Fachhochschule Dortmund

Projektarbeit

Netzwerkemulation zur Untersuchung von DNS

eingereicht von: Ekra, Ano Jean-Michel

Matrikelnummer: 7096374

Studiengang: Bachelor Digitale Technologien

Betreuer: Prof. Dr. Ing. Niemeyer

Inhaltsverzeichnis

1 Einleitung

1.1 Motivation

Das Domain Name System (DNS) ist ein wesentlicher Bestandteil der Internetinfrastruktur. Mit Ausnahme einiger spezifischer Anwendungen (z. B. Peer-to-Peer-Anwendungen usw.) basieren die meisten Internetdienste derzeit auf dem Arbeitsmodell, bei welchem vor den Kommunikationsaktivitäten einige DNS-Abfragen erfolgen. Selbst wenn Ihre Webserver in gutem Zustand sind, die Firewalls ihre Arbeit tun und Ihre Backend-Anwendungsserver und Datenbanken in perfekter Ordnung sind, spielt das alles keine Rolle. Alles kann fehlschlagen, wenn Ihre DNS-Server nicht wie erwartet ordnungsgemäß funktionieren. Technisch gesehen ist das DNS eine besondere Art von verteiltem Verzeichnisdienst, den die Menschen nutzen, um andere Netzinformationssysteme zu erstellen (für den Zugriff) und um auf sie zuzugreifen. Im Grunde ist das DNS eine verteilte Datenbank, die (1)eine lokale Kontrolle ihrer Segmente ermöglicht und (2) die Daten in jedem Segment über ein Client-Server-Schema im gesamten Netz verfügbar macht. Robustheit und angemessene Leistung werden erreicht durch Replikation und Caching (3). Diese verteilte Kontrolle und Konfiguration ist jedoch ein zweischneidiges Schwert; Sie ermöglicht die Skalierung des Systems auf die Größe des Internets, aber auch unglaubliche Fehlkonfigurationen. Das Phänomen ist besonders offensichtlich bei unerfahrenen Administratoren, die ein kleines Netz verwalten. Dies ist der Hauptgrund dafür, dass DNS, obwohl es für den Netzbetrieb so wichtig ist, bei über 70% der DNS-Server Konfigurationsfehler enthalten sind.

Auch in der Vergangenheit gab es immer wieder größere IT-Sicherheitsvorfälle, die die DNS-Infrastruktur bedroht oder für Angriffe ausgenutzt haben. Einer der größten Cyberattacken der letzten Jahre wurde auf eine brasilianische Bank mittels DNS durchgeführt. Die Bank, die nicht näher genannt wurde, besitzt 5 Millionen Kunden und über 500 Filialen. Im Oktober 2016 haben Angreifer die DNS-Einträge der Bank geändert und somit die Kontrolle über die gesamte Infrastruktur der Bank übernommen. Anstelle der legitimen Webseite wurden Kunden auf gefälschte Webseiten umgeleitet und Passwörter gestohlen. Zusätzlich wurden sie beim Besuch mit Malware infiziert. Laut Kaspersky wurden auch Transaktionen über

Bankautomaten auf die Server der Angreifer umgeleitet(4). Da die interne Infrastruktur genauso betroffen war, konnten keine E-Mails verschickt werden, um die Kunden über den Angriff zu informieren. Erst nach sechs Stunden konnte die DNS-Infrastruktur wieder unter die Kontrolle der Bank gebracht werden. Dieser Angriff hat gezeigt, dass eine Übernahme des DNS gesamte Unternehmen für mehrere Stunden handlungsunfähig machen kann. Aufgrund der permanent fortschreitenden Vernetzung von Systemen durch das Internet der Dinge und neuen Architekturmustern wie Micro Services wird DNS und DNS-Sicherheit sogar noch wichtiger(5).

1993 wurden die ersten Sicherheitsbedenken im Umgang mit DNS geäußert. Daher wurde 1999 eigens der Standard Domain Name System Security Extension (DNSSEC) eingeführt, um DNS mittels Signierung der DNS-Daten sicherer zu machen. DNSSEC konnte sich allerdings bis heute nicht flächendeckend durchsetzen. Durch neue Technologien wie DNS-over-TLS und DNS-over-HTTPS, die im März 2016 und Oktober 2018 definiert wurden und von etablierten Unternehmen wie Google oder Mozilla vorangetrieben werden, wurde die Diskussion um DNS-Sicherheit und -Privatsphäre neu entfacht(6).

1.2 Ziel der Arbeit

Ziel der Arbeit ist es, DNS grundlegend aufzuarbeiten und in einer aufzubauenden Emulationsumgebung zu demonstrieren. Damit wird die Grundlage geschaffen, verschiedene Absicherungsverfahren wie DNSSEC, DNS over TLS, DNS over HTTPS, DANE zu implementieren und zu untersuchen.

Zu diesem Zweck sind folgende Arbeitsschritte notwendig:

1. Basierend auf der Literaturrecherche wird das DNS hinsichtlich seiner Funktionsweise und Komponenten beschrieben.

2. Auf Grundlage der Verfahrensbeschreibungen wird ein gemeinsamer Testaufbau und die darauf zu realisierenden Netzwerkszenarien für die späteren Untersuchungen definiert. Als DNS-Software wird Bind9 verwendet. Die Konfigurationen der einzelnen Bind9-Instanzen im Testaufbau werden entwickelt.

3. Basierend auf Docker Container wird der Testaufbau realisiert, der die Emulation der zur Demonstration notwendige Netzelemente und deren Vernetzung ermöglicht.

Die Emulationsumgebung soll so gestaltet werden, dass sie leicht verändert und auch auf andere Untersuchungsgegenstände ausgerichtet werden kann(7).

2 Überprüfung der DNS-Konzepte

2.1 Was ist Domain Name Server

Das Domain Name System (DNS) ist eine verteilte Datenbank, die einen Namensraum darstellt. Der Namensraum enthält alle Informationen, die ein Client benötigt, um einen beliebigen Namen nachzuschlagen. Jeder DNS-Server kann Abfragen zu jedem Namen innerhalb seines Namensraums beantworten. Ein DNS-Server beantwortet Anfragen auf eine der folgenden Arten:

- Wenn sich die Antwort in seinem Cache befindet, beantwortet er die Anfrage aus dem Cache.

- Wenn sich die Antwort in einer vom DNS-Server gehosteten Zone befindet, beantwortet er die Anfrage aus seiner Zone. Eine Zone ist ein Teil des DNS-Baums, der auf einem DNS-Server gespeichert ist. Wenn ein DNS-Server eine Zone hostet, ist er autorisierend für die Namen in dieser Zone (d. h. der DNS-Server kann Anfragen für jeden Namen in der Zone beantworten). Ein Server, der die Zone contoso.com hostet, kann zum Beispiel Anfragen für jeden Namen in contoso.com beantworten.

- Wenn der Server die Abfrage nicht aus seinem Cache oder seinen Zonen beantworten kann, fragt er andere Server nach der Antwort.

In Microsoft Windows Server ist es wichtig, die Kernfeatures von DNS zu verstehen, z. B. Delegierung, rekursive Namensauflösung und Active Directory-integrierte DNS-Zonen, da sie sich direkt auf den Entwurf Ihrer logischen Active Directory-Struktur auswirken(8).

2.2 DNS-Delegation

Damit ein DNS-Server Anfragen zu einem beliebigen Namen beantworten kann, muss er über einen direkten oder indirekten Pfad zu jeder Zone im Namensraum verfügen. Diese Pfade werden mit Hilfe von Delegationen erstellt. Eine Delegation ist ein Eintrag in einer übergeordneten Zone, der einen Nameserver auflistet, der für die Zone auf der nächsten Hierarchieebene autorisierend ist. Delegationen ermöglichen es Servern in einer Zone, Clients auf Server in anderen Zonen zu verweisen. Die folgende Abbildung zeigt ein Beispiel für eine Delegation.

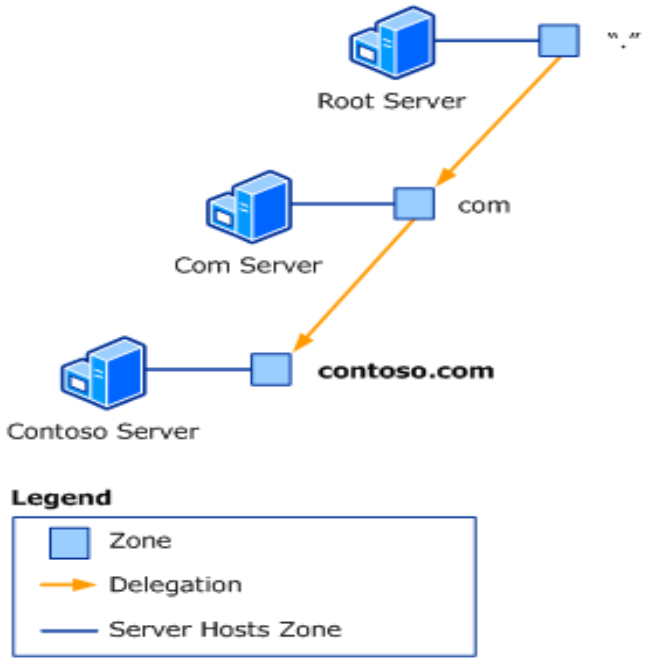


Abbildung 1.1: DNS-Delegation Prozess

Der DNS-Root-Server beherbergt die Root-Zone, die durch einen Punkt (.) dargestellt wird. Die Root-Zone enthält eine Delegation an eine Zone auf der nächsten Hierarchieebene, die com-Zone. Die Delegation in der Root-Zone teilt dem DNS-Root-Server mit, dass er den Com-Server kontaktieren muss, um die com-Zone zu finden. Ebenso teilt die Delegation in der com-Zone dem com-Server mit, dass er zum Auffinden der Zone contoso.com den Contoso-Server kontaktieren muss(9).

Hinweis:

- Bei einer Delegation werden zwei Arten von Einträgen verwendet. Der Nameserver-Ressourcendatensatz (NS) enthält den Namen eines autorisierenden Servers. Die Ressourcendatensätze Host (A) und Host (AAAA) liefern die IP-Version-4- (IPv4) und IP-Version-6- (IPv6) Adressen eines autoritativen Servers.

Dieses System von Zonen und Delegationen bildet einen hierarchischen Baum, der den DNS-Namensraum darstellt. Jede Zone stellt eine Ebene in der Hierarchie dar, und jede Delegation stellt einen Zweig des Baums dar.

Mit Hilfe der Hierarchie von Zonen und Delegationen kann ein DNS-Root-Server jeden Namen im DNS-Namensraum finden. Die Root-Zone enthält Delegationen, die direkt oder indirekt zu allen anderen Zonen in der Hierarchie führen. Jeder Server, der den DNS-Root-Server abfragen kann, kann die Informationen in den Delegationen verwenden, um jeden Namen im Namensraum zu finden.

2.3 Was ist ein Resolving Name Server

Ein Resolving Name Server ist eines der wichtigsten Dinge, die man auf jedem Computer benötigt. Er ist für die Abfrage und das Auffinden von Informationen von Root-Servern, autoritativen NS und Domänennamen zuständig.

Ein resolving Name Server kann für bestimmte Aufgaben wie die Einrichtung eines VPN, eines Einwahlsystems, eines ADSL-Netzes, eines DSL und vielem mehr hilfreich sein.

Wie kann man einen Resolving Nameserver bekommen?

Das hängt vom jeweiligen Internetdienst ab. Wenn man also einen Internetdienst hat, der von einem ISP (Internet Service Provider) bereitgestellt wird, muss man dies gezielt erfragen. Wenn das Internet von einer Organisation verwaltet wird, muss man den Netzwerkadministrator danach fragen, oder denjenigen, der dafür zuständig ist, falls ein dritter Revolver dazwischengeschaltet ist(10).

2.4 DNS-Commands (Linux-Distributionen)

Host ist ein einfaches Werkzeug, um DNS-Abfragen unter Linux durchzuführen. Es wird normalerweise verwendet, um einen Hostnamen in eine IP-Adresse aufzulösen oder umgekehrt. Wenn keine Argumente oder Optionen angegeben werden, gibt host eine kurze Zusammenfassung seiner Befehlszeilenargumente und Optionen aus:

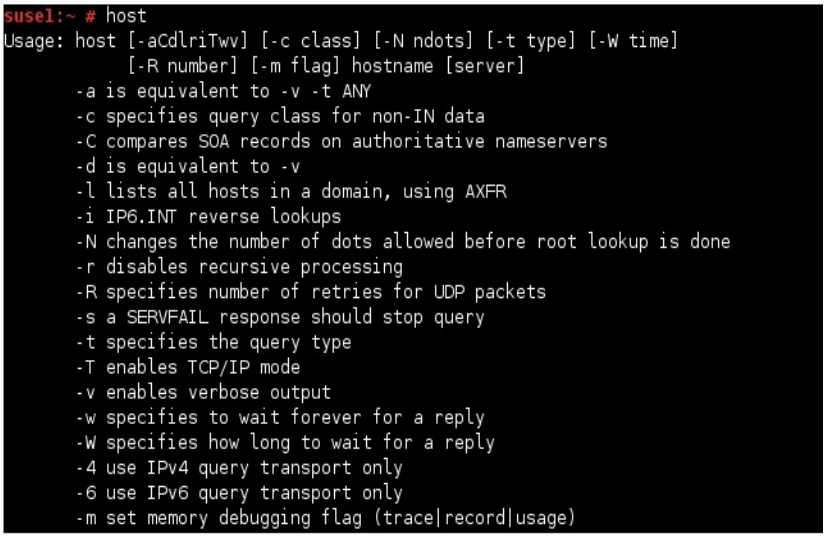


Abbildung 1.2: Host - - Version on Linux

Um die IP-Adresse von linux-bible.com herauszufinden, geben Sie host linux-bible.com ein:

Abbildung 1.3: linux-bible.com IP Address

Nslookup ist eine kleine, aber sehr leistungsfähige Befehlszeilen-Software für die Netzwerkverwaltung. Es hat eine einfache Schnittstelle, aber es ist nützlich. Der Nslookup-Befehl ist auf vielen der gängigen Computer-Betriebssysteme wie Windows, MacOS und Linux-Distributionen verfügbar. Man kann damit DNS-Abfragen durchführen und erhält: Domänennamen oder IP-Adressen oder andere spezifische DNS-Einträge.

Es gibt auch Nslookup Online-Tools. Man kann auf eine solche Website voller Online-Netzwerk-Tools zugreifen und nach der Option für nslookup suchen. Man kann eine Abfrage

für einen bestimmten DNS-Eintrag definieren, um die Domäne, den verwendeten Port und das Timeout in Sekunden zu identifizieren. Für eine bessere Sicherheit wäre zu empfehlen, die Software auf dem Computer zu verwenden(9).

- **So findet man den A-Eintrag der Domain A.:** $ nslookup Google.com

```
(base) ekra@Ekras-MacBook-Pro ~ % nslookup google.com
Server:         185.93.180.131
Address:        185.93.180.131#53

Non-authoritative answer:
Name:   google.com
Address: 216.58.214.174
```

- **um die NS-Einträge einer Domäne zu überprüfen**: $nslookup -type=ns example.com

```
(base) ekra@Ekras-MacBook-Pro ~ % nslookup -type=ns google.com
Server:         185.93.180.131
Address:        185.93.180.131#53

Non-authoritative answer:
google.com      nameserver = ns3.google.com.
google.com      nameserver = ns2.google.com.
google.com      nameserver = ns4.google.com.
google.com      nameserver = ns1.google.com.

Authoritative answers can be found from:
```

- **So fragt man den SOA-Eintrag einer Domain ab.:** $nslookup -type=soa Google.com

```
(base) ekra@Ekras-MacBook-Pro ~ % nslookup -type=ns google.com
Server:         185.93.180.131
Address:        185.93.180.131#53

Non-authoritative answer:
google.com      nameserver = ns3.google.com.
google.com      nameserver = ns2.google.com.
google.com      nameserver = ns4.google.com.
google.com      nameserver = ns1.google.com.

Authoritative answers can be found from:
```

Dig steht für (Domain Information Groper) und ist ein Befehlszeilentool für die Netzwerkadministration zur Abfrage von DNS-Nameservern (Domain Name System) Es ist nützlich für die Überprüfung und Behebung von DNS-Problemen und auch für die

Durchführung von DNS-Lookups und zeigt die Antworten an, die von dem abgefragten Nameserver zurückgegeben werden. Dig ist Teil der BIND-Domain-Name-Server-Software-Suite. Der Dig-Befehl ersetzt ältere Tools wie nslookup und host. Dig ist in den meisten Linux-Distributionen verfügbar.

```
# dig yahoo.com

; <<>> DiG 9.16.1-Ubuntu <<>> yahoo.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 20076
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;yahoo.com.                     IN      A

;; ANSWER SECTION:
yahoo.com.              387     IN      A       98.137.11.163
yahoo.com.              387     IN      A       74.6.143.26
yahoo.com.              387     IN      A       74.6.143.25
yahoo.com.              387     IN      A       74.6.231.20
yahoo.com.              387     IN      A       74.6.231.21
yahoo.com.              387     IN      A       98.137.11.164

;; Query time: 4 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Fri Dec 10 12:58:13 IST 2021
;; MSG SIZE  rcvd: 134
```

Abbildung: 1.6: Abfrage Domain "A" Recorder

Der obige Befehl veranlasst dig, den "A"-Eintrag für den Domänennamen yahoo.com zu suchen. Der Befehl dig liest die Datei /etc/resolv.conf und fragt die dort aufgeführten DNS-Server ab. Die Antwort des DNS-Servers ist das, was dig anzeigt.

Zum Verständnis der Ausgabe der Befehle:

- Zeilen, die mit; beginnen, sind Kommentare, die nicht zu den Informationen gehören.
- In der ersten Zeile steht die Version des Befehls dig (9.16.1).
- Als nächstes zeigt dig den Header der Antwort, die es vom DNS-Server erhalten hat.
- Als Nächstes kommt der Frageteil, der uns einfach die Abfrage mitteilt, die in diesem Fall eine Abfrage nach dem "A"-Eintrag von yahoo.com ist. Das IN bedeutet, dass es sich um eine Internet-Abfrage handelt (in der Klasse Internet).
- Der Antwortabschnitt sagt uns, dass yahoo.com die IP-Adresse 98.137.11.163 hat.

- Schließlich gibt es noch einige Statistiken über die Abfrage. Sie können diese Statistiken mit der Option +nostats ausschalten.

3 Lokalisierungen der DNS-Rootserver

Root-Server sind die wichtigsten DNS-Server auf der ganzen Welt. Sie sind für die Auflösung von DNS-Anfragen für Top-Level-Domains zuständig. Es gibt 13 Root-Server, die über die ganze Welt verteilt sind, hauptsächlich in den Vereinigten Staaten. Diese Server befinden sich unter der Domain: root-servers.org.

Alle Root-Server verwenden BIND (Berkeley Internet Name Domain) als DNS-Server, mit Ausnahme der Server H, L und K, die NSD (Name Server Daemon) verwenden. Verteilte Root-Server verwenden Anycast, um die Last zu verbessern und auszugleichen, was einen dezentralisierten Dienst ermöglicht(11).

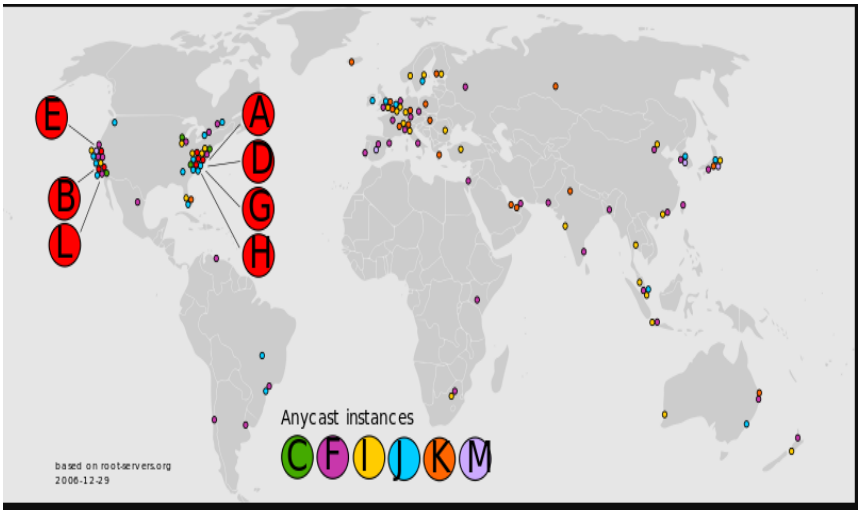


Abbildung: 1.7: Weltweit Root Server

4 Unterschied zwischen Rekursiven und Iterativen DNS-Abfragen

Zwei Begriffe werden oft im Zusammenhang mit DNS (Domain Name System) -Abfragen genannt: Rekursion und Iteration. **Rekursion** im DNS (Domain Name System) ist der Prozess eines DNS-Servers, der andere DNS-Server im Namen des ursprünglichen DNS-Clients abfragt.

Iteration ist der Prozess, bei dem ein DNS-Client wiederholt DNS-Abfragen (Domain Name System) an verschiedene DNS-Server zur Namensauflösung stellt.

Rekursive DNS-Abfrage: Bei einer rekursiven DNS-Abfrage sendet der DNS-Client eine Abfrage an einen DNS-Server zur Namensauflösung. Die Antwort auf die DNS-Abfrage kann eine Antwort auf die Abfrage oder eine Fehlermeldung sein. Wenn der DNS-Server bei einer rekursiven DNS-Abfrage die Antwort nicht kennt, um dem DNS-Client eine genaue Antwort zu geben, kann der DNS-Server im Namen des DNS-Client andere DNS-Server abfragen.

Iterative DNS-Abfrage: Bei der iterativen DNS-Abfrage fragt ein DNS-Client den DNS-Server nach der Namensauflösung und der DNS-Server gibt die beste Antwort, die er hat. Wenn der DNS-Server die Antwort auf die DNS-Abfrage des Clients nicht kennt, kann die Antwort auch ein Verweis auf einen anderen DNS-Server einer niedrigeren Ebene sein. Dieser untergeordnete DNS-Server wird vom übergeordneten DNS-Server als autorisierend für den DNS-Namensraum, auf den sich die DNS-Abfrage bezieht, delegiert. Sobald der DNS-Client die Referenz vom übergeordneten DNS-Server erhalten hat, kann er eine DNS-Anfrage an den untergeordneten DNS-Server senden, den er als Referenz erhalten hat.

Um das Konzept klar zu verstehen, kann man das folgende Beispiel betrachten.

Ich sitze an meinem Schreibtisch und möchte die Website www.omnisecu.com öffnen, um etwas über Netzwerke zu lernen. Ich gebe die URL in meinen Browser ein und drücke "Enter".

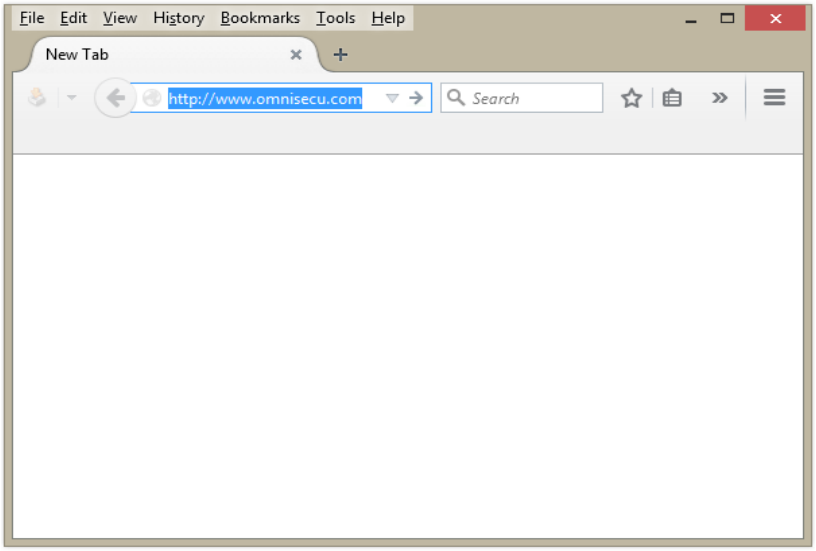


Abbildung: 1.8: Website www.omnisecu.com

Schritt 1: Der DNS-Revolver-Dienst, der im DNS-Client läuft, kontaktiert den lokalen DNS-Server (Rekursive DNS-Server) mit einer rekursiven DNS-Abfrage, um den Fully Qualified Domain Name (FQDN) aufzulösen, www.omnisecu.com. Der lokale DNS-Server muss nun die Abfrage mit der IP-Adresse beantworten, die dem Fully Qualified Domain Name (FQDN) entspricht www.omnisecu.com. Wenn der lokale DNS-Server der autorisierende DNS-Server für den DNS-Namensraum omnisecu.com ist, prüft er die zugehörige Zone auf Ressourcendatensätze, die dem Fully Qualified Domain Name (FQDN) www.omnisecu.com entsprechen, und beantwortet die DNS-Abfrage.

Wenn der lokale DNS-Server nicht der autorisierende DNS-Server für den DNS-Namensraum omnisecu.com ist, überprüft der lokale DNS-Server seinen Cache-Speicher, um eine Antwort auf die DNS-Abfrage zu liefern. Der Cache-Speicher enthält die letzten vom DNS-Server durchgeführten Namensauflösungen.

Schritt 2: Wenn der lokale DNS-Server (rekursiver DNS-Server) keine relevanten Informationen im Zonen- oder Cache-Speicher finden kann, kontaktiert er einen DNS-Server auf der höchsten Ebene (d. h. einen Root Level DNS-Server) mit einer iterativen DNS-Abfrage für www.omnisecu.com.

Schritt 3: Die Root-Level-DNS-Server sind nicht die autoritativen DNS-Server für den DNS-Namensraum omnisecu.com, aber eine Delegation an die autoritativen DNS-Server für die TLD .com ist auf den Root-Level-DNS-Servern konfiguriert. Der Root Level DNS-Server antwortet also dem lokalen DNS-Server mit einem Verweis auf einen autoritativen DNS-Server für den TLD .com Level DNS-Namespace. Der lokale DNS-Server (rekursiver DNS-Server) kontaktiert jetzt den TLD .com Level DNS-Server mit einer iterativen DNS-Abfrage für www.omnisecu.com.

Schritt 4: Die DNS-Server der TLD .com.-Ebene sind nicht die autorisierenden Nameserver für den Namensraum omnisecu.com. Es wird jedoch eine Delegation an die autorisierenden DNS-Server für den Namensraum omnisecu.com auf der TLD .com. Ebene DNS-Servern. Die DNS-Server der TLD .com.-Ebene antworten dem lokalen DNS-Server mit einem Verweis auf einen DNS-Server, der für den DNS-Namensraum omnisecu.com autorisierend ist.

Der lokale DNS-Server kontaktiert nun den autoritativen DNS-Server der Domäne omnisecu.com mit einer iterativen DNS-Abfrage für den FQDN www.omnisecu.com. Die Zone des DNS-Servers omnisecu.com ist mit Ressource Records für alle Server und Workstations innerhalb der Domäne omnisecu.com konfiguriert. Omnisecu.com DNS-Server antwortet nun dem lokalen DNS-Server mit der IP-Adresse, die dem FQDN www.omnisecu.com zugeordnet ist. Schließlich wird diese Information als endgültige DNS-Antwort an den DNS-Client weitergegeben.

Der TCP/IP-Protokollstapel auf dem DNS-Client kann nun die Kommunikation mit www.omnisecu.com unter Verwendung seiner IP-Adresse aufnehmen. Wie Sie in der folgenden Abbildung sehen können, ist die DNS-Abfrage 1 eine rekursive Abfrage, und 8 ist die Antwort darauf. Die DNS-Abfragen 2, 4 und 6 sind iterative DNS-Abfragen und 3, 5 und 7 sind die entsprechenden Antworten(12).

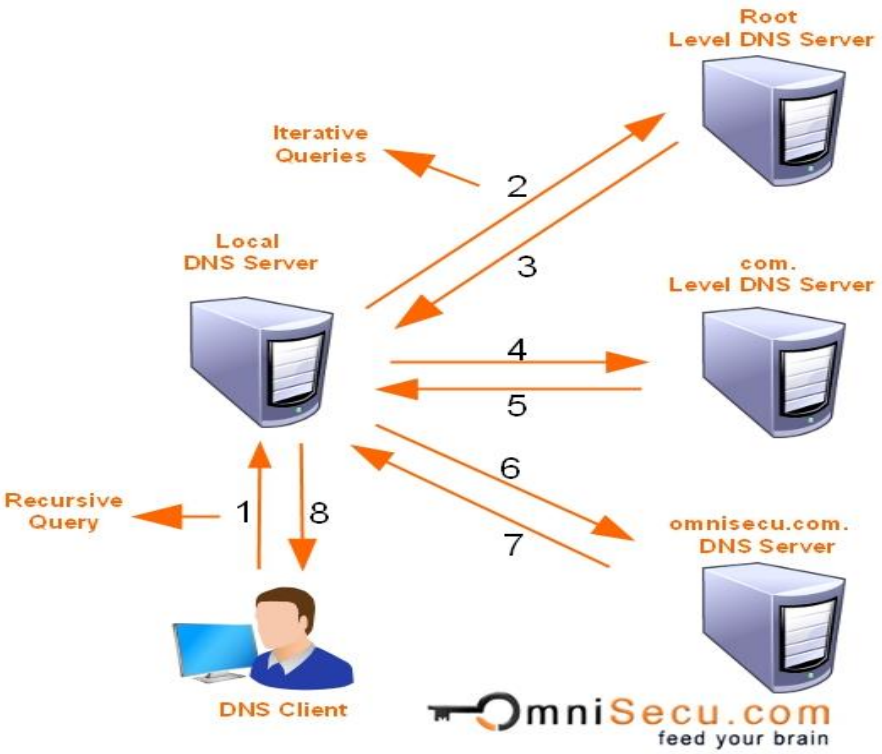
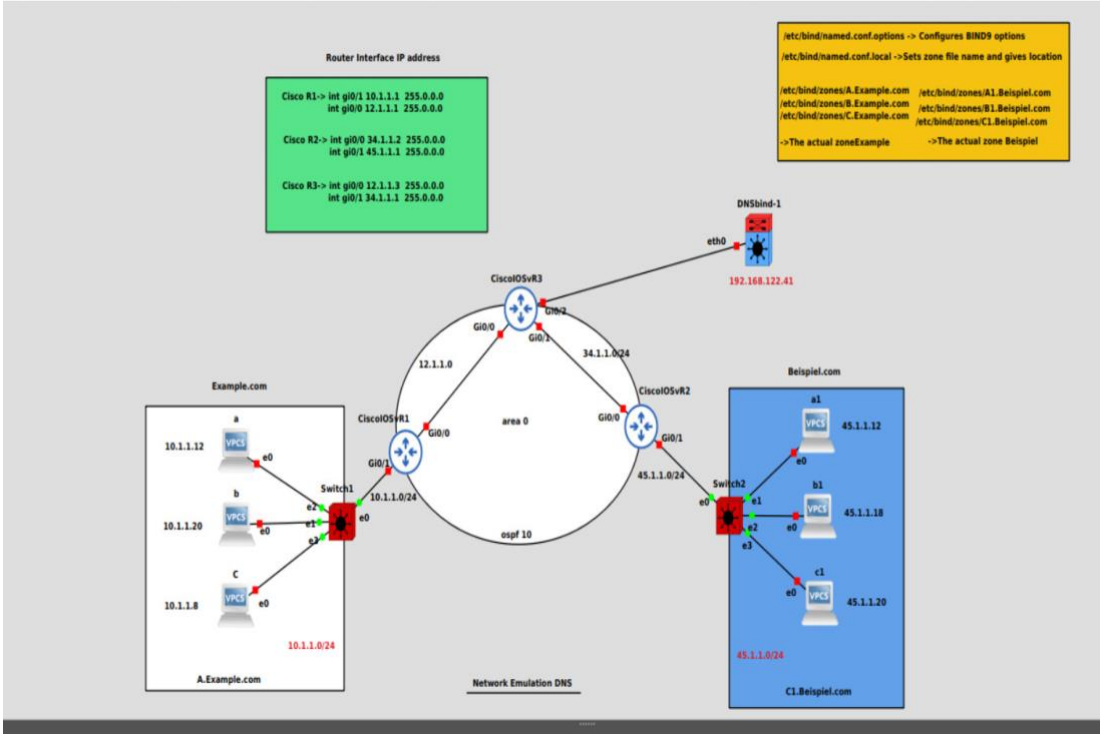


Abbildung: 1.9: Weltweit Root Server

5 Testaufbau: Verfahrensbeschreibung DNS Query mit GNS3

5.1 Topologie



5.2 Erstellen BIND9 Docker Container in GN3

Die Organisation, die das Open-Source-Projekt Bind überwacht, veröffentlicht auch ein offizielles Docker-Image über den Docker-Hub, auf das jeder zugreifen kann. Für System- und Netzwerkingenieure kann die Docker-Container-Technologie anfangs etwas schwierig zu verstehen sein. Bei Docker-Containern handelt es sich nicht um ein komplettes Betriebssystem - vollständige Betriebssysteme sind für die gleichzeitige Ausführung vieler Prozesse ausgelegt. Docker-Container sind für die Ausführung eines einzigen Prozesses konzipiert. Sie enthalten nur die Software und die Bibliotheken, die zur Ausführung dieses Prozesses benötigt werden.

GNS3 hat jedoch eine sehr gute Integration mit Docker. Sie ermöglicht es Ihnen, Ihren Containern vollständige Netzwerkadapter hinzuzufügen und kopiert einige praktische Tools, um die Befehlszeilenumgebung nutzbar zu machen. Da jedoch viele der vertrauten

Betriebssystem-Tools in den meisten Docker-Containern nicht enthalten sind, wie es bei einem Standard-Betriebssystem der Fall wäre, kann es eine Herausforderung sein, die Dinge richtig zum Laufen zu bringen.

Hier wäre eine mögliche Anleitung dazu:

- Erstellen Sie Ihr eigenes Image auf der Grundlage des offiziellen ISC-Bind-Images

Öffnen Sie zunächst eine Shell oder ein Terminal auf der GNS3-VM oder wo auch immer sich der GNS3-Server befindet. Wenn Sie nicht wissen, wie man eine Shell öffnet, finden Sie in den offiziellen GNS3-Dokumenten eine Anleitung dazu:

https://docs.gns3.com/docs/emulators/create-a-docker-container-for-gns3/

- Erstellen Sie ein Verzeichnis, in das Sie Ihre Docker Datei schreiben und das Image erstellen können:

```
mkdir DNSbind-1
cd DNSbind-1

vi Dockerfile
```

Sie können einen beliebigen Texteditor verwenden. Ich nutze am meisten Vi als Text Editor. Wir werden ein Dockerfile schreiben, das so aussieht:

```
FROM internetsystemsconsortium/bind9:9.11
RUN apt-get update
RUN apt-get install vim -y
```

Im Grunde ist alles, was dies tut, das offizielle Bind-Docker-Image zu ziehen und einige Befehle auf dem Image auszuführen. Namentlich aktualisieren wir apt-get und installieren vi. Wir müssen dies tun, weil dieses Docker-Image keinen Texteditor installiert hat und wir die Bind-Konfigurationsdateien bearbeiten müssen. Um es gleich vorweg zu nehmen: Es gibt einen anderen, viel besseren Weg als die manuelle Bearbeitung der Konfigurationsdateien innerhalb des Containers. Sie können die Konfigurationsdateien in denselben Ordner wie die Docker Datei schreiben und sie dem Docker-Image hinzufügen, wenn Sie es erstellen. Ich

denke jedoch, dass es zum Lernen und zur Fehlersuche am besten ist, die Textdateien manuell zu bearbeiten.

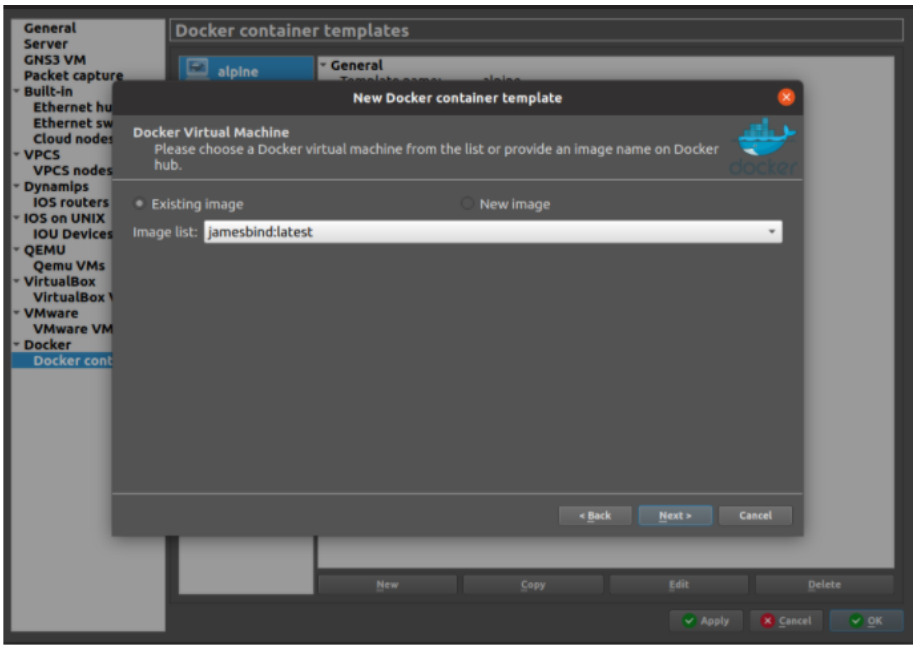
Erstellen Sie Ihr Image (mit dem Schalter -t erhalten Sie einen "Tag", der im Grunde ein Name ist):

docker build -t DNSbind-1.

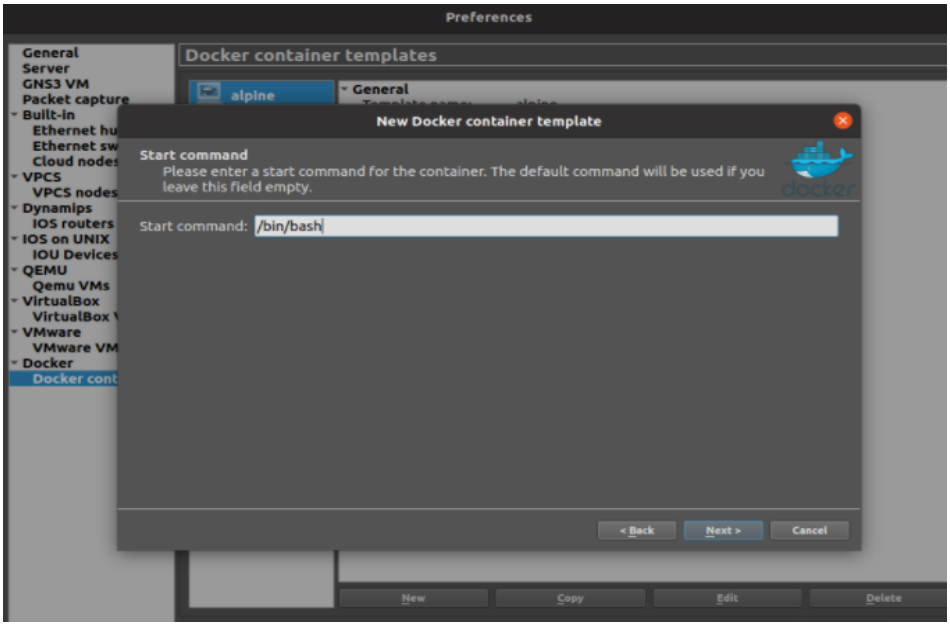
Vergessen Sie nicht den Punkt am Ende, der ist wichtig. Sie sollten nun ein frisches Docker-Image haben, in dem bind und vi installiert sind.

- **Hinzufügen Docker-Image in GNS3**

Im GNS3-Einstellungsfenster können Sie nun Ihr Bild zur Liste der verfügbaren Geräte hinzufügen.



Klicken Sie sich durch und verwenden Sie die Standardeinstellungen, außer wenn Sie zum Fenster "Startbefehl" gelangen. Hier müssen Sie /bin/bash einstellen:



Jetzt können Sie Ihr Bild in GNS3 verwenden!

5.3 Konfiguration Bind9 in Docker Container

Die Installation von BIND9 ist mit dem Paketmanager von Ubuntu recht einfach: **apt-get install bind9**

Es gibt drei Textdateien, die benötigt werden, um eine grundlegende BIND9-Konfiguration zum Laufen zu bringen, diese sind:

- /etc/bind/named.conf.options -> konfiguriert die BIND9-Optionen
- /etc/bind/named.conf.local -> legt den Namen der Zonendatei fest und gibt ihren Speicherort an
- /etc/bind/zones/db.beispiel.com &db.examples.com-> Die eigentliche Zonendatei mit DNS-Einträgen.

Als Erstes müssen Sie **/etc/bind/named.conf.options** bearbeiten, um eine sehr, sehr grundlegende Konfiguration zu erhalten:

```
root@DNSbind-1:/etc/bind#
root@DNSbind-1:/etc/bind# cat named.conf.options


options {
        directory "/var/cache/bind";
        listen-on { any; };

        allow-query{any;};

};
```

Hier ist es wichtig die Funktion **allow-query {any};** eingeben, um die Verbindung und die Kommunikation zwischen bind/named.conf.options und bind/named.conf.local zu ermöglichen.

Zweitens fügen Sie eine Zonen Konfiguration zu **named.conf. local** hinzu, die angibt, wo die Zonendatei gespeichert wird:

```
root@DNSbind-1:/etc/bind# cat named.conf.local
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";



zone "beispiel.com" {
    type master;
    file "/etc/bind/zones/db.beispiel.com";
};
zone "example.com" {
    type master;
    file "/etc/bind/zones/db.example.com";
};
```

Als Letztes erstellen wir "db.beispiel.com" & “db.beispiel.com" "db.example.com" im Verzeichnis "zones", denn das ist das, was ich in **named.conf.local** eingetragen habe. Natürlich werden Sie "jamesmcclay.com" wahrscheinlich durch etwas anderes ersetzen wollen:

```
@               IN      SOA     ns.beispiel.com.    root.beipiel.com. (
                                2               ; Serial
                                604800          ; Refresh
                                86400           ; Retry
                                2419200         ; Expire
                                604800 )        ; Negative Cache TTL
;
@               IN      NS      ns.beispiel.com.
ns              IN      A       192.168.122.41
a1              IN      A       45.1.1.12
b1              IN      A       45.1.1.18
c1              IN      A       45.1.1.20
```

```
@               IN      SOA     ns.example.com.    root.example.com. (
                                2               ; Serial
                                604800          ; Refresh
                                86400           ; Retry
                                2419200         ; Expire
                                604800 )        ; Negative Cache TTL
;
@               IN      NS      ns.example.com.
ns              IN      A       192.168.122.41
a               IN      A       10.1.1.12
b               IN      A       10.1.1.20
c               IN      A       10.1.1.8
```

Die Werte im oberen Bereich sind meist nur Standardwerte für DNS, während die A-Einträge im unteren Bereich wichtiger sind. Dies sind die IP-Adressen, die für Hostnamen innerhalb von "beispiel.com oder example.com" zurückgegeben werden. Es muss sichergestellt werden, dass diese korrekt sind. Sie stimmen alle mit den in der Topologie aufgeführten IPs überein.

Starten Sie BIND9 neu:

systemctl restart bind9

Die Anzeige sollte grün sein, wenn Ihre Konfiguration in Ordnung ist:

```
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/etc/systemd/system/bind9.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2020-04-19 03:42:16 UTC; 1s ago
     Docs: man:named(8)
  Process: 15141 ExecStop=/usr/sbin/rndc stop (code=exited, status=0/SUCCESS)
 Main PID: 15145 (named)
    Tasks: 4 (limit: 507)
   CGroup: /system.slice/bind9.service
           └─15145 /usr/sbin/named -f -u bind
```

Schließlich kann man Bind mit dem Befehl "named -g". Dadurch wird es im Vordergrund ausgeführt, mit Debug-Ausgabe, was sehr praktisch ist. Alternativ kann man auch einfach "named" ausführen und es wird im Hintergrund laufen. Wenn Sie es ausführen, suchen Sie nach einer Zeile, die besagt, dass Ihre Zonendatei geladen wurde. "all zones loaded" scheint eine Lüge zu sein, denn wenn es Fehler in Ihrer Zone gibt, wird es das sagen und dann behaupten, dass alle Zonen geladen wurden. Lesen Sie die Ausgabe sorgfältig:

```
A MINTTL instead
08-Jan-2022 17:25:33.495 zone example.com/IN: loaded serial 2
08-Jan-2022 17:25:33.496 all zones loaded
08-Jan-2022 17:25:33.496 running
```

5.4 Syntax für die Konfiguration der IP-Adresse des DNS-Servers

Man kann mit der folgenden Syntax: die IP-Adresse des DNS-Servers konfigurieren(13).

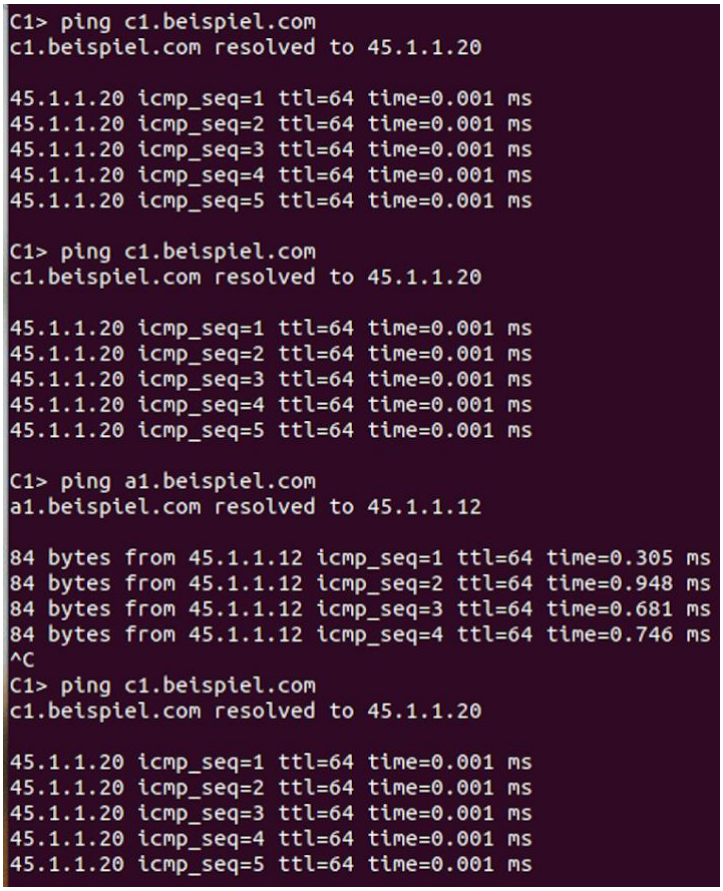
ip dns 192.168.122.41

Oder die Manuelle Konfiguration direkt auf der VPCS-Maschine in GNS3 (Edit Config):

```
set pcname a
ip 10.1.1.12 10.1.1.1 24
ip dns 192.168.122.41
```

5.5 Testen alle Einrichtung

Hier sieht man einige Ping Ergebnisse auf der VPCS-C1-Maschine aus:



5.6 OSPF Konfiguration auf CiscoIOSv15 Router

Open sohrtest Path First (OSPF) ist ein Link-State-Routing-Protokoll, das mit Hilfe eines eigenen SPF-Algorithmus den besten Pfad, zwischen dem Quell- und dem Ziel Router findet.

R1 configuration:

Step 1: Configuring IP address in R1

R1#

R1# configure terminal

R1(config)# interface fastEthernet 0/0

R1(config-if)# ip address 10.1.1.0 255.255.255.0

R1(config-if)# no shutdown

R1# configure terminal

R1# interface fastEthernet 0/1

R1(config-if)# ip address 12.1.1.0 255.255.255.0

R1(config-if)# no shutdown

R1(config-if)#exit

Step 2: Creating loopback interface

R1(config)#interface loopback 1

R1(config-if)# ip address 10.0.0.1 255.0.0.0

R1(config-if)# exit

R1(config)# exit

R1#

Step 3: Configuring OSPF

R1#

R1#configure terminal

R1(config)#router ospf 1

R1(config-router)#network 10.1.1.0 0 255.255.255 area 0

R1(config-router)#network 12.1.1.0 255.255.255.255 area 0

R1(config-router)#exit

R1(config)#exit

R1#

Nach erfolgreicher Konfiguration ist es auch möglich die Gesamt ospf Database zu Überprüfen. Durch den Befehl **R1#show ip ospf database** können wir die Gesamt ospf Konfiguration beobachten(14).

```
R1#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 10)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
2.2.2.2         2.2.2.2         258         0x80000002 0x00916E 1
3.3.3.3         3.3.3.3         255         0x80000003 0x000D8E 2
4.4.4.4         4.4.4.4         260         0x80000002 0x007C48 1

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
12.1.1.3        3.3.3.3         259         0x80000001 0x003ECA
34.1.1.2        4.4.4.4         260         0x80000001 0x005F88

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.0.0.0        2.2.2.2         296         0x80000001 0x00D455
45.0.0.0        4.4.4.4         300         0x80000001 0x00CF2F

                Router Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Link count
2.2.2.2         2.2.2.2         301         0x80000001 0x00AB6B 1

                Summary Net Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum
12.0.0.0        2.2.2.2         296         0x80000001 0x00BA6D
34.0.0.0        2.2.2.2         253         0x80000001 0x00A56B
```

5.7 BIND9 Troubleshooting

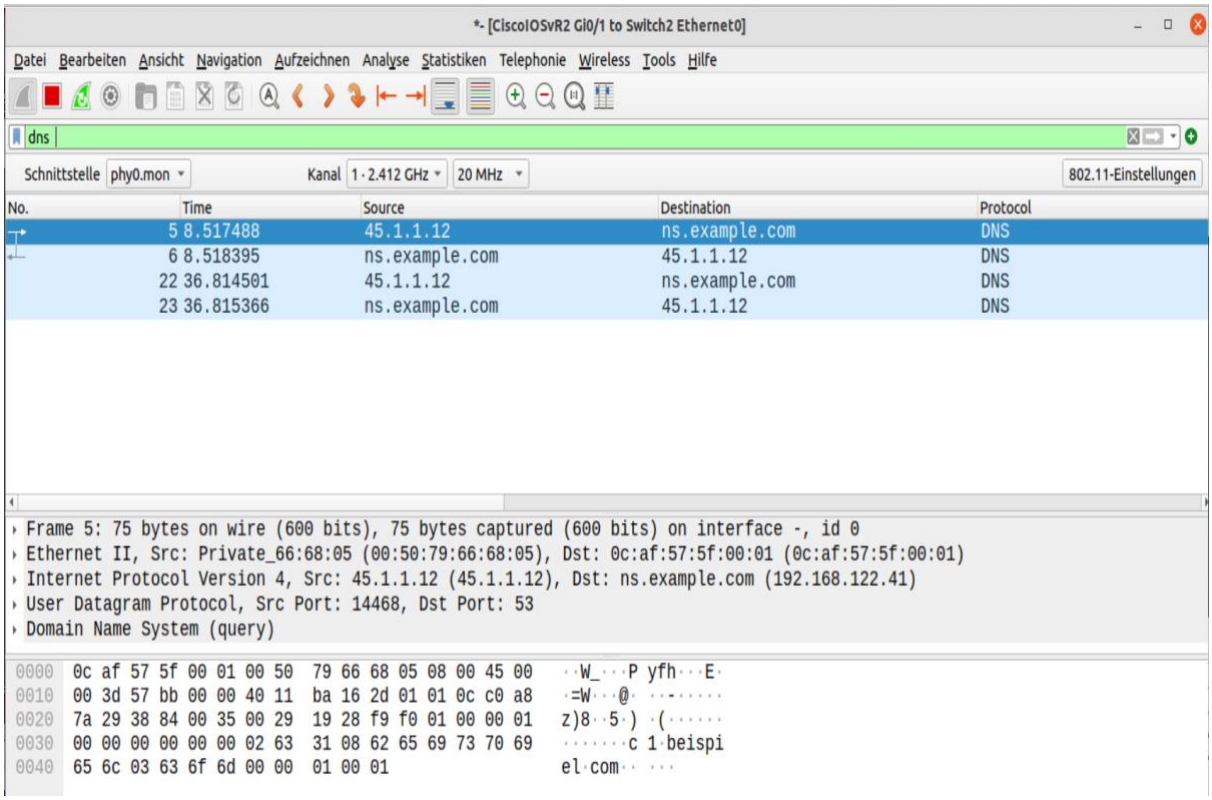
Die Bind-Konfigurationsdateien reagieren sehr empfindlich auf alles, was ausgelassen wird. Man muss überprüfen, ob man ein Semikolon vergessen hat oder ob die Zonendatei richtig formatiert sind und alle erforderlichen Einträge vorhanden sind. Hier wäre es empfehlenswert, beim Testen das **"named -g"** zu verwenden, da es I wichtige Hinweise darauf geben kann, was in Ihrer Konfiguration falsch ist.

Wenn Ihr Bind-Server ohne Konfigurationsfehler läuft und trotzdem etwas nicht funktioniert, könnte es ein Netzwerkproblem sein. Stellen Sie sicher, dass Sie eine Paketaufzeichnung durchführen, um zu sehen, ob die Pakete tatsächlich fließen und den Erwartungen entsprechen! Manchmal mache ich nach langer Fehlersuche einen Paketmitschnitt, nur um festzustellen, dass die Pakete die Netzwerkschnittstelle nie verlassen haben, weil ich etwas vergessen habe, z. B. eine IP-Adresse oder eine Route irgendwo hinzuzufügen(15).

6 DNS-Anfrage und Antwort mit Wireshark

DNS ist ein grundlegendes Protokoll des Internets und ist im Netz häufig zu sehen. Obwohl DNS-Anfragen sehr unterschiedlich sein können, da die Menschen eine Vielzahl verschiedener Websites besuchen, sehen die meisten Anfragen ziemlich "normal" aus. Die Überwachung auf unsinnige Domänennamen, bekannte schlechte Websites, ähnliche Domänen und DNS-Antworten ohne Anfragen kann für die Identifizierung von Angriffsverkehr wertvoll sein.

Hier können wir in einer Wireshark-Paketaufnahme die DNS-Anfrage von 45.0.0.12 und die Antwort von ns.example.com sehen:



7 Fazit

Ziel dieser Arbeit ist es, DNS grundlegend aufzuarbeiten und in einer aufzubauenden Emulationsumgebung zu demonstrieren. im Laufe der Arbeit wurde klar/ offensichtlich was wurde? über welche wichtige Rolle und die Grundlagen des DNS aufgezeigt. Bei der Software Konfiguration BIND9, wurde auch die Funktionsweise des DNS deutlich definiert und geklärt: Mithilfe der drei Bind-Konfigurationsdateien (/etc./bind/named.conf.options, named.conf.local, db.example oder beispiel.com) wurde praktische gezeigt, wie DNS-Zonen in Bind9 kommunizieren und wie empfindlich und manchmal unstabil Bind9 ist. Die Erklärung über den Unterschied zwischen rekursiven und Iterativen Nameserver sollte dem Nutzer eine Möglichkeit zur Entscheidung liefern. Statt den rekursiven Nameserver des Internetzugangsanbieters zu verwenden, sollten sie rekursive Nameserver von Drittanbietern, etwa Google oder OpenDNS nutzen. Der Wechsel zu einem Drittanbieter ist einerseits durch die geringe Dienstqualität (lange Wartezeiten und häufige Server-Ausfälle) der rekursiven Nameserver andererseits durch die Internetzugangsanbieter motiviert. Darüber hinaus wird auf den rekursiven Nameservern der Internetzugangsanbieter zunehmend bewusst in die Namensauflösung eingegriffen, etwa um sog. Internet-Sperren zu realisieren oder um den Nutzer bei der Eingabe eines ungültigen Domainnamens auf werbefinanzierte Suchseiten umzuleiten.

Weiterer Klärung bedürfen Fragen wie: „Welche DNS-Sicherheit und Maßnahmen gibt es, um Infrastruktur-Netzwerke gegen verschiedene Angriffsszenario zu schützen? “. Diese könnte man in der weiteren Forschung untersuchen und vertiefen.

Literaturverzeichnis

1. Background [Internet]. [zitiert 16. Januar 2022]. Verfügbar unter: https://learning.oreilly.com/library/view/dns-and-bind/0596100574/ch01.html

2. Chen C-S, Tseng S-S, Liu CL, Ou CH. Design, and implementation of an intelligent DNS configuration system. In: Proceedings of Fourth International Conference on Advanced Communication Technology, Korea. 2002. S. 230–5.

3. Chen, C.S., Tseng, S.S., Liu, C.L. A distributed intrusion detection model for the domain name system. Special issue on parallel and distributed systems. Journal of Information Science and Engineering 18, 999–1009.

4. Greenberg A. How an Unprecedented Heist Hijacked a Bank's Entire Online Operation. Wired [Internet]. [zitiert 19. Januar 2022]; Verfügbar unter: https://www.wired.com/2017/04/hackers-hijacked-banks-entire-online-operation/

5. sac-105-en.pdf [Internet]. [zitiert 17. Januar 2022]. Verfügbar unter: https://www.icann.org/en/system/files/files/sac-105-en.pdf

6. Hüsgen K. DNS-Sicherheit am Beispiel eines mittelständischen Softwareunternehmens. :150.

7. Projektarbeit 2 Netzwerkemulation zur Untersuchung von DNS [Internet]. Verfügbar unter: https://www.ilias.fh-dortmund.de/ilias/goto_ilias-fhdo_fold_1046888.html

8. iainfoulds. Reviewing DNS Concepts [Internet]. 2022 [zitiert 4. Januar 2022]. Verfügbar unter: https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/reviewing-dns-concepts

9. Recursive and Iterative DNS Queries [Internet]. [zitiert 6. Januar 2022]. Verfügbar unter: https://www.omnisecu.com/tcpip/recursive-and-iterative-dns-queries.php

10. Team Dns net. Resolving Name Server: What is it? Setup guide: Linux, Mac and Windows [Internet]. DNS Articles. 2018 [zitiert 5. Januar 2022]. Verfügbar unter: https://dnspropagation.net/articles/resolving-name-server/

11. Null S. RootServer: Los 13 servidores raíz del mundo! [Internet]. Security Null. 2013 [zitiert 6. Januar 2022]. Verfügbar unter: https://www.securitynull.net/rootserver-los-13-servidores-raiz-del-mundo/

12. iainfoulds. Reviewing DNS Concepts [Internet]. [zitiert 19. Januar 2022]. Verfügbar unter: https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/reviewing-dns-concepts

13. How to Use VPCS in GNS3 - A Step by Step Explanation [Internet]. [zitiert 20. Januar 2022]. Verfügbar unter: http://protechgurus.com/how-to-use-vpcs-in-gns3/

14. Basic OSPF Configuration [Internet]. NetworkLessons.com. 2013 [zitiert 20. Januar 2022]. Verfügbar unter: https://networklessons.com/ospf/basic-ospf-configuration

15. Question Computer [Internet]. Question Computer. [zitiert 8. Januar 2022]. Verfügbar unter: https://www.questioncomputer.com/