

# Network Debugging Toolkit: Netsniff-NG

Daniel Borkmann

`<daniel.borkmann@tik.ee.ethz.ch>`

Swiss Federal Institute of Technology Zurich (ETH Zurich)  
Computer Engineering and Networks Laboratory  
Communication Systems Group


GTALUG, University of Toronto, October 9, 2012

---

This is only part 2/2 of the original talk.

# High-Performance Network Debugging

## netsniff-ng toolkit



- **netsniff-ng**, a high-performance zero-copy analyzer, pcap capturing and replaying tool
- **trafgen**, a high-performance zero-copy network traffic generator
- **mausezahn**, a packet generator and analyzer for HW/SW appliances with a Cisco-CLI
- **bpfc**, a Berkeley Packet Filter (BPF) compiler with Linux extensions
- **ifpps**, a top-like kernel networking and system statistics tool
- **flowtop**, a top-like netfilter connection tracking tool
- **curvetun**, a lightweight multiuser IP tunnel based on elliptic curve cryptography
- **astraceroute**, an autonomous system (AS) trace route utility

# The Toolkit

- *Here*: focus on netsniff-ng, traefgen, mausezahn
- Used to debug and stress-test our dynamic protocol stack
- Rx/Tx zero-copy: no copies between kernel and user space
- Users reported higher capturing/transmission rates on 1-10 Gbps than commonly used tools (tcpdump/libpcap, Wireshark, ...)
- Part of all the big distributions, plus Backtrack, GRML, Xplico, NST, Alpine Linux, Scientific Linux/CERN

# netsniff-ng

- High-performance traffic analyzer, replayer
- PCAP files compatible with tcpdump, Wireshark, ...
- `netsniff-ng --in eth0 --out dump.pcap -s -b 0`
- `netsniff-ng --in wlan0 --rfrw --out dump.pcap -s -b 0`
- `netsniff-ng --in dump.pcap --mmap --out eth0 -s -b 0`
- `netsniff-ng --in eth1 --out /opt/probe/ -s -m -J --interval 30 -b 0`
- `netsniff-ng --in any --filter ip4tcp.bpf --ascii`

## netsniff-ng, Filtering

```
ldh [12] ; Load Ethernet type field
jeq #0x800, Cont, Drop ; Check value against 0x800
Cont: ldb [23] ; Load IPv4 proto
jeq #0x6, Keep, Drop ; Check against 0x6 (TCP)
Keep: ret #0xffffffff ; Return packet
Drop: ret #0 ; Discard packet
```

- `bpfc ip4tcp.bpfa > ip4tcp.bpf`, then pass it to `--filter`
- Or use `tcpdump`: `tcpdump -dd my-filter`
- Filtering done in the Linux kernel (BPF virtual machine)
- Newer kernels: BPF JIT for x86/x86\_64, powerpc, sparc

## trafgen

- Low-level, high-performance traffic generator
- `trafgen --dev eth0 --conf packets.txf -b 0`
- `trafgen --dev wlan0 --rfraw --conf beacon.txf -b 0`
- `trafgen --dev eth0 --conf trafgen.txf -b 0 --num 10 --rand`
- Own configuration language:

```
{ 0x00, 0x01, 0x03, fill(0xff, 60), 0x04 }
```

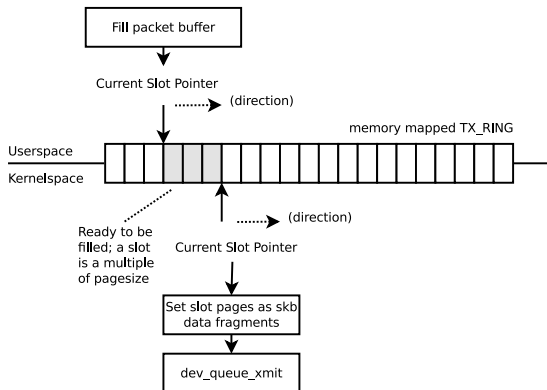
```
{ 0x00, 0x01, 0x03, rnd(60), 0x04 }
```

```
{ drnd(64) }
```

```
{ 0x00, 0b00110011, 0b10101010, rnd(60), 0x04 }
```

## trafgen

- Uses PF\_PACKET sockets with mmap(2)'ed TX\_RING
- Users have reported wire-rate performance from user space
- Low-level packet configuration, more flexible than pktgen



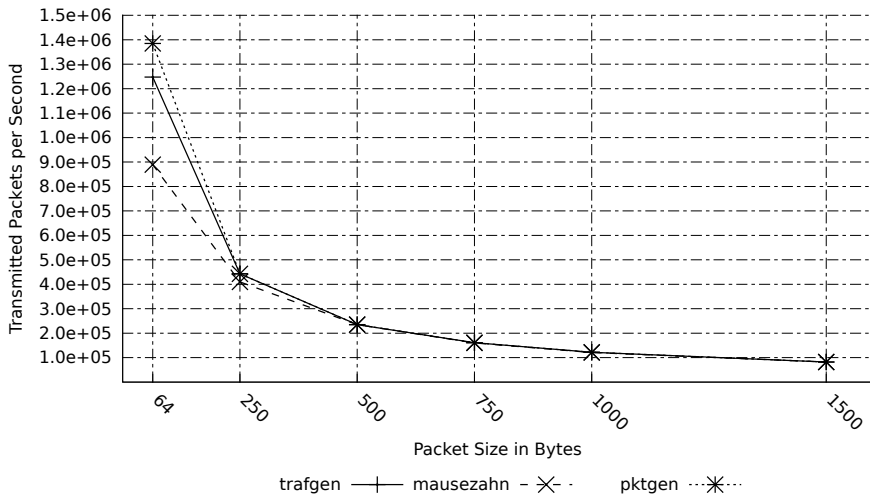
# mausezahn

- High-level, (not so) high-performance traffic generator
- Taken over development and maintainership
- Has a Cisco-like CLI, but also a normal cmdline interface
- Intended for HW/SW appliance in your lab
- `mausezahn eth0 -A rand -B 1.1.1.1 -c 0 -t tcp "dp=1-1023, flags=syn" -P "Good morning! This is a SYN Flood Attack. We apologize for any inconvenience."`
- `mausezahn eth0 -M 214 -t tcp "dp=80" -P "HTTP..." -B myhost.com`



## trafgen, mausezahn, pktgen

Comparison of Traffic Generators



# What's next in netsniff-ng?

- The usual: cleanups, extend documentation, man-pages
- `bpf-h1a`, high-level language for filtering
- DNS traceroute to detect malicious DNS injections on transit traffic
- Compressed on-the-fly bitmap indexing for large PCAP files
- New protocol dissectors/generators for netsniff-ng/mausezahn
- Further performance optimizations (OProfile is your friend)
- Hack `net/packet/af_packet.c` for a better performance

# Source Code

- Toolkit released under GPLv2.0
- Website: <http://www.netsniff-ng.org/>
- Github: <https://github.com/gnumaniacs/netsniff-ng>
- Patches, feedback are welcomed!