

Layer2 Paket senden

Beschreibung:

Kleines python script mit dem man ein layer2 Datenpaket senden kann.

Bedienung:

```
sendframe -h
usage: sendframe.py [-h] -d DEVICE -m {senden,empfangen}
```

```
-d device = eth0 eth1 enp6s0 n2n0 etc
-m senden oder empfangen
```

Beispiel Ziel Computer im empfangsmodus zuerst starten :

```
./sendframe.py -d n2n0 -m empfangen
```

Dann auf dem Quellcomputer paket senden

```
./sendframe.py -d n2n0 -m senden
```

Das Script auch zum [sendframe.py](#)

```
#!/usr/bin/python3

import socket
import sys
import argparse
import signal

def signal_handler(sig, frame):
    print('Programm wurde mit STRG+C beendet.')
```

```

sys.exit(0)

def send_frame(iface):
    # Erstellt einen RAW-Socket
    s = socket.socket(socket.AF_PACKET, socket.SOCK_RAW)
    # Bindet den Socket an die angegebene Schnittstelle
    s.bind((iface, 0))

    # Beispiel eines Ethernet-Frames mit einer leeren Nutzlast
    # Ethernet Header: Ziel MAC, Quelle MAC, Ethertype (0x0800 für IPv4)
    dst_mac = b'\xff\xff\xff\xff\xff\xff' # Broadcast MAC-Adresse
    src_mac = s.getsockname()[4]
    ethertype = b'\x08\x00'
    payload = b'Hello, network!'
    frame = dst_mac + src_mac + ethertype + payload

    # Sendet den Frame
    s.send(frame)
    print(f'Paket gesendet auf {iface}')

def receive_frame(iface):
    # Erstellt einen RAW-Socket
    s = socket.socket(socket.AF_PACKET, socket.SOCK_RAW, socket.ntohs(0x0003))
    # Bindet den Socket an die angegebene Schnittstelle
    s.bind((iface, 0))

    print(f'Warte auf Pakete auf {iface}...')
    while True:
        # Empfängt den Frame
        frame, addr = s.recvfrom(65535)
        src_mac = frame[6:12]
        print(f'Paket empfangen von MAC: {":".join(format(x, "02x") for x in src_mac)}')

def main():
    parser = argparse.ArgumentParser(description='Einfaches Skript zum Senden und Empfangen von Ethernet-Frames')
    parser.add_argument('-d', '--device', required=True, help='Netzwerkgerät (z.B. eth0)')
    parser.add_argument('-m', '--mode', required=True, choices=['senden', 'empfangen'], help='Modus: senden oder empfangen')

```

```
args = parser.parse_args()

# Registriert das STRG+C Interrupt-Signal
signal.signal(signal.SIGINT, signal_handler)

if args.mode == 'senden':
    send_frame(args.device)
elif args.mode == 'empfangen':
    receive_frame(args.device)

if __name__ == '__main__':
    main()
```

Version #2

Erstellt: 30 Mai 2024 15:01:37 von Admin

Zuletzt aktualisiert: 30 Mai 2024 15:09:58 von Admin