OpenNMS 101

Tarus Balog

tarus@opennms.org

http://www.opennms.org/Training

Unit 7: SNMP Traps

SNMP Traps

- OpenNMS Can Receive SNMP Traps
- Traps Must be Mapped to UEIs
- Events can be edited to use masks to make this happen

Many Traps are Pre-defined

Recent Events			
90	12/6/16 12:14:07	Normal	Agent Up with Possible Changes (coldStart Trap) enterprise:.1.3.6.1.4.1.8072.3.2.10 (.1.3.6.1.4.1.8072.3.2.10) args(0):
89	12/6/16 12:14:06	Warning	nsNotifyShutdown trap received

Send Traps With snmptrap

```
snmptrap -v 1 -c public 127.0.0.1 \
    .1.3.6.1.4.1.99999.2 localhost 6 1 ' ' \
    .1.3.6.1.4.1.99999.3.1 s 'They be bored'
```

Event 91

Severity	Normal	Node	localhost
Time	Dec 6, 2016 12:17:29 PM	Interface	127.0.0.1
Service			
UEI	uei.opennms.org/generic/traps/EnterpriseDefault		

Log Message

Received unformatted enterprise event (enterprise:.1.3.6.1.4.1.99999.2 generic:6 specific:1). 1 args: .1.3.6.1.4.1.99999.3.1="They be bored"

Description

This is the default event format used when an enterprise specific event (trap) is received for which no format has been configured (i.e. no event definition exists).

The total number of arguments received with the trap: 1.

They were:

.1.3.6.1.4.1.99999.3.1="They be bored"

Here is a "mask" element definition that matches this event, for use in event configuration files:

```
<mask>
  <maskelement>
    <mename>id</mename>
    <mevalue>.1.3.6.1.4.1.99999.2</mevalue>
  </maskelement>
  <maskelement>
    <mename>generic</mename>
    <mevalue>6</mevalue>
  </maskelement>
  <maskelement>
    <mename>specific</mename>
    <mevalue>1</mevalue>
  </maskelement>
</mask>
```

Formatting SNMP Traps

- OpenNMS will provide the mask needed to match a trap
- The MIB Compiler tool can be used to import trap definitions
- Use the SystemOID to help deduce the device that sent the trap

Update `Class.events.xml`

- `cd /opt/opennms/etc/events`
- Run

  `wget -N https://www.opennms.org/~tarus/Class/Config/Exercise%207/Class.events.xml`

- Reload Eventd

  `send-event.pl uei.opennms.org/internal/reloadDaemonConfig -p 'daemonName Eventd'`

Class.events.xml: Add a "Down" Trap

```xml
<event>
  <mask>
    <maskelement>
      <mename>id</mename>
      <mevalue>.1.3.6.1.4.1.99999.2</mevalue>
    </maskelement>
    <maskelement>
      <mename>generic</mename>
      <mevalue>6</mevalue>
    </maskelement>
    <maskelement>
      <mename>specific</mename>
      <mevalue>1</mevalue>
    </maskelement>
  </mask>
  <uei>uei.opennms.org/class/moody</uei>
  <event-label>OpenNMS defined event: The OpenNMS Class is moody</event-label>
  <descr>
    &lt;p&gt;This event is sent when the OpenNMS Class is moody.&lt;/p&gt;
  </descr>
  <logmsg dest='logndisplay'>
    &lt;p&gt;OpenNMS Class is moody: %parm[#1]%&lt;/p&gt;
  </logmsg>
  <severity>Warning</severity>
</event>
```

Class.events.xml: Add an "Up" Trap

```xml
<event>
  <mask>
    <maskelement>
      <mename>id</mename>
      <mevalue>.1.3.6.1.4.1.99999.2</mevalue>
    </maskelement>
    <maskelement>
      <mename>generic</mename>
      <mevalue>6</mevalue>
    </maskelement>
    <maskelement>
      <mename>specific</mename>
      <mevalue>2</mevalue>
    </maskelement>
  </mask>
  <uei>uei.opennms.org/class/joyful</uei>
  <event-label>OpenNMS defined event: The OpenNMS Class is joyful</event-label>
  <descr>
    &lt;p&gt;This event is sent when the OpenNMS Class is joyful.&lt;/p&gt;
  </descr>
  <logmsg dest='logndisplay'>
    &lt;p&gt;OpenNMS Class is joyful: %parm[#1]%&lt;/p&gt;
  </logmsg>
  <severity>Normal</severity>
</event>
```

Formatted Events

▼ ID	Severity	Time	Node	Interface	Service
97	**Normal** ⊞ ⊟	Dec 6, 2016 12:46:16 PM ◀ ▶	localhost ⊞ ⊟	127.0.0.1 ⊞ ⊟	
		uei.opennms.org/class/joyful ⊞ ⊟ Edit notifications for event			
		OpenNMS Class is joyful: They have Pizza!			
96	**Warning** ⊞ ⊟	Dec 6, 2016 12:46:14 PM ◀ ▶	localhost ⊞ ⊟	127.0.0.1 ⊞ ⊟	
		uei.opennms.org/class/moody ⊞ ⊟ Edit notifications for event			
		OpenNMS Class is moody: They be bored			

Other Event Sources

- Using send-event.pl or other methods to talk directly to port 5817
- Convert SNMP traps to events via trapd
- Convert Syslogs to events via syslogd
- Convert TL/1 messages via tl1d

Exercise #7: Define Trap Events

- Generate the "moody" trap
- Generate the "joyful" trap
- Note that both are unformatted
- Update Class.events.xml
- Repeat both traps
- Note that they are formatted
- Have fun!

Trap Commands

Moody:
```
snmptrap -v 1 -c public 127.0.0.1 \
.1.3.6.1.4.1.99999.2 localhost 6 1 ' ' \
.1.3.6.1.4.1.99999.3.1 s 'They be bored'
```

Joyful:
```
snmptrap -v 1 -c public 127.0.0.1 \
.1.3.6.1.4.1.99999.2 localhost 6 2 ' ' \
.1.3.6.1.4.1.99999.3.1 s 'They have Pizza!'
```