

NAME

akfwebserver – HTTP-Server

SYNTAX

akfwebserver [*Optionen*]

BESCHREIBUNG

Das Programm **akfwebserver** ist ein kleiner Webserver.

Es kann von normalen Benutzern bei Bedarf gestartet werden. Es antwortet per Voreinstellung nur auf Zugriffe von lokalen Netzen, während er Zugriffe aus dem Internet blockiert.

Er kann aber auch als Webserver für das Internet mit einer zentralen Konfigurationsdatei eingesetzt werden. Für HTTPS wird zusätzliche Software benötigt. Zum Beispiel stunnel.

Wenn dieser Server von einem normalen, nicht privilegierten Benutzer aufgerufen wird, verwendet er den Port **8080**. Dieser Port muss dann auch beim Abruf im Browser explizit mit angegeben werden. Zum Beispiel: **http://localhost:8080/**.

Wenn das Programm in einem Terminal im Vordergrund läuft, kann man es einfach mit der Tastenkombination [**Strg**]+[**C**] abbrechen. Ansonsten kann man es auch über die Befehle **killall**(1) oder **kill**(1) beenden.

Das Programm ist mehrsprachig: Größtenteils wird Deutsch, Englisch und Esperanto unterstützt. Verzeichnisdarstellungen unterstützen mehr Sprachen. Die Sprache bei Verzeichnisdarstellungen wird von der jeweiligen Browser-Einstellung beeinflusst (Accept-Language), kann aber auch manuell gewählt werden. Ausgaben am Terminal sind von der Systemeinstellung abhängig.

Achtung: Die Option `--gzip` wurde überarbeitet. Dateien, die mit einem Punkt beginnen, sind abrufbar, außer sie beginnen mit **.ht**. Das Verzeichnis `»/cgi-bin«` wird anders behandelt.

OPTIONEN

`-h`

`--help`

`--Hilfe` Zeigt eine kurze Hilfe an.

`-V`

`--version`

`--Version`

Zeigt die Version an.

`-c Datei`

`--config=Datei`

`--Konfiguration=Datei`

Lies die Konfiguration aus der Datei. Erlaubt sind Optionen ohne die Bindestriche, eine Option pro Zeile. Leerzeilen und Zeilen, die mit `#` beginnen, werden ignoriert. Die Konfigurationsdatei kann auch weitere Konfigurationsdateien aufrufen.

`-d Verzeichnis`

`--directory=Verzeichnis`

`--Verzeichnis=Verzeichnis`

Legt das angegebene *Verzeichnis* als Wurzelverzeichnis fest. Ohne diese Angabe wird das Verzeichnis, in dem das Programm gestartet wurde zum Wurzelverzeichnis.

`--virtual`

`--virtuell`

Aktiviert Namensbasiertes Virtual Hosting.

Im Wurzelverzeichnis müssen dabei Unterverzeichnisse mit den unterstützen Domainnamen kleingeschrieben angelegt werden. Somit gibt es verschiedene Wuzelverzeichnisse für die unterschiedlichen Domainnamen. Wenn ein Verzeichnis für mehrere Domainnamen gelten soll, kann man symbolische Links anlegen. Domainnamen, für die kein Unterverzeichnis gefunden wird, oder in dessen Verzeichnis der Server nicht wechseln kann, erzeugen eine Fehlerseite (403).

Internationale Domainnamen müssen in ASCII-kompatibler Schreibweise (ACE) angelegt werden.

CGI-Variablen werden automatisch angepasst.

Ohne diese Option wird der Domainname ignoriert.

-N *Servername*

--Servername=*Servername*

--servername=*Servername*

Legt den *Servernamen* fest. (Für CGI.)

Internationale Domainnamen müssen in ASCII-kompatibler Schreibweise (ACE) angegeben werden.

Bei Verwendung von Virtual Hosting wird dieser Name nicht verwendet.

-p *Portnummer*

--port=*Portnummer*

--Port=*Portnummer*

Legt die *Portnummer* fest. Dies ist der Port auf dem dieser Server lauscht. [IP]

--Serverport=*Portnummer*

--serverport=*Portnummer*

Legt eine nominelle andere *Portnummer* fest. Falls der Server über ein Frontend gesteuert wird, kann man hiermit dessen Portnummer angeben. Dieser Port wird für die CGI-Variable `SERVER_PORT` verwendet. [IP, CGI]

-l

--localhost

Hört nur auf die **localhost**-Adresse für IPv4. Das ist **127.0.0.1**. Damit sollte der Server nur von dem selben Rechner aus erreichbar sein. [IP]

-a *IP-Adresse*

--address=*IP-Adresse*

--Adresse=*IP-Adresse*

Bindet den Server an die angegebene *IP-Adresse*.

Dieses kann man nutzen, um die Erreichbarkeit auf ein lokales Netzwerk zu begrenzen. Das ist sicherer, als der eingebaute Filter. [IP]

-4

--IPv4 IPv4 verwenden. [IP]

-6

--IPv6 IPv6 verwenden. (Vorgabe) [IP]

Der Server behält sich jedoch das Anrecht vor, die Verbindung nach Gutdünken zu einem gemeinen IPv4 zu degradieren.

Ob im IPv6-Modus auch IPv4-Verbindungen unterstützt werden, hängt vom Betriebssystem und eventuell dessen Einstellungen ab.

-U *Socketpfad*

--socket=*Socketpfad*

--Socket=*Socketpfad*

Unix-Socket anlegen.

Es muss ein absoluter Pfad angegeben werden. Der Server kann dann nur über diesen Socket angesprochen werden. Das funktioniert nur für Programme auf dem selben Gerät. Der Server ist damit nicht über ein Netzwerk erreichbar. Das ist insbesondere sinnvoll, wenn der Server von einem Frontend gesteuert wird.

Als *Socketpfad* eignet sich zum Beispiel `»/tmp/http«`. Der Socket ist für alle Benutzer des Gerätes verwendbar. Falls man auch das einschränken möchte, kann man ihn in einem Verzeichnis mit eingeschränkten Zugriffsrechten anlegen lassen.

--privacy

--Datenschutz

Die IP-Adresse oder der Rechnername steht dann nicht im Logbuch. Bei angemeldeten Benutzern wird nur das Stichwort »login« anstelle des Namens im Logbuch verzeichnet.

Für CGI-Programme ist die IP-Adresse und eventuell der Rechnername aber zugänglich.

-L *Datei*

--log=*Datei*

Logbuch in die *Datei* schreiben

-s

--still nichts ausgeben

-v

--verbose

--wortreich

ausführlichere Ausgabe

-g

--gzip Auch nach komprimierten Dateien mit der Dateierdung ».gz« suchen, wenn der Klient gzip unterstützt. (Accept-Encoding)

Der Dateiname muss mit einer Tilde enden. Dadurch wird er auch beim automatischen Verzeichnisindex nicht angezeigt.

Es findet keine automatische Kompression statt. Diese Dateien müssen zusätzlich zu den unkomprimierten Dateien vorhanden sein.

-G *Datei*

--cgi-file=*Datei*

--cgi-Datei=*Datei*

Die angegebene Datei als CGI-Programm kennzeichnen. Es können so mehrere Dateien gekennzeichnet werden. Diese Dateien müssen ausführbar sein. Der Vorteil dieser Methode liegt darin, dass man schönere Pfade erzeugen kann, und dass CGI-Programme nicht ohne Weiteres als solche erkennbar sind. CGI-Programme können so übrigens auch die Erweiterung ».html« haben.

-C

--cgi Aktiviert CGI-Unterstützung für Dateien mit der Dateierdung **.cgi**. Diese Dateien müssen ausführbar sein. Sie können in jedem exportierten Verzeichnis liegen. Für Verzeichnisaufrufe kann man auch **index.cgi**-Dateien anlegen.

-e *Erweiterung=Aktion*

--extension=*Erweiterung=Aktion*

--Erweiterung=*Erweiterung=Aktion*

Aktion für Datei-*Erweiterung* festlegen. Die *Aktion* sollte ein Pf ad zu einem CGI-Programm sein. Dieses kann auf die Datei über die Variable `PATH_TRANSLATED` zugreifen.

Mit dieser Option kann man unter anderem auch PHP anbinden.

-M *Maske*

--cgi-umask=*Maske*

CGI Programme mit der angegebenen Dateimodus-Erstellungsmaske (Umask) aufrufen. *Mask e* muss eine Oktalzahl sein. Bei besonders sensiblen Daten ist zum Beispiel der Wert 027 oder gar 077 empfehlenswert.

-u *Nutzerkennung*

--Nutzer=*Nutzerkennung*

Wenn der Server mit root-Rechten gestartet wird (reale Nutzerkennung), kann man über diese Option eine andere *Nutzerkennung* angeben, mit deren Rechten er letztendlich laufen soll. Die *Nutzerkennung* kann ein Name, oder eine numerische Angabe sein.

-D

--daemon

Startet den HTTP-Server als Daemon im Hintergrund.

-b

--writable

--beschreibbar

Vorsicht! Das sollte nicht auf öffentlichen Servern verwendet werden!

Macht das exportierte Verzeichnis und Unterverzeichnisse beschreibbar. Es können Dateien hochgeladen (PUT) und entfernt (DELETE) werden.

Dieser Server wird nicht wirklich Dateien löschen, sondern legt Sicherheitskopien an, wenn Dateien entfernt oder bevor sie überschrieben werden. Diese Sicherheitskopien sind über den Server nicht mehr zugänglich. Der Server-Betreiber kann sie wieder herstellen, oder löschen. Er sollte sie regelmäßig löschen, sonst häufen die sich an.

-P

--writeprotected

--Schreibschutz

Vorsicht! Das sollte nicht auf öffentlichen Servern verwendet werden!

wie bei --beschreibbar, aber Schreibzugriffe werden mit Zugangsdaten geschützt. Genauer gesagt gelten die Zugangsdaten hierbei nur für Schreibzugriffe mit PUT oder DELETE, während sie ansonsten für alle Zugriffe gelten.

Achtung: Dieser Schreibschutz betrifft nicht CGI-Programme!

--protected-zone=*Verzeichnis*

--Schutzbereich=*Verzeichnis*

Wenn ein Schutzbereich definiert ist, werden Zugangsdaten für dieses *Verzeichnis* und dessen Unterverzeichnisse benötigt. Das *Verzeichnis* muss mit einem Schrägstrich beginnen. Wenn nur ein Schrägstrich angegeben ist, werden die Zugangsdaten immer benötigt. Es kann zwar nur ein einziges *Verzeichnis* als Schutzbereich deklariert werden, aber dieser Bereich kann in beliebig viele Unterverzeichnisse eingeteilt werden. Diese Einstellung gilt für alle virtuellen Hosts. Diese Option kann nicht zusammen mit --Schreibschutz verwendet werden.

Es können personalisierte Unterverzeichnisse angelegt werden. Diese fangen mit einer Tilde (~) an, gefolgt vom benötigten Benutzernamen.

Bitte die Hinweise im Abschnitt »Zugangsdaten« beachten.

-B

--block-protected-zone

--Schutzbereichsblockierung

Der Schutzbereich wird blockiert.

Man benutze dies, wenn der Server unverschlüsselt angeboten wird, aber auch gleichzeitig verschlüsselt ohne Blockierung. Die betrifft auch die Option --Schreibschutz.

--account=*Name:Kennung*

--Zugang=*Name:Kennung*

Legt Zugangsdaten fest.

Diese Option kann mehrfach verwendet werden, um mehrere Zugänge zu erlauben. Bitte die Hinweise im Abschnitt »Zugangsdaten« beachten.

--open

--offen

--Internet

Vorsicht! Das schaltet den Filter für private Netzwerke ab. Der Server ist dann auch über das Internet erreichbar, wenn der Zugang nicht über -I oder -a eingeschränkt wurde. Man muss dafür eventuell auch den Port an seinem Router freigeben.

Es besteht die Gefahr, dass dadurch ausversehen Daten öffentlich zugänglich gemacht werden, die nicht dafür vorgesehen sind. [IP]

`--hostname`

ermittle die Namen der anfragenden Rechner

Die Namen werden dann im Logbuch verwendet und stehen für CGI-Programme zur Verfügung (REMOTE_HOST).

Dies kostet Zeit und ist im offenen Internet kaum brauchbar. Für lokale Netzwerke kann es aber sehr sinnvoll sein, insbesondere, wenn alle Namen beim Server in der Datei hosts(5) eingetragen sind.

`-H Kopfzeile`

`--header=Kopfzeile`

`--Kopf=Kopfzeile`

HTTP-Kopfzeile hinzufügen. Diese Option kann mehrfach verwendet werden.

Vorsicht: Es wird nicht geprüft, ob diese Kopfzeilen mit anderen in Konflikt stehen.

`--var=VARIABLE=Wert`

Metavariabel hinzufügen. Diese Option kann mehrfach verwendet werden.

Beispiel: `--var=SERVER_ADMIN=webmaster@example.net`

`--servicemode`

`--Wartungsmodus`

Im Wartungsmodus wird grundsätzlich nur eine Fehlerseite angezeigt, die besagt, dass der Server außer Betrieb ist, da daran gearbeitet wird. (HTTP-Status 503) TRACE-Anfragen sind erlaubt und werden normal durchgeführt.

Der Wartungsmodus kann auch über das Signal USR1 aktiviert werden, und mit USR2 deaktiviert werden.

`-T`

`--trace` Erlaube die Methode TRACE.

`-S Sprache`

`--language=Sprache`

`--Sprache=Sprache`

Legt die *Sprache* für lokale Textausgaben, bzw. das Logbuch fest. Gültige Werte sind »de« oder »en«. Diese Einstellung hat keine Auswirkung auf generierte Webseiten.

`--tmpdir=Verzeichnis`

Dieses Verzeichnis wird benutzt, um POST-Daten zwischenspeichern. Es wäre vorteilhaft, wenn das Verzeichnis im Arbeitsspeicher liegt (z.B. mit mfs oder tmpfs).

Falls die zu erwartende Datenmenge bekannt ist und dort nicht genug Platz frei ist, versucht er es nochmal unter »/var/tmp«.

`--onion=http://[...].onion`

`--Onion=http://[...].onion`

Onion-Service bekannt geben (Onion-Location)

Das funktioniert nur, wenn der Server auch über HTTPS erreichbar ist.

`-i`

`--inetd` Nimmt Anfragen auf der Standardeingabe entgegen und gibt Antworten auf der Standardausgabe aus. Wenn die Standardeingabe ein Socket ist, können daraus die Verbindungsdaten ausgelesen werden.

UMGEBUNGSVARIABLEN

TMPDIR

Dieses Verzeichnis wird benutzt, um POST-Daten zwischen zu speichern. Wenn diese Variable nicht gesetzt ist, wird /tmp verwendet. Es wäre vorteilhaft, wenn das Verzeichnis im

Arbeitsspeicher liegt. (Z.B. mit mfs oder tmpfs.)

HOSTNAME, HOST

Falls diese Variable gesetzt ist, wird das als Servername genommen.

LC_ALL, LC_MESSAGES, LANG

diese Variablen beeinflussen die Sprachauswahl für den Bildschirm, oder die Logdatei (Deutsch oder Englisch). Sie haben keinen Einfluss auf generierte Webseiten.

NOTIZEN

Das Wurzelverzeichnis und alle Unterverzeichnisse werden exportiert. In übergeordnete Verzeichnisse kommt man nicht, es sei denn, wenn symbolische Links nach außerhalb verweisen. (CGI-Programme unterliegen jedoch nicht dieser Einschränkung.)

Wenn ein Verzeichnis abgerufen wird, wird erstmal nach Dateien namens **index.shtml**, **index.html** oder **index.cgi** gesucht. Wenn diese Dateien nicht existieren, wird automatisch ein Verzeichnisindex generiert. Symbolische Links werden als normale Dateien, bzw. Verzeichnisse angezeigt.

Dateien oder Verzeichnisse, deren Namen mit **.ht** beginnen, sind überhaupt nicht abrufbar. Dateien, die mit einer Tilde oder auf **.bak** enden, ebenfalls nicht. Der Server tut so, als seien diese Dateien nicht vorhanden (404). Der Zugang zu Gerätetreibern, FIFOs und Sockets ist gesperrt (403).

Reine Textdateien sollten die Dateiendung **.text** oder **.txt** tragen. Der Server geht davon aus, dass diese Dateien in der Kodierung **UTF-8** vorliegen. Wenn man wirklich Textdateien in unterschiedlichen Kodierungen verwenden will, kann man die Dateiendungen **.l1** für Latin1, **.koi8** für KOI8-R, oder **.utf8** für UTF-8 verwenden.

Grundlegende Fähigkeiten

Der Server unterstützt IPv6, IPv4 und Unix-Sockets. Es wird grundsätzlich HTTP/1.1 unterstützt. Alte HTTP/0.9-Anfragen werden aber ebenfalls noch verarbeitet, außer bei virtuellem Hosting. Mit HTTP/1.x sind folgende Methoden anwendbar: GET, HEAD, POST, OPTIONS, TRACE. Mit der Option **--beschreibbar** können zusätzlich PUT und DELETE verwendet werden. Persistente Verbindungen werden bevorzugt. Mit *Rang e* kann man für statische Dateien Teilbereiche in Bytes anfordern. Es werden bedingte Anfragen aufgrund des Änderungsdatums unterstützt (jedoch keine ETags).

root-Rechte

Wenn der Server mit root-Rechten aufgerufen wurde, kann er diese nutzen, um sich an privilegierte Ports zu binden. Insbesondere auch der Standard-Port für HTTP – Port 80 – erfordert root-Rechte. In diesem Fall braucht auch die Konfigurationsdatei nur für root lesbar zu sein. Das ist vor allem wichtig, falls dort Zugangsdaten verzeichnet sind.

Der Server wird aber root-Rechte abschütteln, bevor er Verbindungen entgegen nimmt. Man kann mit der Option **-u** bestimmen, mit welchen Rechten der Server danach laufen soll. Ansonsten wird er versuchen zu ermitteln, von welchem Benutzer er wirklich aufgerufen wurde und im Zweifel die Nutzerkennung »nobody« annehmen. Wenn das alles nicht gelingt, bricht er mit einer Fehlermeldung ab.

Frontends

Für Frontends wird der Header X-Forwarded-For unterstützt, sowie das Proxy-Protokoll, Version 1. Letzteres muss immer erst aktiviert werden.

Beispielkonfiguration für stunnel

Wenn man zum Beispiel HTTPS verwenden will, kann man das Programm stunnel <<https://www.stunnel.org>> verwenden. Um die IP-Adresse des Aufrufers zu erhalten, muss man in der Konfigurationsdatei von stunnel den Parameter »protocol = proxy« setzen.

Im folgenden Beispiel müssen die Pfade angepasst werden.

```
[https]
accept      = 443
connect     = /var/run/www/https
protocol    = proxy
cert        = /etc/letsencrypt/live/example.net/fullchain.pem
```

```
key          = /etc/letsencrypt/live/example.net/privkey.pem
```

Zugangsdaten

Zugangsdaten können im Wurzelverzeichnis in einer Datei namens »**htZugang**« eingetragen werden. Der Dateiname muss mit einem Punkt beginnen. Diese Datei wird zur Laufzeit gelesen, kann also verändert werden. Die Zugriffsrechte sollten eingeschränkt sein. Leerzeilen und Zeilen, die mit einer Raute (#) beginnen, werden ignoriert.

Zugangsdaten können auch über die Option *--Zugang Name:Kennung* angegeben werden. Diese Daten sind für alle virtuellen Hosts gleich und um sie zu ändern, muss der Server neu gestartet werden.

Für Namen und Kennungen darf man nur reine ASCII-Zeichen verwenden; also keine deutschen Umlaute, oder andere Sonderzeichen. Allgemeine Satzzeichen sind aber erlaubt. Leerzeichen sind zwar ebenfalls erlaubt, aber einige Clients könnten Probleme damit haben.

Die Zugangsdaten können im Klartext, oder verschleiert mit Base64 angegeben werden. Verschleierte Daten kann man mit dem Programm **akfnetzpass(1)** erzeugen lassen. Wenn am Anfang ein Doppelpunkt steht, geht er davon aus, dass das verschleiert ist. (Wenn die Daten Müll sind, kommt man damit halt nicht rein.)

Achtung: Die Zugangsdaten werden zwar verschleiert, aber bei HTTP unverschlüsselt übertragen, und das nicht nur einmal, sondern immer wieder. Die Verschleierung schützt nur vor versehentlichem Mitlesen. Ein Angreifer kann das jedoch einfach dekodieren. Man sollte dies also nicht zur Absicherung wirklich sensibler Daten verwenden. Es sei denn, die Übertragung wird durch ein Frontend verschlüsselt.

Der Name darf nicht leer sein. Zugangsdaten mit leerer Kennung sind jedoch erlaubt.

Es können personalisierte Verzeichnisse angelegt werden, die nur von einem Benutzer betreten werden können. Diese fangen mit einer Tilde (~) an, gefolgt vom benötigten Benutzernamen.

Ein CGI-Programm kann feststellen, ob eine Anmeldung erfolgte, indem es prüft, ob die Variable `AUTH_TYPE` existiert und nicht leer ist. Der Name ist dann in der Variablen `REMOTE_USER` zu finden. Wenn Zugangsdaten mitgeschickt werden, obwohl keine Anmeldung nötig war, wird `REMOTE_USER` zugewiesen, aber `AUTH_TYPE` bleibt ungesetzt.

Common Gateway Interface (CGI)

Dieser Server unterstützt CGI/1.1 (RFC 3875). (Er kann aber auch ohne CGI-Unterstützung kompiliert werden.) Das spezielle Verzeichnis »`cgi-bin`« wird immer besonders behandelt.

CGI-Programme sollten vorzugsweise im Verzeichnis »`cgi-bin`« abgelegt werden. Bei diesem Verzeichnis wird kein automatischer Index erstellt. Ausführbare Dateien in diesem Verzeichnis werden als CGI-Programme angesehen. Nicht-ausführbare Dateien sind hingegen nicht zugänglich. Dieses können Konfigurationsdateien oder Daten für CGI-Programme sein. Auch Unterverzeichnisse sind nicht zugänglich.

Mit den Optionen *--cgi* oder *--cgi-Datei* können weitere Dateien als CGI-Programme ausgewiesen werden.

Das Arbeitsverzeichnis wird auf das Verzeichnis gesetzt, in dem das CGI-Programm liegt. Also in der Regel »`cgi-bin`«.

Folgende Variablen werden gemäß der Spezifikation unterstützt: `AUTH_TYPE`, `CONTENT_LENGTH`, `CONTENT_TYPE`, `GATEWAY_INTERFACE`, `PATH_INFO`, `PATH_TRANSLATED`, `QUERY_STRING`, `REMOTE_ADDR`, `REMOTE_HOST`, `REMOTE_USER`, `REQUEST_METHOD`, `SCRIPT_NAME`, `SERVER_NAME`, `SERVER_PORT`, `SERVER_PROTOCOL`, `SERVER_SOFTWARE`, sowie Protokoll-spezifische Variablen, die mit »`HTTP_`« eingeleitet werden.

Die Variable `REMOTE_HOST` wird nur gesetzt, wenn die Option *--hostname* verwendet wurde.

Die Variable `REMOTE_ADDR` bekommt den Wert »`127.0.0.1`« zugewiesen, wenn die IP-Adresse nicht zur Verfügung steht.

Die Variable `REMOTE_IDENT` bleibt ungesetzt, was laut Spezifikation erlaubt ist. Dafür werden aber auch einige zusätzliche Variablen definiert. Programme, die mit anderen Servern laufen können sollen, sollten sich jedoch auf die oben genannten Variablen beschränken.

Erweiterungen: `DOCUMENT_ROOT`, `REQUEST_URI`, `REMOTE_PORT`, `REDIRECT_STATUS`,

REDIRECT_URL

Die Ausgabe des Programmes erfolgt auf der Standardausgabe. Sie muss zunächst einen CGI-Kopf enthalten, dann eine Leerzeile, dann evtl. der Inhalt. Der Kopf muss aus mindestens einer Zeile bestehen. Wenn ein Inhalt folgt, muss der Content-Type angegeben werden. Zum Beispiel »Content-Type: text/plain; charset=UTF-8«. Oft reicht auch nur diese eine Zeile. Besondere Bedeutung bei CGI haben die Zeilen »Status:«, »Location:« und »Content-Type:«. Darüber hinaus können die meisten HTTP-Angaben auch als CGI-Kopfzeilen verwendet werden und werden einfach weitergeleitet.

CGI-Kopfzeilen, die mit »X-CGI-« beginnen, werden ignoriert, und nicht weitergeleitet. Wenn die Variable REQUEST_METHOD den Wert **HEAD** hat, sollte das Programm nur den Kopf ausgeben und keinen Inhalt.

Fehlerausgaben des Programmes werden verworfen (/dev/null). Kommandozeilenparameter (IsIndex) sind auf 50 beschränkt.

Wenn Daten mit der Methode POST übertragen werden, werden diese in einer temporären Datei gespeichert und stehen auf der Standardeingabe zur Verfügung. Die Standardeingabe kann bei diesem Server per **mmap**(2) eingebunden werden. Die Größe der Daten steht in der Variablen CONTENT_LENGTH und der Typ in CONTENT_TYPE.

Für NPH-Skripte muss der Dateiname mit dem Präfix **nph-** anfangen.

PHP kann über die CGI-Schnittstelle angebunden werden. Das ist zwar nicht ideal, aber es funktioniert. Hierfür muss man mit der Option *--Erweiterung* die Dateierweiterung **php** mit dem CGI-Programm für PHP verknüpfen. Die Sicherheitsmaßnahme *cgi.for ce_redirect* wird unterstützt und sollte in PHP aktiviert sein.

AUTOREN

Copyright © 2015-2025 Andreas K. Förster

LIZENZ

Dieses Programm ist freie Software. Sie können es unter den Bedingungen der GNU General Public License, wie von der Free Software Foundation veröffentlicht, weitergeben und/oder modifizieren, entweder gemäß Version 3 der Lizenz oder (nach Ihrer Option) jeder späteren Version.

Die Veröffentlichung dieses Programms erfolgt in der Hoffnung, dass es Ihnen von Nutzen sein wird, aber **ohne irgendeine Garantie**, sogar ohne die implizite Garantie der **Marktreife** oder der **Verwendbarkeit für einen bestimmten Zweck**. Details finden Sie in der GNU General Public License.

Sie sollten ein Exemplar der GNU General Public License zusammen mit diesem Programm erhalten haben. Falls nicht, siehe <<http://www.gnu.org/licenses/>>.

Anmerkung

Auf CGI-Programme, die mit diesem Server ausgeführt werden, hat diese Lizenz erstmal keinen Einfluss. Allerdings bietet akfnetz auch Funktionen zum Schreiben von CGI-Programmen unter C, oder kompatiblen Sprachen. Wenn man diese Funktionen verwendet, muss das CGI-Programm unter der gleichen Lizenz stehen, oder einer Nachfolgeversion; unabhängig davon, mit welchem Server es eingesetzt wird.

SIEHE AUCH

akfnetzpass(1), **akfweb-dl**(1), **akfquery**(1), **lynx**(1)

<https://akfoerster.de/p/akfnetz/>