

.htaccess

Dieses Modul ist eine Erweiterung der grundlegenden Funktionalität von WebSitePanel und implementiert den am Apache-Server übliche Rechtestruktur für Web-Inhalte.

(<http://www.helicontech.com/ape/>)

Dieses Rechtesystem vergibt Rechte auf Ordner-ebene, nicht auf Dateiebene. Wenn man einen Ordner nur für Berechtigte freigeben will, geht man so vor:

- (1) Man legt einen User an (im Beispiel den User „privat“)
- (2) Man legt eine Gruppe Gprivat an (ist nicht erforderlich aber es ist immer eine gute Technik, nicht einem einzelnen User sondern immer nur einer Gruppe von Usern Rechte zu verleihen.)
- (3) Man definiert in jenem Folder, in dem die Zugriffsrechte für den User privat gelten sollen eine Datei htaccess und konfiguriert diese

über das Symbol

Beispiel: Website <http://test.pcnews.at> (zeigt die Datei default.htm von ClubComputer, ungeschützt). Die Datei <http://test.pcnews.at/privat/test.htm> wird nicht angezeigt, es erscheint eine Identifikationsmaske. Nach Eingabe von User: privat und PW: privat1 sieht man den Demoinhalt.

Der Inhalt der Datei .htaccess im Ordner privat ist

```
AuthType Basic
Require valid-user user privat group Gprivat
AuthName "PrivaterFolder"
AuthBasicProvider file
AuthUserFile
D:\HostingSpaces\ffialamember\test.pcnews.at\wwwroot\.htpasswd
AuthGroupFile
D:\HostingSpaces\ffialamember\test.pcnews.at\wwwroot\.htgroups
```

Über diese Rechtfunktionen hinaus bietet aber HeliconApe eine Vielzahl anderer Funktionen, die man am besten durch ein Studium der Hilfe-Seiten erforscht (**Bild 17, Link Help**).

Erweiterungen (Bild 18)

Erweiterungen sind die am Server installierten Websprachen. In diesem Menüpunkt kann man sie ein- und ausschalten.

ASP.NET muss aktiviert bleiben, weil sich das System in der Datei web.config gewisse Einstellungen (zum Beispiel jene für das Startdokument) merkt und damit der Server darauf zugreifen kann, kann man ASP.NET nicht deaktivieren.

In den Erweiterungen muss man nur dann etwas konfigurieren, wenn man eine Web-Anwendung installiert oder wenn man eigene Programme in Betrieb nehmen will. In beiden Fällen muss die entsprechende Sprache aktiviert sein.

Bild 18: Erweiterungen

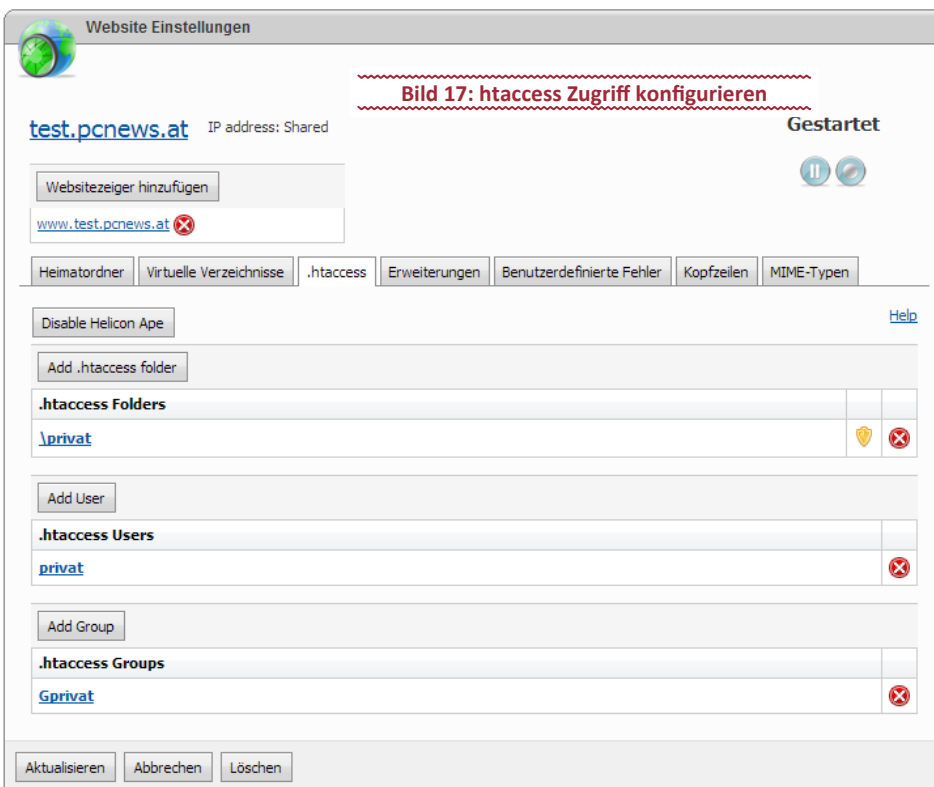
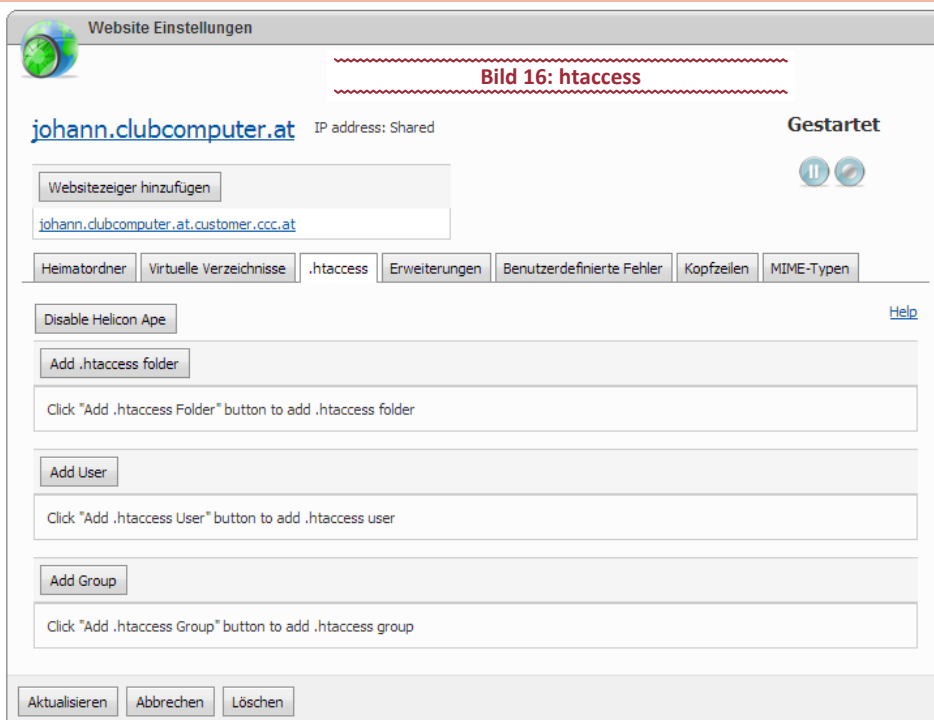
ASP: Aktiviert

ASP.NET:

PHP:

Perl: Aktiviert

CGI-BIN: Installiert



Benutzerdefinierte Fehler (Bild 19)

Gibt der Benutzer des Webs eine Seite ein, die nicht existiert oder enthält die Verlinkung innerhalb des Webs einen Fehler, erscheinen systemnahe Fehlerseiten, die für den Benutzer verwirrend sind. Man kann für die häufig vorkommenden Fehler, allen voran 404 „not found“ eigene Fehlerseiten definieren, die im Stil der eigenen Webseite formatiert sind und die auch Hinweise enthalten, was der Benutzer jetzt tun soll. Und so wird eine konkrete Fehlerseite angegeben:

Man kann aber statt einer Datei auch eine Umleitung angeben.

Bild 19

Benutzerspezifische Fehlermeldungen hinzufügen

Code	Handler-Typ	Fehlerinhalt
	Datei	

Kopfzeilen (Bild 20)

Hier könnte man Zeilen für typische Header-Zeilen hinzufügen, wie zum Beispiel, author, description, language, keywords usw. Damit erspart man sich, diese Angaben in jeder einzelnen Datei angeben zu müssen, sie werden vom Server automatisch in HTML-Dateien eingefügt.

Bild 20: Kopfzeilen definieren

Benutzerspezifischen Header hinzufügen

Kopfzeilenname	Kopfzeilenwert
X-Powered-By	ASP.NET