nf+bpf

Florian Westphal
4096R/AD5FF600 fw@strlen.de
80A9 20C5 B203 E069 F586
AE9F 7091 A8D9 AD5F F600

Red Hat

Oct 2022

Problem statement

- distro kernels use RETPOLINE=y
- ... which makes indirect calls very expensive
- while nftables already uses tricks to avoid indirections...
- ... the entire netfilter pipeline (conntrack, xtables, selinux hooks, ..) relies on them

xdp bpf dispatcher

- kernel has a bpf program dispatcher, added for xdp
- `DEFINE_BPF_DISPATCHER(foo)`
  - gets bpf context, bpf program to run as arguments
  - nop-sled, run-time patching to call autogenerated/jitted program
  - the autogenerated program will then call the bpf prog directly
  - ... no indirect call anymore
  - but limited to 48 programs atm, once exceeded: indirect calls.

Translate nf_hook_slow to bpf

- instead of having `nf_hook_slow` iterate + call the registered hooks ...
- ... generate a bpf program that calls the registered hooks in-sequence
- indirect calls are rewritten to direct calls.

Preliminary results

- Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60GHz, 14 cores (SMT/HT off)
- pktgen in ingress mode: pktgen → prerouting → forward → postrouting → dummy0
- 64 byte udp packets
- 14 Threads (one per core)
- `CONFIG_RETPOLINE=y`
- results were averaged over 10 runs each

Preliminary results (2)

unpatched 6.0.0-rc:

- netfilter off: 676 MB/s, 1322047 pps
- iptables-nft -m conntrack: 447 MB/s, 875025 pps
- nft ct state new: 480 MB/s, 940362 pps

with bpf patchset:

- iptables-nft -m conntrack: 455 MB/s, 891358 pps
- nft ct state new: 490 MB/s, 958544 pps

Note: Selinux registers netfilter hooks too, so "netfilter off" is not really correct

Preliminary results (3)

- with flowtable 2552 Mb/s (XXX: retest with SELINUX=n)
- with flowtable+bpf: 2941 Mb/s

... but those results are with a grain of salt: this bypasses selinux forward hooks too, not just conntrack+nft filter

Todo

- suppress call to `ip_defrag`: inline `iphr->frag_off` check before call
- elide `nf_queue` handling for netdev family
- retest everything, also with `RETPOLINE=n`
- there are other means to speed things up, figure out which things make sense and which things should better not be used

Todo (2)

- iptables has `xt_bpf` match: call cbf or ebpf prog (filter type)
- plan would be to add new netfilter program type
- programs can be attached to the (raw) hook points
- has access to all info needed to mimic `-i foo -o bar`
- could be used by a hypothetical `iptables-bpf`
- ... or by nft: send both nft expressions and bpf prog, kernel only uses the bpf prog
- such bpf chains would have to be immutable (full chain replacement)