



## Übung 07: IP Forwarding

### Vorbereitung: Installation von Docker

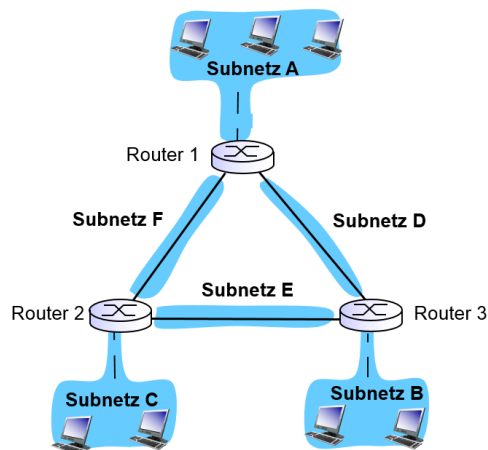
Sie benötigen ein Linux, idealerweise die VM aus Betriebssysteme.

- Legen Sie die Dateien Dockerfile und docker-compose.yaml in ein beliebiges Verzeichnis unter Linux. Beide Dateien werden im Moodle in einem zip-Archiv bereitgestellt:  
<https://learning-campus.th-rosenheim.de/course/view.php?id=1094>
- Installieren Sie *Docker*<sup>1</sup> und *Docker-Compose* (kann bis zu 10 Minuten dauern)  
`sudo apt install docker.io docker-compose`

### Aufgabe 1: Aufteilung von IP Adressen auf Subnetze

Gegeben sei ein Netzwerk mit 6 Subnetzen, siehe Zeichnung. Weisen Sie jedem Subnetz IP Adressen zu unter Einhaltung der folgenden Bedingungen:

- Alle Adressen müssen im Bereich 214.97.254.0/23 liegen
- Subnetz A muss mindestens 250 Hosts unterstützen
- Subnetz B muss mindestens 120 Hosts unterstützen
- Subnetz C muss mindestens 53 Hosts unterstützen
- Die Subnetze D, E und F benötigen jeweils mindestens ein /30 Subnetz.

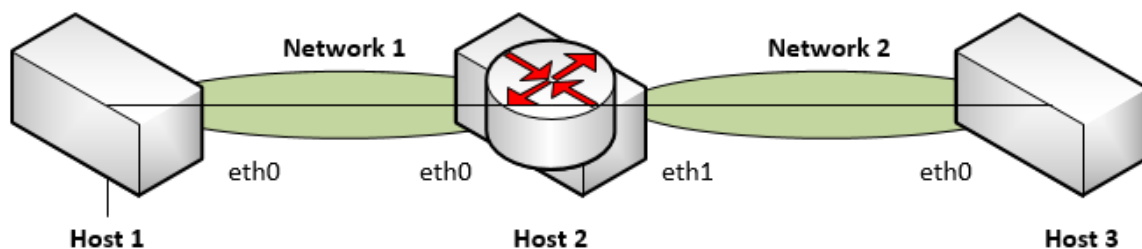


Subnetz	CIDR Notation (a.b.c.d/x)
Subnetz A	
Subnetz B	
Subnetz C	
Subnetz D	
Subnetz E	
Subnetz F	

### Aufgabe 2: IP Forwarding unter Linux

Sie konfigurieren ein vorgegebenes Netzwerk selbst. Die virtuellen Hosts im Netzwerk sind *Docker*-Container. Durch *Docker-Compose* wird ein Netzwerk aus 3 Hosts erzeugt, die durch die Ethernet-Netze Network1 und Network2 miteinander verbunden, siehe. Abbildung.

Ziel: Host1 und Host3 sollen sich gegenseitig erreichen können, obwohl sie in unterschiedlichen Subnetzen sind. Host2 ist ein Router, hat 2 Ethernet Interfaces und muss Pakete weiterleiten.



<sup>1</sup> Mit Docker Container lassen sich „extrem leichtgewichtige, modulare virtuelle Maschinen“ realisieren, die alle unter einem Betriebssystem laufen, ideal um ein kleines Netzwerk nachzubilden.

- a) Nach der Vorbereitung: Erzeugen Sie das Netzwerk für diese Aufgabe, indem Sie den folgenden Befehl im Verzeichnis mit den Dateien `Dockerfile` und `docker-compose.yaml` ausführen:  
`sudo docker-compose up` (Dauer: ca. 3 Minuten)
- b) Öffnen Sie für jeden der 3 Hosts ein Terminal:  
Host1: `sudo docker exec -it host1 /bin/bash`  
Host2: `sudo docker exec -it host2 /bin/bash`  
Host3: `sudo docker exec -it host3 /bin/bash`
- c) Weisen Sie den 3 Hosts **sinnvolle** IP Adressen zu, tragen Sie die Adressen in die Zeichnung ein!  
Anforderung: Öffentliche IPs verwenden, keine privaten IPs!
- d) Konfigurieren Sie nun auf allen Hosts diese IP Adressen!  
- Tool: `ip [addr add]`, Internet-Recherche!  
- Überprüfen Sie mit `ping`, ob sich alle(!) Hosts gegenseitig erreichen können.
- e) Warum funktioniert der `ping` von Host1 zu Host3 nicht? Um Ihre Antwort zu begründen, schauen Sie sich die Routingtabelle von Host1 an.  
Hinweis: `route`
- f) Fügen Sie Host1 manuell eine **statische Route** zu Host3 hinzu: Teilen sie Host1 mit, dass er Host3 über Host2 erreichen kann. Anschließend erneut die Routingtabelle anschauen. Funktioniert der `ping` nun?  
- Hinweis: `ip route add`. Internet-Recherche!  
- Damit Host2 empfangene Pakete weiterleitet, muss Forwarding explizit aktiviert werden:  
`sysctl -w net.ipv4.ip_forward=1`
- g) Korrigieren Sie das Problem aus f).
- h) Das Tool `netcat` wird oft als „Schweizer Taschenmesser“ von TCP/IP bezeichnet. Es erlaubt Daten über TCP und UDP Sockets zu versenden.  
• Starten Sie auf Host3 den „Server“ durch `netcat -l -p 9000`.  
• Auf Host1 öffnen Sie eine Verbindung zum „Server“ mit `netcat <dstip> 9000`. Versenden Sie Textnachrichten („Chat Messages“) zwischen den beiden Hosts.
- i) Schauen Sie sich die ARP-Tabelle auf Host1 mit dem Kommando `arp -n` an. Welche Einträge stehen in der ARP Tabelle? Warum existiert dort ein Eintrag für Host2, warum keiner für Host3?
- j) Löschen Sie den Eintrag für Host2 aus der ARP-Tabelle von Host1 mit dem Kommando `arp -d <ip addr>`. Überzeugen Sie sich, dass der ARP-Tabelle nun leer ist.
- k) Senden Sie nun erneut einen `ping` an Host3. Wurde der ARP-Eintrag für Host 2 erneut hinzugefügt?
- l) Zum Aufräumen: `sudo docker-compose down`