Protokoll Nr.4	Höhere Technische Bundeslehranstalt Fischergasse 30 A-4600 Wels	Abteilung **IT**

Protokoll

Übungs Nr.: 4	Titel der Übung: DHCP- und DNS-Server		
Katalog Nr.: 3	Verfasser: Christian Bartl	Jahrgang:	**4.AIT**
An dieser Übung haben mitgearbeitet:	Thomas Fischl	Gruppe:	**1**
		Datum der Übung:	02.02.2006
		Abgabe Datum:	09.02.2006

Übungsleiter: Prof. Hell

	Beurteilung:

DHCP- und DNS-Server

Angabe

NWSY – DHCP- und DNS-Server unter Linux installieren, konfigurieren und (mit Windows) testen

ACHTUNG:
Für diese Übung darf der Linux-PC nicht an das Schulnetz angeschlossen werden, da er als DHCP-Server Konflikte verursachen kann/wird.

Im Netzwerklabor ist auf den bereits mit Linux installierten Wechselfestplatten ein DHCP-Server und dazu ergänzend ein DNS-Server einzurichten und in Linux und Windows zu testen.
Als Vorbereitung sind Recherchen durchzuführen um auf folgende Fragen Antworten zu finden. Diese sind im Protokoll zu dokumentieren.

1. Welche Aufgabe erfüllt das DHCP-Protokoll und welche Dienste/Informationen werden von einem DHCP-Sever darüber zur Verfügung gestellt?
2. Listen Sie die Vorteile des Einsatzes eines DHCP-Servers in einem LAN auf.
3. Worin besteht die Gefahr, dass zwei DHCP-Server im selben Subnetz aktiv sind?
4. Ist DHCP typischerweise ein globaler Dienst oder ein lokaler Dienst innerhalb eines LAN/WAN?
5. Was ist das Domain Name System und welche Dienste werden von diesem angeboten?
6. Beschreiben Sie den globalen Aufbau des Domain Name Systems.
7. Ist der Betrieb mehrerer DNS-Server im selben Subnetz problematisch und wenn ja, warum?
8. Welche zusätzlichen Aspekte ergeben sich aus dem gemeinsamen Einsatz eines DNS- und DHCP-Servers innerhalb eines Firmennetzwerks?
9. Welche Bedeutung haben Zonen-Angaben für den DNS-Server und was ist darunter zu verstehen?

Folgende Anforderungen werden an die zu erstellende Konfiguration gestellt:

1. Der DHCP-Server ist im Klasse-C-Netz 192.168.<PC-ID> mit der Domäne unetz<PC-ID>.htl-wels mit der festen IP-Adresse 192.168.<PC-ID>.254 einzurichten.
2. Die dynamisch zu vergebenden Adressen umfassen 192.168.1.1 bis 192.168.1.200. Die restlichen IP-Adressen sind reserviert für spezielle Server im Netz (Firewall, Gateway,…).
3. Die dynamische Konfiguration der DNS-Datenbank ist vorerst noch auszuschalten. Diese macht erst bei Betrieb eines DNS-Servers Sinn.
4. Nach Installation, Konfiguration und Inbetriebnahme des DHCP-Servers ist dieser über einen Windows-PC zu testen, indem versucht wird, dynamisch eine IP-Adresse anzufordern.
5. Als nächstes ist der Linux-PC für die oben festgelegte Domäne um einen Nameserver zu erweitern.
6. Nach erfolgtem erfolgreichem Test des DNS-Servers stellen Sie diesen und den DHCP-Server um auf dynamische Aktualisierung der DNS-Eintrage.

Unterlagen/Links:
Internet Systems Consortium: www.isc.org/index.pl?/sw/dhcp/
DHCP-Mini-HOWTO: http://www.tldp.org/HOWTO/DHCP/index.html
DNS-DHCP-Setup: http://www.arda.homeunix.net/dnssetup.html
DNS,DHCP,LDAP,...: http://www.bind9.net/

Manual-Pages: dhcpd, dhcpd.conf, dhcpd.leases, dhcp-options, dhc-eval, named, named.conf, hosts, host.conf, resolv.conf,...
...

Protokoll:
Das Vorgehen beim Einrichten ist ebenso zu protokollieren wie die durchgeführten Tests (Screenshots). Auch die in anderen Programmen vorgenommenen Einstellungen (Firewall) sind zu dokumentieren und zu erläutern.

Durchführung

1. **Konfigurieren des DHCP Servers**
   Dazu muss die /etc/dhcpd.conf editiert werden. Hier werden die Netze eingetragen, aus denen die IP Adressen verteilt werden. Man kann mit dem DHCP Server auch noch Zusatzinformationen mitschicken. Z.B.: Domainnamen, DNS Server, usw.

```
option broadcast-address 192.168.1.255;
option subnet-mask 255.255.255.0;

option domain-name-servers 192.168.1.254;
option domain-name "unet19.htl.at";

subnet 192.168.1.0 netmask 255.255.255.0 {
        range 192.168.1.10 192.168.1.20;
        default-lease-time 86400;
        max-lease-time 86400;
        # 86400 sekunden => 24 Stunden => 1 Tag
}
```

   Jetzt muss noch festgelegt werden, auf welchem Interface sich Dhcp Server verbinden soll. In diesem Fall wäre es das eth1. Dies wird in der Datei /etc/sysconfig/dhcpd eingestellt. Natürlich muss dieses Interface eine passende IP Adresse bekommen, sonst lässt sich der Dhcp Server nicht starten.

```
DHCPD_INTERFACE="eth1"
```

   DHCP-Server starten: rcdhcpd start

2. **Konfigurieren des DNS Servers**
   Um den DNS Server zu konfigurieren, muss man die /etc/named.conf editieren. Es müssen die Zonen eingetragen werden, für die der DNS Server zuständig ist. Für jede Zone die konfiguriert wird, muss ein eignes Zonefile erstellt werden, aber dazu später. Die erste Zone ist für die Vorwärtsauflösung der „unet19.htl.at“ Domain und die Zweite ist für die Rückwertsauflösung.

```
zone "unet19.htl.at" {
        type master;
        file " unet19.htl.at.zone";

};

zone "1.168.192.in-addr.arpa" {
        type master;
        file 192.168.1.zone";
};
```

   Nun fehlen noch die vorher erwähnten Zonenfiles. Diese Zonenfiles müssen im Verzeichnis /var/lib/named/ selbst angelegt werden.

/var/lib/named/unet19.htl.at.zone

```
unet19.htl.at          IN      SOA     unet19.htl.at. root.unet19.htl.at. (
                       2005122808      ; Serial
                       8H      ; Refresh       8 hours
                       2H      ; Retry         2 hours
                       1W      ; Expire        1 week
                       1D      ; Minimum       1 day
                       )

IN      NS      Fischi.unet19.htl.at.

Notebook        IN      A       192.168.1.20
```

/var/lib/named/192.168.1.zone

```
1.168.192.in-addr.arpa          IN      SOA     unet19.htl.at.  root.unet19.htl.at. (
                                2005122807      ; Serial
                                8H      ; Refresh       8 hours
                                2H      ; Retry         2 hours
                                1W      ; Wxpire        1 week
                                1D )    ; Minimum       1 day

IN      NS      Fischi.unet19.htl.at.

20      IN      PTR     Notebook.unet19.htl.at.
```

3. Konfiguration des DDNS Servers

Der unterschied zum DNS Server ist der, dass die Einträge in die Zonefiles nicht mehr selbst statisch eingetragen werden müssen, sonder selbständig generiert wird. Dazu muss der DNS Server mit dem DHCP Server kommunizieren. Wenn der DHCP Server eine IP Adresse verteilt, fragt er das Gerät nach seinem Namen. Dieser Name wird nun eingetragen und der DNS Server kann diesen Namen auflösen. Um dieses zu realisieren erfordert es die dhcpd.conf und named.conf zu bearbeiten.

/etc/dhcp.conf

```
ddns-update-style interim;
ignore client-updates;
include "/etc/named.keys";


#unet19.htl.at. subnet

subnet 192.168.1.0 netmask 255.255.255.0 {
        range 192.168.1.10 192.168.1.20;
        default-lease-time 86400;
        max-lease-time 86400;
```

```
        zone unet19.htl.at. {
                primary 192.168.1.254;
                key DHCP_UPDATER;
        }

        zone 1.168.192.in-addr.arpa. {
                primary 192.168.1.254;
                key DHCP_UPDATER;
        }
}
```

/etc/named.conf

```
        options {

                # The directory statement defines the name server's working directory

                directory "/var/lib/named";

                notify no;
        };


        #Unet19 zone einstellungen für das lokale netzwerk

        zone "unet19.htl.at" {
                type master;
                file "dyn/unet19.htl.at.zone";
                allow-update {
                        key DHCP_UPDATER;
                };
        };

        zone "1.168.192.in-addr.arpa" {
                type master;
                file "dyn/192.168.1.zone";
                allow-update {
                        key DHCP_UPDATER;
                };
        };

        include "/etc/named.conf.include";
```

/etc/named.conf.include

```
        include "/etc/named.keys";
```

/etc/named.keys

```
        # generated by genDDNSkey on Thu Feb  2 09:53:42 CET 2006

        key DHCP_UPDATER {
```

```
algorithm hmac-md5;
secret
"04glJCUEFXPmZsgT01eH7HfaZxvtcxHm9o0ksxg9paiQAj84dgia9jqF6a49hjSuYI
l97Wb+1OU3xlGAVohiBw==";
};
```

Dieses File wird mittels eines Skripts erzeugt.

linux:/usr/bin # genDDNSkey -help

Usage:

genDDNSkey <options>

Options:

```
-f|--key-file <FILENAME>      includable key is written to this file
                        (default: /etc/named.keys)
-n|--key-name <NAME>          name of the key (default: DHCP_UPDATER)
-d|--key-dir <NAME>           public / private key directory
                        (default is key-file directory)
-r|--random                   random device to use (default: /dev/random)
--force                       overwrite an existing key file
--help                        print usage info
```

See /usr/share/doc/packages/dhcp-server/DDNS-howto.txt (in dhcp-server package) about configuration of a DHCP server to do DDNS updates.

Die verändert Zonefiles findet man am Ende dieses Protokolls.

Achtung: Die Basisverzeichnisse und auch die Unterverzeichnisse für den Dns Server und den Dhcp Server benötigen schreib und lese Rechte für die Benutzer. Die betroffenen Verzeichnisse sind /var/lib/named und /var/lib/dhcp.

linux:/var/lib # chown dhcpd dhcp -R

drwxr-xr-x 7 dhcpd root 168 Dec 25 14:17 dhcp

```
linux:/var/lib # ls -l dhcp
total 1
drwxr-xr-x  7 dhcpd root  168 Dec 25 14:17 .
drwxr-xr-x 51 root  root 1328 Feb  8 21:01 ..
drwxr-xr-x  2 dhcpd root  112 Feb  8 20:52 db
drwxr-xr-x  2 dhcpd root   72 Feb  8 20:45 dev
drwxr-xr-x  2 dhcpd root  200 Feb  8 20:51 etc
drwxr-xr-x  2 dhcpd root  112 Feb  8 20:51 lib
drwxr-xr-x  3 dhcpd root   72 Dec 25 14:17 var
```

```
linux:/var/lib # chown named named –R

drwxr-xr-x   9 named   root        312 Feb  8 21:39 named

linux:/var/lib # ls -l named
total 13
drwxr-xr-x   9 named root   312 Feb  8 21:39 .
drwxr-xr-x  51 root  root  1328 Feb  8 21:01 ..
-rw-r--r--   1 named root   192 Jul  4  2001 127.0.0.zone
drwxr-xr-x   2 named root   120 Feb  8 20:45 dev
drwxr-xr-x   2 named named   48 Sep  9 20:46 dyn
drwxr-xr-x   3 named root    72 Dec 25 13:54 etc
-rw-r--r--   1 named root   158 Jul  4  2001 localhost.zone
drwxr-xr-x   2 named named   48 Sep  9 20:46 log
drwxr-xr-x   2 named root    48 Sep  9 20:46 master
-rw-r--r--   1 named root  2517 Sep  9 20:46 root.hint
drwxr-xr-x   2 named named   48 Sep  9 20:46 slave
drwxr-xr-x   4 named root   120 Dec 25 13:54 var
```

Fehlerbehebung: Wenn die Fehlermeldung auftritt, dass der Dns Key nicht korrekt ist, oder so ähnlich, sollte man nachsehen, ob die .keys Files in die etc Verzeichnisse der Dhcp und Dns Servers sind. Wenn dies nicht der Fall ist, können diese händisch nachkopiert werden. Diese File sollte in den Verzeichnissen /var/lib/named/etc/xxx.keys und /var/lib/dhcp/etc/xxx.keys liegen.

Wenn sie nicht vorhanden sind, kopiert man einfach die named.keys aus dem /etc Verzeichnis in diese beiden Verzeichnisse.

Am ende müssen dem User named noch Rechte auf die named.keys im Verzeichnis /var/lib/named/etc zugeteilt werden:

```
/var/lib/named/etc#chown named named.keys
```

4. DHCP- und DNS-Server starten

DNS-Server starten: rcnamed start
DHCP-Server starten: rcdhcpd start

Starten mit start
Stoppen mit stop
Status abfragen mit status

tail –f dateiname gibt immer das Ende der Datei aus und aktualisiert auch die Ausgabe.

Konfigurationsdateien

/etc/sysconfig/dhcpd

```
## Path:        Network/DHCP/DHCP server
## Description: DHCP server settings
## Type:        string
## Default:     ""
## ServiceRestart: dhcpd
#
# Interface(s) for the DHCP server to listen on.
#
# Instead of the interface name, the name of its configuration can be given.
# If the configuration file is named
#    /etc/sysconfig/network/ifcfg-eth-id-00:50:fc:e4:f2:65
# then id-00:50:fc:e4:f2:65 would be suitable to identify the configuration.
#
# Examples: DHCPD_INTERFACE="eth0"
#           DHCPD_INTERFACE="eth0 eth1 eth2 tr0 wlan0"
#           DHCPD_INTERFACE="internal0 internal1"
#           DHCPD_INTERFACE="id-00:50:fc:e4:f2:65 id-00:a0:24:cb:cc:5c wlan0"
#
DHCPD_INTERFACE="eth1"

## Type:        yesno
## Default:     yes
## ServiceRestart: dhcpd
#
# Shall the DHCP server dhcpd run in a chroot jail (/var/lib/dhcp)?
#
# Each time you start dhcpd with the init script, /etc/dhcpd.conf will
# be copied to /var/lib/dhcp/etc/.
#
# Some files that are important for hostname to IP address resolution
# (/etc/{hosts,host.conf,resolv.conf,localtime}, /lib/libnss_dns.so.2,
# /lib/libresolv.so.2) will also be copied to the chroot jail by the
# init script when you start it (about 100kB altogether).
#
# The pid file will be in /var/lib/dhcp/var/run/dhcpd.pid.
#
DHCPD_RUN_CHROOTED="yes"

## Type:        string
## Default:     ""
## ServiceRestart: dhcpd
#
# Since version 3, dhcpd.conf can contain include statements.
# If you enter the names of any include files here, _all_ conf
# files will be copied to $chroot/etc/, when dhcpd is started in the
# chroot jail. (/etc/dhcpd.conf is always copied.)
#
# For your convenience, you can also specify entire directories, like
# "/etc/dhcpd.conf.d".
#
# Example: "/etc/dhcpd.conf.shared /etc/dhcpd.conf.bootp-clients"
#
DHCPD_CONF_INCLUDE_FILES="/etc/named.keys"

## Type:        string
## Default:     "dhcpd"
## ServiceRestart: dhcpd
```

/etc/dhcpd.conf

```
option broadcast-address 192.168.1.255;
option subnet-mask 255.255.255.0;

option domain-name-servers 192.168.1.254;
option domain-name "unet19.htl.at";

ddns-update-style interim;
ignore client-updates;
include "/etc/named.keys";


#unet19.htl.at. subnet

subnet 192.168.1.0 netmask 255.255.255.0 {
        range 192.168.1.10 192.168.1.20;
        default-lease-time 86400;
        max-lease-time 86400;

        zone unet19.htl.at. {
                primary 192.168.1.254;
                key DHCP_UPDATER;
        }

        zone 1.168.192.in-addr.arpa. {
                primary 192.168.1.254;
                key DHCP_UPDATER;
        }
}
```

/etc/named.conf

```
options {

        # The directory statement defines the name server's working directory

        directory "/var/lib/named";

        notify no;
};


# The following zone definitions don't need any modification.  The first one
# is the definition of the root name servers.  The second one defines
# localhost while the third defines the reverse lookup for localhost.

zone "." in {
        type hint;
        file "root.hint";
};

zone "localhost" in {
        type master;
        file "localhost.zone";
        allow-update { none; };
};
```

```
zone "0.0.127.in-addr.arpa" in {
        type master;
        file "127.0.0.zone";
        allow-update { none; };
};

#Unet19 zone einstellungen für das lokale netzwerk

zone "unet19.htl.at" {
        type master;

        file "dyn/unet19.htl.at.zone";
        allow-update {
                key DHCP_UPDATER;
        };
};

zone "1.168.192.in-addr.arpa" {
        type master;

        file "dyn/192.168.1.zone";
        allow-update {
                key DHCP_UPDATER;
        };
};
# Include the meta include file generated by createNamedConfInclude.  This
# includes all files as configured in NAMED_CONF_INCLUDE_FILES from
# /etc/sysconfig/named

include "/etc/named.conf.include";
```

/var/lib/named/dyn/unet19.htl.at.zone

```
$ORIGIN.
$TTL 86400
unet19.htl.at   IN      SOA     unet19.htl.at. root.unet19.htl.at. (
                        2005122808      ; Serial
                        8H      ; Refresh       8 hours
                        2H      ; Retry   2 hours
                        1W      ; Expire 1 week
                        1D      ; Minimum       1 day
                        )

        IN      NS      Fischi.unet19.htl.at.



$ORIGIN 1.168.192.in-addr-arpa.
```

/var/lib/named/dyn/192.168.1.zone

```
$ORIGIN .
$TTL 600
1.168.192.in-addr.arpa  IN      SOA     unet19.htl.at.      root.unet19.htl.at. (
                        2005122807      ; Serial
                        8H      ; Refresh       8 hours
                        2H      ; Retry         2 hours
```

```
                        1W      ; Wxpire        1 week
                        1D )    ; Minimum       1 day

        IN      NS      Fischi.unet19.htl.at.

$ORIGIN unet19.htl.at.
```

Theorie

1. Welche Aufgabe erfüllt das DHCP-Protokoll und welche Dienste/Informationen werden von einem DHCP-Sever darüber zur Verfügung gestellt?

   Das DHCP-Protokoll dient dazu Rechnern in einem Netzwerk automatisch Netzwerkeinstellungen zuzuweisen. DHCP verwendet das BOOTP-Protokoll mit dem sich laufwerkslose Workstations realisieren lassen. DHCP übermittelt an die Clients folgende Informationen:
   - IP-Adresse
   - Subnetmask
   - Gateway-Server
   - Time Server
   - Name Server
   - Domain Name Server
   - WINS-Server
   - Domain Name
   - Default IP TTL
   - Broadcast Address
   - SMTP Server
   - POP3 Server

2. Listen Sie die Vorteile des Einsatzes eines DHCP-Servers in einem LAN auf.

   Mitte Hilfe eines DHCP-Servers ersparen sich die Clients im Netzwerk die Suche der richtigen Konfigurationseinstellungen und das Konfigurieren selbst. Einsatzgebiete sind: WLAN-Hotspots, große Netzwerke in denen sich die Topologien ständig ändern, Netzwerke in denen ständig Clients hinzugefügt und wieder entfernt werden, …

3. Worin besteht die Gefahr, dass zwei DHCP-Server im selben Subnetz aktiv sind?

   Es ist nicht vorhersehbar welcher der beiden Server die Netzwerkeinstellungen vergibt und damit kann es passieren, das Clients eine falsche Konfiguration erhalten.

4. Ist DHCP typischerweise ein globaler Dienst oder ein lokaler Dienst innerhalb eines LAN/WAN?

   DHCP ist ein lokaler Dienst innerhalb eines LAN's oder WAN'S (z.b.: im WAN eines Providers, im Schnulnetz)

5. Was ist das Domain Name System und welche Dienste werden von diesem angeboten?

   Ein Domain Name System ist ein verteiltes hierarchisches System zur Auflösung von Computernamen in IP-Adressen und umgekehrt.

6. Beschreiben Sie den globalen Aufbau des Domain Name Systems.

   Komponenten des DNS
   Das DNS besteht aus drei Hauptkomponenten:
   - Domain-Namensraum
   - Nameservern
   - Resolver

Domain-Namensraum

Der Domain-Namensraum hat eine baumförmige Struktur. Die Blätter und Knoten des Baumes werden als Labels bezeichnet. Ein kompletter Domainname eines Objektes besteht aus der Verkettung aller Labels. Label sind Zeichenketten (alphanumerisch, als einziges Sonderzeichen ist '-' erlaubt), die mindestens ein Zeichen und maximal 63 Zeichen lang sind. Die einzelnen Label werden durch Punkte voneinander getrennt. Ein Domain Name wird mit einem Punkt abgeschlossen (der hinterste Punkt wird normalerweise weggelassen, gehört rein formal aber zu einem vollständigen Domain Namen dazu). Ein korrekter, vollständiger Domain Name (auch Fully Qualified Domain Name (FQDN) genannt) lautet etwa www.wikipedia.de. (der letzte Punkt gehört zum Domain Name).
Ein Domain Name darf inklusive aller Punkte maximal 255 Zeichen lang sein.
Ein Domain Name wird immer von rechts nach links delegiert und aufgelöst, das heißt je weiter rechts ein Label steht, umso höher steht es im Baum. Der Punkt am rechten Ende eines Domainnamens trennt das Label für die erste Hierarchieebene von der root. Diese erste Ebene wird auch als Top Level Domain (TLD) bezeichnet.
Die DNS-Objekte einer Domäne (zum Beispiel die Rechnernamen) werden als Satz von Resource Records meist in einer Zonendatei gehalten, die auf einem oder mehreren autoritativen Nameservern vorhanden ist. Anstelle von Zonendatei wird meist der etwas allgemeinere Ausdruck Zone verwendet.

Nameserver

Nameserver sind Programme, die Anfragen zum Domain-Namensraum beantworten. Man unterscheidet zwischen autoritativen und nicht-autoritativen Nameservern.
Ein autoritativer Nameserver ist verantwortlich für eine Zone. Seine Informationen über diese Zone werden deshalb als gesichert angesehen. Für jede Zone existiert mindestens ein autoritativer Server, der Primary Nameserver. Dieser wird im SOA Resource Record, einer Zonendatei aufgeführt. Aus Redundanz- und Lastverteilungsgründen werden autoritative Nameserver fast immer als Server-Cluster realisiert, wobei die Zonendaten identisch auf einem oder mehreren Secondary Nameservern liegen. Die Synchronisation zwischen Primary und Secondary Nameservern erfolgt per Zonentransfer.
Ein nicht-autoritativer Nameserver bezieht seine Informationen über eine Zone von anderen Nameservern sozusagen aus zweiter oder dritter Hand. Seine Informationen werden als nicht gesichert angesehen. Da sich DNS-Daten normalerweise nur sehr selten ändern, speichern nicht-autoritative Nameserver die einmal von einem Resolver angefragten Informationen im lokalen RAM ab, damit diese bei einer erneuten Anfrage schneller vorliegen. Diese Technik wird als Caching bezeichnet. Jeder dieser Einträge besitzt ein eigenes Verfallsdatum (TTL time to live), nach dessen Ablauf der Eintrag aus dem Cache gelöscht wird. Die TTL wird dabei durch einen autoritativen Nameserver für diesen Eintrag festgelegt und wird nach der Änderungswahrscheinlichkeit des Eintrages bestimmt (sich häufig ändernde DNS-Daten erhalten eine niedrige TTL). Das kann unter Umständen aber auch bedeuten, dass der Nameserver in dieser Zeit falsche Informationen liefern kann, wenn sich die Daten zwischenzeitlich geändert haben. Ein Spezialfall ist der caching only Nameserver. In diesem Fall ist der Nameserver für keine Zone verantwortlich und muss alle eintreffenden Anfragen über weitere Nameserver auflösen.

Strategien

Damit ein nicht-autoritativer Nameserver Informationen über andere Teile des Namensraumes finden kann, bedient er sich folgender Strategien.

Delegierung
Teile des Namensraumes einer Domain werden oft an Subdomains mit dann eigens zuständigen Nameservern ausgelagert. Ein Nameserver einer Domäne kennt die zuständigen Nameserver für diese Subdomains aus seiner Zonendatei und delegiert Anfragen zu diesem untergeordneten Namensraum an einen dieser Nameserver.

Weiterleitung
Falls der angefragte Namensraum außerhalb der eigenen Domäne liegt, wird die Anfrage an einen fest konfigurierten Nameserver weitergeleitet.

Auflösung über die Root-Server
Falls kein Weiterleitungsserver konfiguriert wurde oder dieser nicht antwortet, werden die Root-Server befragt. Dazu werden in Form einer statischen Datei die Namen und IP-Adressen der Root-Server hinterlegt. Es gibt 13 Root-Server (Server A bis M). Die Root-Server beantworten ausschließlich iterative Anfragen. Sie wären sonst mit der Anzahl der Anfragen schlicht überlastet.
DNS-Anfragen werden normalerweise auf Port 53 UDP beantwortet. Falls die Antwort sehr umfangreich ausfällt (größer 512 Bytes), wird diese auf Port 53 TCP übermittelt. Zonentransfers werden stets auf Port 53 TCP durchgeführt.

Nameserversoftware

- BIND (Berkeley Internet Name Domain) ist der Ur-Nameserver und heute noch die meistgenutzte Nameserversoftware, nicht zuletzt da er die Referenzimplementation der meisten RFCs zu DNS darstellt. BIND ist Open Source Software.
- djbdns gilt als sehr sicher und erfreut sich steigender Beliebtheit, wird aber von Dan Bernstein nicht mehr weiterentwickelt, weil er es als fertig ansieht.
- PowerDNS war eine kostenpflichtige Implementierung, die inzwischen auch unter der GPL erhältlich ist und vor allem für das direkte Betreiben von Zonen aus SQL-Datenbanken und LDAP-Verzeichnissen bekannt ist.
- MyDNS ist eine weitere Open-Source-Software, die insbesondere auf MySQL- und PostgreSQL-Datenbanken spezialisiert ist.
- Xyria:DNSd ist ein performance-optimierter DNS-Server, der etwa doppelt so schnell ist wie Bind. Xyria:DNSd ist derzeit noch recht minimalistisch und unterstützt keine Zonetransfers (ausser etwa via SSH), dafür aber extrem sicher und stabil.
- NSD ist optimiert für Server die ausschließlich autoritative Antworten liefern sollen.

Resolver
Resolver sind Ansammlungen von Bibliotheken, die Informationen aus den Nameservern abrufen können. Sie bilden die Schnittstelle zwischen Anwendung und Nameserver. Der Resolver übernimmt die Anfrage einer Anwendung, ergänzt sie falls notwendig zu einem FQDN und übermittelt sie an den fest konfigurierten Nameserver.
Ein Resolver arbeitet entweder iterativ oder rekursiv und informiert den Nameserver über die verwendete Arbeitsweise. Übliche Resolver von Clients arbeiten ausschließlich rekursiv, sie werden dann auch als Stub-Resolver bezeichnet.

Bei einer rekursiven Anfrage schickt der Resolver eine Anfrage an einen ihm bekannten Nameserver und erwartet von ihm eine eindeutige Antwort. Diese Antwort enthält entweder den gewünschten Resource Record oder „gibt es nicht“. Rekursiv arbeitende Resolver überlassen also die Arbeit zur vollständigen Auflösung anderen.
Bei einer iterativen Anfrage bekommt der Resolver entweder den gewünschten Resource Record oder die Adresse eines weiteren Nameserver, den er als nächsten fragt. Der Resolver hangelt sich so von Nameserver zu Nameserver bis er bei einem autoritativen Nameserver landet.
Die so gewonnene Antwort übergibt der Resolver an das Programm, das die Daten angefordert hat, beispielsweise an den Webbrowser.
Bekannte Programme zur Überprüfung der Namensauflösung sind nslookup, host und dig. Weitere Informationen zur iterativen/rekursiven Namensauflösung finden sich unter rekursive und iterative Namensauflösung.

Beispiel
Im Beispiel wird www.example.net „per Hand“ aufgelöst. Die Adresse von A.root-servers.net (198.41.0.4) wird dabei als bekannt vorausgesetzt, die Ausgabe ist auf das Wesentliche gekürzt.

```
$ dig +norecurse @198.41.0.4 www.example.net
   net.                 172800  IN      NS      A.GTLD-SERVERS.net.
   A.GTLD-SERVERS.net.  172800  IN      A       192.5.6.30

$ dig +norecurse @192.5.6.30 www.example.net
   example.net.         172800  IN      NS      a.iana-servers.net.
   a.iana-servers.net.  172800  IN      A       192.0.34.43

$ dig +norecurse @192.0.34.43 www.example.net
   www.example.net.     172800  IN      A       192.0.34.166
```

Bei den von den nicht-zuständigen Nameservern zusätzlich ausgegebenen A-Records handelt es sich um Glue Records.

Erweiterung des DNS
Bisher waren die Label – wie beschrieben – auf alphanumerische Zeichen und das Zeichen '-' eingeschränkt. Dies hängt vor allem damit zusammen, dass das DNS (wie auch das Internet ursprünglich) in den USA entwickelt wurde. Allerdings gibt es in vielen Ländern Zeichen, die nicht in einem Label verwendet werden durften (im deutschen Sprachraum zum Beispiel die Umlaute ä, ö und ü) oder Zeichen aus komplett anderen Schriftsystemen (zum Beispiel Chinesisch). Namen mit diesen Zeichen waren ursprünglich nicht möglich.
1999 beschrieb Paul Vixie im RFC 2671 einige kleinere, abwärtskompatible Erweiterungen am Domain Name System, die als EDNS Version 0 bezeichnet werden. Durch Verwendung von bis dahin reservierten, aber ungenutzten Header-Codes, kann der Anfragende festlegen, dass er UDP-Antworten größer als 512 Bytes entgegennehmen kann. Außerdem wurde es möglich andere Label-Typen zu nutzen. RFC 2673 beschreibt das Binary Label, mit welchem der volle Zeichensatz einer aus Bytes bestehenden Zeichenfolge genutzt werden kann. Da sowohl Resolver als auch Nameserver die Erweiterungen implementieren müssten, erlang EDNS keine weltweit flächendeckende Nutzung.
Ein anderer Ansatz zur Vergrößerung des Zeichenvorrats ist das 2003 in RFC 3490 beschriebene IDNA. Um das neue System mit dem bisherigen kompatibel zu halten, werden die erweiterten Zeichensätze mit erlaubten Zeichen kodiert, also auf derzeit

gültige Namen abgebildet. Die erweiterten Zeichensätze werden dabei zunächst gemäß dem Nameprep-Algorithmus (RFC 3491) normalisiert, und anschließend per Punycode (RFC 3492) auf den für DNS verwendbaren Zeichensatz abgebildet. Das Voransetzen des durch die IANA festgelegten IDNA-Prefix xn-- vor das Ergebnis der Kodierung ergibt das vollständige IDN-Label. IDNA benötigt eine Anpassung des Resolvers und unter Umständen auch der Netzwerkanwendungen, die Nameserver-Infrastruktur braucht jedoch nicht verändert zu werden. Im deutschsprachigen Raum können seit März 2004 deutsche, liechtensteinische, österreichische und schweizer Domains (.de, .li, .at und .ch) mit Umlauten registriert und verwendet werden. Auch bei einigen anderen Top-Level-Domains, insbesondere im asiatischen Raum, ist die Nutzung von IDNA möglich.
Eine weitere aktuelle Erweiterung des DNS stellt ENUM (RFC 2916) dar. Diese Anwendung ermöglicht die Adressierung von Internet-Diensten über Telefonnummern, also das „Anwählen“ von per Internet erreichbaren Geräten mit dem aus dem Telefonnetz bekannten Adressschema. Aus dem breiten Spektrum der Einsatzmöglichkeiten bietet sich insbesondere die Verwendung für Voice over IP Services an.

DynDNS
Es kann nur Rechnern mit fester, sich also nie ändernden IP-Adresse ein fester Rechnername zugeordnet werden. Da jedoch sehr viele Nutzer mit Heimrechnern eine variable IP-Adresse haben (mit jeder Einwahl in das Internet wird eine andere IP-Adresse aus einem Pool zugeteilt = DHCP oder BOOTP), gibt es inzwischen DynDNS-Betreiber (zum Beispiel DynDNS.org oder MyDyn.de), die dafür sorgen, dass man auch mit solch rasch ändernden Adressen möglichst immer über denselben Rechnernamen erreichbar ist.
Siehe auch: Liste der TCP/IP-basierten Netzwerkdienste

Weitere Verwendungszwecke
Im E-Mail-Verkehr wird das DNS verwendet, um abzufragen, ob ein Mailserver ein Open Relay darstellt. Da über offene Relays häufig Spam versandt wird, soll das Spam-Aufkommen durch Ablehnen einer Verbindung oder Verwerfen der E-Mail reduziert werden. Dazu fragt der Mailserver bei einer Realtime Blackhole List (RBL) bzw. DNS-based Blackhole List (DNSBL) an, ob die IP-Adresse der eingehenden SMTP-Verbindung als Open Relay eingetragen ist.
Mit dem Telephone Number Mapping werden Telefonnummern im Domain Name System abgelegt, um die IP-Telefonie zu erleichtern. Mobilfunkbetreiber benutzen DNS bei ihren Triple-A-Systemen, um zum Beispiel intern in ihrem System zu einer IP-Adresse eines Kunden die Mobile Subscriber ISDN Number abzufragen.
Es gibt auch Ansätze, das Domain Name System zum Tunneln von Nutzdaten zu verwenden und so eine Firewall zu umgehen.

DNS-Security
Das DNS ist ein zentraler Bestandteil einer vernetzten IT-Infrastruktur. Eine Störung kann erhebliche Kosten nach sich ziehen und eine Verfälschung von DNS-Daten Ausgangspunkt von Angriffen sein. Mehr als zehn Jahre nach der ursprünglichen Spezifikation wurde DNS um Security-Funktionen ergänzt. Folgende Verfahren sind verfügbar:

- Bei TSIG (Transaction Signatures) handelt es sich um ein einfaches, auf symmetrischen Schlüsseln beruhendes Verfahren, mit dem der Datenverkehr zwischen DNS-Servern gesichert werden kann.

- Bei DNSSEC (DNS Security) wird von einem asymmetrischen Kryptosystem Gebrauch gemacht, mit dem nahezu alle DNS-Sicherheitsanforderungen erfüllt werden können. Neben der Server-Server-Kommunikation wird auch die Client-Server-Kommunikation gesichert.

Domain-Registrierung
Um DNS-Namen im Internet bekannt machen zu können, muss der Besitzer die Domain, die diese Namen enthält, registrieren. Durch eine Registrierung wird sichergestellt, dass bestimmte formale Regeln eingehalten werden und dass Domain-Namen weltweit eindeutig sind. Domain-Registrierungen werden von Organisationen (Registrars) vorgenommen, die dazu von der IANA bzw. ICANN autorisiert wurden. Registrierungen sind gebührenpflichtig.
Detaillierte Informationen finden sich unter Domain-Registrierung.

7. Ist der Betrieb mehrerer DNS-Server im selben Subnetz problematisch und wenn ja, warum?

   Nein, ist es nicht, wenn alle DNS-Server denselben Namensraum verwenden.

8. Welche zusätzlichen Aspekte ergeben sich aus dem gemeinsamen Einsatz eines DNS- und DHCP-Servers innerhalb eines Firmennetzwerks?

   Müssen untereinander Kommunizieren können, um DNS-Namen den richtigen IP-Adressen zuwiesen zu können.

9. Welche Bedeutung haben Zonen-Angaben für den DNS-Server und was ist darunter zu verstehen?

   Die DNS-Objekte einer Domäne (zum Beispiel die Rechnernamen) werden als Satz von Resource Records meist in einer Zonendatei gehalten, die auf einem oder mehreren autoritativen Nameservern vorhanden ist. Anstelle von *Zonendatei* wird meist der etwas allgemeinere Ausdruck **Zone** verwendet.

Links:

http://ops.ietf.org/dns/dynupd/secure-ddns-howto.html