Open Flow Typed Message Queues

Nikolay Merinov
Inango Systems
Agenda

• About Inango
• Issue Description
• Solution
• Configuration Options
• Israeli software company
• Advanced solutions for communications systems
• Member and contributor to several open-source projects
  • ONOS
  • Yocto
  • RDK
Issue Reproduction

• "mininet" network with "linear,2" topology
• ONOS server with "openflow" and "fwd" applications, "ipv6Forwarding" enabled
• A traffic generator is used to generate ICMPv6 packets from a mininet host:
  • Sending 100,000 IPv6 echo requests per second
    • Source: Host IP, Destination: Random IPv6 address
Issue Reproduction (cont.)
Issue Reproduction (cont.)
Issue reproduction (cont.)

Before test:
- FC00::1
- FC00::2

During test:
- FC00::1
- Traffic generator
- FC00::2

ONOS Summary:
Version: 2.1.0.34d116c
Devices: 2
Links: 2
Hosts: 2
Topology SCCs: 1
Intents: 0
Flows: 10

Infrastructure Link:
A type: Device
A id: of:0000000000000004
A name: of:0000000000000004
A port: 2

B type: Device
B id: of:0000000000000005
B name: of:0000000000000005
B port: 1

A to B: direct / active / not expected
B to A: [no link]
Issue Description

- Under heavy load ONOS stops reading the OpenFlow messages from the socket
- Operation System buffer is exhausted (overflow)
- Operation System sets TCP Window Size to 0
- OpenFlow Agent keeps on sending packets during buffer overflow and so the messages for the ONOS are dropped
- **We loose LLDP packets and reconfigure the network!**
Requirements for the Solution

- Never stop reading messages
- Use several internal buffers for different types of messages
- Drop messages when the internal storage is full
Solution Description

• Define 8 queues for the OpenFlow messages processing
• Queues 0 – 6 can be configured for processing the specific Ethernet types
• A default queue processes all the messages that were not classified to queues 0 – 6
• Size of each queue may be configured independently
Solution Description (cont.)

• Queues are processed in a weighted round robin (WRR) manner
  • Configured with different number of messages processed per time

• **WARNING:** Messages sequences that are passed to different queues may be reordered
  • Message to queue classification should be configured carefully
Queues Configuration

- Queue sizes and weights are configured through the **Component Configuration** ONOS mechanism
- Classifiers can be configured through ONOS **NetworkConfig** API
Queues Configuration (cont.)

- Add new configuration fields to the component
  `org.onosproject.openflow.controller.impl.OpenFlowControllerImpl`

  - `queueSizeN0, ..., queueSizeN6, defaultQueueSize` – queue sizes
  - `bulkSizeN0, ..., bulkSizeN6, defaultBulkSize` – num of messages per period

- Configuration example:
  ```
  cfg set \
  org.onosproject.openflow.controller.impl.OpenFlowControllerImpl \ 
  defaultQueueSize 15000
  ```
Classifiers Configuration

- Queue classifiers can be configured in the following way:

```json
{  "devices":
   {  "of:0000000000000001":
      {  "classifiers": [
         {  "ethernet-type" : "LLDP",
             "target-queue"  : 0 },
         {  "ethernet-type" : "0x1234",
             "target-queue"  : 1 }
         ]
      }
   }
}
```
Classifiers list can be configured for each OpenFlow device

- *ethernet-type* field can be either Ethernet type symbolic name supported by ONOS, or a number in form of “0xABCD”
- *target-queue* field contains the target queue number. Number “7” means default queue
Default Configuration

- queueSizeN0 = 1000
- queueSizeN1, ..., queueSizeN6 = 1
- defaultQueueSize = 5000
- bulkSizeN0, ..., bulkSizeN6, defaultBulkSize = 100
- For each new OpenFlow Agent BDDP and LLDP packets are targeted to queue 0
- All other packets are targeted to the default queue
Performance Impact

Single-Node CBench Throughput
Last 3 Builds

Last Updated: Aug 28, 2019 at 06:15 PM PDT

- 2019-08-28 09:59:36: 1,771,860
- 2019-08-28 18:14:47: 1,703,650

Responses / sec

Build Date
Thank You

Follow Up Links: www.inango.com
ONOS Supported Ethernet Types

- ARP
- RARP
- IPV4
- IPV6
- LLDP
- VLAN
- QINT
- BDDP
- MPLS_UNICAST
- MPLS_MULTICAST
- EAPOL
- SLOW