

Konfiguration des DNS und BIND v4*

Tobias Brinkert
eMail: <t.brinkert@semibyte.de>
Homepage: <www.semibyte.de>

24.05.2005
Version: 0.52

Zusammenfassung

Unter Unix und Linux kommt für den Domain Name Services das Softwarepaket *Berkeley Internet Name Domain (BIND)* zum Einsatz. BIND besteht aus zwei Teilen, die Client-Seite wird *Resolver* genannt und generiert die DNS-Abfragen (Queries) und die Server-Seite übernimmt ein Daemon namens **named** und beantwortet die Anfragen des Resolvers.

Der Begriff *Zone* wird häufig mit dem Begriff *Domain* gleichgesetzt. Der Begriff »Zone« wird jedoch dafür verwendet, wenn man von der Domain-Datenbankdatei spricht während der Begriff »Domain« im allgemeineren Zusammenhang verwendet wird. Eine Sammlung von Domain-Informationen, die in einer Domain-Datenbankdatei enthalten sind, wird als »Zone« bezeichnet.

Inhaltsverzeichnis

1.	Konfiguration des Resolvers	2
1.1	Resolver-Konfigurationsdatei (/etc/resolv.conf)	2
1.1.1	Einträge	2
1.2	Beispiele	3
1.2.1	Name-Server auf lokalen Host	3
1.2.2	reiner Resolver	3
2.	Konfiguration von named	4
2.1	named.boot	4
2.1.1	Konfigurationsbefehle in named.boot	4
2.1.2	Beispiele	7
2.2	Records der Zonendateien	8
2.2.1	Steuerungseinträge	8
2.2.2	Standard Resource Records	8
2.2.3	Beispiele	13
	Literatur	16

*Dieses Referat ist eine Zusammenfassung aus dem Kapitel 8 und Anhang C des Buches [1]

1. Konfiguration des Resolvers

Die Konfiguration des Resolvers wird in der Datei `/etc/resolv.conf` vorgenommen. Der Resolver ist kein eigener Prozeß, sondern die Datei `/etc/resolv.conf` wird eingelesen, wenn ein den Resolver nutzender Prozeß gestartet wird. Falls keine `/etc/resolv.conf`-Datei vorgefunden wird, wird der auf dem lokalen Host laufende **named**-Server befragt. Besser ist jedoch, dem Resolver dies explizit anzugeben, da es sonst zu unvorhersagbaren Erscheinungen kommen kann.

1.1 Resolver-Konfigurationsdatei (`/etc/resolv.conf`)

Die Konfigurationsdatei erlaubt Einträge für bis zu 3 Namenservern, wobei zwei von ihnen als Backup dienen, falls der erste Server nicht antwortet. Weiter werden Standard-Domain und verschiedene andere Verarbeitungsoptionen definiert. Diese Datei liegt im Klartext vor.

1.1.1 Einträge

nameserver *Adresse*

- IP-Adressen von Nameservern
- werden nach der Reihenfolge ihres Auftretens abgefragt
- sind keine **nameserver**-Einträge enthalten oder die Datei `/etc/resolv.conf` existiert nicht, wird der lokale Host befragt
- soll der lokale Host befragt werden (weil er selber Nameserver ist), muß er über die offizielle IP-Adresse angesprochen werden

domain *Name*

- definiert den Standard-Domain-Namen
- Hostnamen, die keinen Punkt enthalten, wird der Standard-Domain-Name angehängt
- wenn die Umgebungsvariable LOCALDOMAIN gesetzt ist, überschreibt sie den **domain**-Eintrag
- es werden auch übergeordnete Parent-Domains befragt (z.B. wenn **domain** `hgII.rsa` wird erst nach **Host.hgII.rsa** und wenn die Anfrage zu keinem Ergebnis führt, nach **Host.rsa** angefragt)
- eine Parent-Domain muß aus mindestens zwei Feldern bestehen

search *Domain ...*

- definiert eine Reihe von Domain, die durchsucht werden, wenn ein Hostname keinen Punkt enthält
- es werden nur die explizit aufgeführten Domains durchsucht

- wenn die Umgebungsvariable LOCALDOMAIN gesetzt ist, überschreibt sie den **search**-Eintrag
- nie **search** und **domain** in der selben Konfigurationsdatei verwenden

sortlist *Netzwerk ...*

- angegebene Netzwerkadressen werden gegenüber anderen Adressen bevorzugt
- normalerweise werden die Adressen in der Reihenfolge an die Anwendung übergeben, wie sie vom Resolver empfangen wurden
- Ausnahme: Adressen aus dem eigenen Netzwerk werden vorgezogen

options *Option ...*

- optionale Einstellungen für den Resolver
- derzeit zwei gültige Schlüsselwörter

debug

- Aktivierung des Debugging

ndots:n

- Angabe der Zahl von Punkten, die ein Hostname enthalten darf, um die Standard-Domain nicht anzuwenden (Standard ist 1)

1.2 Beispiele

1.2.1 Name-Server auf lokalen Host

Der lokale Host ist *ServerII.hgII.rsa* mit IP-Adresse 192.168.3.1 und auf ihm läuft ein Nameserver. Zuerst wird der Resolver auf dem lokalen Host nachfragen und wenn die Queries fehlschlagen bei *ServerI.hgI.rsa* und anschließend bei *Proxy.hgII.rsa* nachfragen. Da sich dieser Server in der Domain **hgII.rsa** befindet, wird bei Hostnamen ohne Domain-Angabe zuerst für die Domain **hgII.rsa**, danach **hgI.rsa** und zum Schluß **rsa** Einträge gesucht.

```
#
# Domain Name Resolver Konfigurationsdatei
# für ServerII.hgII.rsa
#
search hgII.rsa hgI.rsa rsa
# zuerst auf lokalen Host ServerII
nameserver 192.168.3.1
# dachnach bei ServerI
nameserver 192.168.2.1
# und zuletzt beim Proxy-Server Proxy.hgII.rsa
nameserver 192.168.3.2
```

1.2.2 reiner Resolver

Der Backup-Rechner *Backup.hgI.rsa* befragt, da er selber kein Nameserver ist, zuerst *ServerI.hgI.rsa* danach *ServerII.hgII.rsa* und zuletzt bei *Proxy.hgII.rsa*. Queries werden nie lokal aufgelöst.

```
#
# Domain Name Resolver Konfigurationsdatei
# für Backup.hgI.rsa
#
search hgI.rsa hgII.rsa rsa
# zuerst auf ServerI
nameserver 192.168.2.1
# dachnach bei ServerII
nameserver 192.168.3.1
# und zuletzt beim Proxy-Server Proxy.hgII.rsa
nameserver 192.168.3.2
```

2. Konfiguration von named

Bei der Konfiguration von **named** werden, im Gegensatz zur Resolver-Konfiguration, mehrere Konfigurationsdateien benötigt. Dies umfaßt

named.boot allgemeine **named**-Parameter und zeigt auf die Quellen mit Domain-Informationen, die von diesem Server genutzt werden

named.ca verweist auf die Root-Domain-Server

named.local lokale Auflösung der Loopback-Adresse genutzt

named.hosts Zonendatei, in der Hostnamen auf IP-Adressen abgebildet werden

named.rev Zonendatei der Reverse Domain, die IP-Adressen auf Hostnamen abbildet

Für die Datei *named.ca* gibt es allgemein bekannte Alternativnamen wie *named.root* und *root.ca*. Die Namen *named.hosts* und *named.rev* sollten jedoch nicht genommen werden sondern beschreibene Namen.

Normalerweise gibt es keine reinen Caching-Only, Primäre- oder Sekundäre-Nameserver, sondern Mischformen. So hat zum Beispiel der Caching-Only-Server normalerweise noch die Aufgabe als Primärer-DNS für die Loopback-Domain, ebenso der sekundäre Server. Ebenso besitzen Primäre und sekundäre Server häufig die Cache-Option.

2.1 named.boot

Diese Datei zeigt **named** die Quellen für die benötigten DNS-Informationen. Hierbei kann es sich um lokal vorliegende Dateien sowie um Dateien auf entfernten Servern handeln. Angelegt werden müssen nur die Dateien, die in den *primary*- und *cache*-Anweisung referenziert werden.

2.1.1 Konfigurationsbefehle in named.boot

directory *verzeichnis-pfad*

- Definiert ein Verzeichnis für alle nachfolgenden Dateireferenzen;

- alle Dateinamen, die nicht mit »/« beginnen, werden relativ zum angegebenen Verzeichnis gesucht;

primary *domain-name dateiname*

- deklariert Server als Primary für die angegebene Domain bzw. Reverse Domain und gibt an, in welcher Datei die Zonen-Daten zu finden sind;
- Format: Schlüsselwort `primary`, dem Domain-Namen und dem Namen der Zonen-Datei;

secondary *domain-name server-adressliste dateiname*

- deklariert Server als Secondary für die angegebene Domain bzw. Reverse Domain und gibt an, von welchen Master-Servern die Daten abgerufen werden sollen und in welche Datei sie abgelegt werden sollen;
- Format: Schlüsselwort `secondary`, dem Domain-Namen, Adresse eines oder mehrerer Master-Server für diese Domain und dem Namen einer lokalen Datei, in der die empfangenen Informationen abgelegt werden;

cache . *dateiname*

- Zeigt auf die Cache-Datei;
- weist `named` an, einen Cache mit Antworten der Nameserver aufzubauen und zu pflegen;
- Cache mit Liste von Root-Servern, die in der Datei `named.ca` definiert sind, zu füllen;

forwarders *server-adresse server-adresse . . .*

- leitet alle Queries, die nicht selber beantwortet werden können, an die angegebenen IP-Adressen weiter;
- baut einen großen DNS-Cache auf ausgewählten Servern im lokalen Netzwerk auf;
- reduziert die Zahl der Queries, die ins WAN geschickt werden müssen;

slave

- zwingt den lokalen Server, nur die in der `forwarders`-Befehlszeile angegebenen Server zu verwenden;
- **forwarders**-Zeile muß in `named.boot` enthalten sein;

sortlist *netzwerk netzwerk . . .*

- sorgt dafür, daß **named** Adressen der aufgeführten Netzwerke gegenüber Adressen anderer Netzwerke vorzieht;

xfrnets *adresse[&maske] . . .*

- beschränkt Zonentransfers auf Hosts mit der angegebenen *adresse*;
- wird in Punktnotation angegeben und als Netzwerkadresse interpretiert;
- ist ein Bit in der Maske gesetzt, bestimmt es, welche Hosts einen Zonen-datei-Transfer empfangen dürfen;

include *datei*

- fügt den Inhalt der *datei* an der Stelle ein, an der der Befehl in der boot-Datei steht;

bogusns *adresse adresse ...*

- verhindert, daß Queries an den durch *adresse* bezeichneten Nameserver gesendet werden;
- muß eine IP-Adresse sein

limit *name wert*

- ändert interne Quotas von BIND;
- *wert* ist eine Zahl, die die neuen Quotas festlegt
- **k** (Kilobyte), **m** (megabyte) oder **g** (Gigabyte) können ganz auch Bedarf an den neuen Quota-Wert angehängt werden;
- Werte für *name*:
datasize setzt die Quota für die Datengröße des Prozesses
transfer-in legt die Anzahl von Transfer-Unterprozessen fest, die bind gleichzeitig ausführen kann;
transfers-per-ns legt die max. Anzahl von simultanen Zonentransfers fest
- es können mehrere limit-Befehle in einer Boot-Datei stehen, jeweils einer für jede gesetzte Quota

options

- Aktiviert BIND-Optionen;
Optionen:

forward-only:

- zwingt den lokalen Server, immer den Forwarder zu benutzen;
- lokaler Server versucht nicht, die Query selbst aufzulösen, selbst dann nicht, wenn er von den Forwardern keine Antwort erhält;

query-log

- protokolliert alle Queries via syslogd;

fake-iquery

- NS täuscht bei inversen Queries eine Antwort vor, statt einen Fehler zurückzugeben;

no-recursion

- beantwortet Queries nur für Zonen, über die er die Autorität besitzt;
- alle anderen Queries werden mit einem Verweis auf einen anderen Server beantwortet;

no-fetch-glue

- holt für eine Antwort keine fehlenden Binde-Records ein;

check-names *quelle* *aktion*

- weist den Nameserver na, die Hostnamen mit dem in RFC 952 definierten Standards für Hostnamen zu vergleichen;
- Sicherstellung, daß die Namen keine nicht durckbaren Zeichen enthalten;
- Quelle kann die Werte **primary** für die primäre Zonendatei, **secondary** für die sekundäre Zonendatei oder **response** für die bei einer rekursiven Suche empfangene Antwort annehmen;
- Werte für *aktion*:
 - **fail**: Daten ablehnen; sie nicht laden, im Cache speichern oder weiterleiten;
 - **warn**: Fehlermeldung an Systemlog senden;
 - **ignore**: Daten verarbeiten, as wäre kein Fehler aufgetreten;
- möglich sind mehrere **check-names**-Befehle, eine für jede @atenquelle und die Aktion kan für jede *quelle* unterschiedlich sein;

max-fetch *wert*

- gleiche Funktion wie **limit transfers-in**;

2.1.2 Beispiele

Konfiguration eines reinen Cache-Nameservers

```
;
; eine Caching-Only Server-Konfiguration
;
directory /var/named
primary 0.0.127.IN-ADDR.ARPA named.local
cache . named.ca
```

Konfiguration eines primären Servers

```
;
; primäre Server-Konfiguration
;
directory /var/named
primary 0.0.127.IN-ADDR.ARPA named.local
primary hgII.rsa named.hosts.hgII.rsa
primary 3.168.192.IN-ADDR.ARPA named.rev.192.168.3
cache . named.ca
```

Konfiguration eines sekundären Servers

```
;
; sekundäre Server-Konfiguration
;
directory /var/named
secondary hgII.rsa 192.168.3.1 named.hosts.hgII.rsa
secondary 3.168.192.IN-ADDR.ARPA 192.168.3.1 named.rev.192.168.3
primary 0.0.172.IN-ADDR.ARPA named.local
cache . named.ca
```

2.2 Records der Zonendateien

2.2.1 Steuerungseinträge

\$INCLUDE *dateiname*

- gibt eine Datei an, deren Inhalt in die Zonendatei eingefügt werden soll;
- *dateiname* liegt relativ zu dem in der *directory*-Anweisung der *named.boot* angegebenen Verzeichnis;

\$ORIGIN *domainname*

- ändert den Standard-Domainnamen, der in der Zonendatei von nachfolgenden Records verwendet wird;

2.2.2 Standard Resource Records

Die oben angeführten Konfigurationsbefehle werden in der Datei *named.boot* verwendet. Alle anderen zur Konfiguration von **named** verwendeten Dateien speichern Domain-Informationen. Diese Dateien verwenden alle das gleiche grundlegende Format und die gleichen Arten von Datenbank-Records, die Standard Resource Records, kurz RRs.

Liste der DNS Resource Records

Resource Record (ausgeschriebener Name)	Record- Typ	Funktion
Start of Authority	SOA	Markiert den Anfang der Daten einer Zone und definiert die gesamte Zone betreffende Parameter
Nameserver	NS	Bestimmt den Nameserver einer Domain
Address	A	Wandelt einen Hostnamen in einer Adresse um.
Pointer	PRT	Wandelt eine Adresse in einen Hostnamen um.
Mail Exchange	MX	Bestimmt, wohin die Mail für eine gegebene Domain ausgeliefert werden sollen.
Canonical Name	CNAME	Definiert einen Alias für einen Hostnamen.
Host Information	HINFO	Beschreibt Hardware und Betriebssystem des Hosts.
Well-Known Service	WKS	Bietet Netzwerkdienste an.
Text	TXT	Speichert beliebige Zeichenketten ab.

Format des DNS Resource Record

`[name] [ttl] klasse typ daten`

name

- Name des Domain-Objekts;
- es kann sich um einen einzelnen Host oder um eine gesamte Domain handeln;
- endet der eingesetzte String nicht mit einem Punkt, so wird er relativ für die aktuelle Domain betrachtet;
- er wird auf den zuletzt genannten Namen angewandt, wenn das Feld leer bleibt;
- ein oder zwei Punkte im Namensfeld bezeichnen die root-Domain;
- ein einzelnes at-Zeichen (@) im Namensfeld verweist auf den momentanen Ursprung
 - ein vom System über den aktuellen Domain-Namen abgeleiteter oder ein explizit mit dem \$ORIGIN-Befehl eingestellter Domain-Name
- ein Sternchen (*) dient als Wildcard

ttl

- Time-to-Live, Lebensdauer;
- definiert die Zeitspanne in Sekunden, für die die Informationen dieses RRs im Cache erhalten bleiben;
- normalerweise bleibt dieses Feld leer und die Standard-TTL wird verwendet;
- Standard-TTL (Minimum) wird im SOA-Record für die gesamte Zone definiert;
- bis zu acht Zeichen Länge

klasse

- definiert die Adreßklasse des RR
- IN**
- identifiziert das Record als Internet DNS-RR (es gibt noch andere)

HS

- Hesiod-Server

CH

- chaosnet-Informationen

typ

- Art des RR (siehe Tabelle in [2.2.2](#))
- Angabepflicht

daten

- spezifische Daten für diesen Typ von RR

;

- Kommentarzeichen

()

- Fortsetzungszeichen
- verwendet, um Daten über eine einzelne zeile hinweg weiterzuführen

\x

- ein auf ein Backslash folgendes nicht-numerisches Zeichen wird als Literal betrachtet und jede besondere Bedeutung, die das Zeichen normalerweise haben könnte, wird ignoriert

\ddd

- die drei Dezimalzahlen werden als absoluter Bytewert interpretiert

Klasse: SOA (Start of Authority)

Das SOA-Record markiert den Anfang einer Zone und ist üblicherweise das erste Record einer Zonendatei. Jede Zone hat nur ein SOA-Record, d. h., das nächste SOA-Record markiert den Anfang einer anderen Zone.

Format:

```
[zone] [ttl] IN SOA ursprung kontakt (  
    serial  
    refresh  
    retry  
    expire  
    minimum  
)
```

zone

- Name der Zone (üblicherweise @);

ttl

- bleibt leer;

IN

- Adreßklasse für alle Internet-RRs;

SOA

- RR-Typ, alle hierauf folgenden Informationen sind Teil des Datenfeldes und spezifisch für das SOA-Record;

ursprung

- Hostname des primären Master-Servers, normalerweise als voll qualifizierter Domain-Name angegeben;

kontakt

- E-Mail-Adresse der für diese Domain verantwortlichen Person;

serial

- Versionsnummer der Zonendatei (bis zu 8 Ziffern);
- muß erhöht werden bei jeder Änderung, damit für einen Secondary-Server bekannt wird, daß sich die Zonen-Datei verändert hat;

refresh

- legt die Zeitspanne fest, die der sekundäre Server warten soll, bevor er auf dem primären Server prüft, ob die Zone aktualisiert wurde;
- bis zu 8 Ziffern und gibt die Sekunden an;

retry

- gibt an, wie lange sekundäre Server warten sollen, bevor sie erneut versuchen, den primären Server anzusprechen, wenn dieser auf eine Zonen-Refresh-Anforderung nicht geantwortet hat;
- bis zu 8 Ziffern und gibt die Sekunden an;

expire

- legt fest, wie lange die Zonendaten vom sekundären Server vorgehalten werden, ohne einen Zonen-refresh empfangen zu haben;

minimum

- Standard-TTL für alle Resource-Records, bei denen kein expliziter TTL-Wert angegeben wurde;

Klasse: NS (Nameserver-Record)

Nameserver-Records geben die autoritativen Server einer Zone an.

Format:

```
[domain] [ttl] IN NS server
```

domain

- Name der Domain, für die der im Server-Feld angegebene Host den autoritativen Nameserver darstellt.

server

- Hostname eines Computers, der autoritativen Nameservice für diese Domain anbietet
- muß voll qualifizierter Hostname sein

Klasse: A (Adress-Record)

A-Records werden benutzt, um Hostnamen in IP-Adressen umzuwandeln.

Format:

```
[host] [ttl] IN A adresse
```

host

- Name des Hosts, dessen Adresse im Datenfeld dieses Records steht
- häufig relativ zur aktuellen Domain

adresse

- IP-Adresse des Hosts in Punktnotation, also z.B. 192.168.3.10

Klasse: MX (Mail Exchanger-Record)

Das MX-Record leitet Mails an einen Mailserver weiter. Hierbei kann die Mail für einen einzelnen Rechner oder für eine vollständige Domain weitergegeben werden

Format:

```
[name] [ttl] IN MX präferenz host
```

name

- Name eines Hosts oder einer Domain, an die die Mail adressiert ist;

präferenz

- legt die Reihenfolge fest, die bei Angabe von mehreren Mailserver benutzt wird
- je kleiner der Wert ist, desto höher ist die Präferenz

host

- Name des Mailserverns, an den die Mail ausgeliefert wird

Klasse: CNAME (Canonical Name-Record)

Das Resource-Record CNAME definiert einen Alias für den offiziellen Namen eines Hosts.

Format:

```
spitzname [ttl] IN CNAME host
```

Klasse: PTR (Domain Name Pointer-Record)

PTR Resource Records werden zur Umwandlung von numerischen IP-Adressen in Hostnamen verwendet.

Format:

```
name [ttl] IN PTR host
```

name

- Nummer relativ zur aktuellen *in-addr.arpa*-Domain;
- in umgekehrter Reihenfolge;

host

- voll qualifizierter Domain-Name des Computers;

2.2.3 Beispiele

named.local

Die Datei *named.local* wird verwendet, um die Adresse 172.0.0.1 (die »loopback-Adresse«) auf den Namen *localhost* abzubilden. Es handelt sich um die Zonendatei für die Reverse Domain 0.0.127.IN-ADDR.ARPA.

```
@      IN      SOA      ServerII.hgII.rsa. root.ServerII.hgII.rsa. (
                                1          ; Serial
                                360000     ; Refresh alle 100 Stunden
                                3600      ; Retry nach einer Stunde
                                3600000    ; Expire nach 1000 Stunden
                                360000     ; Standard-TTL von 100 Stunden
                                )
                                IN      NS      ServerII.hgII.rsa.
0      IN      PTR      loopback.
1      IN      PTR      localhost.
```

named.rev.192.168.3

Die Datei *named.rev.192.168.3* und *named.local* sind in ihrer Struktur sehr ähnlich, da bei Dateien IP-Adressen auf Hostnamen abbilden. Daher enthalten sie PTR-Records. Diese Beispieldatei ist die Zonendatei für 3.168.192.IN-ADDR.ARPA auf dem Server *ServerII.hgII.rsa*. Der SOA-Record wird in jeder Zone verwendet und referenziert immer den richtigen Domain-Namen, weil es die Domain referenziert, die für diese Zonendatei in *named.boot* angegeben wurde. Die NS-Records definierten die Nameserver für die Domain und werden generell unmittelbar nach SOA aufgeführt. Somit ist sichergestellt, daß vor den Nameserver-Einträgen der Domain-Name nicht geändert werden kann.

```
;
; Abbildung von Adressen auf Hostnamen
@      IN      SOA      ServerII.hgII.rsa. root.ServerII.hgII.rsa. (
                                10099     ; Serial
                                43200     ; Refresh
                                3600      ; Retry
                                3600000    ; Expire
                                2592000    ; Minimum
                                )
                                IN      NS      ServerII.hgII.rsa.
                                IN      NS      ServerI.hgI.rsa.
1      IN      PTR      ServerII.hgII.rsa.
10     IN      PTR      EDV1Lehrer.hgII.rsa.
11     IN      PTR      EDV1RTN01.hgII.rsa.
12     IN      PTR      EDV1RTN02.hgII.rsa.
```

named.hosts.hgII.rsa

Ebenso wie *named.rev.192.168.3* beginnt auch *named.hosts.hgII.rsa* mit einem SOA- und einigen NS-Records. Der erste MX-Record bestimmt einen Mailserver für die gesamte Domain und den zweiten MX-Record bestimmt den Backup-Mailserver, falls der erste nicht zu erreichen ist. Der CNAME-Record definiert einen Aliasnamen, in diesem Beispiel kann »*ServerII*« auch

über »Logon« angesprochen werden. Dem A-Record für »Proxy« folgt ein MX-Record, der alle an »benutzer@proxy.hgII.rsa« gerichteten Mail an »ServerII« weiterleitet. Dieses MX-Record ist notwendig, weil die MX-Records am Anfang der Zonendatei nur Mails weiterleiten, wenn sie direkt an »benutzer@nuts.com« gehen. Wenn an »Proxy« gerichtete Mails weitergeleitet werden sollen, muß dies explizit durch einen MX-Record angegeben werden.

```

;
; Adressen und weitere Host-Informationen
@           IN      SOA      ServerII.hgII.rsa. root.ServerII.hgII.rsa. (
                                10099      ; Serial
                                43200      ; Refresh
                                3600       ; Retry
                                3600000    ; Expire
                                2592000    ; Minimum
                                )
;
; Definition der Name- und Mail-Server
;
                IN      NS      ServerII.hgII.rsa.
                IN      NS      ServerI.hgI.rsa.
                IN      MX      10 ServerII.hgII.rsa.
                IN      MX      20 ServerI.hgI.rsa.
;
; Definition von Localhost
;
localhost    IN      A        127.0.0.1
;
; Definition der Hosts dieser Zone
;
ServerII     IN      A        192.168.3.1
              IN      MX      5 ServerII.hgII.rsa.
logon        IN      CNAME    ServerII.hgII.rsa.
Proxy        IN      A        192.168.3.2
              IN      MX      5 ServerII.hgII.rsa.
EDV1Lehrer  IN      A        192.168.3.10
EDV1RTN01   IN      A        192.168.3.11
EDV1RTN02   IN      A        192.168.3.12

```

Cache-Initialisierungsdatei (named.ca)

Die cache-Anweisung in *named.boot* verweist auf eine cache-Initialisierungsdatei. Jeder einen Cache pflegende Server besitzt eine solche Datei. Sie enthält Informationen, die beim Start des Nameservers zum Aufbau eines Caches mit Ddomaindaten notwendig sind. Die root-Domain wird in der cache-Anweisung in Form eines einzelnen Punktes angegeben, und *named.ca* enthält die Namen und adressen der Root-Server.

Diese Datei kann von **rs.internic.net** per anonymous FTP oder Gopher heruntergeladen werden. Sie liegt im Verzeichnis »domain« mit dem Namen »named.boot«. Sie sollte alle paar Monate heruntergeladen werden, damit der Cache mit korrekten Informationen versorgt wird.

```

;
; formerly NS.INTERNIC.NET
;
.                3600000    IN      NS      A.ROOT-SERVERS.NET.

```

Konfiguration des DNS und BIND v4

```
A.ROOT-SERVERS.NET.      3600000      A      198.41.0.4
;
; formerly NS1.ISI.EDU
;
.                          3600000      NS      B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.      3600000      A      128.9.0.107
;
; formerly C.PSI.NET
;
.                          3600000      NS      C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET.      3600000      A      192.33.4.12
;
; formerly TERP.UMD.EDU
;
.                          3600000      NS      D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET.      3600000      A      128.8.10.90
;
; formerly NS.NASA.GOV
;
.                          3600000      NS      E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET.      3600000      A      192.203.230.10
;
; formerly NS.ISC.ORG
;
.                          3600000      NS      F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET.      3600000      A      192.5.5.241
;
; formerly NS.NIC.DDN.MIL
;
.                          3600000      NS      G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET.      3600000      A      192.112.36.4
;
; formerly AOS.ARL.ARMY.MIL
;
.                          3600000      NS      H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET.      3600000      A      128.63.2.53
;
; formerly NIC.NORDU.NET
;
.                          3600000      NS      I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET.      3600000      A      192.36.148.17
;
; temporarily housed at NSI (InterNIC)
;
.                          3600000      NS      J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET.      3600000      A      198.41.0.10
;
; housed in LINX, operated by RIPE NCC
;
.                          3600000      NS      K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET.      3600000      A      193.0.14.129
;
; temporarily housed at ISI (IANA)
;
.                          3600000      NS      L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET.      3600000      A      198.32.64.12
;
; housed in Japan, operated by WIDE
;
```

```
.           3600000      NS      M.ROOT-SERVERS.NET.  
M.ROOT-SERVERS.NET. 3600000      A       202.12.27.33  
; End of File
```

Literatur

- [1] Craig Hunt. *TCP/IP Netzwerk-Administration*. O'Reilly Verlag, Deutsche Ausgabe der 2. edition, January 1999.

Liste der Versionen

Version	Datum	Bearbeiter	Bemerkung
0.5	25.06.2000	Bri	Dokumenterstellung
0.51	11.02.2005	Bri	Neusatz des Dokuments
0.52	24.05.2005	Bri	Adressänderungen aufgrund Domainwechsel