

noxa.de

[Über uns](#)[noxaMail >>](#)[noxaDynDNS >>](#)[Jabber](#)[Weitere Dienste >>](#)[Policy >>](#)[Spenden](#)[Kontaktformular](#)[Impressum](#)

noxaDNS Interface

noxa.de > [Dienste](#) > [noxaDynDNS](#) > InterfaceDieses Dokument beschreibt das Interface des noxaDNS/noxaDynDNS-Backends in der Version 1.0. Es ist abwärtskompatibel zu [0.9.2](#).

Ein > bei Beispielen bezeichnet die Ausgaben des Servers, < entsprechend eine Eingabe des Nutzers/Clients.

[] bei Parametern bezeichnen optionale Angaben. | bezeichnet Alternativen.

Es werden folgende Befehle unterstützt:

Befehle Standard Command Set

Übersicht Befehle			
Befehl	Parameter	Wirkung	setzt voraus
HELP		zeige Befehlsübersicht	
EHELP		zeige Befehlsübersicht für Advanced Command Set	EHLO
MOTD		zeigt Message Of The Day an	
QUIT		beende Verbindung	
HELO	fqhn[(client-clientversion)]	Begrüßung und Austausch eines Salts für Hashs	
EHLO	fqhn[(client-clientversion)]	Begrüßung und Austausch eines Salts für Hashs, aktiviert Advanced Command Set	
USER	user pass	Anmeldung mit Username und Passwort im Klartext	EHLO/HELO
USER	user md5	s.o., aber mit einem MD5-Hash als Kennwort (md5(pass+salt))	EHLO/HELO
USER	user sha1	s.o., aber mit einem SHA1-hash als Kennwort (sha1(pass+salt))	EHLO/HELO
UPDATE[!]	host domain type [value [ttl]]	Update eines Records. Ohne value werden Standardwerte benutzt; bei A-Records etwa die IP-Adresse der Verbindung.	EHLO/HELO, USER
CREATE[!]	host domain [type [value [ttl]]]	erzeuge neuen Host oder Record	EHLO/HELO, USER
DELETE[!]	host domain [type]	lösche Host oder Record	EHLO/HELO, USER
LIST[!]	[pattern [sortorder [limit [typepattern]]]]	liste Einträge auf	EHLO/HELO, USER
CHPASS	newpass	ändere Kennwort	EHLO/HELO, USER
GET	setting	Wert für Einstellung anzeigen	EHLO/HELO, USER
SET	setting new value	Wert für Einstellung setzen	EHLO/HELO, USER

! hinter LIST, UPDATE, CREATE oder DELETE forciert den Befehl, auch wenn ein Host oder eine Zone nicht dem Nutzer zugeordnet ist. ! benötigt besondere Rechte.

Ausgabe von HELP

```

210-Available Commands: (Protocol Version 1.0)
210-HELP - show this help message
210-EHELP - show help for Advanced Command Set (needs EHLO first)
210-MOTD - show message of the day
210-QUIT - exit interface
210-
210-Authorization:
210-HELO fqdn - great each other, get salt
210-EHLO fqdn - great each other, get salt,
                use Advanced Command Set
210-
210-USER user pass - authorize yourself
210-USER user md5 - md5(pass+salt)
210-USER user sha1 - sha1(pass+salt)
210-
210-Updates:
210-UPDATE host domain type [value [ttl]] - update record
210-CREATE host domain [type [value [ttl]]] - create new host
210-DELETE host domain [type]
210- delete type from host entry or purge entire host
210- (use UPDATE!, CREATE!, or DELETE! to force
210- even if not owned or attached to)
210-
210-Misc:
210-LIST [pattern [sortorder [limit [typepattern]]]]
210- list records for this account matching pattern, if given
210-LIST! - as above, but list even unattached zones
210-CHPASS new - change password to new

```

```

210-GET setting          - get value of setting
210-SET setting newvalue - set value
210-
210-Available Record Types:
210-A AAAA MX CNAME NS TXT PTR
210-Available Sort Orders:
210-ASC DESC VALUE VALUEDESC
210-Available Special Pattern Chars:
210-%
210-Available Settings:
210-EMAIL DEFAULT_TTL
210-Available Domains:
210-noxadydns.de h3c.de
210-Blank Char Substitute:
210-#
210 .

```

Befehle Advanced Command Set

Das Advanced Command Set wird durch EHLO statt HELO bei der Begrüßung aktiviert.

Folgende zusätzlichen Kommandos werden unterstützt:

Übersicht Befehle			
Befehl	Parameter	Wirkung	setzt voraus
EHELP		zeige Befehlsübersicht für Advanced Command Set	EHLO
ZONE_ADD	zone master	fügt Zone zum Nameserver als Slave-Zone hinzu	EHLO, USER, Rechte
ZONE_DEL	zone	löscht Zone samt aller Records vom Nameserver	EHLO, USER, Rechte
ZONE_CREATE	zone email.address	erstellt neue Master-Zone mit email@address als Kontakt	EHLO, USER, Rechte
ZONE_CREATE	zone email.address	erstellt neue Master-Zone mit email@address als Kontakt	EHLO, USER, Rechte
ZONE_LIST	[pattern [sortorder [limit [typepattern]]]]	listet Zones auf	EHLO, USER, Rechte
ZONE_ATTACH	zone [user]	weist Zone dem Nutzer user zu, oder einem selbst, falls keine Angabe	EHLO, USER, Rechte
ZONE_DETACH	zone	löscht Zuweisung für Zone	EHLO, USER, Rechte
ZONE_ALIST	zone	liste Zuweisungen auf	EHLO, USER, Rechte
BUMP	zone [serial]	erhöhe Serial oder setze Serial auf Wert	EHLO, USER, Rechte
DELAY_BUMP	true false	automatisches BUMP nach Änderungen deaktivieren oder reaktivieren	EHLO, USER, Rechte
FORWARD	server1[,server2[,serverN]] COMMAND [param1 [param2 [paramN]]]	leite Kommando an weitere(n) Server	EHLO, USER, Rechte
USER_CREATE	user firstname lastname email passwd [userlevel [maxhosts]]	neuen Nutzer anlegen	EHLO, USER, Rechte
USER_CHANGE	user newuserlevel [newmaxhosts [newpasswd]]	Werte für bestehenden Nutzer ändern	EHLO, USER, Rechte
USER_DEL	user	Nutzer löschen	EHLO, USER, Rechte
USER_LIST	[pattern [sortorder [limit]]]	Nutzer auflisten	EHLO, USER, Rechte
USER_HLIST	[pattern [sortorder [limit]]]	Nutzerhosts auflisten	EHLO, USER, Rechte
HOST_ATTACH	fqhost [user]	weist Host dem Nutzer zu, oder einem selbst, falls keine Angabe	EHLO, USER, Rechte
HOST_DETACH	fqhost	löscht Zuweisung für den Host	EHLO, USER, Rechte
SQL	sqlstatement	führt das angebene SQL-Statement auf der SQL-Datenbank aus	EHLO, USER, Rechte
FETCH	[numrows]	holt ein oder mehrere Ergebnisse für das zuletzt ausgeführte Statement	EHLO, USER, Rechte

Ausgabe von EHELP

```

211-Advanced Command Set Available Commands: (Protocol Version 1.0)
211-EHELP - show this help message

```

```

211-HELP - show other help message
211-MOTD - show message of the day
211-QUIT - exit interface
211-
211-Zone Administration:
211-ZONE_ADD zone master - add zone to NS as SLAVE
211-ZONE_DEL zone - delete zone and all records of zone from NS
211-ZONE_CREATE zone email.address
211- create new MASTER zone with email@address as contact
211-ZONE_LIST [pattern [sortorder [limit [typepattern]]]]
211- list zones matching pattern, if given
211-ZONE_ATTACH zone [user]
211- attach zone to user, if omitted attach to ourself
211-ZONE_DETACH zone - detach zone
211-ZONE_ALIST [pattern [sortorder [limit]]]
211- list attachments matching pattern
211-
211-Advanced Administration:
211-BUMP zone [newserial]
211- increment serial of zone or set serial (i.e., to change schema)
211-DELAY_BUMP true|false
211- turn off/on automatic bumping of serial on CREATE|UPDATE|DELETE
211-FORWARD server1[,server2[,serverN]] COMMAND [param1 [param2 [paramN]]]
211- forward command to server(s)
211-
211-User Administration:
211-USER_CREATE user firstname lastname email passwd [userlevel [maxhosts]]
211- create new user
211-USER_CHANGE user newuserlevel [newmaxhosts [newpasswd]]
211- change userlevel or more
211-USER_DEL user - delete user
211-USER_LIST [pattern [sortorder [limit]]]
211- list users matching pattern
211-USER_HLIST [pattern [sortorder [limit]]]
211- list user hosts matching pattern
211-HOST_ATTACH fqhost [user]
211- attach host to user, if omitted attach to ourself
211-HOST_DETACH fqhost - detach host
211-
211-Guru Meditation Commands:
211-SQL sqlstatement - execute SQL statement
211-FETCH [numrows] - fetch results for last SQL statement
211-
211-Available Advanced Record Types:
211-SOA PTR
211-Available Zone Sort Orders:
211-ASC DESC TYPE MASTER NOTIFIED
211-Available Attachment Sort Orders:
211-ASC DESC USER USERDESC
211-Available User Sort Orders:
211-ASC DESC USERLEVEL USERLEVELDESC MAXHOSTS MAXHOSTSDESC
211-Available User Host Sort Orders:
211-ASC DESC USER USERDESC
211 .

```

Rückgabe- und Fehlercodes

Die Codes sind in 4 Gruppen eingeteilt:

Codegruppen

```

2xx positive Antworten, Statusmeldungen
3xx positive Antworten mit Rückgabewerten
4xx temporäre Fehler
5xx fatale Fehler, Eingabe-/Ausgabefehler

```

Auswahl von Fehlercodes für noxaDynDNS

Status- und Fehlercodes

Code	Aussage	vorheriger Befehl
4xx		
450	temporärer Fehler: Hier sollte die Statusabfrage http://noxa.de/?status bemüht werden. Sollte dort kein Fehler aufgeführt sein, bitten wir um Meldung an hostmaster@noxa.de .	*
451	Kein Update nötig, da neuer und alter Wert übereinstimmen. Wir bitten darum, unnötige Updates zu unterlassen.	UPDATE
453	Dienst kurzzeitig deaktiviert.	*
5xx		

500	Unbekannter Fehler.	*
510	Es fehlen Parameter.	*
511	Es wurde kein gültiger Hostname angegeben.	CREATE/UPDATE /DELETE
512	Es wurde eine ungültige Domain angegeben.	CREATE/UPDATE /DELETE
513	Es wurde ein unbekannter Typ angegeben.	CREATE/UPDATE
514	Es wurde keine gültige IP-Adresse angegeben. IPv4/IPv6 Version beachten.	CREATE/UPDATE
515	Es wurde ein unvollständiger oder ungültiger Hostname angegeben.	CREATE/UPDATE /DELETE
52x		
520	Passwort ungültig.	CHPASS
53x		
530	Hostname existiert nicht oder wird von einem anderen Benutzer verwaltet.	CREATE/UPDATE
531	Zuweisung von Typ zu Hostname existiert nicht oder ist bereits vorhanden beim Versuch, die Zuweisung anzulegen.	CREATE/UPDATE
532	CNAME existiert für den Hostnamen, ein weiterer Record kann nicht angelegt werden.	CREATE
533	CNAME soll angelegt werden, aber es existieren bereits andere Records.	CREATE
54x		
540	Hostname existiert nicht oder wird von einem anderen Benutzer verwaltet.	DELETE
55x		
550	Der Benutzername und/oder das Kennwort sind falsch.	USER
560	Zu viele Hostnamen.	CREATE

Alle Status und Fehlercodes

220	%s noxaDNS Backend %s (%s), Protocol Version %s	(on connection)
221	%s disconnecting	(QUIT)
230	user [%s] created	(USER_CREATE)
231	user [%s] changed	(USER_CHANGE)
232	user [%s] deleted	(USER_DEL)
240		(message of the day MOTD)
250	%s, welcome [%s]. {%s}	(HELO/EHLO)
250	Oh, hello again, good friend..	(HELO/EHLO)
250	password for [%s] changed	(CHPASS)
250	success, user [%s] logged in; userlevel (%s) maxhosts (%s)	(USER)
260	host [%s] domain [%s] type [%s] value [%s] updated	(UPDATE)
270	host [%s] domain [%s] type [%s] value [%s] created	(CREATE)
271	host [%s] attached to [%s]	(HOST_ATTACH)
272	host [%s] detached	(HOST_DETACH)
280	host [%s] domain [%s] type [%s] deleted	(DELETE)
290	[%s] is now slave for zone [%s] with master [%s]	(ZONE_ADD)
291	zone [%s] deleted	(ZONE_DEL)
292	zone [%s] created	(ZONE_CREATE)
293	zone [%s] attached to [%s]	(ZONE_ATTACH)
294	zone [%s] detached	(ZONE_DETACH)
295	serial bumped on zone [%s]	(BUMP)
296	session value for DELAY_BUMP changed to [%s]	(DELAY_BUMP)
300		(results from LIST)
301		(results from USER_HLIST)
310		(results from GET)
320		(results from SET)
330		(results from USER_LIST)
350		(results from ZONE_LIST)
351		(results from ZONE_ALIST)
380		(results from SQL)
381		(results from FETCH)
390		(results from FORWARD)
391		(results per server from FORWARD (in 390))
450	temporary failure	(*)
451	no changes made	(UPDATE)
452	cannot get old SOA for zone, can't update serial	(UPDATE, CREATE, DELETE)

453 service currently n/a	(*)
500 please authorize yourself first	(Standard and Advanced Command Set)
500 polite people say EHLO or at least HELO first..	(Standard and Advanced Command Set)
501 already authorized	(USER)
502 unknown command	(*)
503 ProgrammingError: syntax error in your SQL statement [%s] [%s]	(SQL)
504 OperationalError: SQL statement could not be executed [%s] [%s]	(SQL)
505 unknown error occured on SQL statement [%s] [%s]	(SQL)
506 rows needs to be a number	(FETCH)
507 ProgrammingError: guess there was no previous SQL statement	(FETCH)
508 unknown error on fetching results	(FETCH)
	(Standard and Advanced Command Set)
510 missing values, see HELP/EHLP	
511 invalid hostname	
512 invalid domain, valid public domains: %s	
513 unknown type, valid: %s	
514 invalid value, not an IPv4 address	
514 invalid value, not an IPv6 address	
515 invalid value, not a full hostname	
516 invalid value, too long	
517 unknown sort order, valid: %s	
518 invalid ttl	
520 invalid password	(CHPASS)
521 invalid setting, possible keys: %s	(SET, GET)
522 invalid value, possible values: %s	(SET)
530 host [%s] already exists and is not delegated to you	(CREATE)
530 host [%s] does not exist or is not delegated to you	(UPDATE)
531 mapping [%s] for host [%s] already exists	(CREATE)
531 mapping [%s] for host [%s] does not exist	(UPDATE)
532 CNAME exists for host [%s], can not add [%s] record	(CREATE)
533 CNAME requested, but other records exist	(CREATE)
534 unallowed record type for a reverse zone	(CREATE)
535 unallowed record type for a normal zone	(CREATE)
540 host [%s] is not delegated to you or does not exist	(DELETE)
541 no mapping [%s] for host [%s]	(DELETE)
550 authorization failed for [%s]	(Advanced Command Set)
551 your are not authorized to execute this command.	(Advanced Command Set)
560 too many hosts, maximum is %d	(CREATE)
570 need fully qualified domain name	(ZONE_ADD, ZONE_DEL, ZONE_CREATE)
571 need IPv4 or IPv6 address for master	(ZONE_ADD)
572 zone already exists	(ZONE_ADD, ZONE_CREATE)
573 zone does not exist	(ZONE_DEL)
574 email contact is invalid	(ZONE_CREATE)
575 cannot attach zone, already attached or non-existing zone	(ZONE_ATTACH)
577 cannot detach zone or host	(ZONE_DETACH)
580 invalid user name	(USER_CREATE)
581 invalid email address	(USER_CREATE)
583 user [%s] already exists	(USER_CREATE)
584 userlevel needs to be a number	(USER_CREATE)
585 maxhosts needs to be a number	(USER_CREATE)
586 you cannot set userlevels higher than yours	(USER_CREATE)
587 user [%s] not found	(USER_DEL)
595 cannot bump non-existing zone [%s]	(BUMP)
596 cannot bump zone [%s]	(BUMP)
597 invalid serial	(BUMP)
598 expected boolean, TRUE or FALSE	(DELAY_BUMP)

Verfügbare Record-Typen

Derzeit werden A, AAAA, MX, TXT, NS, CNAME, PTR und SOA-Records unterstützt. Von jedem Typ bis auf CNAME und SOA können bis zu 9 Records existieren und unter Angabe einer Ziffer 1 <= x <= 9 als Suffix beim Typen angesprochen werden. Wird die Ziffer weggelassen, so wird 1 angenommen.

Hat bzw. soll ein Host einen CNAME-Record bekommen, so darf kein anderer Record unter dem selben Host existieren.

Bei MX-Records entspricht die Suffix-Ziffer der Priorität.

Wird beim Update von A-Records kein Wert übergeben, so wird automatisch die IP-Adresse der Verbindung eingetragen.

Die Standard TTL (Time To Live) der Records beträgt 3 Minuten.

Verbindung aufnehmen zum noxaDNS-Server

Das Backend ist erreichbar unter telnet://server:64200/ und telnets://server:64201/ (SSL).

Wir empfehlen die grundsätzliche Verwendung einer SSL-gesicherten Verbindung. Sollte dies nicht möglich sein, sollte zumindest die angebotene Challenge Response Authentication mit MD5/SHA1 benutzt werden, damit das Passwort nicht im Klartext uebertragen wird.

Die Backend-Server verwenden untereinander SSL-Verbindungen zur Kommunikation. Sollte dies fehlschlagen, verwenden diese die MD5/SHA1 Authentication. Des weiteren sind diese über IPv6 zu erreichen.

Derzeit sind folgende Server verfügbar:

- dyndns.noxa.de (aka enterprise.noxa.de, a.dns.noxa.de, 212.60.197.71)
- defiant.noxa.de (aka b.dns.noxa.de, 195.122.181.41)
- ds9.noxa.de (aka h.dns.noxa.de, 212.60.197.61)
- g.dns.noxa.de (80.237.205.100, *nicht öffentlich*)

noxaDynDNS-Nutzer verwenden bitte nur dyndns.noxa.de.

Um zu telnets://dyndns.noxa.de/ Kontakt aufzunehmen eignet sich openssl(1) oder ein SSL-fähiger telnet-Client:

```
(Die Schritte 1 und 2 sind optional und nur einmalig erforderlich)

# (1) CA-Cert von noxa.de holen (zur Verifikation):
# z.B. mit curl(1), wget(1) oder lynx(1):
<0>[4:59]niklas@mayhem:~$ curl -s -o noxa-ca.crt http://noxa.de/ca.crt
<0>[4:59]niklas@mayhem:~$ wget -q -O noxa-ca.crt http://noxa.de/ca.crt
<0>[5:00]niklas@enterprise:~$ lynx -dump http://noxa.de/ca.crt > noxa-ca.crt

# (2) Fingerprint überprüfen:
# http://www.noxa.de/?sslcert
<0>[5:00]niklas@mayhem:~$ openssl x509 -noout -fingerprint < noxa-ca.crt
SHA1 Fingerprint=92:11:9B:B6:64:33:B8:0F:D0:AD:A1:F2:CD:76:A3:AA:F3:44:5F:EF

# (3) Verbindung aufnehmen:
<0>[5:01]niklas@mayhem:~$ openssl s_client -quiet -CAfile noxa-ca.crt -connect dyndns.noxa.de:64201
depth=1 /C=DE/O=noxa.de Network/CN=noxa.de/emailAddress=hostmaster@noxa.de
verify return:1
depth=0 /C=DE/O=noxa.de Network/CN=*.noxa.de/emailAddress=hostmaster@noxa.de
verify return:1
220 enterprise.noxa.de noxaDNS Backend 1.0 ($Revision: 1.1 $), Protocol Version 1.0
```

Beispielabläufe noxaDynDNS

Anmelden und Ändern eines Records

Nach dem Verbindungsaufbau zu dyndns.noxa.de auf Port 64200 liefert der Server eine Meldung der Form:

```
> 220 enterprise.noxa.de noxaDNS Backend <version> (<revision>), Protocol Version <version>
```

Nun erwartet der Server eine Antwort in der Form HELO <fqhn>:

```
< HELO centaur.noxa.de
> 250 enterprise.noxa.de, Welcome [centaur.noxa.de]. {97110}
```

Jetzt kann die Authentifizierung durch eine von drei verschiedenen Methoden stattfinden: Im Klartext, als MD5 oder als SHA1-Hash. Der Salt für die letzteren wurde bei der Begrüßung in den geschweiften Klammern übergeben.

Wir verwenden die Klartext-Variante:

(User: andrea, Passwort: geheim)

```
< USER andrea geheim
> 250 success, user [andrea] logged in; level (0), maxhosts (10)
```

Mit LIST kann in Erfahrung gebracht werden, welche Hosts bereits auf den Account delegiert sind:

```
< LIST
> 300-entries for [andrea] matching () type (%)
> 300- [andrea.h3c.de] {h3c.de}
> 300- * [MX] [b.mx.noxa.de] (180) (1)
> 300- * [A] [127.0.0.1] (180) (1)
> 300- [rhcp-fansite.noxadyndns.de] {noxadyndns.de}
> 300- * [MX] [a.mx.noxa.de] (14400) (1)
> 300- * [MX] [c.mx.noxa.de] (14400) (2)
> 300- * [A] [212.60.197.61] (7200) (1)
> 300- * [A] [195.122.181.61] (7200) (2)
> 300 .
```

Wir haben also die Hosts `andrea.h3c.de` und `rhcp-fansite.noxadydns.de`. Letzterer hat zwei MX-Records und zwei IP-Adressen eingetragen, mit höheren TTL-Werten, da es sich dabei um statische IP-Adressen handelt.

`andrea.h3c.de` ist der Privatrechner zu Hause und soll nun geändert werden.

Update des A-Records:

```
< UPDATE andrea h3c.de A 80.146.61.21
> 260 host [andrea] domain [h3c.de] type [A] value [80.146.61.21] updated
```

Sofern die IP-Adresse der Verbindung mit dem gewünschten Wert übereinstimmt, kann diese auch weggelassen werden. Der Server setzt diese dann automatisch ein:

```
< UPDATE andrea h3c.de A
> 260 host [andrea] domain [h3c.de] type [A] value [80.146.61.21] updated
```

Die Ausgabe von LIST ist nun:

```
< LIST andrea.h3c.de
> 300-entries for [andrea] matching (andrea.h3c.de) type (%)
> 300-[andrea.h3c.de]
> 300- * [MX] [b.mx.noxa.de] (180) (1)
> 300- * [A] [80.146.61.21] (180) (1)
> 300- .
```

Wir haben als Parameter bei List den Host angegeben, den wir sehen wollen. So verlieren wir bei vielen Hosts und Records nicht die Übersicht.

Mit QUIT beenden wir die Verbindung:

```
< QUIT
> 221 enterprise.noxa.de disconnecting
Connection closed by foreign host.
```

Host oder Record anlegen

Bei neuen Accounts ist noch kein Hostname vorhanden. Hostnames und Records dazu werden mit dem Befehl CREATE erstellt:

```
< CREATE cyber noxadydns.de
> 270 host [cyber] domain [noxadydns.de] type [A] value [127.0.0.1] created

< LIST %noxadydns.de
> 300-entries for [andrea] matching (%noxadydns.de) type (%)
> 300-[cyber.noxadydns.de] {noxadydns.de}
> 300- * [A] [127.0.0.1] (180) (1)
> 300-[rhcp-fansite.noxadydns.de] {noxadydns.de}
> 300- * [MX] [a.mx.noxa.de] (14400) (1)
> 300- * [MX] [c.mx.noxa.de] (14400) (2)
> 300- * [A] [212.60.197.61] (7200) (1)
> 300- * [A] [195.122.181.61] (7200) (2)
> 300- .
```

Uns wurde der Hostname `cyber.noxadydns.de` delegiert. Der Server hat außerdem automatisch einen A-Record für uns angelegt. Wir haben diesmal das Wildcard-Zeichen `%` bei LIST verwendet, um alle Hosts unter `noxadydns.de` zu sehen.

Diesen ändern wir und fügen mit einem weiteren CREATE noch zwei MX-Records hinzu. Beim CREATE können wir auch gleich die gewünschten Werte übergeben, dies spart uns ein UPDATE. Wir übergeben als TTL 7200 Sekunden, das sind zwei Stunden:

```
< UPDATE cyber noxadydns.de A 131.6.137.10
> 260 host [cyber] domain [noxadydns.de] type [A] value [131.6.137.10] updated

< CREATE cyber h3c.de MX1 andrea.h3c.de 7200
> 270 host [cyber] domain [noxadydns.de] type [MX1] value [andrea.h3c.de] created
< CREATE cyber h3c.de MX3 b.mx.noxa.de 7200
> 270 host [cyber] domain [noxadydns.de] type [MX3] value [b.mx.noxa.de] created

< LIST cyber.noxadydns.de
> 300-entries for [andrea] matching (%noxadydns.de) value (%)
> 300-[cyber.noxadydns.de] {noxadydns.de}
> 300- * [A] [131.6.137.10] (180) (1)
> 300- * [MX] [andrea.h3c.de] (7200) (1)
> 300- * [MX] [b.mx.noxa.de] (7200) (3)
> 300- .
```

Host oder Record löschen

Records und ganze Hosts lassen sich mit dem Befehl DELETE löschen.

Dabei ist zu beachten, dass es möglich ist, einen Hostname delegiert zu haben, der keine Records besitzt.

Mit anderen Worten: Das Löschen aller Records eines Hosts löscht den Host nicht. Er wird auch weiterhin bei LIST angezeigt.

Wir löschen den A-Record des Hosts cyber.noxadyndns.de:

```
< DELETE cyber noxadyndns.de A1
> 280 host [cyber] domain [noxadyndns.de] type [A1] deleted

< LIST cyber.noxadyndns.de
> 300-entries for [andrea] matching (cyber.noxadyndns.de) type (%)
> 300-[cyber.noxadyndns.de] {noxadyndns.de}
> 300- * [MX] [andrea.h3c.de] (7200) (1)
> 300- * [MX] [b.mx.noxa.de] (7200) (3)
> 300 .
```

Da die MX-Records noch vorhanden sind, können weiterhin Mails empfangen werden.

Wenn wir keinen Gefallen an cyber.noxadyndns.de mehr haben, kann er mit DELETE auch komplett gelöscht werden. Dazu lassen wir die Angabe des Typs weg:

```
< DELETE cyber noxadyndns.de
> 280 host [cyber] domain [noxadyndns.de] type [] deleted
```

Authentifikation via MD5/SHA1

Wir bieten neben der Klartext-Authentifikation auch MD5- und SHA1-Authentifikation an. Dazu liefert der Server beim HELO in geschweiften Klammern ({}), einen Salt, der ans Passwort angehängt und zusammen damit gehashed wird.

Der Hashwert wird, anstelle des Klartextpassworts, mit dem Befehl USER übergeben.

Beispiel:

```
Connected to ds9.noxa.de.
Escape character is '^'.
> 220 ds9.noxa.de noxaDNS Backend 1.0 ($Revision 1.42 $), Protocol Version 1.0
< HELO bender.noxadyndns.de
> 250 ds9.noxa.de, welcome [bender.noxadyndns.de]. {443493}
```

Wir haben für diese Session den Salt 443493. Nun berechnen wir den Hashwert für unser Passwort geheim:

```
<0>[21:03]niklas@mayhem:~$ echo -n 'geheim443493' | openssl sha1
935b403e4cde930c5d85a64b5650c42bb9bb3325

# oder, falls wir openssl nicht haben, jedoch das Programm md5:
<0>[19:04]niklas@martyr:~$ echo -n 'geheim443493' | md5
ae0f418ae57dd0d7cf5bc12ede7e4773

# oder, falls wir lieber Python benutzen:
<0>[21:05]niklas@mayhem:~$ python2.3
Python 2.3.5 (#1, Jul 11 2006, 09:47:28)
[GCC 4.1.2 20060628 prerelease (NetBSD nb1 20060628)] on netbsd3
Type "help", "copyright", "credits" or "license" for more information.
>>> import sha
>>> digest = sha.new()
>>> digest.update('geheim')
>>> digest.update('443493')
>>> digest.hexdigest()
'935b403e4cde930c5d85a64b5650c42bb9bb3325'
```

Nun können wir uns authentifizieren.

Wir benutzen den SHA1-Hashwert:

```
USER niklas 935b403e4cde930c5d85a64b5650c42bb9bb3325
250 success, user [niklas] logged in; level (255), maxhosts (2000)
```

Beispielabläufe noxaDNS

Die Grundlagen aus dem vorigen Abschnitt werden vorausgesetzt.

Secondary/Slave-NS für eine Zone anlegen

Der aktuelle noxaDNS-Server soll Slave für example.com werden, der authoritative NS für die Zone sei 23.42.23.42:

```
< ZONE_ADD example.com 23.42.23.42
```

```
> 290 [h.dns.noxa.de] is now slave for zone [example.com] with master [23.42.23.42]

< ZONE_LIST
(falls genügende Rechte für dieses Kommando!)
> 350-zones matching () type (%)
> 350-[example.com] master [23.42.23.42] type [SLAVE] ntfy'd serial [None]
> 350 .

< ZONE_ALIST
> 351-attachments matching ()
> 351-[example.com] user [jessica]
> 351 .
```

Die noxaDNS-Server ds9.noxa.de und defiant.noxa.de sollen Slave für die Reverse-Delegation des IPv6-Netzes 2001:6f8:96c::/48 werden, der authoritative NS für die Zone sei 212.60.197.71. Wir benutzen dazu zusätzlich das FORWARD Kommando:

```
< FORWARD ds9.noxa.de,defiant.noxa.de ZONE_ADD c.6.9.0.8.f.6.0.1.0.0.2.ip6.arpa 212.60.197.71
> 390-executing command [ZONE_ADD c.6.9.0.8.f.6.0.1.0.0.2.ip6.arpa 212.60.197.71] on [2] server(s):
> 390-391-response from [ds9.noxa.de]:
> 390-391-[290 [h.dns.noxa.de] is now slave for zone [c.6.9.0.8.f.6.0.1.0.0.2.ip6.arpa] with master [212.60.197.71]]
> 390-391 .
> 390-391-response from [defiant.noxa.de]:
> 390-391-[290 [b.dns.noxa.de] is now slave for zone [c.6.9.0.8.f.6.0.1.0.0.2.ip6.arpa] with master [212.60.197.71]]
> 390-391 .
> 390 .

< FORWARD ds9.noxa.de ZONE_LIST c.6.9.%.ip6.arpa
(falls genügende Rechte für dieses Kommando!)
> 390-391-response from [ds9.noxa.de]:
> 390-391-[350-zones matching (c.6.9.%.ip6.arpa) type (%)]
> 390-391-[350-[c.6.9.0.8.f.6.0.1.0.0.2.ip6.arpa] master [212.60.197.71] type [SLAVE] ntfy'd serial [None]]
> 390-391-[350 .]
> 390-391 .
> 390 .

< FORWARD ds9.noxa.de ZONE_ALIST
> 390-391-response from [ds9.noxa.de]:
> 390-391-[351-attachments matching ()]
> 390-391-[351-[example.com] user [jessica]]
> 390-391-[351-[c.6.9.0.8.f.6.0.1.0.0.2.ip6.arpa] user [jessica]]
> 390-391-[351 .]
> 390-391 .
> 351 .
```

Neue Zone mit primärem und sekundärem NS anlegen

Die Zone example.com soll auf dem aktuellen noxaDNS-Server angelegt werden (der dann Primärer NS wird) und defiant.noxa.de soll Slave werden. Es werden zusätzlich ein paar Records angelegt:

```
< ZONE_CREATE example.com hostmaster.example.com
> 292 zone [example.com] created
< LIST %example.com
> 300-entries for [ines] matching (%example.com) type (%)
> 300-[example.com] {example.com}
> 300- * [NS] [a.dns.noxa.de] (14400) (1)
> 300- * [SOA] [a.dns.noxa.de hostmaster.example.com 1153683996 14400 1800 2419200 1800] (14400) (0)
> 300 .

< FORWARD defiant.noxa.de ZONE_ADD example.com 212.60.197.41
(212.60.197.41 ist die IP-Adresse von a.dns.noxa.de, siehe SOA)
> 390-executing command [ZONE_ADD example.com 212.60.197.41 of a.dns.noxa.de] on [1] server(s):
> 390-391-response from [defiant.noxa.de]:
> 390-391-[290 [b.dns.noxa.de] is now slave for zone [example.com] with master [212.60.197.41]]
> 390-391 .
> 390 .

< SET DEFAULT_TTL 14400
> 320 value for key [DEFAULT_TTL] is now [14400]

< CREATE . example.com NS2 b.dns.noxa.de
> 270 host [.] domain [example.com] type [NS2] value [b.dns.noxa.de] created
< CREATE . example.com A 127.0.0.1
> 270 host [.] domain [example.com] type [A] value [127.0.0.1] created
< CREATE www example.com CNAME example.com
> 270 host [www] domain [example.com] type [CNAME] value [example.com] created
< CREATE . example.com MX1 a.mx.noxa.de 7200
> 270 host [.] domain [example.com] type [MX1] value [a.mx.noxa.de] created
< CREATE . example.com MX3 c.mx.noxa.de 7200
```

```

> 270 host [.] domain [example.com] type [MX3] value [c.mx.noxa.de] created
< CREATE . example.com TXT1 v=spf1#a#mx#-all
> 270 host [.] domain [example.com] type [TXT1] value [v=spf1 a mx -all] created

< LIST %example.com
> 300-entries for [ines] matching (%example.com) type (%)
> 300-[example.com] {example.com}
> 300- * [A] [127.0.0.1] (14400) (1)
> 300- * [MX] [a.mx.noxa.de] (7200) (1)
> 300- * [MX] [c.mx.noxa.de] (7200) (3)
> 300- * [NS] [a.dns.noxa.de] (14400) (1)
> 300- * [NS] [b.dns.noxa.de] (14400) (2)
> 300- * [SOA] [a.dns.noxa.de hostmaster.example.com 1153684315 14400 1800 2419200 1800] (14400) (0)
> 300- * [TXT] [v=spf1 a mx -all] (14400) (1)
> 300-[www.example.com] {example.com}
> 300- * [CNAME] [example.com] (14400) (0)
> 300 .

```

Serial-Schema einer Zone von Unixtimestamp auf YYYYMMDDnn umstellen

Standardmäßig führt noxaDNS neue Zones mit Unixtimestamp-Serials, d.h., Sekunden seit der Epoche (1.1.1970).

Das noxaDNS-Backend erkennt, welches Schema eine Zone besitzt und benutzt dieses entsprechend. Bei YYYYMMDDnn wird also automatisch der Slot hochgezählt und das Datum angepasst.

DynDNS-Domains werden sinnvollerweise auf Unixtimestamp geführt, da es leicht zu über 100 Updates am Tag kommen kann. Für andere Domains kann das Datumsschema sinnvoller sein.

Um die Serial per Hand zu ändern, gibt es das Kommando BUMP. Bitte beachten Sie, dass Versuche, die Serial über Änderung der SOA fehlschlagen werden, da dort der Serial-Wert automatisch ersetzt wird.

BUMP ohne Parameter aufgerufen erhöht die Serial nur, mit Parameter wird diese entsprechend gesetzt:

```

< BUMP example.com 2006072300
> 295 serial bumped on zone [example.com]
< LIST example.com ASC 0 SOA
> 300-entries for [svenja] matching (example.com) type (SOA)
> 300-[example.com] {example.com}
> 300- * [SOA] [a.dns.noxa.de hostmaster.example.com 2006072300 14400 1800 2419200 1800] (14400) (0) 300 .

```

Da die neue Serial größer war als die alte ist die Konvertierung hiermit abgeschlossen, die Slaves werden notifiziert und holen sich die neue Zone. Der umgekehrte Fall, eine Konvertierung hin zu Unixtimestamps, verläuft analog, jedoch muss hier zunächst die Serial auf allen Slaves niedrig, am besten auf 1, gesetzt werden.

Dazu kann FORWARD slave1,slave2 BUMP example.com 1 erhalten.

Um exzessive Serial-Updates zu verhindern, kann das automatische Update sitzungsweise über DELAY_BUMP true abgeschaltet werden. Ein manuelles BUMP domain nach getaner Arbeit führt dann zur einmaligen Erhöhung.

(nikas/noxa.de)

Impressum

Copyright © 2004, 2005, 2006, 2007 noxa.de
noxa.de Network ist ein **nicht-kommerzielles** Team von Netzbegeisterten.



Verbunden über IPv4 (87.178.38.70)