Übungsblatt 5 – Netzwerkdienste

Aufgabe A – DHCP

1. Auf dem *freeBSD* mit grafischer Oberfläche läuft bereits DHCP für das Interface *em0* im Bridge bzw. NAT-Mode. Dies nutzen wird für den ersten Teil.
   Überprüfen sie, ob der DHCP-Client auf diesem System ordnungsgemäß läuft.
2. Lassen sie sich vom DHCP-Server für das *freeBSD* eine neue IP-Adresse geben. Beobachten sie zeitgleich via Wireshark welche Nachrichten hierfür via BOOTP ausgetauscht werden.
   (a) Wie lautet die Sender-Adresse des DHCP-Client?
   (b) Warum nutzt der DHCP-Cleint seine Sender-Adresse?
   (c) An welche Ziel-IP-Adresse hat der DHCP-Client seine Nachrichten versandt?
   (d) An welche Ziel-MAC-Adresse sendet der DHCP-Client seine Nachrichten?
   (e) An welche Ziel-IP-Adresse hat der DHCP-Server seine Nachrichten versandt?
   (f) An welche Ziel-MAC-Adresse sendet der DHCP-Server seine Nachrichten?
   (g) Welche IP-Adresse wurde dem Client vom Server angeboten?
   (h) Welche Lease-Time wurde durch den DHCP-Server angeboten?
   (i) Welche IP-Adresse wählte und sendete der Client für die Antworten des DHCP-Servers?
   (j) Welcher IP-Adresse bestätigte (acknowledges) der DHCP-Server dem DHCP-Client?
3. Zeichen sie ein Sequenzdiagramm der IP-Adressvergabe die sie via Wireshark aufgezeichnet haben.

Aufgabe B – DHCP II

Im Folgenden soll ein weiteres Netzwerk unseren zwei bestehenden hinzugefügt werden. Das neueste Netzwerk soll jedoch dynamisch und automatisch IP-Adressen vergeben. Demnach haben wir zwei statische Netzwerke und ein durch DHCP organisiertes Netzwerk.

1. Vorbereitend:
   (a) Legen sie ein weiteres *Host-Only-Network* in virtualBox an. Dies soll ebenfalls in der IPv4-Range 172.16.X.Y liegen und maximal /24 sein.
   Achtung: Beim Erstellen nicht das Feld DHCP Server aktivieren, wir bauen einen eigenen DHCP-Server.

   (b) Fügen sie dem *freeBSD*-Router eine weitere Netzwerkschnittstelle für das eben angelegte Netzwerk hinzu.

   (c) Klonen sie mindestens ein *freeBSD* ohne grafische Oberfläche und geben dieser Maschine ebenfalls Zugang zum neu erstellten Netzwerk.

2. Der DHCP-Server ist bereits vorinstalliert, d.h. wir können direkt mit der Konfiguration starten. Kopieren sie zunächst die die Datei `/usr/local/etc/dhcpd.conf.example` nach */usr/local/etc/dhcpd.conf.example*

3. Passen diese Datei für ihre Bedürfnisse an, sodass dieser Rechner als DHCP-Server arbeitet.

4. Passen sie den DHCP-Client entsprechend an, sodass die VM ohne grafische Oberfläche eine IP-Adresse beziehen kann.

5. Starten sie den DHCP-Server als Daemon.

6. Testen sie, ob der DHCP-Client eine IP-Adresse bekommt und Zugang ins Internet hat. Schneiden sie den Traffic in Wireshark mit, um dies zu überprüfen.

7. Falls alles funktioniert hat, tragen sie den DHCP-Server persistent in die *rc.conf* ein!

Aufgabe C – Domain Name System (DNS) I

1. DNS-Requests:

   (a) Fragen sie mit jedem der Kommando der Hausaufgaben jeweils einmal einen Hostnamen (bspw. `www.htw-berlin.de`), einen Domainnamen (htw-berlin.de) und eine IP-Adresse (bspw. 141.45.5.100) ab.

   (b) Schauen sie sich die Ausgabe von *dig* bei der Abfrage der IP-Adresse genauer an – dort werden sie in der „Question Section“ sehen, dass nach dem A-Resource-Record mit dem Namen 141.45.5.100 gefragt wurde. Wenn Sie den Namen zu dieser IP-Adresse suchen – welchen Resource-Record müssen sie anstelle des A-Records erfragen?

   (c) In welcher Form müssen sie dann die IP-Adresse angeben? (Test mit `dig -t <record-type> <richtiges-format-ip-adresse>`).

   (d) Denken sie sich einen Domainnamen aus, den es wahrscheinlich geben könnte, welcher aber in den letzten Stunden nicht aufgelöst worden ist. Erfragen sie diesen Namen zweimal kurz hintereinander via *dig* und vergleichen sie die beiden Ausgaben. Worin unterscheiden sich beide Einträge? Falls eine größere zeitliche Differenz vorhanden ist, worin liegt die Ursache?

   (e) Erfragen sie mit *dig* und *nslookup* den zuständigen Mail-Server für die Domain `htw-berlin.de`.

(f) Erzwingen sie mit *dig* und *nslookup* eine Namensauflösung ohne den Standard-DNS-Server des Betriebssystems, sondern mit einem öffentlichen Nameserver (bspw.: 9.9.9.9) erfolgt. Testen sie dies am Besten zuerst mit *dig*, da dieses Werkzeug immer den genutzten Namensserver angibt.

2. DNS-Resolver: Das Listing zeigt die „resolv.conf“ eines Servers.

```
nameserver 141.45.3.100
search f4.htw-berlin.de
```

Was bedeuten die Einträge mit den Schlüsselwörtern: „nameserver“ und „search“?

Aufgabe D – Domain Name System (DNS) II

Im Folgenden soll ein DNS-Server für eine eigene Domain aufgesetzt werden.

1. Sichern sie zunächst ihre VM als Snapshot an, falls etwas schiefgehen sollte!
2. Im Moodle-Kurs sind einige Beispieldateien hinterlegt, wie eine Zone konfiguriert werden kann, hinterlegt.
3. Mit dem Kommando `rndc-confgen -a` können sie einen Schlüssel für die Administration der *rndc (Remote Name Daemon Control)* Werkzeuge hinterlegen.
4. Da dieser Schlüssel für den Nutzer *bind* und nicht *root* laufen soll, muss noch die Zugehörigkeit im Dateisystem angepasst werden:

```
sudo chown root:bind /usr/local/etc/namedb/rndc.key
sudo chmod 640 /usr/local/etc/namedb/rndc.key
```

5. Der Bind-Server ist bereits vorinstalliert. Mit der Zeile `named_enable="YES"` in der `rc.conf` kann dieser aktiviert werden. Falls der Dienst direkt gestartet werden soll kann dies mit `service named onestart` geschehen.
6. Mithilfe von *tail* können sie schauen, ob *rndc* korrekt läuft:

```
tail /var/log/messages
Oct 22 15:11:11 ns1 named[1161]: ------------------------------------
Oct 22 15:11:11 ns1 named[1161]: BIND 9 is maintained by Internet Systems Consortium,
Oct 22 15:11:11 ns1 named[1161]: Inc. (ISC), a non-profit 501(c)(3) public-benefit
Oct 22 15:11:11 ns1 named[1161]: corporation. Support and training for BIND 9 are
Oct 22 15:11:11 ns1 named[1161]: available at https://www.isc.org/support
Oct 22 15:11:11 ns1 named[1161]: ------------------------------------
Oct 22 15:11:11 ns1 named[1161]: command channel listening on 127.0.0.1#953
Oct 22 15:11:11 ns1 named[1161]: command channel listening on ::1#953
Oct 22 15:11:11 ns1 named[1161]: all zones loaded
Oct 22 15:11:11 ns1 named[1161]: running
```

7. Zentrale Stell für die Konfiguration ist die Datei `/usr/local/etc/namedb/named.conf`. Diese Datei ist bereits vorhanden. Der Bind-Server läuft zunächst nur lokal, daher muss der Eintrag `listen-on` angepasst werden, sodass auch ihre IP-Adressen hinterlegt sind.

8. Der oben erzeugte Schlüssel `rndc.key` muss noch in *named.conf* eingetragen werden, da wir sonst den Binder-Server nicht administrieren können. Mit Folgender Modifikation kann der Schlüssel eingetragen werden:

```
include "/usr/local/etc/namedb/rndc.key";

controls {
        inet 127.0.0.1 allow { localhost; } keys { "rndc-key"; };
};
```

   Falls der DNS-Server von außen administriert werden soll, können an dieser Stelle auch andere IPs hinterlegt werden.

9. Der DNS-Server ist nun vorbereitet! Jetzt müssen die Zonen angelegt werden. Im Moodle-Kurs habe ich eine kleinere Beispielkonfiguration hinterlegt.
   Bevor sie die Einträge ihrer `named.conf` anpassen, machen sie sich die Bedeutung Folgender Einträge klar:

   a)

```
options {
        directory "/usr/local/etc/named/working";
        forwarders { 62.104.191.241; 62.104.196.134; };
        listen-on port 53 { 127.0.0.1; 172.16.0.1; };
        allow-query { 127.0/16; };
        cleaning-interval 120;
        notify no;
};
```

   b)

```
zone "localhost" in {
        type master;
        file "localhost.zone";
};
```

   c)

```
zone "0.0.127.in-addr.arpa" in {
        type master;
        file "127.0.0.zone";
};
```

d)

```
zone "crypto.all." in {
        type master;
        file "crypto.zone";
};
```

e)

```
zone "0.16.172.in−addr.arpa" in {
        type master;
        file "172.16.0.zone";
};
```

f)

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

10. Eine *zone*-Datei ist wie folgt aufgebaut:

```
$TTL 2D
@  IN SOA @ root (
                                    42  ; serial (d. adams)
                                    1D  ; refresh
                                    2H  ; retry
                                    1W  ; expiry
                                    2D )  ; minimum

                IN NS  @
                IN A  127.0.0.1
```

```
$TTL 2D
crypto.all. IN SOA  mceliece root.localhost. (
                                    2001091300 ; serial
                                    1D  ; refresh
                                    2H  ; retry
                                    1W  ; expiry
                                    2D )  ; minimum

                IN NS  diffie
                IN MX  10 hellman

diffie  IN A  172.16.0.1
```

```
hellman IN A  172.16.0.24
peikerts  IN A  172.16.0.23
bernstein  IN A  172.16.0.25

www  IN CNAME mceliece
ftp  IN CNAME www
```

```
$TTL 2D
0.16.172.in-addr.arpa. IN SOA mceliece.crypto.all. root.localhost. (
                                        2001091300 ; serial
                                        1D  ; refresh
                                        2H  ; retry
                                        1W  ; expiry
                                        2D )  ; minimum

                    IN NS  mceliece.cyrpto.all.

11  IN PTR  diffie.crypto.all.
24  IN PTR  hellman.crypto.all.
23  IN PTR  peikerts.crypto.all.
25  IN PTR  bernstein.crypto.all.
```

Wie sind diese Dateien zu interpretieren?

11. Anhand des obigen Beispiels können sie sich eine eigene Zone ausdenken, oder mein Beispiel anpassen. Das heißt die Zonen des Beispiel für die `named.conf` müssen angepasst oder ähnlich erstellt werden.

12. Tragen sie die nun erstellten Zonen in die `named.conf` Datei ein!

13. Legen sie die Dateien entsprechend ihrer Konfiguration im richtigen Ordner ab (bspw. unter `/usr/local/etc/named/working`).

14. Überprüfen sie den Status des DNS-Servers mit dem Kommando `rndc status`. Gibt es Fehlermeldungen?

15. Mit dem Kommando `rndc reload` können sie den Server die neuen Konfigurationen geben, sodass dieser den Server neu lädt.

16. Tragen sie ihren DNS-Server in der */etc/resolv.conf* als weiteren Name-Server ein.

17. Ein Rechner eines anderen Netzes sollte nun VMs ihrer Zone via namen auflösen können. D.h. statt:

```
ping -c 1 172.16.0.25
```

könnten sie nun:

```
ping -c 1 bernstein.cyrpto.all
```

erreichen.