SharkFest'25 EUROPE

# Handcrafted packets

## build network packets with Scapy

Uli Heilmeier

# Demo

# Showcase ESP issue I

```
sf_esp_frame=IP(dst="1.2.3.4")/ESP(spi=1234,seq=5678,data=os.urandom(1300))
send(sf_esp_frame)
sf_esp_fragments=fragment(sf_esp_frame,fragsize=1200)
send(sf_esp_fragments)
sf_esp_fragments=fragment(sf_esp_frame,fragsize=200)
send(sf_esp_fragments)
```

# Showcase ESP issue II

```
sf_udp_frame=IP(dst="1.2.3.4")/UDP(sport=1025,dport=53)/Raw(os.urandom(1300)
   )
send(sf_udp_frame)
send(fragment(sf_udp_frame,fragsize=1200))
send(fragment(sf_udp_frame,fragsize=200))

sf_gre_frame=IP(dst="1.2.3.4")/GRE()/Raw(os.urandom(1300))
send(fragment(sf_gre_frame,fragsize=200))
```

# Layers I



```
frame=IP()
frame.show()
frame/UDP()
frame=frame/UDP()
frame.show()
frame.show2()
frame=frame/DNS(qd=DNSQR(qname="www.wireshark.org"))
fame.show()
frame.show2()
frame["DNSQR"].qtype="AAAA"
```

# Layers II

```
frame[1]
frame[0]
frame[3]
frame[UDP].chksum
frame.show()
frame[UDP].chksum=0xffff
frame.show()
del(frame[UDP].chksum)
frame.show()
frame.summary()
```

# Layers III

```
ls()
ls("dns")
explore()
explore("mqtt")
help("dns")
```

# Visualize / Save

```
frame.command()
frame.pdfdump(filename="showcase_frame.pdf")
hexdump(frame)
wireshark(frame)
tcpdump(frame,prog="tshark",args=["-V"])
wrpcapng("example_frame.pcapng",frame)
wrpcap("oldschool.pcap",frame)
lsc()
```

# Stacking layers

```
>>> dnspkt.show2()
###[ Ethernet ]###
  dst= ff:ff:ff:ff:ff:ff
  src= 00:00:00:00:00:00
  type= 0x86dd
###[ IPv6 ]###
     version= 6
     tc= 0
     fl= 0
     plen= 39
     nh= UDP
     hlim= 64
     src= ::
     dst= 2001:db8::1
###[ UDP ]###
        sport= domain
        dport= domain
        len= 39
        chksum= 0x8a7b
###[ DNS ]###
           id= 0
           qr= 0
           opcode= QUERY
           aa= 0
```

# Reading / Filtering



```
ultimate=rdpcap("The_Ultimate_PCAP_v20250325.pcapng")
len(ultimate)
ultimate=rdpcap("The_Ultimate_PCAP_v20250325.pcapng",count=100)
len(ultimate)
ultimate=rdpcap("The_Ultimate_PCAP_v20250325.pcapng")

dns_frames = [frame for frame in ultimate if frame.haslayer(DNS)]
len(dns_frames)
```

# Fuzzing / Random

```
dns_frames[0][DNS].id=RandNum(1,65535)
fuzz_frame = IP(dst="1.1.1.1")/UDP()/fuzz(DNS())
fuzz_frame.show()
send(fuzz_frame,loop=1,count=10)
[frame for frame in IP(ttl=(1,5))/ICMP()]
corrupt_bits(dns_frames[0][DNSQR].qname,n=1)
```

# Sniffing

```
sniffed_frames = sniff(iface=["eth0"],filter="icmp",prn=Packet.summary)
len(sniff_frames)
```

# Sending / Receiving

```
nc_ts=IP(dst="192.168.64.1")/TCP(dport=8901,sport=RandShort(),flags="S",opti
   ons=[("WScale",32),("MSS",1460),("Timestamp", (2294967294,0))])
ans = sr1(nc_ts)
pkt=IP(dst="192.168.65.0/24")/TCP(dport=[23,21],sport=RandShort(),flags="SAU
   FP")
ans,unans=sr(pkt,timeout=1
help(send)
help(sendp)
help(sr)
help(srp)
```

# Config



```
conf.route
conf.route.add(net="10.10.10.10/32",gw="192.168.64.1")
conf.route.resync()

conf.iface
```

# Sockets



```
sck = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sck.connect(("192.168.64.1", 8901))
ssck = StreamSocket(sck)
ssck.basecls = Raw
ssck.sr1(Raw(b'GET / HTTP/1.0\r\n\r\n'))
sck.close()
```

# Installation



- Debian/Ubuntu/Kali:
  ```
  apt install scapy
  ```
- Python:
  ```
  python -m venv .venv
  source .venv/bin/activate
  pip3 install scapy bypthon
  ```
- Living on the edge:
  ```
  git clone https://github.com/secdev/scapy.git
  cd scapy
  ./run_scapy #on Linux/macOS
  .\run_scapy.bat #on Windows
  ```

# Good to know



- More protocols in contrib dir:
  `load_contrib("lacp")`
- Nicer UI with bpython with history, tab completion etc.
- setcap cap_net_raw,cap_net_admin=ep $(which python3)
- More infos: https://scapy.net

# Your turn

# Feedback

# Questions?


Email: uh@heilmeier.eu
: chaos.social/@uhei


#sf25eu