

Paket DYNDNS - Update für DynamicDNS-Dienste Version 3.10.19

Tobias Gruetzmacher

E-Mail: fli41@portfolio16.de

Das fli4l-Team

E-Mail: team@fli4l.de

2. Februar 2020

Inhaltsverzeichnis

1. Dokumentation des Paketes DYNDNS	3
1.1. DYNDNS - Dynamische Updates für Domain Name Services	3
A. Anhang zum Paket DYNDNS	10
A.1. DYNDNS	10
A.1.1. Hinzufügen von neuen Providern	10
A.1.2. Dank	12
A.1.3. Lizenz	13
Abbildungsverzeichnis	14
Tabellenverzeichnis	15
Index	16

1. Dokumentation des Paketes DYNDNS

1.1. DYNDNS - Dynamische Updates für Domain Name Services

Dieses Paket ist dafür gedacht, automatisch bei jeder Einwahl einen dynamischen Hostname zu aktualisieren. Folgende Dienste werden unterstützt:

Anbieter	FreeDNS (afraid.org)
DYNDNS_x_PROVIDER	AFRAID
Homepage	http://freedns.afraid.org

Wichtig: Als Passwort ist hier der letzte Teil (hinter dem Fragezeichen) der URL anzugeben, die man auf der Homepage von Afraid.org abrufen kann (Einloggen ⇒ „Dynamic DNS“ ⇒ Die URL, die sich hinter dem Link „Direct URL“ versteckt). Alle anderen Angaben werden ignoriert.

Anbieter	Companity
DYNDNS_x_PROVIDER	COMPANITY
Homepage	http://www.staticip.de/

Anbieter	DDNSS
DYNDNS_x_PROVIDER	DDNSS
Homepage	http://www.ddnss.de/

Anbieter	DHS International
DYNDNS_x_PROVIDER	DHS
Homepage	http://www.dhs.org/

Anbieter	DNS2Go
DYNDNS_x_PROVIDER	DNS2GO
Homepage	http://dns2go.com/

Anbieter	DNS-O-Matic
DYNDNS_x_PROVIDER	DNSOMATIC
Homepage	http://www.dnsomatic.com

Anbieter	DtDNS
DYNDNS_x_PROVIDER	DTDNS
Homepage	http://www.dtdns.com/

1. Dokumentation des Paketes DYNDNS

Anbieter	DynAccess
DYNDNS_x_PROVIDER	DYNACCESS
Homepage	http://dynaccess.de/

Wichtig: DynAccess bietet im Rahmen der fli4l-DynAccess-Kooperation für die Subdomains *.dyn-fli4l.de, *.dyn-fli4l.info und *.dyn-eisfair.de Sondertarife an. Informationen hierzu gibt es auf der Internet-Seite <http://www.dyn-fli4l.de/> bzw. <http://www.dyn-eisfair.de/>.

Anbieter	DynDNS.org
DYNDNS_x_PROVIDER	DYNDNS
Homepage	http://dyn.com/

Anbieter	DynDNS.org (custom)
DYNDNS_x_PROVIDER	DYDNSDC
Homepage	http://dyn.com/standard-dns/

Anbieter	DynDNS DK
DYNDNS_x_PROVIDER	DYDNSDK
Homepage	http://dyndns.dk/

Anbieter	dyndns:free
DYNDNS_x_PROVIDER	DYDNSFREE
Homepage	http://dyndnsfree.de/

Anbieter	eisfair.net
DYNDNS_x_PROVIDER	DYNEISFAIR
Homepage	http://www.intersales.de/it-infrastruktur/dyneisfair.html

Wichtig: Mit der Benutzung dieses Dienstes unterstützt man die Arbeit der fli4l- und eisfair-Entwickler.

Anbieter	DyNS
DYNDNS_x_PROVIDER	DYNSCX
Homepage	http://www.dyns.cx/

Anbieter	GnuDIP Dynamic DNS
DYNDNS_x_PROVIDER	GNUDIP
Homepage	http://gnudip2.sourceforge.net/

Anbieter	Provider Hurricane Electric
DYNDNS_x_PROVIDER	HE
Homepage	https://dns.he.net/

1. Dokumentation des Paketes DYNDNS

Anbieter	IN-Berlin e.V.
DYNDNS_x_PROVIDER	INBERLIN
Homepage	http://www.in-berlin.de

Anbieter	KONTENT
DYNDNS_x_PROVIDER	KONTENT
Homepage	http://www.kontent.de/

Anbieter	Nerdcamp.net
DYNDNS_x_PROVIDER	NERDCAMP
Homepage	http://nerdcamp.net/dynamic/dns.cgi

Anbieter	No-IP.com
DYNDNS_x_PROVIDER	NOIP
Homepage	http://www.no-ip.com/

Anbieter	noxaDynDNS
DYNDNS_x_PROVIDER	NOXA
Homepage	http://www.noxa.de/

Anbieter	OVH.DE
DYNDNS_x_PROVIDER	OVHDE
Homepage	http://www.ovh.de/

Anbieter	PHPDYN
DYNDNS_x_PROVIDER	PHPDYN
Homepage	http://www.webnmail.de/phpdyn/

Wichtig: *diese Lösung muß man selber hosten*

Anbieter	Regfish.com
DYNDNS_x_PROVIDER	REGFISH
Homepage	http://www.regfish.de/

Anbieter	SelfHost.de
DYNDNS_x_PROVIDER	SELFHOST
Homepage	http://selfhost.de/cgi-bin/selfhost

Anbieter	Securepoint Dynamic DNS Service
DYNDNS_x_PROVIDER	SPDNS
Homepage	http://www.spdns.de/

1. Dokumentation des Paketes DYNDNS

Anbieter	Strato
DYNDNS_x_PROVIDER	STRATO
Homepage	http://www.strato.de/

Anbieter	T-Link.de
DYNDNS_x_PROVIDER	TLINK
Homepage	http://www.t-link.de/

Anbieter	twodns.de
DYNDNS_x_PROVIDER	TWODNS
Homepage	http://www.twodns.de/

Anbieter	ZoneEdit.com
DYNDNS_x_PROVIDER	ZONEEDIT
Homepage	http://zoneedit.com/

Wir versuchen diese Daten aktuell zu halten. Trotzdem übernehmen wir keine Haftung für die Richtigkeit dieser Daten. Wer einen Fehler oder eine Änderung entdeckt sollte eine Mail an das fli4l-Team (E-Mail: team@fli4l.de) schicken.

Diese Liste ist komplett, andere Provider werden ohne Änderung nicht unterstützt. Wie man das Paket um eigene Anbieter erweitern kann, steht im Anhang.

Der dynamische Hostname wird automatisch bei jeder Einwahl ins Internet aktualisiert. Das Paket beinhaltet eine Sperre, die das mehrmalige aktualisieren der gleichen IP verhindert, da dies bei einigen DynDNS-Anbietern nicht gerne gesehen wird und im Extremfall zur Sperrung des Accounts führen kann.

Hinweis: Es kann einige Minuten dauern, bis die Änderung des dynamischen Hostnamens wirksam wird.

Bevor man mit der Einrichtung dieses Paketes beginnen kann, muss man sich bei einem der oben genannten Anbietern einen Account holen. Falls man das schon hat, kann man sofort loslegen. Hat man noch keinen Account, so kann man sich an obiger Tabelle orientieren, um einen Hostname zu finden, der den Ansprüchen genügt und den persönlichen Geschmack trifft.

Für die nun folgende Konfiguration benötigt man folgende Daten:

- Name des Anbieters
- Benutzername
- Passwort
- Den DynDNS-Hostnamen

Die benötigten Angaben können je nach Anbieter variieren, es wird versucht eine möglichst konsistente Konfiguration zu bieten. Manchmal ist z.B. der Hostname gleich dem Benutzernamen, in so einem Fall werden wir versuchen, immer das Host-Feld zu benutzen und den Benutzernamen einfach ignorieren. Jetzt aber los:

OPT_DYNDNS Steht dieser Parameter auf 'yes', wird OPT_DYNDNS aktiviert.

DYNDNS_SAVE_OUTPUT Wird dieser Parameter auf 'yes' gestellt, wird das Ergebnis der DynDNS-Anfrage(n) in einer Datei gespeichert und kann über den Webserver¹ abgefragt werden.

DYNDNS_N Hat man bei mehreren DynDNS-Anbietern einen Account und will deswegen bei jeder Einwahl mehrere Namen updaten, so ist dieser Wert entsprechend anzupassen.

DYNDNS_x_PROVIDER Hier wird der Name des zu benutzenden Providers angegeben (siehe Tabelle weiter oben und Hinweis in der Config-Datei).

DYNDNS_x_USER Benutzername bei dem DynDNS-Anbieter. Häufig ist dies eine E-Mail-Adresse, ein selbstgewählter Name oder gleich dem DynDNS-Hostname.

DYNDNS_x_PASSWORD Hier ist das Passwort des DynDNS-Accounts anzugeben. Aufpassen, dass niemand anderes beim Editieren der Config-Datei zusieht!

DYNDNS_x_HOSTNAME Hier ist der *komplette* DynDNS-Hostname des Accounts einzutragen. Beispielsweise könnte hier folgendes stehen:

- cool.nerdcamp.net
- user.dyndns.org
- fli4luser.fli4l.net

DYNDNS_x_UPDATEHOST Hier wird für den Provider PHPDYN angegeben, auf welchem Host der Updater installiert ist. Dies ist nötig, da dies kein herkömmlicher Provider ist sondern nur ein Script, welches einen PowerDNS Server mit MySQL aktualisiert und welches unter der GPL steht.

DYNDNS_x_CIRCUIT Hier kann angegeben werden, bei welchen Circuits dieser Hostname aktualisiert wird. Die einzelnen Circuits werden mit Leerzeichen voneinander getrennt. Es kann z.B. erwünscht sein, den Hostnamen nur bei der DSL-Einwahl zu benutzen. Hier ein paar Beispiele:

```
DYNDNS_1_CIRCUIT='1 2 3'           # Nur ISDN: Circuits 1 bis 3
oder
DYNDNS_1_CIRCUIT='pppoe'           # Nur DSL: pppoe-Circuit
oder
DYNDNS_1_CIRCUIT='dhcp'            # Update bei DHCP-Providern
                                   # (opt_dhcp wird benötigt)
oder
DYNDNS_1_CIRCUIT='pppoe 1'         # DSL und ISDN
oder
DYNDNS_1_CIRCUIT='static'          # fli4l hinter z.B. LTE Router
```

DYNDNS_x_RENEW Manche Provider erwarten, dass alle n Tage ein Update ausgeführt wird, auch wenn sich die IP nicht verändert hat. Dieses Intervall kann man hier angeben. Gibt man keinen Wert an, wird nach 29 Tagen ein Update durchgeführt.

¹OPT_HTTPD im Paket HTTPD (Seite ??) auf <http://www.fli4l.de/download/stabile-version/>

Zu beachten ist hierbei, dass ein Update nur bei einer Einwahl angestoßen wird - also bei einer Einwahl über DSL oder ISDN oder einer Erneuerung einer Lease bei einem via DHCP konfigurierten Interface, wie man es bei einem Kabelmodem findet. Findet über längere Zeit keine Einwahl statt, muß man das Update auf andere Weise anstoßen.

DYNDNS_x_EXT_IP Mit dieser Variable wird die Methode, mit der die externe IP Adresse ermittelt wird, konfiguriert. Im Moment gibt es die Möglichkeit mit 'no' überhaupt keinen externen Dienst nach der IP Adresse zu befragen sondern direkt die externe IP Adresse anhand des WAN Interfaces zu bestimmen. Das funktioniert in der Regel aber nur bei WAN Verbindungen, die direkt auf dem fli4l terminieren, wie z.B. DSL via PPPoE. Mit der Einstellung 'dyndns' wird die beim Update verwendete IP Adresse über den externen Dienst von checkip.dyndns.org ermittelt. Wird die Einstellung 'stun' benutzt wird die Liste der STUN Server der Reihe nach abgefragt bis eine erfolgreiche Antwort geliefert wird. Die Nutzung eines externen Dienstes zur Ermittlung der IP Adresse ist notwendig, wenn der Router selbst nicht derjenige ist, der die externe IP erhält. Dabei ist zu beachten, dass der Router in diesem Falle momentan eine Änderung der externen IP nicht mitbekommt, den dyndns-Namenseintrag also nicht zeitnah aktualisieren kann.

DYNDNS_x_LOGIN Manche Provider erfordern, dass sich der Benutzer regelmäßig auf ihrer Internet-Seite unter seinem Benutzerkonto anmeldet, damit der Dienst nicht deaktiviert wird. Wenn diese Variable auf 'yes' gesetzt ist, erledigt das der fli4l für Sie. Allerdings funktioniert das nur, wenn das dyndns-Paket für den jeweiligen Provider vorbereitet wurde. Momentan ist eine solche regelmäßige Anmeldung nur für den Provider "DYNDNS" erforderlich und möglich. Bedenken Sie bitte auch, dass die Nutzung dieser Funktion `OPT_EASYCRON='yes'` im Paket easycron erfordert.

DYNDNS_LOGINTIME Nutzen Sie einen Provider, bei dem sich der fli4l regelmäßig anmelden soll, um eine Deaktivierung des Dienstes zu verhindern (s.o.), dann können Sie mit dieser Variable einstellen, wann diese Anmeldung stattfinden soll. Benötigt wird eine Zeitangabe im Cron-Format; zu Details lesen Sie sich bitte die Dokumentation des easycron-Pakets durch. Die Standardbelegung lautet `0 8 * * *`, was einer täglichen Anmeldung um acht Uhr morgens entspricht.

DYNDNS_ALLOW_SSL Ist diese Variable auf 'yes' gesetzt, wird das Update wenn möglich über SSL (verschlüsselte Verbindung) durchgeführt.

DYNDNS_LOOKUP_NAMES Ein Update der IP sollte eigentlich nur erfolgen, wenn sich die IP geändert hat. Viele fli4l-Router haben jedoch keinen permanenten Speicher, auf der die Information über die registrierte IP gesichert werden kann, daher steht diese Information direkt nach dem Booten dort nicht zur Verfügung. Um trotzdem unnötige Updates zu vermeiden, kann fli4l in dieser Situation (und nur in dieser Situation) beim Namensdienst nach der aktuell registrierten IP fragen. Die ermittelte IP wird dann zwischengespeichert und für jedes weitere Update genutzt.

Zu beachten ist dabei, dass nach einem Reboot das Update-Intervall neu beginnt, wenn fli4l den Namensdienst zur Ermittlung der IP nutzt.

DYNDNS_DEBUG_PROVIDER Ist diese Variable auf 'yes' gesetzt, wird ein trace des Update-Vorgangs aufgezeichnet, so dass man im Nachhinein bei einem Problem prüfen kann, was schief gegangen ist. Default: `DYNDNS_DEBUG_PROVIDER='no'`

OPT_STUN Mit 'yes' wird die Funktionalität zur Ermittlung der externen IP-Adresse über STUN-Server aktiviert

STUN_SERVER_N Mit dieser Variable wird die Anzahl der STUN-Server definiert.

STUN_SERVER_x FQDN des STUN-Server, optional kann der FQDN um den zu verwendenden Port ergänzt werden

```
STUN_SERVER_1='stun.1.google.com:19302'  
STUN_SERVER_2='stun1.1.google.com:19302'  
STUN_SERVER_3='stun2.1.google.com:19302'  
STUN_SERVER_4='stun3.1.google.com:19302'  
STUN_SERVER_5='stun4.1.google.com:19302'  
STUN_SERVER_6='stun01.sipphone.com'  
STUN_SERVER_7='stun.ekiga.net'  
STUN_SERVER_8='stun.fwdnet.net'  
STUN_SERVER_9='stun.ideasip.com'
```

A. Anhang zum Paket DYNDNS

A.1. DYNDNS

A.1.1. Hinzufügen von neuen Providern

Das Hinzufügen von neuen Providern ist eigentlich sehr leicht, da die Update-Skripts komplett von den Provider-Daten getrennt sind. Für einen neuen Provider müssen folgende Dateien angepasst werden:

Datei `opt/etc/dyndns/provider.NAME`

Dies ist die Datei, in der definiert wird, wie ein Update bei diesem speziellen Provider funktioniert. Meistens besteht sie nur aus einer Liste von Variablen, da es aber ein ganz normales Shell-Skript ist, können hier auch komplexere Operationen durchgeführt werden, das sollte aber selten nötig sein. In dieser Datei können folgende Variablen benutzt werden:

\$ip Die IP des Interfaces, das den dynamischen Hostnamen erhalten soll.

\$host Der komplette Hostname, wie ihn der Benutzer in seiner Konfiguration angegeben hat.

\$subdom Alle Komponenten des Hostnamen bis zum vorletzten Punkt (`name.provider.dom`)

\$domain Die letzten beiden Komponenten des Hostnamen (`name.provider.dom`)

\$provider Der symbolische Name des Providers, wie ihn der Benutzer in seiner Konfigurationsdatei angegeben hat.

\$user Der Benutzername für diesen Dienst.

\$pass Das dazugehörige Passwort.

Diese Variablen können zur klareren Abgrenzung gegenüber anderem Text mit geschweiften Klammern geschrieben werden, aus `$ip` wird z.B. `${ip}`. Bei Verwendung von Anführungszeichen ist zu beachten, dass innerhalb von einfachen Anführungszeichen die oben genannten Variablen *nicht* expandiert werden, bei doppelten Anführungszeichen schon. Als Faustregel kann man also sagen: Immer einfache Anführungszeichen benutzen, aber sobald man Variablen benutzt, doppelte Anführungszeichen benutzen.

Die folgenden Variablen müssen in dieser Datei definiert werden, damit das Paket weiß, wie ein Update bei dem entsprechenden Provider funktioniert:

provider_update_type Dies bestimmt die Art der Anfrage, die an den Server des Providers geschickt wird. Momentan werden unterstützt:

http Es wird automatisiert eine bestimmte Webseite des Providers abgerufen und so der DynDNS-Eintrag aktualisiert.

A. Anhang zum Paket DYNDNS

netcat Es wird einfach ein bestimmter Text an den Server des Providers geschickt, der das Update auslöst.

gnudip Ein relativ einfaches aber sicheres Updateverfahren, welches über zwei HTTP-Anfragen ausgeführt wird.

provider_host Der Hostname des Providers, der bei einem Update kontaktiert wird.

provider_port Der Port auf dem Host des Providers, der angesprochen werden soll. Der Standard-Port für HTTP ist 80.

Je nach Update-Typ müssen weitere Variablen angegeben werden:

HTTP provider_url Hier wird die relative URL (ohne Hostname, aber mit / am Anfang zum Script des Providers abgelegt. Für Beispiele bitte die Dateien der anderen Provider ansehen.

provider_auth (optional) Benötigt der Provider eine Anmeldung per Basic Authentication, so sind hier die entsprechenden Daten anzugeben. Das Format ist "USER:PASSWORD".

Netcat provider_data Dies ist der Text, der an den Server des Providers geschickt wird. Siehe z.B. `provider.DYNEISFAIR`.

GNUDip provider_script Der Pfad zum GNUDip-Script auf dem Server, dies ist meist etwas wie z.B. `'/cgi-bin/gdipupdt.cgi'`.

Datei `opt/dyndns.txt`

Hier müssen eine oder mehr Zeilen für den neuen Provider eingefügt werden. Meistens reicht eine Zeile in der Art:

```
dyndns_%_provider    NAME    etc/dyndns/provider.NAME
```

Wird für den Provider HTTP und Basic Authentication benutzt, so braucht man noch das base64-Tool:

```
dyndns_%_provider    NAME    files/usr/local/bin/base64
```

Sollten noch andere Tools benötigt werden, bitte mir vorher eine Mail schicken, damit geprüft werden kann, ob das für das OPT_DYNDNS geeignet ist.

Datei `check/dyndns.exp`

In dieser Datei muss an der langen Zeile, die mit `DYNPROVIDER =` beginnt, der Providernamen mit einem senkrechten Strich von den anderen abgetrennt, hinten angefügt werden.

Datei `doc/<Sprache>/tex/dyndns/dyndns_main.tex`

In der Dokumentation einen neuen Abschnitt eintragen. Auch hier sind die Provider alphabetisch nach dem Kurznamen, den der Benutzer in der Config-Datei eingibt, sortiert. Das `prov`-Makro ist am Anfang der Datei dokumentiert, genug Beispiele sollten vorhanden sein.

A.1.2. Dank

Als allererstes möchten wir dem danken, der dieses Paket ins Leben gerufen hat und lange Zeit dieses Paket betreut hat: Thomas Müller (E-Mail: opt_dyndns@s2h.cx) hat hier hervorragende Arbeit geleistet, ohne ihn wäre das Paket in der heutigen Form nicht möglich gewesen.

Desweiteren möchten wir auch Marcel Döring (E-Mail: m@rcel.to) danken, der das Paket einige Zeit lang gepflegt hat.

Bei der Entwicklung des Paketes haben sehr viele Leute geholfen und Ideen beigetragen. Mein Dank gilt allen diesen fleißigen Helfern.

Außerdem danken wir Frank Meyer und dem Rest des fli4l-Tems für ihre unermüdliche Arbeit, um den besten Router der Welt zu basteln (Bitte nicht ganz Ernst nehmen ;-).

Weiterhin möchten wir folgenden Leuten danken, die sich mit Tips, neuen Providern, Fehlerberichten, etc. an dem Paket beteiligt haben:

- Paul Bischof für den Provider AFRAID.
- Jens Fischer schrieb das Paket `opt_dtdns`, welches mich erst auf die Idee brachte, ein Paket für DynDNS.org zu schreiben.
- Till Jäger schrieb das Paket `opt_cjb`, welches in `opt_dyndns` übernommen habe.
- Tobias Gruetzmacher hat auf <http://portfolio16.de/index.de> Informationen zu weiteren DynDNS-Anbietern zusammengetragen, die hier verwendet werden.
- Die Anbieter dynamischer DNS, die auf ihren Webseiten zum Teil sehr gute, zum Teil weniger gute Beschreibungen des zu verwendenden Protokolls veröffentlicht haben.
- Die Programmierer diverser Update-Programme für DynDNS Anbieter, aus deren Code schamlos geklaut wurde. ;-)
- Heiko Ambos von dynaccess.de hat mich bei der Entwicklung der Unterstützung für diesen Anbieter unterstützt.
- Dennis Neuhäuser, der die Idee hatte, die Antworten der Dienste per Webserver verfügbar zu machen statt sie auf der Konsole auszugeben und auch gleich eine erste Implementation dafür geschickt hat.
- Lars Winkler der freundlicherweise die Änderungen, um das Paket unter 2.0pre2 zum Laufen zu bringen zur Verfügung gestellt hat.
- Markus Kraft und Tobias Gruetzmacher haben die Grundlage für die Anpassung an fli4l 2.0 gelegt.
- Georg Bärwald für die Daten zu Selfhost.de
- Mark C. Storck für die Daten zu Storck.org
- Arne Biermann für den Hinweis auf den Anbieter hn.org
- Detlef Paschke für die Daten zu dyn.ee und dyndns.dk
- Martin Kisser für seine Idee zum Vermeiden von Updates, wenn die IP sich nicht geändert hat.

- Björn Hoffmann für die Daten von DnsArt.com
- Christian Busch für die Daten von no-ip.com.
- Ralf Gill für das Update der Daten von selfhost.de.
- Michael (HeinB) für eine weitere Möglichkeit sich mit fli4l selbst in den Fuss zu schiessen.
;-)
- Marcus Mönnig, dito.

A.1.3. Lizenz

Copyright ©2001-2002 Thomas Müller (E-Mail: opt_dyndns@s2h.cx)

Copyright ©2002-2003 Tobias Gruetzmacher (E-Mail: fli4l@portfolio16.de)

Copyright ©2004-201x fli4l-Team (E-Mail: team@fli4l.de)

Dieses Programm ist freie Software. Sie können es unter den Bedingungen der GNU General Public License, wie von der Free Software Foundation herausgegeben, weitergeben und/oder modifizieren, entweder unter Version 2 der Lizenz oder (wenn Sie es wünschen) jeder späteren Version.

Die Veröffentlichung dieses Programms erfolgt in der Hoffnung, dass es Ihnen von Nutzen sein wird, aber OHNE JEDE GEWÄHRLEISTUNG - sogar ohne die implizite Gewährleistung der MARKTREIFE oder der EIGNUNG FÜR EINEN BESTIMMTEN ZWECK. Details finden Sie in der GNU General Public License.

Sie sollten eine Kopie der GNU General Public License zusammen mit diesem Programm erhalten haben. Falls nicht, schreiben Sie an die

Free Software Foundation Inc.
59 Temple Place
Suite 330
Boston MA 02111-1307 USA.

Der Text der GNU General Public License ist auch im Internet unter <http://www.gnu.org/licenses/gpl.txt> veröffentlicht. Eine inoffizielle deutsche Übersetzung findet sich unter <http://www.gnu.de/documents/gpl.de.html> Diese Übersetzung soll jedoch nur zu einem besseren Verständnis der GPL verhelfen, rechtsverbindlich ist die englischsprachige Version.

Abbildungsverzeichnis

Tabellenverzeichnis

Index

DYNDNS_ALLOW_SSL, [8](#)
DYNDNS_DEBUG_PROVIDER, [8](#)
DYNDNS_LOGINTIME, [8](#)
DYNDNS_LOOKUP_NAMES, [8](#)
DYNDNS_N, [7](#)
DYNDNS_SAVE_OUTPUT, [6](#)
DYNDNS_x_CIRCUIT, [7](#)
DYNDNS_x_EXT_IP, [8](#)
DYNDNS_x_HOSTNAME, [7](#)
DYNDNS_x_LOGIN, [8](#)
DYNDNS_x_PASSWORD, [7](#)
DYNDNS_x_PROVIDER, [7](#)
DYNDNS_x_RENEW, [7](#)
DYNDNS_x_UPDATEHOST, [7](#)
DYNDNS_x_USER, [7](#)

OPT_DYNDNS, [6](#)
OPT_STUN, [8](#)

STUN_SERVER_N, [9](#)
STUN_SERVER_x, [9](#)