet. Bitten Sie Ihren Hoster, das und die Sperre des Ordners `daten` in der Serverkonfiguration einzurichten, oder tragen Sie es selbst ein, wenn Sie Zugriff darauf haben. Für ein Script im Verzeichnis `/bilder/` sieht das so aus. Die zweite Regel muss vor dem Block stehen, der PHP-Dateien an PHP weiterreicht:

```
location ^~ /bilder/daten/ { deny all; }
location ~* ^/bilder/img/.*\.(php\d*|phtml?|pht|phar|phps|cgi|pl|py|sh)$ { deny all; }
```

Prüfen Sie danach wie im Abschnitt Sicherheit beschrieben, ob `/bilder/daten/` mit Fehler 403 antwortet. Dasselbe gilt für Apache-Server, bei denen der Hoster `.htaccess`-Dateien abgeschaltet hat.

**Die Meldung lautet: Die Anmeldung ist gerade nicht möglich.**

Das Script kann im Ordner `daten` die Anmeldesperre oder den Schlüssel nicht schreiben. Setzen Sie die Rechte des Ordners auf `755`, nur wenn Ihr Hoster es verlangt, auf `777`. Liegt dort eine Datei `anmeldesperre.php`, die Ihrem FTP-Zugang gehört, löschen Sie sie; das Script legt sie neu an.

**Die Meldung lautet: Die Sitzung ist abgelaufen.**

Die Seite war zu lange geöffnet, oder Ihr Browser blockiert Cookies für die Domain. Laden Sie die Seite neu und versuchen Sie es noch einmal. Das Script braucht ein Cookie, um Sie angemeldet zu halten.

**Die Meldung lautet: Die Datei daten/zugang.php ist beschädigt.**

Die Datei mit dem Passwort-Hash lässt sich nicht lesen, etwa nach einer unvollständigen Übertragung per FTP. Löschen Sie sie und legen Sie ein neues Passwort fest wie unter Passwort vergessen beschrieben.

**Es ist ein unerwarteter Fehler aufgetreten.**

Einzelheiten stehen im Fehlerprotokoll `daten/fehler.log.php`, das Sie per FTP herunterladen und mit einem Texteditor öffnen. Mit dieser Angabe hilft Ihnen auch unser Support-Forum weiter.

## 11. Nutzungshinweis und Copyright

Für jeglichen Schaden, der Ihnen durch die Benutzung dieses Scripts entsteht, übernehmen wir keine Haftung oder juristische Verantwortung.

Der Copyright-Hinweis *Programmierung: www.php-space.info* mit Link im Fuß der Seite und die Hinweise in den Dateiköpfen müssen erhalten bleiben. Eine Version ohne Copyright-Hinweis können Sie über das Bestellformular für Script-Erweiterungen erwerben: www.php-space.info/script-erweiterung-bestellen/

## 12. Support, Vorschläge und Erweiterungen

Bei Fragen zum Script hilft Ihnen unser Support-Forum. Dort nehmen wir auch Vorschläge und Kritik gerne entgegen.

- Support-Forum: www.php-space.info/forum/
- Info zum Script: www.php-space.info/bilder-upload-script/
- Installationsservice: www.php-space.info/script-installation-service/

Sie brauchen eine Erweiterung für das Script oder eine individuelle Programmierung? Nehmen Sie Kontakt mit uns auf, wir machen Ihnen ein passendes Angebot.

## 13. Sprachdateien

Alle Texte der Oberfläche stehen in den Sprachdateien im Ordner `language`. Mitgeliefert werden sieben Sprachen:

| Datei | Sprache |
|---|---|
| `language_de_sie.php` | Deutsch, Anrede mit Sie (Standard) |
| `language_en.php` | Englisch |
| `language_es.php` | Spanisch |
| `language_fr.php` | Französisch |
| `language_nl.php` | Niederländisch |
| `language_pl.php` | Polnisch |
| `language_arabic.php` | Arabisch, von rechts nach links |

Die Sprache wählen Sie in der `config.php`, zum Beispiel `$language_file = 'language_en.php';`. Das Script lädt immer zuerst die deutsche Datei und legt die gewählte Sprache darüber. Fehlt in einer älteren oder selbst übersetzten Datei ein Text, erscheint an dieser Stelle der deutsche. Die Sprachdatei legt auch Sprache, Schreibrichtung, Dezimalzeichen und Datumsformat der Seite fest. Der Copyright-Hinweis im Fuß bleibt in jeder Sprache gleich.

Einzelne Sprachen gibt es auf der Seite des Scripts auch als eigenes Archiv, etwa `language_en.zip`. Entpacken Sie es im Hauptverzeichnis des Scripts, dann landet die Datei im Ordner `language`.

### Sprachdatei übersetzt?

Für eine neue Sprache kopieren Sie `language_de_sie.php`, zum Beispiel nach `language_it.php`. Der Dateiname muss mit `language_` beginnen und darf danach nur Kleinbuchstaben und Unterstriche enthalten. Übersetzen Sie nur die Texte rechts vom Gleichheitszeichen. Die Namen in den eckigen Klammern und Platzhalter wie `~anzahl~` bleiben unverändert. Speichern Sie die Datei in UTF-8. Wenn Sie uns Ihre Übersetzung kostenlos zur Verfügung stellen, erhalten Sie als Dankeschön eine Version ohne Copyright-Hinweis. Übersetzungen bieten wir kostenlos zum Download an: [www.php-space.info/bilder-upload-script/](www.php-space.info/bilder-upload-script/)