IETF ALARM Module

mbj@tail-f.com
stefan@wallan.se
The Draft
Alarm Data Model
Features

- Alarm list
  - A list of all alarms.
  - Cleared alarms stay in the list until explicitly removed.
  - Optional alarm history (state changes) [YANG feature]

- Operator actions on alarms [YANG feature]
  - Acknowledging and closing alarms

- Alarm inventory
  - A management application can read all alarm types implemented by the system.

- Alarm shelving [YANG feature]
  - Shelving (blocking) alarms according to specific criteria.

- Administrative actions on alarms
  - Purging alarms from the list according to specific criteria.

- X.733 Mapping [Augmenting module]
What is an alarm?

An alarm signifies an undesirable state in a resource that requires corrective action.

Not events in general

(KEY)
(resource, alarm-type, severity, clearance-flag)
GigabitEthernet0/25, linkAlarm, Major, -
GigabitEthernet0/25, linkAlarm, Major, Clear
GigabitEthernet0/25, linkAlarm, Minor, -
GigabitEthernet0/25, linkAlarm, Minor, Clear

Alarm State on a resource

This module focuses here!

Alarm Notifications

X733 and other alarm modules focus here!
The Alarm List

• Stateful
  • NOT a notification log
• \( f(\text{alarm-type, resource}) \) -> current alarm state
• Clearance is separate from severity
• Resource state separate from operator state
  • For example: no manual clear
Alarm Type

ietf-alarms.yang

typedef alarm-type-id {
  type identityref {
    base alarm-identity;
  }
  description
  "Identifies an alarm type. The description of the alarm type
  id MUST indicate if the alarm type is abstract or not. An
  abstract alarm type is used as a base for other alarm type ids
  and will not be used as a value for an alarm or be present in
  the alarm inventory.";
}

typedef alarm-type-qualifier {
  type string;
  description
  "If an alarm type can not be fully specified at design time by
  alarm-type-id, this string qualifier is used in addition to
  fully define a unique alarm type.

  The definition of alarm qualifiers is considered being part
  of the instrumentation and out of scope for this module.
  An empty string is used when this is part of a key.";
}

module example-xyz-alarms {
  namespace "urn:example:xyz-alarms";
  prefix xyz-al;

  import ietf-alarms {
    prefix al;
  }

  identity xyz-alarms {
    base al:alarm-identity;
  }

  identity communications-alarm {
    base xyz-alarms;
  }

  identity quality-of-service-alarm {
    base xyz-alarms;
  }

  identity processing-error-alarm {
    base xyz-alarms;
  }

  identity equipment-alarm {
    base xyz-alarms;
  }

  identity environmental-alarm {
    base xyz-alarms;
  }

  // communications alarms
  identity link-alarm {
    base communications-alarm;
  }

  // QoS alarms
  identity high-jitter-alarm {
    base quality-of-service-alarm;
  }
}
Alarm Type and Alarm Type Qualifier

- Alarm Type Qualifier
- Only used when alarm types are not known at design time
  - Example: Digital input with configured detector type
- Industry practice of X.733 “specific problem”

```java
// Alternative 1: concrete alarm type identity
import ietf-alarms {
    prefix al;
}
identity environmental-alarm {
    base al:alarm-type;
    description "Abstract alarm type";
}

identity smoke {
    base environmental-alarm;
    description "Concrete alarm type";
}

// Alternative 2: concrete alarm type qualifier
import ietf-alarms {
    prefix al;
}
identity environmental-alarm {
    base al:alarm-type;
    description "Abstract alarm type";
}
identity external-detector {
    base environmental-alarm;
    description
    "Abstract alarm type, a run-time configuration procedure sets the type of alarm detected. This will be reported in the alarm-qualifier."
}
```
Identifying Resources

• Instance Identifier
• SNMP OID
• String for other naming schemes like DN
• Part of key

• The alarm allows for alternate naming like referring to the interface SNMP OID
Alarm Life-Cycle

- Resource View
  - Severity, Clearance, Text
- Operator View
  - Ack, Close
- Admin View
  - Delete alarms
  - Compress alarm history
Alarm Inventory

• Possible alarms?
• For which resources?
• Can the operator expect an alarm clearance?

```plaintext
+-- ro alarm-inventory
  |   +-- ro alarm-type* [alarm-type-id alarm-type-qualifier]
  |     +-- ro alarm-type-id      alarm-type-id
  |     +-- ro alarm-type-qualifier alarm-type-qualifier
  |     +-- ro resource*          string
  |     +-- ro has-clear          boolean
  |     +-- ro severity-levels*    severity
  |     +-- ro description        string
```
Alarm Shelving

- **“Filtering/Blocking”**
- Move the blocked alarms to the shelf
Notifications

• Alarm state change notification
• Operator state change notification
• Alarm Inventory changed
RPCs/Actions

• Purge
  • Delete alarms based on specific criteria
  • For example: all alarms with:
    • operator state closed
    • clearance flag true
    • Last changed: older than 24h

• Compress
  • Compress the alarm history

• Set operator state for an alarm
X733 Mapping

• Augments alarm list, alarm inventory and alarm notification with X733 parameters
• Optional feature to configure the X733 mapping

```module: ietf-alarms-x733
augment /al:alarms/al:alarm-inventory/al:alarm-type:
  +--ro event-type? event-type
  +--ro probable-cause? uint32
augment /al:alarms/al:control:
  +--rw x733-mapping [alarm-type-id alarm-type-qualifier-match]
      {configure-x733-mapping}?
        +--rw alarm-type-id al:alarm-type-id
        +--rw alarm-type-qualifier-match string
        +--rw event-type? event-type
        +--rw probable-cause? uint32
augment /al:alarms/al:alarm-list/al:alarm:
  +--ro event-type? event-type
  +--ro probable-cause? uint32
augment /al:alarms/al:shelved-alarms/al:shelved-alarm:
  +--ro event-type? event-type
  +--ro probable-cause? uint32
augment /al:alarm-notification:
  +---- event-type? event-type
  +---- probable-cause? uint32```