nftables layer 7

NFWS 2017 - Faro, Portugal

Pablo Neira Ayuso <pablo@netfilter.org>

Payload instruction (1)

- Allows you to fetch an arbitrary amount of bytes from skbuff
- You can to indicate [base, offset, length]
  - Base:
    - Link layer (NFT_PAYLOAD_LL_HEADER)
    - Network layer (NFT_PAYLOAD_NETWORK_HEADER)
    - Transport layer (NFT_PAYLOAD_TRANSPORT_HEADER)
  - Offset, in bytes
  - Length, in bytes
- In iptables, u32 match.

Payload instruction (2)

- Before nft_do_chain(), struct nft_pkt is set.
- Link and network offset taken from skbuff.
- Transport offset in calculated:
  - IPv4
    - take iph->protocol and iph->ihl * 4
  - IPv6: ipv6_find_hdr()
    - Parses extension headers
    - Transport layer points to first non-extension header
      - Skips RT and FRAG extensions
  - In both cases, AH is considered a transport header.

Extending payload instruction

- Add application base
  - NFT_PAYLOAD_CONTENT: offset to content
- Keep this away from fast path:
  - Set offset to content in struct nft_pkt from payload.
  - On demand, set it first time user requests it.
- Add protocol definitions in userspace
  - nftables/src/proto.c
- Example:
  - nft add rule x y iif wlan0 dhcp opcode reply counter drop
- Works for:
  - UDP-based protocols
  - Fixed headers.

nftables layer 7

NFWS 2017 - Faro, Portugal

Pablo Neira Ayuso <pablo@netfilter.org>