!!! ACHTUNG - evtl. veraltet - ACHTUNG !!!

Diese Seite wurde zuletzt am 9. Juli 2014 um 10:57 Uhr geändert.

Der virtuelle Server, der DNS anbieten soll, soll einen DNS-Server auf einem eigenen loopback-Device und einen DNS-Cache auf der eigentlichen IP-Adresse beherbergen.

Die anderen virtuellen Server richten Ihre Anfragen dann an den DNS-Cache, welcher sowohl das interne Netz als auch das Internet cached.

in der Konfiguration (/etc/vservers/...) des Hosts folgende Dateien mit darunter stehendem Inhalt anlegen:

/apps/init/runlevel:

```
 3
```

/apps/init/style:

```
 plain
```

djbdns installieren:

```
 aptitude install djbdns
 adduser --system --ingroup nogroup --home /var/log/dns --no-create-home
dnslog
 adduser --system --ingroup nogroup --home /etc/dnscache --no-create-home
dnscache
 adduser --system --ingroup nogroup --home /etc/tinydns --no-create-home
tinydns
```

tinydns konfigurieren:

```
 tinydns-conf tinydns dnslog /etc/tinydns 127.0.0.1
 cd /service/tinydns/root
 ./add-ns vserver.lan 127.0.0.1
 ./add-ns 10.in-addr.arpa 127.0.0.1
 ./add-mx vserver.lan 10.0.0.25
 ./add-host server.vserver.lan 10.0.0.10
 ./add-host php.vserver.lan 10.0.0.80
 ./add-alias www.vserver.lan 10.0.0.80
 [[...]]
 make
 ln -s /etc/tinydns /etc/service
```

und gleich weiter mit dem dnscache:

```
 dnscache-conf dnscache dnslog /etc/dnscache 10.0.0.53
 echo 127.0.0.1 > /etc/dnscache/root/servers/vserver.lan
 echo 127.0.0.1 > /etc/dnscache/root/servers/10.in-addr.arpa
 echo 1 > /etc/dnscache/env/FORWARDONLY
```

```
touch /etc/dnscache/root/ip/10
ln -s /etc/dnscache /etc/service
```

mit folgendem Befehl kann man gucken, wie lange die Dienste laufen (wenn sie immer nur ein paar Sekunden laufen, stimmt etwas nicht):

```
svstat /service/tinydns
```

oder

```
svstat /service/dnscache
```

wenn dann noch eine Auflösung erfolgreich ist:

```
dnsip www.vserver.lan
10.0.0.80
dnsip www.heise.de
193.99.144.85
dnsip www
www.vserver.lan 10.0.0.80
dnsqr a php.vserver.lan
1 php.vserver.lan:
50 bytes, 1+1+0+0 records, response, noerror
query: 1 php.vserver.lan
answer: php.vserver.lan 86223 A 10.0.0.80
```

kann er in die Datei „/etc/**resolv.conf**“ der anderen virtuellen Server eingetragen werden:

```
search vserver.lan
nameserver 10.0.0.53
```

Möchte man zwei „primäre“ DNS-Server auf Basis von djbdns anbieten, könnte man sie wie folgt untereinander abgleichen...

Zusätzliche Pakete installieren:

```
aptitude install incron openssh-server rsync
echo "root" > /etc/incron.allow
```

... passwortfreies Login per Schlüssel als root nur vom anderen DNS-Server einrichten (ssh-keygen, ssh-coppy-id, /etc/hosts.deny, /etc/hosts.allow,...)...

/usr/local/sbin/tinydns-rsync-data:

```
#!/bin/bash
# dependencies:
# - incron (/etc/tinydns/root IN_MOVED_TO,IN_CLOSE_WRITE
/usr/local/sbin/tinydns-rsync-data $#)
# - ip
# - rsync
```

```
 # - ssh (as root, without password, only from the other nameserver!)
 # - make
 # simple configuration
 DIR="/etc/tinydns/root/"
 FILE="data"
 LOG="/var/log/dns/edit_rsync_make_data.log"
 # functions
 mylog() {
   [[|-n "$1" ]] && echo -e "$( date +"%Y %m %d %H %M %S" ) - $1" >> "$LOG"
 }
 # simple checks
 ## expect exactly one parameter!
 [[|$# -eq 1 ]] || {
   mylog "wrong call of $0... TERMINATING!\n- - - - - - - - - - - - - - - -
- - - - - - - - -"
   exit 1
 }
 ## do nothing, if the the affected file ist not "data"!
 [[|"$1" = "data" ]] || exit 0
 mylog "/etc/tinydns/root/data was changed! starting roll out changes..."
 # identify ip of the other nameserver
 case "$( ip addr )" in
   *10.0.0.22* ) REMOTE_IP="10.0.0.23" ;;
   *10.0.0.23* ) REMOTE_IP="10.0.0.22" ;;
 esac
 mylog "the ip of my twin was set to $REMOTE_IP..."
 # rsyncing (the "reverse"-rsync has nothing to do and will not trigger this
script!)
 rsync -aze ssh "$DIR$FILE" "$REMOTE_IP:$DIR$FILE"
 mylog "rsync with my twin done..."
 # aktivate changes on both nameservers
 cd "$DIR"
 make
 mylog "data.cdb was updated..."
 mylog "done!\n- - - - - - - - - - - - - - - - - - - - - - - - - - - -"
 exit 0
```

Neue inotify-Regel per „incrontab -e“ hinzufügen

```
 /etc/tinydns/root IN_MOVED_TO,IN_CLOSE_WRITE /usr/local/sbin/tinydns-rsync-
data $#
```

(/etc/tinydns/root/data direkt überwachen läuft ins Leere, weil Tools wie rsync, Editoren, /usr/bin/tinydns-edit erst in eine andere Datei schreiben, die alte löschen und die neue in die alte umbenennen, was zu dann zu einem einem „inotify-IGNORE“ führen würde!)

Einen der beiden DNS-Server wie gewohnt pflegen und glücklich sein!