IETF ACL YANG Enhancements

Sonal Agarwal
Mahesh Jethanandani

Agenda

- Support for different combinations of statistics and ACL application on an interface
- Inclusion of match & action parameters
- Open Issues found on further review
- These slides do no reflect discussion after -14

Support for different combinations of statistics

What's in the current model?

Draft 11 provides stats support on a per ACE level. The problem is that it aggregates stats across all interfaces, which is not granular.

What's new in draft 14?

Statistics support on a per interface/per ace basis, in both ingress and egress directions.

```
+--rw interfaces
   +--rw interface* [interface-id]
      +--rw interface-id    if:interface-ref
      +--rw ingress
      |  +--rw acl-sets
      |     +--rw acl-set* [set-name type]
      |        +--rw set-name    -> ../../../../../../acl/a
      |        +--rw type        -> ../../../../../../acl/a
      |        +--rw ace* [rule-name] {interface-state or i
      |           +--rw rule-name          -> ../../../../.
      |           +--rw matched-packets?   yang:counter64
      |           +--rw matched-octets?    yang:counter64
      +--rw egress
         +--rw acl-sets
            +--rw acl-set* [set-name type]
               +--rw set-name    -> ../../../../../../acl/a
               +--rw type        -> ../../../../../../acl/a
               +--rw ace* [rule-name] {interface-state or i
                  +--rw rule-name          -> ../../../../.
                  +--rw matched-packets?   yang:counter64
                  +--rw matched-octets?    yang:counter64
```

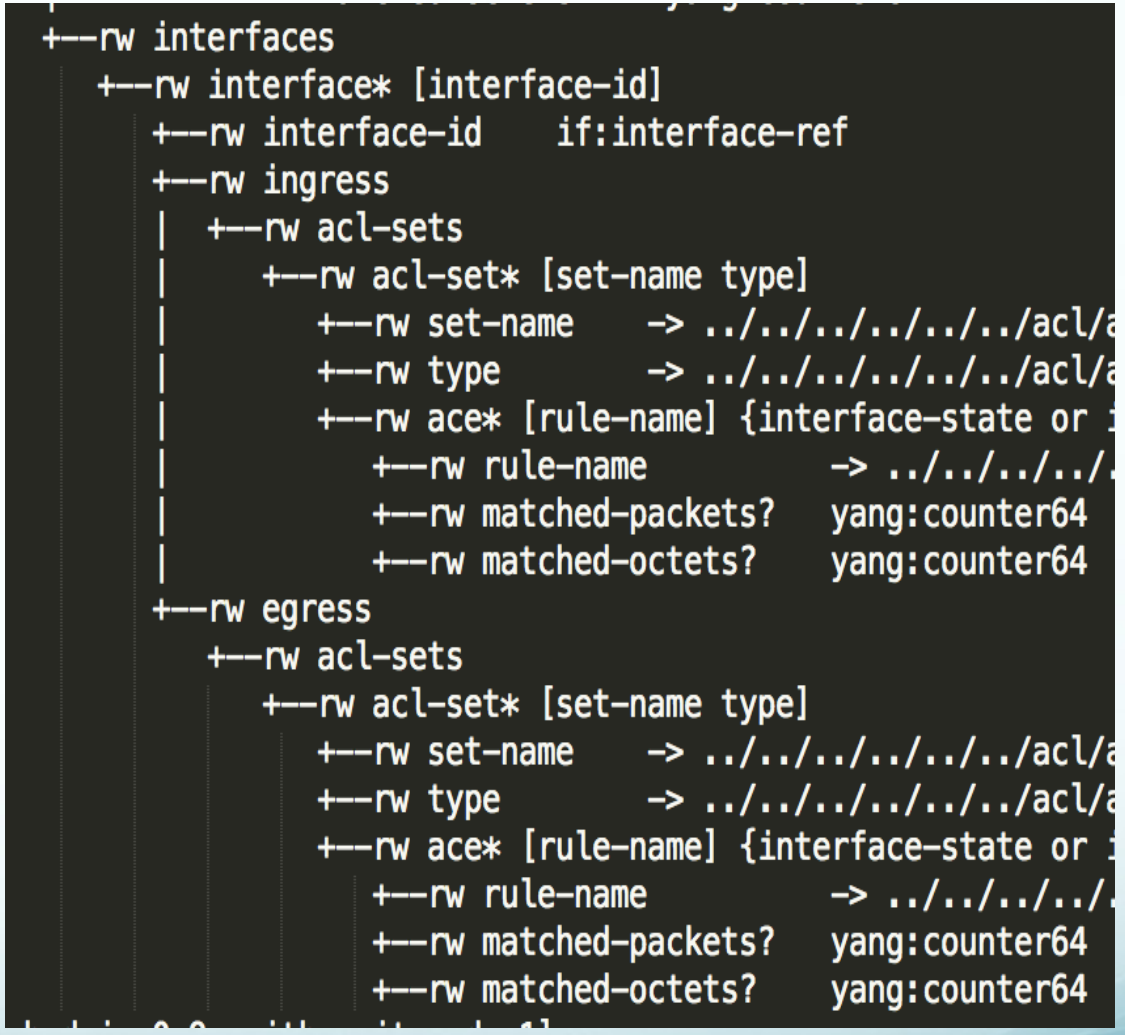
Support for ACL-set application on interface

What's in the current model?

Draft 11 does not provide a way to support ACL application on an interface.

What's new in draft 14?

Support for application of a list of ACL's in both ingress and egress directions of an interface.



Inclusion of match and action parameters

- Added logging option as an action
- Added dscp and ecn leafs
- Added operations to source and destination port containers

```
identity log-action {
    description
      "Base identity for defining the destination for logging actions”;
}
identity log-syslog {
   base log-action;
   description
     "System log (syslog) the information for the packet";
}

typedef operator {
   type enumeration {
      enum lt {
         description
            "Less than.";
      }
      enum gt {
         description
            "Greater than.";
      }
      enum eq {
         description
            "Equal to.";
      }
      enum neq {
         description
            "Not equal to.";
      }
}
```

Inclusion of match and action parameters

- Added headers for tcp

```
+--rw tcp-acl {tcp-acl}?
|  +--rw sequence-number?          uint32
|  +--rw acknowledgement-number?   uint32
|  +--rw data-offset?              uint8
|  +--rw reserved?                 uint8
|  +--rw flags?                    bits
|  +--rw window-size?              uint16
|  +--rw urgent-pointer?           uint16
|  +--rw options?                  uint32
```

- Added udp headers

```
|  +--rw udp-acl {udp-acl}?
|  |  +--rw length?   uint16
```

- Added icmp headers

```
+--rw icmp-acl {icmp-acl}?
|  +--rw type?              uint8
|  +--rw code?              uint8
|  +--rw rest-of-header?    uint32
```

- Added input-interface

```
|  +--rw interface?          if:interface-ref
```

Open issues

https://github.com/netmod-wg/acl-model/issues