Fachschule für Technik

Schriftliche Abschlussprüfung

Fachrichtung Elektrotechnik
Walther-Rathenau-Gewerbeschule Freiburg

Fach:ITV	**Abschlussprüfung:**	**2020**
	Fachlehrer:	**Dienert**
Prüfungsdauer: 150 Min.	**Klasse:**	**FTI4T**

Hilfsmittel:	**4 A4-Seiten mit beliebigem Inhalt, Schriftgrösse nicht kleiner als 10pt. Schülerrechner mit Linux im Klassenarbeitsmodus.**

Hinweise

In den folgenden Aufgaben finden Sie vorbreitete Tabellen für die Lösungen, die sie handschriftlich füllen können. Alternativ können Sie die Lösungen auch am PC in eine Datei schreiben. Die Datei muss dann bei Abgabe der Prüfung ausgedruckt und der Ausdruck abgegeben werden.

Netzwerk

Abb. 1, Seite 12, zeigt eine Übersicht über das Netzwerk eines kleinen, neugegründeteten Elektronik-Unternehmens.

Auf dem Host `Ostello` laufen Dienste, die aus dem Internet und aus dem internen Firmennetz erreichbar sein müssen. Damit das interne Netzwerk maximal geschützt ist, wird `Ostello` in einer sog. *DMZ* betrieben.

Um Lizenzkosten zu sparen und maximale Freiheit bei der Konfiguration zu haben, setzt das Unternehmen auf allen Hosts (*R1, R2, Ostello*) ausschliesslich Linux als Betriebssystem ein.

Die DMZ ist mit einer zweistufigen *Firewall* realisiert. Sie besteht aus den Routern `R1, R2` und dem Switch `DMZ`. Die IP-Pakete werden auf `R1` und `R2` mit dem `netfilter`-Modul von Linux gefiltert.

Das Netzwerk in der de-militarisierten Zone hat die Netzadresse **141.31.147.112/29**.

Der Host `Ostello` in der DMZ hat die IPv4-Adresse **141.31.147.113/29**.

Auf dem Router `R1` wird der Dienst *IPv6 Router Advertisement Daemon (radvd)* betrieben.

Die Konfigurationsdatei des *radvd* hat folgenden Inhalt:

```
interface dmz0{
  AdvSendAdvert on;
  prefix 2001:7c0:c0ca:c01a::0/64{
    AdvOnLink on;
    AdvAutonomous on;
  };
};
```

Die Interfaces von `R1, R2` und `Ostello` haben folgende Layer2-Adressen:

R1, dmz0:	**dc:a6:32:0a:70:12**
R2, ether0:	**c8:60:00:c7:0b:1f**
R2, VLAN10,VLAN20:	**08:00:27:cc:be:bc**
Ostello, enp0s8:	**74:da:38:5e:36:b8**

Tabelle 1: Layer2-Adressen

Das Netz zwischen `R1` und dem ersten Router im Internet (`iNetHost`) hat eine Präfixlänge von **30**. `R1` hat die Hostadresse **129.143.15.130/30**.

1 Aufgabe: IPv4- und IPv6-Adressen

1. Versehen Sie die Schnittstellen `wan0` von `iNetHost` mit je einer IPv4-Adresse aus dem Verbindungsnetz. Geben Sie auch die Netzadresse des Verbindungsnetzes an.

   Tragen Sie alles in die Lösungstabelle 2 ein. 2 P

Name, Vorname:		
Hostname	Interface	Adresse
R1	wan0	129.143.15.130/29
iNetHost		
Netz	—	

Tabelle 2: Wide-Area-Network Anbindung

   Tragen Sie auch das letzte Oktett der Adressen in die Zeichnung ein.

2. Versehen Sie `R1`, `R2` und `Ostello` mit IPv4-Adressen aus dem angegebenen DMZ-Netz. `Ostello` soll die niedrigste, `R1` die zweithöchste und `R2` die höchste Hostadresse aus dem DMZ-Netz bekommen.

   Alle Adressen müssen in der CIDR-Notation (Netzadresse/Präfixlänge) angegeben werden.

   Geben Sie auch die Broadcast-Adresse und die Subnetzmaske an.

   Tragen Sie alles in die Lösungstabelle 3 ein. 5 P

Name, Vorname:		
Hostname	Interface	Adresse/Präfix bzw. Maske
R1		
R2		
Ostello		
BC-Adresse	-	
Maske	-	

Tabelle 3: DMZ

Tragen Sie zusätzlich die letzten Oktette der Adressen in die Zeichnung ein.

3. Auf allen Hosts in der Aufgabe ist IPv6 aktiviert. Geben Sie die **globalen** IPv6 Unicast-Adressen von `R1`, `R2` und `Ostello` mit ihrer Präfixlänge an (Lösungstabelle 4). Gehen Sie davon aus, dass der Interface-Identifier von den Hosts im **EUI-64-Format** gebildet wird (no Privacy Extension). Notieren Sie die IPv6-Adressen in möglichst kurzer Form! 6 P

   Auf R1 läuft der Dienst `radvd` mit der oben angegebenen Konfigurationsdatei.

Name, Vorname:		
Hostname	Interface	IPv6-Adresse/Präfix
R1	dmz0	
R2	ether0	
Ostello	enp0s8	

Tabelle 4: Globale IPv6 Unicast-Adressen

2 Aufabe: Routing

Geben Sie von allen Hosts die Routingtabellen an. Es darf kein Standardgateway angegeben werden, sondern alle möglichen Zielnetze müssen aufgeführt werden. D.h. für die beiden Hosts in den VLANs steht die Adresse von `iNetHost` *stellvertretend* für alle möglichen Adressen im Internet.

Die **direkt verbundenen** Netze sollen in den Routingtabellen der jeweiligen Hosts **nicht** angegeben werden.

Der Router R2 ist über einen Trunk mit einem Switch verbunden, auf dem die beiden VLANs `VLAN10` und `VLAN20` konfiguriert sind.

Das VLANs haben folgende Netzadressen:

VLAN10	5.5.10.0/24
VLAN20	5.5.20.0/24

Tabelle 5: Netze in den VLANs

In allen Routingtabellen sind diese beiden Zielnetze zu einem übergeordneten Netz zusammenzufassen. Das übergeordnete Netz soll so klein wie möglich und so gross wie nötig sein. Wenn Sie dieses sog. *Supernetting* nicht lösen können, nehmen Sie **5.5.10.0/24** als Zielnetz an. Die GW-Adressen in diesen Netzen sollen jeweils die höchsten Hostadressen sein.

Geben Sie für je einen beliebigen Host aus den VLANs die Routingtabellen an. Die Interfaces der Hosts müssen nicht angegeben werden.

Tabelle für Lösungen: 20 P

Hostname	Zielnetz/Präfix	Gateway	Interface
iNetHost			
R1			
Ostello			
R2			
Host in VLAN10			———-
			———-
Host in VLAN20			———-
			———-

Tabelle 6: IPv4 Routingtabellen

3 Aufgabe: Erreichbarkeit testen

Nachdem alle Routingtabellen erstellt sind, soll die Erreichbarkeit von `Ostello` aus dem Internet getestet werden. Dazu wird auf dem Rechner `iNetHost` das Kommando

```
traceroute 141.31.147.113
```

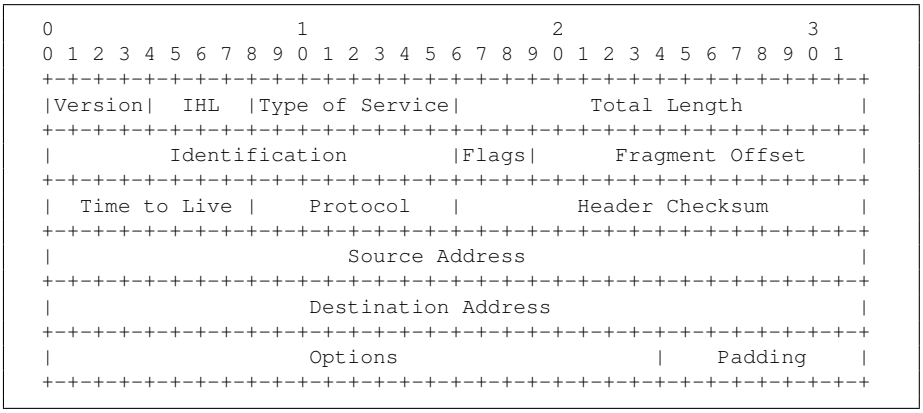
ausgeführt.

Gleichzeitig werden die gesendeten Pakete auf `iNetHost` untersucht. In der folgenden Aufzeichnung ist immer nur das erste gesende Paket und das erste Anwortpaket dargestellt. Die Wiederholungen und die Antwortpakete wurden gekürzt:

```
IP 129.143.15.129.47358 > 141.31.147.113.33434: UDP, length 32
0x0000:  0800 277b ca6f 0800 2780 4c12 0800 4500  ..'{.o..'.L...E.
0x0010:  003c 4346 0000 0111 c4ca 818f 0f81 8d1f  .<CF............
0x0020:  9371 b8fe 829a 0028 b1da 4041 4243 4445  .q.....(..@ABCDE
0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455  FGHIJKLMNOPQRSTU
0x0040:  5657 5859 5a5b 5c5d 5e5f                 VWXYZ[\]^_

IP 129.143.15.130 > 129.143.15.129: ICMP time exceeded in-transit, length 68
0x0000:  0800 2780 4c12 0800 277b ca6f 0800 45c0  ..'.L...'{.o..E.
0x0010:  0058 4183 0000 4001 1641 818f 0f82 818f  .XA...@..A......
0x0020:  0f81 0b00 a6da 0000 0000 4500 003c 4346  ..........E..<CF
0x0030:  0000 0111 c4ca 818f 0f81 8d1f 9371 b8fe  .............q..
0x0040:  829a 0028 1d5f 4041 4243 4445 4647 4849  ...(._@ABCDEFGHI
0x0050:  4a4b 4c4d 4e4f 5051 5253 5455 5657 5859  JKLMNOPQRSTUVWXY
0x0060:  5a5b 5c5d 5e5f                           Z[\]^_

IP 129.143.15.129.43196 > 141.31.147.113.33438: UDP, length 32
0x0000:  0800 277b ca6f 0800 2780 4c12 0800 4500  ..'{.o..'.L...E.
0x0010:  003c 434a 0000 0211 c3c6 818f 0f81 8d1f  .<CJ............
0x0020:  9371 a8bc 829e 0028 b1da 4041 4243 4445  .q.....(..@ABCDE
0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455  FGHIJKLMNOPQRSTU
0x0040:  5657 5859 5a5b 5c5d 5e5f                 VWXYZ[\]^_

IP 141.31.147.113 > 129.143.15.129: ICMP 141.31.147.113
udp port 33438 unreachable, length 68
0x0000:  0800 2780 4c12 0800 277b ca6f 0800 45c0  ..'.L...'{.o..E.
0x0010:  0058 c13d 0000 3f01 0807 8d1f 9371 818f  .X.=..?.....q..
0x0020:  0f81 0303 aed7 0000 0000 4500 003c 434a  ..........E..<CJ
0x0030:  0000 0111 c4c6 818f 0f81 8d1f 9371 a8bc  .............q..
0x0040:  829e 0028 2d9d 4041 4243 4445 4647 4849  ...(-.@ABCDEFGHI
0x0050:  4a4b 4c4d 4e4f 5051 5253 5455 5657 5859  JKLMNOPQRSTUVWXY
0x0060:  5a5b 5c5d 5e5f                           Z[\]^_
```

Beschreiben Sie, wie man mit dem Kommando `traceroute` die Struktur von Netzwerken aus der Ferne untersuchen kann und auf welchem Prinzip `traceroute` beruht. Verwenden Sie dazu die oben gezeigten Mitschnitte des Netzverkehrs. Aufgezeichnet wurden Frames ab einschliesslich der MAC-Zieladresse. Falls sich Ihre Antwort direkt auf Felder im Mitschnitt bezieht, können Sie diese dort direkt markieren (Hinweis nächste Seite!). 6 P

Der Aufbau eines IPv4-Headers hat folgendes Format (RFC791):



4 Aufgabe: Paketfilter

`Ostello` ist ein Host, auf dem die Dienste

- httpd
- smtpd
- bind
- ssh

gestartet sind. `Ostello` ist von Rechnern aus VLAN10 und 20 und aus dem Internet erreichbar und muss auf Anfragen aus allen diesen drei Netzen antworten können. Wichtige Hilfsmittel sind evtl. die man-Pages zu `iptables` und `iptables-extensions` .

1. Auf R1 soll das Paketfilter-Modul `netfilter` mit dem Kommando `iptables` konfiguriert werden. Dabei steht das Modul `conntrack` zur Verfügung.

   Erstellen Sie Filterregeln, so dass vom Internet aus **nur** auf die Dienste `httpd, smtpd, ssh` und `bind` auf `ostello` zugegriffen werden kann.

   Beim Aufstellen der Filterregeln muss nur IPv4 berücksichtigt werden. Es müssen auch Regeln existieren, mit denen die Antworten der Dienste an eine Anfrage aus dem Netz zugelassen werden.

   Erstellen sie ein Shell-Skript, das beim Aufruf die korrekten Regeln setzt. Verwenden Sie zum Erstellen des Skripts einen Editor wie `nano, vi, emacs` oder `gedit`. 10 P

2. Auf dem Router R1 sollen TCP/IP-Pakete aus den beiden Netzen 5.5.10.0/24 und 5.5.20.0/24 maskiert werden. D.h. R1 soll die Quelladressen in Pakten aus VLAN10 oder VLAN20 gegen seine eigene Adresse an der Schnittstelle wan0 austauschen.

   Fügen Sie dem Skript die dafür notwendige Regel hinzu. Die beiden Netze 5.5.10.0/24 und 5.5.20.0/24 sollen dabei zu einem Supernet zusammengefasst werden.

Falls Sie kein Supernet bilden können, schreiben Sie die Regel nur für 5.5.10.0/24. 4 P

3. Drucken Sie das Skript mit folgendem Kommando aus:

```
a2ps --column=1 -R -M A4 --delegate=0 --font-size 10 <nameDerDatei>
```

5 Aufgabe: IEEE802.3 und IEEE802.1Q

Auf `R2` werden Frames, die aus dem `VLAN10` stammen mitgeschnitten (d.h. vlan-ID = 10 = 0x0a). Die Frames stammen vom Rechner mit der MAC-Adresse **08:00:27:87:71:9d** und der IP-Adresse **5.5.10.64** und sind an den Webserver auf `iNetHost` adressiert.

Tragen Sie die Antworten auf folgende Fragen unten in die Lösungstabelle ein oder erstellen Sie eine Datei und drucken diese aus. Alle Feldwerte *müssen* als Hexadezimal in 16bit Blöcken angegeben werden.

1. Welchen Zielport haben die TCP-Segmente im Datenteil der Frames? Achtung! **Punkte gibt es nur für den korrekten Hexadezimalwert der Portnummer!!!** 1 P
2. welche MAC-*Quell*adresse haben die Frames? 1 P
3. welche MAC-*Ziel*adresse (siehe Tabelle 1) haben die Frames? 1 P
4. geben Sie das VLAN-Tag nach IEEE802.1Q an. Der Ethernet-Frame habe Standard-Priorität PCP=0. 4 P
5. geben Sie die IP-Quell-Adresse in hexadezimaler Form an. 1 P
6. mit welchem Protokoll ermittelt der Rechner 5.5.10.64 die MAC-*Ziel*adresse? 2 P

Die Switchports #1-#4 sind dem VLAN20 und die Ports #6-#8 dem VLAN10 zugeordnet. Unterscheiden sich die Frames, die an diesen Ports empfangen und gesendet werden von ihrem prinzipiellen Aufbau her von einem EthernetII-Frame? Begünden Sie Ihre Antwort. 2 P

Name, Vorname:	
Feld	Wert **alles HEX!!!**
Zielport	
Quell-MAC	
Ziel-MAC	
VLAN-Tag	
IP-Quell-Adresse (hexad.)	
Feld	Wert **Text**
Protokoll	

Tabelle 7: Lösungstabelle

6 Aufgabe: Zonendatei

1. Auf `ostello` ist der Nameserver `bind9` installiert. Er ist verantwortlich für die Domäne

schroeder-nt.de

   Erstellen Sie die Resource-Records für die Vorwärts- und Rückwärtsauflösung für IPv4 und IPv6 für den Host `ostello`. Vom Internet aus soll der Name *www.schroeder-nt.de* ebenfalls in die Adresse von Ostello aufgelöst werden. Erstellen Sie hierzu noch einen Alias-Eintrag.

   Gehen Sie davon aus, dass `$ORIGIN` **nicht** gesetzt ist und dass die IPv6-Adresse von Ostello den Wert `2001:7c0:c0ca:c01a::e110` hat.

   Tragen Sie alles in die Lösungstabelle 8 ein oder erstellen Sie eine Datei und drucken diese aus. 12 P

Name, Vorname:	
Vorwärtsaufl.	Resource-Record
_____-	
_____-	
_____-	
Rückwärtsaufl.	Resource-Record
_____-	
_____-	

Tabelle 8: Lösungstabelle Resource-Records

2. Beschreiben Sie den Aufbau des FQDN `ostello.schroeder-nt.de.` einschliesslich der Sonderzeichen. 2 P

3. Beschreiben Sie den Sinn und Zweck der Variablen `$ORIGIN` und geben Sie ein Beispiel für ihre Verwendung. 2 P

7 Aufgabe: Sicherheit durch SSL

Auf `Ostello` läuft ein Webserver. Er soll Seiten mit SSL-Verschlüsselung ausliefern. Beim Aufruf von `https://www.schroeder-nt.de` soll eine Testseite erscheinen.

Folgende Vorarbeiten wurden bereits erledigt:

1. Auf `Ostello` wurde mit `openssl` eine *Certification Authority* (CA) mit einem öfffentlichen (`ca-root.crt`) und privaten Schlüssel (`ca-key.key`) erstellt.
2. Das Modul `ssl` wurde mit `a2enmod ssl` aktiviert.
3. Eine Konfigurationsdatei für den virtuellen Host wurde erstellt und in `/etc/apache2/sites-available/www.conf` gespeichert:

```
<IfModule mod_ssl.c>
<VirtualHost _default_:443>
ServerAdmin webmaster@localhost

DocumentRoot /var/www/schroederNt
ServerName www.schroeder-nt.de

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

SSLEngine on

SSLCertificateFile    /etc/ssl/certs/pruefung.crt
SSLCertificateKeyFile /etc/ssl/private/pruefung.key

</VirtualHost>
</IfModule>
```

Bitte bearbeiten Sie folgende Aufgaben:

1. Was ist ein virtueller Host? 4 P
2. Wie unterscheidet der Webserver zwischen mehreren virtuellen Hosts? 2 P
3. Beschreiben Sie *in Worten* welche Schritte prinzipiell notwendig sind, um Mithilfe der CA und `openssl` das Zertifikat (öffentlicher Schlüssel) und den privaten Schlüssel für den Virtuellen Host `www.schroeder-nt.de` zu erstellen. 6 P
4. Angenommen, diese beiden Schlüssel sind erstellt und liegen hier:

   ```
   /root/pruefung/pruefung.crt
   /root/pruefung/pruefung.key
   ```

   Wie können Sie als User *root* diese Schlüssel installieren? Geben Sie die dazu notwendigen Kommandozeilenbefehle an. 2 P
5. Mit welchem Kommandozeilenbefehl kann der virtuelle Host aktiviert werden?
6. Wie können Sie von `iNetHost` aus die Seite testen? Welche Bedingung muss dazu erfüllt sein? 2 P

8 Aufgabe: mqtt

Im Datenschrank, in dem die Netzwerkinfrastruktur der Firma schroeder-NT untergebracht ist, soll die Temperatur, der Zustand der elektrischen Lüfter und die Batteriespannung der USV mit Sensoren überwacht und die Drehzahl der Lüfter beinflusst werden können.

Herr Schröder möchte, dass das Erfassen der Daten und die Drehzahlsteuerung der Lüfter über das mqtt-Protokoll erfolgen soll. 10 P

Beschreiben Sie das Kommunikationsmodell von mqtt in Textform mit Hilfe einer Skizze mit allen obigen Sensoren und Aktoren. Gehen Sie dabei auch auf die Unterschiede zur Kommunikation beim http-Protokoll ein.

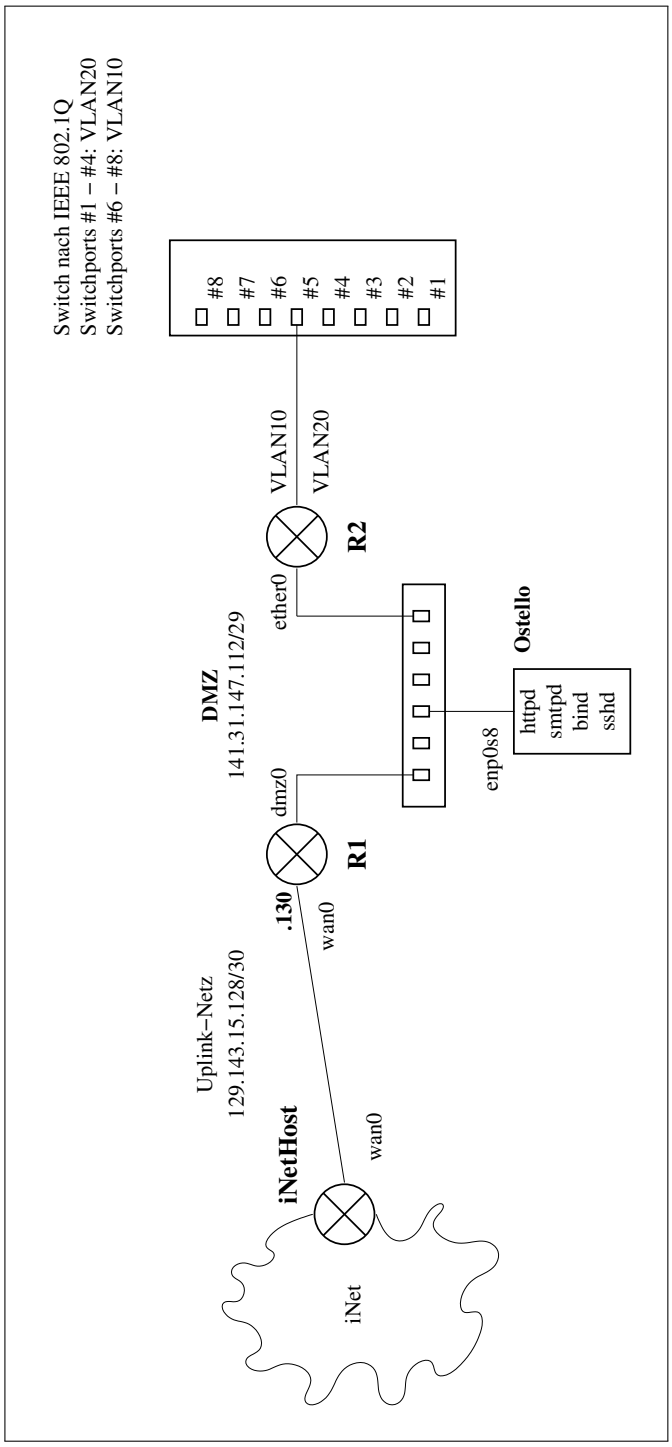


Abbildung 1: Netzwerktopologie

Lösungen

1 Aufgabe

1.1

Name, Vorname:		
Hostname	Interface	Adresse
R1	wan0	129.143.15.130/29
iNetHost	wan0	129.143.15.129/29
Netz	—	129.143.15.128/29

Tabelle 9: Wide-Area-Network Anbindung

1.2

Name, Vorname:		
Hostname	Interface	Adresse/Präfix bzw. Maske
R1	dmz0	141.31.147.117/29
R2	ether0	141.31.147.118/29
Ostello	enp0s8	141.31.147.113/29
BC-Adresse	—	141.31.147.119/29
Maske	—	255.255.255.248

Tabelle 10: DMZ

1.3

Name, Vorname:		
Hostname	Interface	IPv6-Adresse/Präfix
R1	dmz0	2001:7c0:c0ca:c01a:dea6:32ff:fe0a:7012/64
R2	ether0	2001:7c0:c0ca:c01a:ca60:ff:fec7:b1f/64
Ostello	enp0s8	2001:7c0:c0ca:c01a:76da:38ff:fe5e:36b8/64

Tabelle 11: Globale IPv6 Unicast-Adressen

2 Aufgabe

Tabelle für Lösungen:

Hostname	Zielnetz/Präfix	Gateway	Interface
iNetHost	5.5.0.0/19	129.143.15.130	wan0
	141.31.147.112/29	129.143.15.130	wan0
R1	5.5.0.0/19	141.31.147.118	dmz0
Ostello	5.5.0.0/19	141.31.147.118	enp0s8
	129.143.15.128/30	141.31.147.117	enp0s8
R2	129.143.15.128/30	141.31.147.117	ether0
Host in VLAN10	129.143.15.128/30	5.5.10.254	-
	141.31.147.112/29	5.5.10.254	-
Host in VLAN20	129.143.15.128/30	5.5.20.254	-
	141.31.147.112/29	5.5.20.254	-

Tabelle 12: IPv4 Routingtabellen

3 Aufgabe

Zur Funktion (Zitat aus der manpage):

```
traceroute tracks the route packets taken from an IP network on
their way to a given host. It utilizes the IP protocol's time to
live (TTL) field and attempts to elicit an ICMP TIME\_EXCEEDED
response from each gateway along the path to the host.

This program attempts to trace the route an IP packet would follow
to some internet host by launching probe packets with a small ttl
(time to live) then listening for an ICMP "time exceeded" reply from
a gateway.  We start our probes with a ttl of one and increase by
one until we get an ICMP "port unreachable" (or TCP reset), which
means we got to the "host", or hit a max (which defaults to 30
hops). Three probes (by default) are sent at each ttl setting and
a line is printed showing the ttl, address of the gateway and round
trip time of each probe. The address can be followed by additional
information when requested. If the probe answers come from different
gateways, the address of each responding system will be printed.
If there is no response within a certain timeout, an "*"
(asterisk) is printed for that probe.
```

Die Analyse der aufgezeichneten Pakete zeigt, dass der Host 129.143.15.129 (`iNetHost`) zuerst ein Paket mit TTL=01 sendet:

0x0010: 003c 4346 0000 01 11 c4ca 818f 0f81 8d1f

Beim ersten Router wird TTL dekrementiert und da TTL=0, dann verworfen. Gleichzeitig wird eine `ICMP time exceeded` - Meldung an den Sender geschickt.

Beim nächsten Paket hat den Wert TTL=02.

0x0010: 003c 434a 0000 02 11 c3c6 818f 0f81 8d1f

Das Paket wird nach dem ersten Routingvorgang also nicht verworfen und erreicht Host 141.31.147.113 (`Ostello`). Da das Paket an einen nicht aktiven Port auf dem Host adressiert ist:

IP 129.143.15.129.43196 > 141.31.147.113.33438
antwortet dieser mit `udp port 33437 unreachable`

4 Aufgabe

```
# optional, nicht bewerten:
# routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward

iptables -F  # F = flush=klospuelung

# Alle Benutzereigenen Ketten loeschen
iptables -X

#alle Policies (Sprungziele der Ketten INPUT, OUTPUT, FORWARD) auf DROP
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP

# optional, ist als loesung nicht erforderlich
iptables -A INPUT -s 127.0.0.1 -j ACCEPT
iptables -A OUTPUT -d 127.0.0.1 -j ACCEPT

# optional, ist als loesung nicht erforderlich
#alles vom router zu ostello und umgekehrt erlauben
iptables -A OUTPUT -d 141.31.147.113 -j ACCEPT
iptables -A INPUT -s 141.31.147.113 -j ACCEPT

# stateful inspection
# hinweis -m = modul laden; d.h:  -m conntrack = conntrack modul laden

iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

# hinweis -p protocol; d.h. -p tcp -> fuer tcp erlauben
iptables -A FORWARD -p tcp --dport 80 -d 141.31.147.113 \
-m conntrack --ctstate NEW -j ACCEPT
iptables -A FORWARD -p tcp --dport 443 -d 141.31.147.113 \
-m conntrack --ctstate NEW -j ACCEPT
iptables -A FORWARD -p tcp --dport 25 -d 141.31.147.113 \
-m conntrack --ctstate NEW -j ACCEPT
iptables -A FORWARD -p udp --dport 53 -d 141.31.147.113 \
-m conntrack --ctstate NEW -j ACCEPT
iptables -A FORWARD -p tcp --dport 22 -d 141.31.147.113 \
-m conntrack --ctstate NEW -j ACCEPT

# illegale anfragen verbieten
iptables -A FORWARD  -p tcp -m conntrack --ctstate INVALID -j DROP

#nat fuer spaeter
iptables -A FORWARD -o wan0 -i dmz0 -s 5.5.20.0/24 \
-m conntrack --ctstate NEW -j ACCEPT
iptables -t nat -A POSTROUTING -o wan0 -j MASQUERADE

#hosts in den vlans10 und 20 sollen per ssh erreichbar sein
#netz in vlan10: 5.5.10.0/24
#netz in vlan20: 5.5.20.0/24
# -> supernet: 5.5.0.0 - 5.5.31.255
iptables -A FORWARD -p tcp --dport 22 -d 5.5.0.0/19 \
-m conntrack --ctstate NEW -j ACCEPT

# optional, policy ist ohnehin auf drop
# alles, was bisher nicht gefiltert wurde wegschmeissen
iptables -A INPUT -j DROP

iptables -L
```

5 Aufgabe

5.1

Name, Vorname:	
Feld	Wert **alles HEX!!!**
Zielport	0050
Quell-MAC	0800 2787 719d
Ziel-MAC	0800 27cc bebc
VLAN-Tag	8100 000a
IP-Quell-Adresse (hexad.)	0505 0a40
Feld	Wert **Text**
Protokoll	Address Resolution Protocol

Tabelle 13: Lösungstabelle

5.1 Nein.
Die Ports #1-#4 und #6-#8 sind sog. *Access*- oder *Untagged*-Ports und senden und empfangen normale EthernetII-Frames **ohne** VLAN-Tag. So können auch Geräte, die Frames vom Typ 802.1Q nicht verstehen, daran betrieben werden.

6 Aufgabe

6.1

Name, Vorname:	
Vorwärtsaufl.	Resource-Record
	ostello.schroeder-nt.de. IN A 141.31.147.113
	ostello.schroeder-nt.de. IN AAAA 2001:7c0:c0ca:c01a::e110
	www.schroeder-nt.de. IN CNAME ostello.schroeder-nt.de
Rückwärtsaufl.	Resource-Record
	113.147.31.141.in-addr.arpa. IN PTR ostello.schroeder-nt.de.
	0.1.1.e.0.0.0.0.0.0.0.0.0.0.0.0.a.1.0.c.a.c.0.c.0.c.7.0.1.0.0.2.ip6.arpa. IN PTR ostello.schroeder-nt.de.

Tabelle 14: Lösungstabelle Resource-Records

6.2

`ostello.schroeder-nt.de.`

ostello DNS-Label, hier Name des Hosts

. Trennzeichen

schroeder-nt DNS-Label mit einzigem erlaubtem Sonderzeichen: **-**

de. Top-Level-Domain **de** . Der Punkt nach **de** ist das Trennzeichen zur Root-Domain, die als Label einen *leeren String* hat.

6.3

Alle Namen in der Zonendatei, die *nicht mit einem Dezimalpunkt abgeschlossen sind*, werden um den Inhalt der Variablen $ORIGIN ergänzt.

Beispiel (optional), Einträge in der Zonendatei:

```
$ORIGIN wara.de.

msv IN A 141.31.147.114

#das naechste funktioniert nicht:
capo.wara.de IN A 141.31.147.113

#so muss es heissen
capo.wara.de. IN A 141.31.147.113
```

D.h. der FQDN **msv.wara.de.** wird aufgelöst in die Adresse 141.31.147.114

capo.wara.de. wird aber nicht gefunden, da beim zweiten Eintrag der Punkt am Ende vergessen wurde. Der eigentliche Eintrag lautet:

`capo.wara.de.wara.de. IN A 141.31.147.113`

7 Aufgabe

7.1

Ein virtueller Host ist ein Verzeichnis mit allen Inhalten und Seiten einer Web-Site. Jede dieser Sites ist über eine eigene URL erreichbar.

Mehrere verschiedene virtuelle Hosts liegen dabei auf einem physikalischen Host in verschiedenen Verzeichnissen.

Pro virtuellem Host muss auf den Nameservern im Internet ein Eintrag existieren, wobei alle Einträge der virtuellen Hosts in die selbe Adresse des physikalischen Hosts aufgelöst werden müssen.

Auf dem physikalischen Host wird ein Webserver betrieben, der für jeden virtuellen Host eine eigene Konfiguration aufweist. Die einzelnen Konfigurationen werden oft in einzelne Dateien aufgeteilt.

7.2

Die beim Webserver eintreffenden http-Requests müssen im Header einen Eintrag wie z.B.

`Host: www.schroeder-nt.de`

haben. In der Konfiguration des Webservers wird der Hostname mit dem Verzeichnis in dem der gesamte Inhalt der Site liegt, verknüpft ()

```
DocumentRoot /var/www/schroederNt
ServerName www.schroeder-nt.de
```

7.3

1. mit `openssl` wird zuerst der private Schlüssel erzeugt. Z.B.: `pruefung.key`
2. aus dem privaten Schlüssel wird mit `openssl` eine sog. Certificate Signing Request - Datei erzeugt. Z.B. `pruefung.csr` . Die csr-Datei enthält den öffentlichen Schlüssel.
3. die csr-Datei wird der CA zum Signieren übergeben.

   D.h. der öffentliche Schlüssel des Antragstellers wird in einen Hashwert umgewandelt, dieser Hashwert wird mit dem *privaten* Schlüssel der CA verschlüsselt.

   Dieser verschlüsselte Hashwert wird zusammen mit dem öffentlichen Schlüssel des Antragstellers in eine Zertifikatsdatei kombiniert. Z.B. `pruefung.crt`.

   Dieser ganze Vorgang kann ebenfalls mit `openssl` vorgenommen werden, in der Regel wird aber die csr-Datei an eine offizielle, kostenpflichtige CA geschickt.

7.4

```
cp /root/pruefung/pruefung.crt /etc/ssl/certs/pruefung.crt
cp /root/pruefung/pruefung.key /etc/ssl/private/pruefung.key
```

7.5

```
#im verzeichnis /etc/apache2/sites-available
a2ensite www
#anschliessend neustart von apache2 oder reload der config
```

7.6

Vom Host `iNetHost` aus mit einem http-UserAgent (aka *Browser*) die Seite

```
https://www.schroeder-nt.de
```

aufrufen. Dabei muss sichergestellt sein, dass auf `iNetHost` der Domain-Name in die IP-Adresse 141.31.147.113 aufgelöst wird. Dazu kann man in `/etc/hosts` einen entsprechenden Eintrag erstellen oder `bind9` auf `Ostello` konfigurieren.

8 Aufgabe

Bei mqtt kommt kein klassischen Client-Server-Modell wie z.B. bei http zum Einsatz, sondern die Kommunikation erfolgt über einen zentralen Dienst, den sog. *Broker*:

- es besteht keine direkte Verbindung zwischen den Endpunkten (Z.B. Temperaturmessstelle und Temperaturanzeige) bei MQTT.
- MQTT verwendet ein *Publish/Subscribe*-Muster: die Endpunkte werden dabei durch einen Broker (Makler) voneinander entkoppelt und kennen sich nicht.

- Der Broker filtert und speichert die Meldungen der *Publisher* und verteilt sie korrekt an die *Subscriber*.
- Die Entkopplung ist *räumlich* aber auch *zeitlich*: Veröffentlichen und Abonnieren müssen nicht gleichzeitig stattfinden.
- Die Hauptlast der Meldungsverarbeitung liegt beim Broker. Die Endpunkte (Publisher/Subscriber) können daher auf Microcontrollern mit geringen Resourcen betrieben werden.