

Einrichten eines GRE-Tunnels

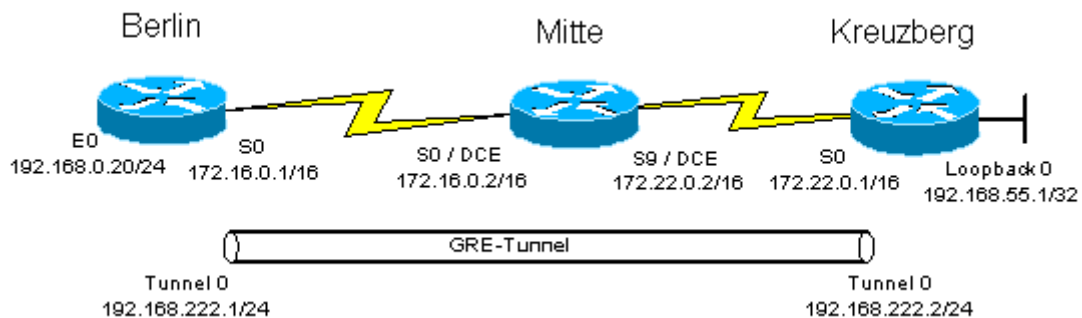
Das Generic Routing Encapsulation (GRE) ist ein Netzwerkprotokoll, das von der Firma Cisco entwickelt wurde und im RFC 1701 definiert ist. Über GRE können andere Protokolle eingepackt werden und so in einem IP-Tunnel transportiert werden.

GRE verwendet das IP Protokoll 47 und der GRE-Header hat den folgenden Aufbau:

C	R	K	S	s	Recur	Flags	Ver	Protocol Type
Checksum (optional)							Offset (optional)	
Key (optional)								
Sequence Number (optional)								
Routing (optional)								

Ein GRE-Paket setzt sich also aus einem IP-Header, einem GRE-Header und der eigentlichen Nutzlast zusammen.

Zeichnung



Konfiguration

```
berlin(config)#int tunnel 0
```

```
berlin(config-if)#ip address 192.168.222.1 255.255.255.0
```

```
berlin(config-if)#tunnel source 172.16.0.1
```

Die Quelle für den Tunnel, also die IP des physikalischen Interfaces von Berlin

```
berlin(config-if)#tunnel destination 172.22.0.1
```

Die Zieladresse für den Tunnel, also die IP des physikalischen Interfaces von Kreuzberg

```
berlin(config-if)#tunnel mode gre ip
```

```
berlin(config-if)#no shutdown
```

```
berlin(config-if)#exit
```

```
berlin(config)#ip route 192.168.55.0 255.255.255.0 tunnel 0
```

Der Traffic zu 192.168.55.0/24 wird durch den Tunnel geschickt

```
berlin(config)#ip route 172.22.0.0 255.255.255.0 s0
```

Die Zieladresse des Tunnels (also Kreuzberg) muss auf Berlin bekannt sein (entweder über ein Routing Protokoll oder wie in diesem Beispiel über statische Routen)

```
kreuzberg(config)#int tunnel 0
```

```
kreuzberg(config-if)#ip address 192.168.222.2 255.255.255.0
```

```
kreuzberg (config-if)#tunnel source 172.22.0.1
```

Die Quelle für den Tunnel, also die IP des physikalischen Interfaces von Kreuzberg

```
kreuzberg (config-if)#tunnel destination 172.22.0.1
```

Die Zieladresse für den Tunnel, also die IP des physikalischen Interfaces von Berlin

```
kreuzberg (config-if)#tunnel mode gre ip
```

```
kreuzberg (config-if)#no shutdown
```

```
kreuzberg (config-if)#exit
```

```
kreuzberg (config)#ip route 192.168.0.0 255.255.255.0 tunnel 0
```

Der Traffic zu 192.168.0.0/24 wird durch den Tunnel geschickt

```
kreuzberg (config)#ip route 172.16.0.0 255.255.255.0 s0
```

Die Zieladresse des Tunnels (also Berlin) muss auf Kreuzberg bekannt sein (entweder über ein Routing Protokoll oder wie in diesem Beispiel über statische Routen)

Kontrolle

Ein Ping aus dem 192.168.0.0/24 Netz nach 192.168.55.1 wird durch den Tunnel geschickt, ein Ping auf 172.22.0.1 über den Router Mitte.

Am besten kann man dies über den Befehl traceroute (auf Windows-Systemen tracert) sehen.

```
berlin#sh ip int brief
```

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0	192.168.0.20	YES	NVRAM	up	up
Serial0	172.16.0.1	YES	NVRAM	up	up
Serial1	unassigned	YES	NVRAM	down	down
Tunnel0	192.168.222.1	YES	manual	up	up

```
berlin#
```

```
berlin#show interfaces tunnel 0
```

```
Tunnel0 is up, line protocol is up
```

```
Hardware is Tunnel
```

```
Internet address is 192.168.222.1/24
```

```
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
```

```
reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation TUNNEL, loopback not set
```

```
Keepalive set (10 sec)
```

```
Tunnel source 172.16.0.1, destination 172.22.0.1
```

```
Tunnel protocol/transport GRE/IP, key disabled, sequencing disabled
```

```
Checksumming of packets disabled, fast tunneling enabled
```

```
Last input 00:04:31, output 00:04:31, output hang never
```

```
Last clearing of "show interface" counters never
```

```
Queueing strategy: fifo
```

```
Output queue 0/0, 0 drops; input queue 0/75, 0 drops
```

```
5 minute input rate 0 bits/sec, 0 packets/sec
```

```
5 minute output rate 0 bits/sec, 0 packets/sec
```

```
12 packets input, 1112 bytes, 0 no buffer
```

```
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
```

```
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
```

```
13 packets output, 2480 bytes, 0 underruns
```

```
0 output errors, 0 collisions, 0 interface resets
```

```
0 output buffer failures, 0 output buffers swapped out
```

```
berlin#
```