Zabbix and the art of SNMP traps

Action = Reaction
whoami

Brian van Baekel

• Zabbix consultant
• Zabbix trainer
• Netherlands
• United Kingdom
• United States
Goal of this talk

• Explain the (very) basics of SNMP polling
• Explain the (very) basics of SNMP trapping
• Explain how to capture traps
• Explain how to react on those traps
What is SNMP

**Simple Network Management Protocol**

- Introduced early 90s
- 3 version: v1, v2c, v3
- Used to monitor:
  - Routers
  - Switches
  - Firewalls
  - Printers
  - Applications?
  - etc

- Various components:
  - Polling
  - Trapping
  - Commands/Control
Polling

- Zabbix server is requesting data from a remote device
  - Snmpget: single request, single metric
  - GetBulk: single request, many metrics
### MIB? OID? ?????

- **MIB:** Management Information Base
- **OID:** Object IDentifier

### SysUptime MIB

- **sysUpTime**
  - **OBJECT-TYPE**
  - **SYNTAX** TimeTicks
  - **ACCESS** read-only
  - **STATUS** mandatory
  - **DESCRIPTION**
    
    "The time (in hundredths of a second) since the network management portion of the system was last re-initialized."
  
  ::= { system 3 }

### SysUptime OID

- **.1.3.6.1.2.1.1.3**
Trapping

- Remote device is pushing data to Zabbix
- Event driven
  - Port flapping
  - Temperature too high/low
  - Administrative login
  - etc

SNMP device → snmptrapd → Trap receiver → Trap file → Zabbix SNMP trapper
# How it looks in Zabbix

- **Polling:**

<table>
<thead>
<tr>
<th>Name</th>
<th>Triggers</th>
<th>Key</th>
<th>Interval</th>
<th>History</th>
<th>Trends</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Bits received</td>
<td>Triggers: net.if.in[iHICInOctets.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>365d</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Bits sent</td>
<td>Triggers: net.if.out[iHICOutOctets.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>365d</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Inbound packets discarded</td>
<td>net.if.in.discards[iHIDiscards.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>365d</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Inbound packets with errors</td>
<td>Triggers: net.if.in.errors[iHINerrors.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>365d</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Interface type</td>
<td>Triggers: net.if.type[iHIType.4]</td>
<td></td>
<td>4h</td>
<td>7d</td>
<td>0</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Operational status</td>
<td>Triggers: net.if.status[iHIOperStatus.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>0</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Outbound packets discarded</td>
<td>net.if.out.discards[iHIDiscards.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>365d</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Outbound packets with errors</td>
<td>Triggers: net.if.out.errors[iHOutErrors.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>365d</td>
<td>SNMP agent</td>
</tr>
<tr>
<td>Network interfaces discovery: Interface lan(Lan): Speed</td>
<td>Triggers: net.if.speed[iHISpeed.4]</td>
<td></td>
<td>5m</td>
<td>7d</td>
<td>0</td>
<td>SNMP agent</td>
</tr>
</tbody>
</table>
# How it looks in Zabbix

- **Trapping:**

  Capture all (fallback)

<table>
<thead>
<tr>
<th>Name</th>
<th>Triggers</th>
<th>Interval</th>
<th>History</th>
<th>Trends</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template Module Generic SNMP: SNMP traps (fallback)</td>
<td>snmptrap.fallback</td>
<td></td>
<td></td>
<td></td>
<td>SNMP trap</td>
</tr>
<tr>
<td><strong>Network interfaces:</strong> Link status trap for dmz</td>
<td>snmptrap(&quot;((IF-MIB::linkDown</td>
<td>IF-MIB::linkUp prescribe::space:)</td>
<td>&quot;dmz&quot;)</td>
<td>90d</td>
<td></td>
</tr>
<tr>
<td><strong>Network interfaces:</strong> Link status trap for internal</td>
<td>snmptrap(&quot;((IF-MIB::linkDown</td>
<td>IF-MIB::linkUp prescribe::space:)</td>
<td>&quot;internal&quot;)</td>
<td>90d</td>
<td></td>
</tr>
<tr>
<td><strong>Network interfaces:</strong> Link status trap for modem</td>
<td>snmptrap(&quot;((IF-MIB::linkDown</td>
<td>IF-MIB::linkUp prescribe::space:)</td>
<td>&quot;modem&quot;)</td>
<td>90d</td>
<td></td>
</tr>
<tr>
<td><strong>Network interfaces:</strong> Link status trap for office</td>
<td>snmptrap(&quot;((IF-MIB::linkDown</td>
<td>IF-MIB::linkUp prescribe::space:)</td>
<td>&quot;office&quot;)</td>
<td>90d</td>
<td></td>
</tr>
</tbody>
</table>

---

*Zabbix summit 2021*
How it looks in Zabbix

Raw trap(tcpdump):

09:27:01.569250 IP 192.168.1.251.snmp > 192.168.0.3.snmptrap:
Trap(81) .1.3.6.1.6.3.1.1.5 192.168.1.251 linkUp 119078322
  .1.3.6.1.2.1.2.1.1.1=1
  .1.3.6.1.2.1.2.1.7.1=1
  .1.3.6.1.2.1.2.1.8.1=1

Zabbix:
Conceptual

• Poll frequency should be as low as possible. 10-15 minutes interval?
• Traps are important to get fast status updates
• Don’t rely on traps only
• Upon received trap we want to get all(or a subset) of items updated ASAP.
• Devices are typically not capable to facilitate fast polling
Conceptual

• We can get a status update with the “execute now” function in Zabbix

• We can utilize the API of Zabbix

• Assumption: Zabbix configuration is working already
Configuration

- Make sure all SNMP trap triggers are using tags:

- Severity can be anything
- Multiple tags are not a problem
Create an API token (Administration -> General -> API tokens):
Configuration

Create a new frontend script (Administration -> Scripts):

- **Name**: Update all items
- **Scope**: Action operation, Manual host action, Manual event action
- **Type**: Webhook, Script, SSH, Telnet, IPMI
- **Execute on**: Zabbix agent, Zabbix server, Zabbix server (proxy)
- **Commands**: `python3 /usr/lib/zabbix/frontendscripts/update_all_items.py '{HOST.HOST}'`
- **Description**: 

---

Zabbix summit 2021
Configuration

Create an action:

Condition:

Operation:

- Operation: Update all items
- Steps: 1 - 1 (0 - infinitely)
- Step duration: 0 (0 - use action default)
- Target list:
  - Current host
  - Host: type here to search
  - Host group: type here to search
- Conditions:
  - Add
Configuration

• Last but not least, the API script
  • Download from https://github.com/OpensourceICTSolutions/zabbix-update_all_items
  • Place it in /usr/lib/Zabbix/frontendscripts/ on your Zabbix server

Prerequisites:
• Python3
• python pip
  • Requests module
Result

Normal operations

Polling (once/15min or slower)

1. Zabbix
2. SNMP device
3. Zabbix
**Result**

Upon trap receival

1. **SNMP trap**
2. Trap received. Trigger goes in problem state
3. Action takes place due to problem event: Calling the Python script
4. Python script: execute check now on all items of involved host
5. Immediate poll
6. Within seconds after receiving the trap, we know the exact state of the device as all items are updated
Thank you

Questions?