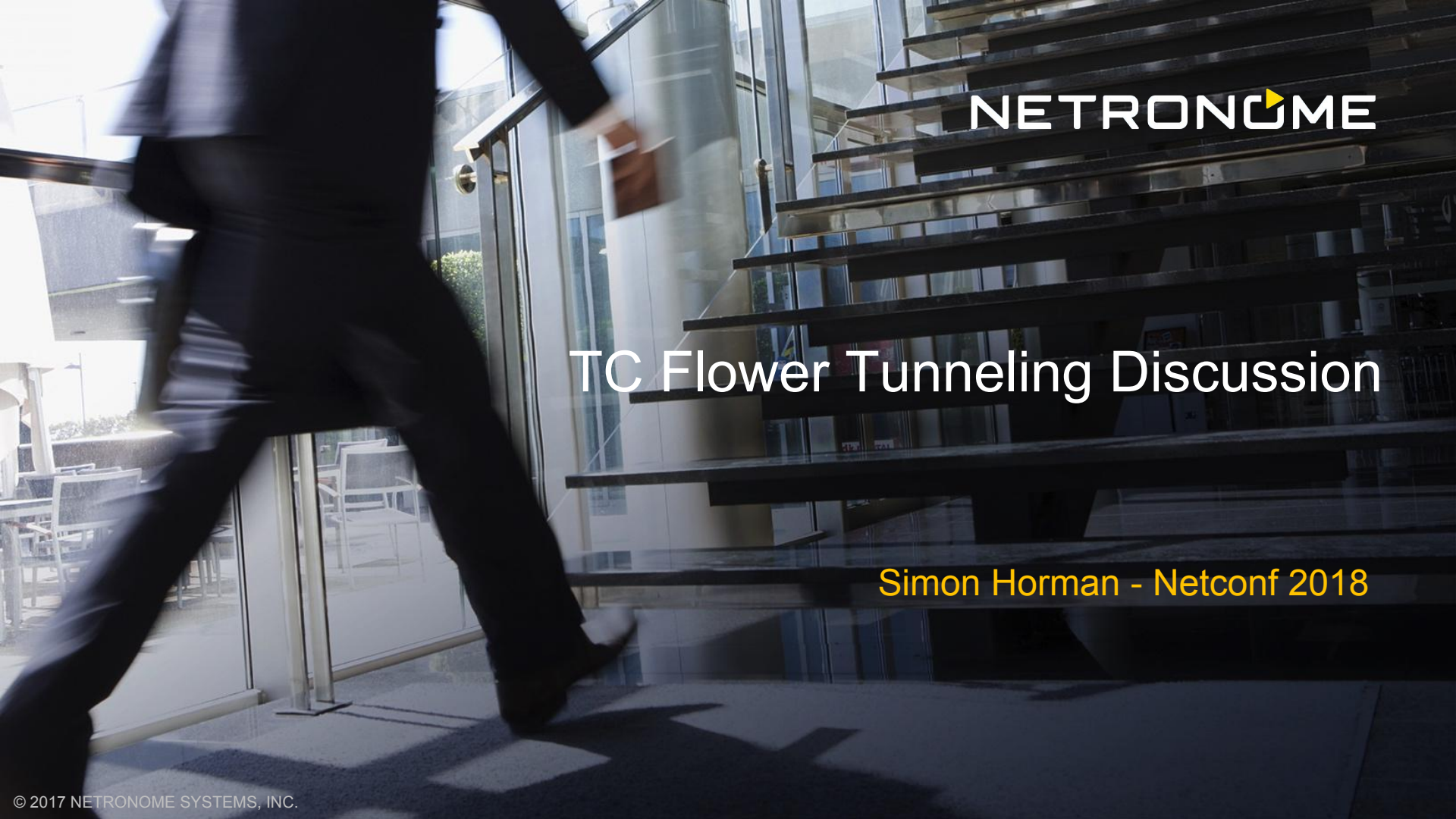

TC Flower Tunneling Discussion

Simon Horman - Netconf 2018

Enc Key Dissection

NETRONOME

- Used only by Flower
- Allows differentiation between inner and outer header keys
- Classification and thus dissection run after decapsulation
- Packet data keys populated using decapsulated packet
  - Inner-header before decap
  - Dissected by skb_flow_dissect()
- “Enc” keys:
  - Seeded from tunnel metadata
  - In turn seeded from outer-header during decapsulation
  - Dissected by skb_flow_dissect_tunnel_info()
  - As decided at previous Netconf

“Enc” Keys

- FLOW_DISSECTOR_KEY_ENC_KEYID
- FLOW_DISSECTOR_KEY_ENC_IPV4_ADDRS
- FLOW_DISSECTOR_KEY_ENC_IPV6_ADDRS
- FLOW_DISSECTOR_KEY_ENC_CONTROL
- FLOW_DISSECTOR_KEY_ENC_PORTS

struct flow_dissector_key_control

NETRONOME

- Used by:
  - FLOW_DISSECTOR_KEY_CONTROL
  - FLOW_DISSECTOR_KEY_ENC_CONTROL
- Has the following fields:
  - .thoff
  - .addr_type
  - .flags
- Only .addr_type used in FLOW_DISSECTOR_KEY_ENC_CONTROL use-case
- Suggested cleanup: create struct flow_dissector_key_enc_control
  - Negligible memory saving?

.addr_type

NETRONOME

- Takes the following values:
  - 0
  - FLOW_DISSECTOR_KEY_IPV6_ADDRS
  - FLOW_DISSECTOR_KEY_IPV4_ADDRS
  - FLOW_DISSECTOR_KEY_TIPC
    - Only in FLOW_DISSECTOR_KEY_CONTROL use-case
- Above are reuse of enum flow_dissector_key_id where
  - 0 = FLOW_DISSECTOR_KEY_CONTROL
- Suggested cleanup: create enum flow_dissector_key_type
  - Cleaner
  - Allow elements to diverge for different use-cases

NFP Decap Offload with Flower

NETRONOME

- Requirements of match
  - ...ENC_CONTROL.addr_type = ...IPV4_ADDRS
  - Exact match on ...ENC_IPV4_ADDRS
  - ...ENC_PORTS is 4789 (VXLAN) or 6081 (Geneve)
    - Used to decide between VXLAN and Geneve
- Current “Enc” Key setup:
  - Also facilitates IPv6
  - Seems to require known-port matching to differentiate between UDP-based encap protocols
  - Does not seem to facilitate matching non-UDP-based encap protocols

Proposal

NETRONOME

- New attribute: FLOW_DISSECTOR_KEY_ENC_L3_PROTO
  - 47 in the case of GRE
  - Unused for UDP-based encapsulation - current ABI
  - But could also be 17 assuming applications ignore unknown attributes
- Alternate: FLOW_DISSECTOR_KEY_ENC_TYPE
  - Enum or string for VXLAN, Geneve, GRE, …
  - This is known during decapsulation
    - VXLAN netdev decap code knows it's VXLAN
  - Would allow UDP encap protocols to be distinguished other than by port
  - Dependent on egdev staying
- Omit FLOW_DISSECTOR_KEY_ENC_PORTS for non-UDP encapsulation
- Usage of other FLOW_DISSECTOR_KEY_ENC_* attributes is unchanged

NFP Encap Offload with Flower

NETRONOME

- Uses tunnel_key action to set tunnel metadata
  - TCA_TUNNEL_KEY_ENC_IPV4_SRC
  - TCA_TUNNEL_KEY_ENC_IPV4_DST
  - TCA_TUNNEL_KEY_ENC_IPV6_SRC
  - TCA_TUNNEL_KEY_ENC_IPV6_DST
  - TCA_TUNNEL_KEY_ENC_KEY_ID
  - TCA_TUNNEL_KEY_ENC_DST_PORT
  - TCA_TUNNEL_KEY_NO_CSUM
- Tunnel type derived from TCA_TUNNEL_KEY_ENC_DST_PORT and checked against type of egress netdev
- Extension to non-UDP seems possible by looking of type of egress netdev

Egress Encap Offload and Bonds

NETRONOME

- Currently for offload egress must be a representor not a bond
- Bonding issues discussed in Jakub's presentation

Geneve Options

NETRONOME

- Would like to allow matching on and setting Geneve options
- Proposed patch for set portion (tunnel_key action):
  - TLVs exposed to user-space: class, type, data
  - Jiri Benc requested check of tunnel type
    - Only set Geneve options for Geneve tunnels
- For match portion Flower needs to be enhanced
  - Maskable TLVs: class, type, data
  - In-order matching
    - Makes sense for fast datapath
    - UAPI should allow extension for any-order matching
  - Should also match on tunnel type?
    - Only match Geneve options for Geneve tunnels