HTL ST. PÖLTEN

Titel: Labor 13 – DNS Server

Klasse: 3BHIF

Name: Haiden

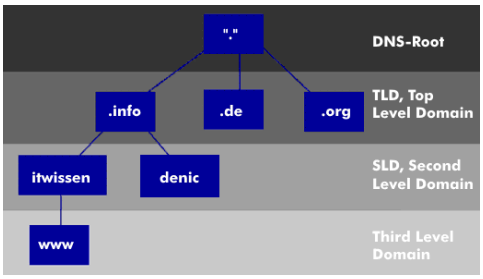
Gruppe: 01

Aufgabe: 13.05.2020 Abgabe: 27.05.2020

Inhaltsverzeichnis

1 Allgemeines

1.1 Wie ist der DNS Namensraum aufgebaut?



Der DNS Namensraum ist hierarchisch aufgebaut. Am ganz oberen Ende steht die Root-Ebene. Sie wird durch den Punkt gekennzeichnet. Danach folgen in zweiter Ebene die TLDs, das heißt Namen wie Endungen wie .de, .org, .com usw…. Danach folgen die Second-Level-Domains, das sind dann die Namen, die man selbst vergeben kann. Third-Level oder auch Subdomains sind in 3.Ebene und darunter angesiedelt. Dies kann man unendlich fortführen, z.B. kann man srv01.frankfurt.host01.nhaiden.dev haben.

1.2 Wie läuft eine Namensauflösung ab?

Immer wenn man eine Anfrage mit einem Domain-Namen stellt, geht dies an den Resolver, einer Komponente des Betriebssystems. Der Resolver speichert IPs im Cache. Wird die benötigte IP allerdings nicht vom Resolver gefunden, leitet dieser die Anfrage an einen DNS Server im Internet bzw. internen Netzwerk weiter. Hat dieser den Namen auch nicht im Cache, gibt er die Anfrage an einen weiteren DNS-Server weiter. Dies geschieht so lange, bis der Name aufgelöst wurde. Die aufgelöste IP-Adresse wird an den Anfragenden über die Server zurückgeschickt und die Seite kann nun aufgerufen werden.

1.3 Welche spezielle Bedeutung hat der Bind Nameserver?

Bind ist ein Nameserver, der von der Universität Berkeley in Kalifornien, USA entwickelt wurde. Er gilt als der Nameserver unter den Nameservern und wurde auf alle Betriebssysteme, die es gibt, portiert. Bind ist der Grundstock des Internets und somit auch größerer Netze.

1.4 Welche Sicherheitsmechanismen bringt DNS (DNSSEC, DOH, DOT)? Kurze Beschreibung.

1.4.1 DNSSEC

DNSSEC ist ein Sicherheitsstandard für DNS, der den Standard um einige Sicherheitsfunktionen erweitert. Erstmal wurde er 1999 vorgestellt, doch da war der den großen Netzen noch nicht gewachsen. Bis 2005 wurde er überarbeitet und wird seit 2010 auf den Top-Root Servern eingesetzt.

DNSSEC ergänzt DNS um eine Quellenauthentifizierung via Public-Key System, ein asymmetrisches Verschlüsselungsverfahren. Ressource Records werden hierbei mit einer Signatur, einem Public Key versehen, der bei Namensauflösung mit einem Private Key des Anfragenden abgeglichen wird.

1.4.2 DOH – DNS Over HTTPS

Da bei DNS die Daten unverschlüsselt übertragen werden, gibt es den neuen Standard DOH (DNS over HTTPS). Er soll die Daten des Anfragenden über HTTPS und somit verschlüsselt übertragen, damit Internetnutzer nicht dem DNS Hijacking ausgesetzt werden können.

1.4.3 DOT – DNS Over TLS

DOT verfolgt das selbe Prinzip wie DOH – Dateien sollen verschlüsselt über einen Tunnel übertragen werden, auf den nur der Anfragende und der Server Zugriff haben, damit der Client kein Opfer von DNS Hijacking werden kann. Für DOT ist ein eigener Port vorgesehen, nämlich Port 853.

1.5 Welche davon setzen Sie ein?

Man kann diese Sicherheitsmechanismen bereits in Browsern wie Firefox aktivieren, nur werden sie noch nicht von allen Websites unterstützt, deswegen setze ich es noch nicht ein.

1.6 Welche Bedeutung haben die Dateien /etc/hosts und /etc/resolv.conf?

1.6.1 Die Hosts Datei

Die Hosts Datei stammt aus den frühen Zeiten des Internets, als es noch kein DNS zur Namensauflösung gab. Hier stehen IP Adressen drinnen, welche zu spezifischen Hostnames gemapped werden. Es ist eine reine Textdatei.

1.6.2 Resolv.Conf

Die Datei /etc/resolv.conf wird dazu verwendet um zu konfigurieren, welcher DNS Resolver vom Betriebssystem verwendet werden soll. Dieser Job wird oft nicht von System-Admins übernommen, sondern von Verwaltungs-Daemons wie SystemD oder resolvconf unter FreeBSD / Unix.

2 Übung

2.1 Installieren Sie das Paket bind9

```
niklas@ubuntu18VM:~$ sudo apt install bind9 dnsutils -y
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen.... Fertig
Die folgenden zusätzlichen Pakete werden installiert:
  bind9utils libirs160 net-tools python3-ply
Vorgeschlagene Pakete:
  bind9-doc resolvconf rblcheck python-ply-doc
Die folgenden NEUEN Pakete werden installiert:
  bind9 bind9utils dnsutils libirs160 net-tools python3-ply
…
```

2.2 Erstellen Sie ein neues Zonenfile für Ihr Heimnetzwerk. Benennen Sie die Zone nach Belieben (z.B. ihrname.lan)

Es gibt Dateien die man als Template für seine eigenen ZoneFiles benutzen kann. Ich habe die db.local genommen und habe eine Kopie angefertigt. Ich habe sie als niklasNetwork.io und die entsprechenden Werte in dem SOA Record umgeändert:

```
cp db.local db.niklasNetwork.io
root@ubuntuVM:/etc/bind# cat db.niklasNetwork.io
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     niklasNetwork.io. admin.niklasNetwork.io. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
```

Damit die Dateien in dem Bind9-DNS Server geladen werden, muss man sie in der named.conf.local eintragen:

```
zone "niklasNetwork.lan" {
        type master;
        file "/etc/bind/db.niklasNetwork.lan";
};
```

2.3 Was bedeuten die Einträge im SOA Record?

```
@       IN      SOA     localhost. root.localhost. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
```

Ein SOA Eintrag besteht aus folgenden Teilen (von oben nach unten):

Der **Zone-Origin** (localhost.) ist der FQDN (Fully-Qualified-Domain-Name). Wichtig ist der Punkt am Ende, deswegen localhost.

Danach kommt der **Zone-Contact**, hier trägt man die E-Mail Adresse des Administrators ein, der den DNS Server verwaltet. Statt @ schreibt man hier einen Punkt. (root.localhost.). Hier sollte man auch nicht den Punkt am Ende vergessen.

Serial ist eine beliebige Seriennummer, die der Administrator festlegen. Am besten ist hier ein Datumsformat wie YYYYMMDDSS, wobei Y = Jahr, M = Monat, D = Tag, SS = Seriennummer. Sie wird erhöht, wenn der Administrator etwas geändert hat.

Die anderen Werte stehen auf bereits voreingestellten Werten, die man nicht unbedingt ändern muss.

Refresh ist die Zeit in Sekunden, die ein sekundärer Server wartet, bis der primäre Server nachfragt, ob sich die Zonendateien verändern hat.

Retry – die Zeit die ein sekundärer Server wartet, bis er die Anfrage an einen primären Server wiederholt.

Expire – gibt die Zeit in Sekunden an die ein sekundärer Server auf einen erfolgreichen Kontakt zum primären Server wartet, bis er die Zonendatei für ungültig erklärt.

NX – gibt die Zeit an, die ungültige DNS-Anfragen zwischengespeichert werden sollen. (0 Sekunden – 3 Stunden).

2.4 Welche Einträge können im Zonenfile vorkommen?

RR oder auch Resource Records genannt, gibt in es in vielen verschiedenen Varianten, hier sind die häufigsten aufgelistet:

RR	Wert	Beschreibung
NS	FQDN eines DNS Servers	
A	IP-Adresse	Eintrag für eine IP-Adresse: Name → IP
CNAME	Richtiger Name	Alias-Definition
MX	Priorität Name	Mailserver, der für die Domain die Emails annimmt, Priorität ist eine Zahl (niedrigere Zahl: zuerst probieren), Name ist Name des Mail-Servers
PTR	FQDN	Ist die Umkehrung des A-Records (IP → Name)

2.5 Machen Sie Einträge für Ihre Geräte im Netzwerk (MX Eintrag nur bei Bedarf)

Ich habe zwei A-Einträge für zwei VMs, die ich zum Test aufgesetzt habe, gemacht. Diese haben die IP-Adressen 192.168.0.132 und 192.168.0.133.

Ein Eintrag hat folgenden Aufbau:

<NAME> IN <TYP> <WERT>

Name ist hierbei der Name, über den der Server aufgelöst werden soll, z.B. srv01.

TYP ist der Typ des Records.

WERT ist der Wert des Records, z.B. bei einem A-Record eine IP-Adresse.

Meine Einträge:

```
srv01    IN      A       192.168.0.132
srv02    IN      A       192.168.0.133
```

2.6 Überwachen sie das Logfile (syslog) beim Neustarten des Dienstes. Wurde das Zonenfile fehlerfrei geladen?

Um den DNS-Server neuzustarten, einfach systemctl restart bind9 eingeben. Danach kann man den Status, ob er läuft, mit dem Kommando systemctl status bind9 überprüfen:

```
root@ubuntuVM:/etc/bind# systemctl restart bind9

root@ubuntuVM:/etc/bind# systemctl status bind9

● named.service - BIND Domain Name Server

     Loaded: loaded (/lib/systemd/system/named.service; enabled;
vendor preset: enabled)

     Active: active (running) since Tue 2020-05-26 16:44:00 CEST; 6s
ago

       Docs: man:named(8)

   Main PID: 3120 (named)

      Tasks: 26 (limit: 9480)

     Memory: 56.6M

     CGroup: /system.slice/named.service

             └─3120 /usr/sbin/named -f -u bind


Mai 26 16:44:00 ubuntuVM named[3120]: managed-keys-zone: Key 20326
for zone . is now trusted (acce>Mai 26 16:44:00 ubuntuVM
named[3120]: resolver priming query complete
```

3 Testen

3.1 Testen Sie die Namensauflösung mit dem Befehl nslookup. Achten Sie dabei darauf, dass der Befehl ihren Dienst verwendet (server 127.0.0.1)

Mit NSLookup kann man sich Informationen zu DNS-Einträgen anzeigen lassen. Um sich Einträge von einem spezifischen DNS-Server anzeigen zu lassen.

nslookup <DOMAIN> <DNS-SERVER>

```
root@ubuntuVM:/etc/bind# nslookup srv01.niklasNetwork.lan localhost
Server:         localhost
Address:        127.0.0.1#53

Name:   srv01.niklasNetwork.lan
Address: 192.168.0.132
```

3.2 Installieren Sie das Paket dnsutils

```
root@ubuntuVM:/etc/bind# sudo apt install dnsutils -y
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen.... Fertig
Die folgenden Pakete wurden automatisch installiert und werden nicht
mehr benötigt:
  apt-clone archdetect-deb cryptsetup-bin dctrl-tools dmeventd
dmraid dpkg-repack
  gir1.2-timezonemap-1.0 gir1.2-xkl-1.0 kpartx kpartx-boot libaio1
libdebian-installer4
  libdevmapper-event1.02.1 libdmraid1.0.0.rc16 liblvm2cmd2.03
libreadline5 libtimezonemap-data
  libtimezonemap1 lvm2 python3-icu python3-pam rdate thin-
provisioning-tools
Verwenden Sie »sudo apt autoremove«, um sie zu entfernen.
Die folgenden NEUEN Pakete werden installiert:
  dnsutils
```

3.3 Testen Sie die Namensauflösung mit dem Befehl `dig (@127.0.0.1)`.

```
root@ubuntuVM:/etc/bind# dig @127.0.0.1 srv01.niklasNetwork.lan

; <<>> DiG 9.16.1-Ubuntu <<>> @127.0.0.1 srv01.niklasNetwork.lan
; (1 server found)
```

```
;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25377

;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0,
ADDITIONAL: 1


;; OPT PSEUDOSECTION:

; EDNS: version: 0, flags:; udp: 4096

; COOKIE: f43cfb35f903d186010000005ecd31fb50683d703f885765 (good)

;; QUESTION SECTION:

;srv01.niklasNetwork.lan.       IN      A


;; ANSWER SECTION:

srv01.niklasNetwork.lan. 604800 IN      A       192.168.0.132


;; Query time: 0 msec

;; SERVER: 127.0.0.1#53(127.0.0.1)

;; WHEN: Di Mai 26 17:12:59 CEST 2020

;; MSG SIZE  rcvd: 96
```

3.4 Wie unterscheidet sich dig von nslookup? Wie können Sie mit dig eine kurze Antwort erzwingen?

Nslookup gilt als veraltet und wurde früher benutzt, um sich die Namen anzuschauen, welchen IP-Adressen sie zugeordnet waren. Dagegen bietet Dig eine viel ausführlichere Antwort.

Um sich nur die IP des verwendeten Namens anzeigen zu lassen, kann man den +short <RECORD TYPE> – Parameter bei Dig verwenden. Nach +short schreibt man den Typ des Records, welchen Wert man sich anzeigen lassen möchte.

```
root@ubuntuVM:/etc/bind# dig @127.0.0.1 srv01.niklasNetwork.lan
+short A

192.168.0.132
```

3.5 Wenn sich derselbe Name (mit anderer Adresse) sowohl am Nameserver und in der Datei /etc/hosts befindet, welcher Eintrag wird dann verwendet (Testen!)? Welche Auswirkung hat das?

Wenn man in die /etc/hosts Datei einen Server mit dem selben Namen einträgt, wird immer zuerst die hosts Datei herangezogen. (Beispiel an einem Ping-Test):

```
niklas@ubuntuVM:~$ ping srv01.niklasNetwork.lan
PING srv01.niklasNetwork.lan (192.168.0.134) 56(84) Bytes Daten.
```

Dig listet nach wie vor den DNS-Eintrag mit der richtigen IP-Adresse auf:

```
niklas@ubuntuVM:~$ dig @127.0.0.1 srv01.niklasNetwork.lan +short A
192.168.0.132
```

3.6 Kann Ihr Nameserver nur Namen aus Ihrer Zone auflösen (Testen)? Wie macht er das?

Standardmäßig kann der DNS Server von allen Zonen, die geladen wurden, Namen auflösen. Wenn man aber nun mal einen Namen anpingen möchte, der nicht in einen der Zonen steht oder im Cache des DNS Servers steht, bekommt man einen Fehler. Der Server schaut sich die Dateien und geht sie durch beim Starten.

```
niklas@ubuntuVM:~$ ping google.com
ping: google.com: Temporärer Fehler bei der Namensauflösung
```

3.7 Tragen Sie einen Forwarder (z.B. 8.8.8.8) ein. Testen.

Forwarder sind nun dazu da, um einen DNS Request, den der Server selber nicht bearbeiten kann, an einen fremden DNS-Server weiterzuleiten, der sie möglicherweise im Cache hat oder sie schnell auflösen kann.

Forwarders kann man in der Datei named.options.conf eintragen:

```
forwarders {
      1.1.1.1;
};
```

Danach kann man mit Dig testen:

```
niklas@ubuntuVM:/etc/bind$ dig @127.0.0.1 google.com +short A
172.217.22.78
```

3.8 Welche Variante ist effizienter?

Die Variante mit dem Forwarder ist effizienter, da man sonst alle Namen manuell in die Zonen-Dateien eintragen müsste.