Automations for Monitoring and Troubleshooting your Cisco IOS Network

Presented by Dan Jerome
An Analogy

From: Full control by a single central authority
To: Operating a system of self-managing components

With increasing scale, complexity, differentiation and availability requirements, operators rely on Embedded Automations.

<table>
<thead>
<tr>
<th>Airplane</th>
<th>Router</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instruments</td>
<td>Embedded Automations</td>
</tr>
<tr>
<td>21,000 sensors</td>
<td>OIDs in MIBs</td>
</tr>
</tbody>
</table>
The Human Factor ...
## Device Manageability Instrumentation

**Cisco IOS® Device Manageability Instrumentation (DMI)**

### Fault
- IP OAM—Ping, Trace, BFD, ISG per session
- **802.3ah**—Link monitoring and remote fault indication
- **802.1 ag**—Continuity check, L2 ping, trace, AIS
- MPLS OAM—LSP ping, LSP trace, VCCV
- EEM—Embedded Event Manager
- EVENT-MIB—OID-based triggers, events, or SNMP Set, IETF DISMON
- EXPRESSION-MIB—OID expression-based triggers, IETF DISMON
- ...   

### Configuration
- Config CLI—diff, logging, lock, replace, rollback
- E-LMI—parameter and status signaling
- E-DI—Enhanced Device Interface, CLI, Perl, IETF Netconf
- EMM—Embedded Menu Manager
- NETCONF—IETF NETCONF XML PI
- CNS and WSMA
- TR-069
- KRON—command scheduler
- AutolInstall—bootstrapping
- IOS.sh—IOS Shell
- SmartInstall
- Auto SmartPorts
- ...   

### Performance
- Auto IP SLA—delay, jitter, loss probability
- CBQoS MIB—class-based QoS
- NBAR
- RMON
- EPC—Embedded Packet Capture
- ERM—Embedded Resource Manager
- GOLD—Generic Online Diagnosis
- Smart Call Home—preventive maintenance
- VidMon—Video Monitoring
- ...   

### Accounting
- Flexible NetFlow—IETF IPFIX
- BGP policy accounting—includes AS information
- Periodic MIB bulk data collection and transfer
- ...   

### Security
- Auto Secure—one-touch device hardening
- LDP Auth—message authentication
- Routing Auth—MD5 authentication, BGP, OSPF
- ...   

---

**Device Manageability Instrumentation Has Evolved**

---
Packaging Embedded Automations

**Problem:** Automations may consist of multiple elements – how to deploy them in a professional and efficient manner?

**Solution I:** Write detailed requirements and step-by-step instructions

**Solution II:** Create an installable EASy package

- Package Description
- Pre-Requisite Verification
- Pre-Installation Config
- Pre-Installation Exec
- Environment Variables
- Configuration
- Files
- Post-Requisite Verification
- Post-Installation Config
- Post-Installation Exec
- Uninstall

EASy Installer + MyPackage.tar = Menu Guided Installation

Router# easy-installer tftp://10.1.1.1/mypackage.tar flash:/easy

Configure and Install EASy Package 'mypackage-1.03'

1. Display Package Description
2. Configure Package Parameters
3. Deploy Package Policies
4. Exit

Enter option: 2

See: [http://www.cisco.com/go/easy](http://www.cisco.com/go/easy)
Embedded Automation Systems

**Embedded Automation Systems (EASy)**

1. Browse and Download EASy Packages
   www.cisco.com/go/easy

2. Make Sure to also download EASy Installer

3. Browse Other Embedded Automations
   www.cisco.com/go/ciscobeyond

4. Learn About The Technology Under The Hood
   www.cisco.com/go/instrumentation
   www.cisco.com/go/eem
   www.cisco.com/go/pec

5. Discuss, Ask Questions, Suggest Answers
   supportforums.cisco.com

6. Upload your own Examples to CiscoBeyond
   www.cisco.com/go/ciscobeyond

7. Engage via ask-easy@cisco.com
Agenda

- Using SNMP for Monitoring
- How to Analyze Transient Conditions?
- What about the Service?
- Who is doing What on the Network?
- What if I need a Packet Capture?
- Summary
SNMPv2c: Review

Version = SNMPv2c
Community string = 'clear text'
SNMP PDU = Get, GetNext, Set, GetBulk

Version = SNMPv2c
Community string = 'clear text'
SNMP PDU = GetResponse, Trap, Inform
What’s new in SNMPv3?

SNMPv3 defines two security-related capabilities:

β The user-based security model (USM)
   – provides authentication (user/password)
   – privacy (encryption)
   Note: operates at the message level

β The view-based access control model (VACM)
   – determines whether a given principal (user) is allowed access to particular MIB objects to perform particular functions
   Note: operates at the PDU level

Available from: IOS 12.0(3)T, 12.0(6)S
Where to start with MIBs?

MIB Locator:  
http://www.cisco.com/go/mibs
Which OIDs are actually being used?

Example: CiscoView polling

Router#show snmp statistics oid

<table>
<thead>
<tr>
<th>time-stamp</th>
<th>#of times requested</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>97</td>
<td>sysUpTime</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>9</td>
<td>cardTableEntry.7</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>9</td>
<td>cardTableEntry.1</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>4</td>
<td>cardTableEntry.9</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>16</td>
<td>ifAdminStatus</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>16</td>
<td>ifOperStatus</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>6</td>
<td>ciscoEnvMonSupplyStatusEntry.3</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>17</td>
<td>ciscoFlashDeviceEntry.2</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>8</td>
<td>ciscoFlashDeviceEntry.10</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>2</td>
<td>ltsLineEntry.1</td>
</tr>
<tr>
<td>16:16:50 CET Jan 12 2005</td>
<td>2</td>
<td>chassis.15</td>
</tr>
<tr>
<td>16:16:27 CET Jan 12 2005</td>
<td>11</td>
<td>ciscoFlashDeviceEntry.7</td>
</tr>
<tr>
<td>16:16:27 CET Jan 12 2005</td>
<td>2</td>
<td>cardIfIndexEntry.5</td>
</tr>
<tr>
<td>16:16:24 CET Jan 12 2005</td>
<td>1</td>
<td>ciscoFlashDevice.1</td>
</tr>
</tbody>
</table>

Available from: IOS 12.0(22)S, 12.4(20)T
Is there a way to quickly export SNMP Statistics?

**Problem:** Sometimes we need data from one or multiple MIBs, but
- we may not want to (re-)configure an NMS
- don’t want to constantly poll
- need to gather data during temporary loss of connectivity

**Solution:** Use Bulk File MIB to define the data we need and periodically transfer it to a convenient location
- group data from multiple MIBs
- single, common polling interval
- buffer data
- transfer using RCP, FTP, TFTP
- format ASCII or Binary

Feature Name: Periodic MIB Data Collection and Transfer Mechanism

Available from: IOS 12.0(24)S, 12.2(25)S, 12.3(2)T, IOS XE 2.1, IOS XR 3.2
Platforms: ASR1k, x8xx ISR, x900x ISR, 72xx, 73xx, 76xx, 10xxx, ME3400, C4k, C6k, …
Service Planning

Configuration – Example

1. Define Lists of relevant OIDs (Names for IF-MIB, ASN.1 for all others)

Router(config)# snmp mib bulkstat object-list my-if-data
Router(config-bulk-objects)# add ifIndex
Router(config-bulk-objects)# add ifDescr
Router(config-bulk-objects)# add ifAdminStatus
Router(config-bulk-objects)# add ifOperStatus
Router(config-bulk-objects)# exit

2. Specify Polling Schema

Router(config)# snmp mib bulkstat schema my-if-schema
Router(config-bulk-sc)# object-list my-if-data
Router(config-bulk-sc)# poll-interval 1
Router(config-bulk-sc)# instance exact interface FastEthernet0
Router(config-bulk-sc)# exit

3. Configure the Transfer Mechanism – and enable it!

Router(config)# snmp mib bulkstat transfer my-fa0-transfer
Router(config-bulk-tr)# schema my-if-schema
Router(config-bulk-tr)# transfer-interval 5
Router(config-bulk-tr)# url primary tftp://10.10.10.10/folder/
Router(config-bulk-tr)# retain 30
Router(config-bulk-tr)# buffer-size 4096
Router(config-bulk-tr)# enable
What if it’s not in a MIB?

**Problem:** Collect data via SNMP, even if there is no MIB support currently available.

**Solution:** Expression-MIB provides the capability to process data into more relevant information via SNMP

- Expression-MIB can be configured using SNMP directly since 12.0(5)T.
- Initially Cisco Implementation was based on OID 1.3.6.1.4.1.9.10.22 but current Cisco implementation is based on RFC2982-MIB, OID 1.3.6.1.2.1.90.
- In 12.4(20)T Expression-MIB feature is enhanced to add CLIs to configure expressions.

Expression-MIB can gather data from Command Line Interface (CLI show commands), even if there is no MIB support

EVENT-MIB adds ability to send an event based on value of expression

EEM 3.1 provides similar capability without the need to involve Expression-MIB or Event-MIB

How to Analyze Transient Conditions?
“Troubleshooting starts **before** troubleshooting starts.

Be prepared.”

Source unknown
Embedded Event Manager (EEM)

1. Something happens on the causing an Event to trigger

2. An EEM Event Detector receives notification

3. An EEM Policy is activated that initiates a pre-defined set of actions
EEM Architecture

Actions

Embedded Event Manager

EEM Applets
multi-event-correlation

Event Detectors

Syslog ED
SNMP EDs
Timer EDs
none ED
HW EDs
Watchdog ED
Interface Counter ED
XML ED
RPC ED
CLI ED
OR ED
ER Ed
ERT ED
ETT ED
RF ED
GOLD ED
NetFlow ED
IPSLA ED
Route ED
CDP ED
LLDP ED
802.1x ED
MAC ED

Syslog Event
Remote:
• Notification
Local:
• Notification
• Get/Set

• Cron
• Count
down

• Fan
• Temp
• Env
• ...

Process Scheduler Database
Interface Descriptor Blocks

Syslog
email notification
SNMP set
Counter
SNMP get
SNMP
notification
Reload or
switch-over
Application
specific
CLI Applets
IOS.sh
Policies
TCL Policies

SNMP
get
SNMP
notification

Syslog
Event

ED
GOLD
ED
RH
ED

Remote:
• Notification
Local:
• Notification
• Get/Set

• Cron
• Count
down

• Fan
• Temp
• Env
• ...

Process Scheduler Database
Interface Descriptor Blocks

Applications
specific
Reload
or
switch-over
 CLI
Applets
 IOS.sh
Policies
TCL Policies

Syslog
Event

ED
GOLD
ED
EEM Architecture
EEM Applets and Policies

**CLI Applets**
- Part of the Cisco IOS Configuration
- Based on CLI Commands
- Simple Actions
- Programmatic Applet Extensions

**IOS.sh Policies**
- Separate ASCII File *my-policy.sh*
- Based on Cisco IOS CLI and Shell Commands
- Effective shell-like simple scripting
- Registered via the Cisco IOS Config

**TCL Policies**
- Separate ASCII File *my-policy.tcl*
- Based on Cisco IOS CLI and Safe TCL Commands
- Flexible and powerful scripting capabilities
- Registered via the Cisco IOS Config
Embedded monitoring of different components of the system via a set of software agents (event detectors)

Event detectors (ED) notify EEM when an event of interest occurs; based on this, a policy will trigger an action to be taken

Advantages: Local programmable actions, triggered by specific events – growing set of detectors and actions:

- EEM 1.0 introduced in 12.0(26)S, 12.3(4)T
- EEM 2.0 introduced in 12.2(25)S
- EEM 2.1 introduced in 12.3(14)T
- EEM 2.2 introduced in 12.4(2)T
- EEM 2.3 introduced in 12.4(11)T
- EEM 2.4 introduced in 12.4(20)T
- EEM 3.0 introduced in 12.4(22)T
- EEM 3.1 introduced in 15.0(1)M
- EEM 3.2 introduced in 12.2(52)SE
- stay tuned ...

- Adds multi-event correlation
- Adds programmatic Applets
<table>
<thead>
<tr>
<th>Event Detector</th>
<th>Description (ED Triggers, based on ...)</th>
<th>EEM Version in IOS</th>
<th>IOS XR</th>
<th>IOS XE</th>
<th>NX-OS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1.0 2.0 2.1 2.2 2.3 2.4 3.0 3.1 3.2</td>
<td>3.6 4.0</td>
<td>2.1 2.2</td>
<td>4.0 4.1</td>
</tr>
<tr>
<td>Syslog</td>
<td>RegExp match of local syslog message</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNMP Notif</td>
<td>SNMP MIB Variable Threshold</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Watchdog</td>
<td>IOS process or subsystem activity events</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interface Counter</td>
<td>(Interface) Counter Threshold</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Timer</td>
<td>Designated Time or Interval</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Counter</td>
<td>Change of a designated counter value</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application specific</td>
<td>An IOS subsystem or policy script</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLI</td>
<td>RegExp match of input via command line interface</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OIR</td>
<td>Hardware online insertion and removal OIR</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>none</td>
<td>No trigger, used in conjunction with exec command</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERM</td>
<td>Embedded Resource Manager (ERM) events</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EOT</td>
<td>Enhanced Object Tracking variable (EOT) events</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RF</td>
<td>IOS Redundancy Facility (switchover)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GOLD</td>
<td>Generic Online Diagnostics (GOLD) events</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNMP Proxy</td>
<td>Incoming remote SNMP Notification</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XML RPC</td>
<td>Incoming XML message</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Routing</td>
<td>State change of Routing Protocols</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Netflow</td>
<td>Traffic Flow information from Netflow</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPSLA</td>
<td>IPSLA events (supersedes EOT for EEM / IPSLA)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLI enhanced</td>
<td>Integrates CLI Ed with the XML PI</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNMP Object</td>
<td>Intercept SNMP GET/SET requests</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neighbor Disco</td>
<td>CDP, LLPD, Link up/down events</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identity</td>
<td>802.1x and MAB authentication events</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAC</td>
<td>MAC Address Table entry changes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hardware</td>
<td>Register for environmenta monitoring hardware</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Statistics</td>
<td>Threshold crossing of a statistical counter</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sysmgr</td>
<td>Process start and stop events</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fan (absent / bad)</td>
<td>Presence and State of a Fan</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Module failure</td>
<td>Occurrence of a Module Failure Event</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storm Control</td>
<td>Occurrence of a Storm Control Event</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Temperature</td>
<td>Temperature Sensor Thresholds</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**EEM 2.0: EOT Event Detector**

**Problem:** A Notification is required upon failure of a specific route

**Solution:** Track the Route using Enhanced Object Tracking (EOT) and Embedded Event Manager (EEM)

```
track 400 ip route 1.1.1.1/32 reachability
  delay down 10 up 10
!
event manager environment my_server 172.27.121.177
event manager environment my_from router-abc@customer.com
event manager environment my_to attach@cisco.com
event manager environment my_route 1.1.1.1/32
!
event manager applet email_track_iproute
event track 400 state down
action 1.0 syslog msg "Prefix to [$my_route] has been withdrawn!"
action 1.1 mail server "$my_server" to "$my_to" from "$my_from"
  subject "EEM: Prefix to Remote Site [$my_route] is DOWN" body ""
action 1.2 syslog msg "EEM: Path Failure alert email sent!"
```

**Note:** New Routing Event Detector in EEM 3.0
EEM 2.4: Proxy Event Detector

- Router or switch can RECEIVE an SNMP trap
- EEM event upon trap receipt
- Execute (trigger) EEM script to take local action
- Script sees varbind info in trap

Example:
- UPS on battery backup
  ==> Shut non-critical POE ports to conserve power
- Only 5 minutes remaining
  ==> Shutdown service modules gracefully

Example: managed Services
Previous to EEM v2.4, there was a one-to-one correspondence between a single event and the triggered policy.

In other words, a policy could only be triggered by a single event and any event correlation had to be coded by the user.

Multiple Event Support ushers in an event correlation specification such that multiple events may be considered together to trigger a policy.

For example:

If (Event 1 OR Event 2) AND Event 3, then Trigger Policy A
EEM 2.4: Multiple Event Correlation

Problem: A Syslog message is required upon state change of either Ethernet1/0 or Ethernet1/1

Solution: Use Embedded Event Manager (EEM) Multiple Event Correlation with a `correlate` statement within the trigger block to define the logic between individual events and optional `occurs` clauses to define the number of times a specific event must be raised before being used in the correlation (inner level), or the number of times the total correlation must be true before invoking the action (outer level):

```plaintext
event manager applet example
  event tag e1 syslog pattern ".*UPDOWN.*Ethernet1/0.*"
  event tag e2 syslog pattern ".*UPDOWN.*Ethernet1/1.*"
  trigger occurs 1
    correlate event e1 or event e2
    attribute e1 occurs 1
    attribute e2 occurs 1
  action 1.0 syslog msg "Critical interface status change"
  set 2.0 _exit_status 0
```
EEM 3.0: Programmatic Applet Example

event manager applet route-watch
  event routing network 10.1.1.0/24 type add protocol ospf
  action 001 cli command "enable"
  action 002 set done 0
  action 003 while $done eq 0
  action 004 wait 5
  action 005 cli command "ping ip 10.1.1.1"
  action 005 regexp "!!!!!!" "$_cli_result"
  action 006 if $_regexp_result eq 1
  action 007 cli command "config t"
  action 008 cli command "int Tunnel0"
  action 009 cli command "shut"
  action 010 cli command "end"
  action 011 set done 1
  action 012 end
  action 013 end

The applet will trigger when the route 10.1.1.0/24 is learned via OSPF

The applet will try and ping host 10.1.1.1, and when it is successful, it will take down the backup tunnel interface
Example: Integrating CleanAir and Security

**Problem:** A new rogue WLAN device in sensitive areas should be detected by Cisco CleanAir and automatically focus/pan/zoom a security camera.

**Solution:** Use Network Automation based on Cisco IOS Embedded Event Manager to receive an SNMP Notification from WLC and trigger the Video Operations Manager via HTTP.
Using EEM step-by-step

1. Which problem do you want to solve?
2. Which event detector and action do you need?
   - Upgrade to the right IOS image

3. Check whether a suitable script/applet is available already
   - http://www.cisco.com/go/ciscobeyond
   - http://www.cisco.com/go/eem
   - http://www.cisco.com/go/easy

4. Work from an existing example

5. Deploy and Monitor
   - CiscoWorks LMS (from 3.1) via RME
     http://www.cisco.com/go/lms
   - Davra Networks EEMLive
     http://www.davranetworks.com/

6. If customization/new development/testing is required
   - Cisco Advanced Services

7. Don’t forget to ask to (and share with) the EEM forum

show event manager detector <detector-type> detailed
What about the Service?
IP Service Level Agreements (IP SLA)

- Active probing by injecting synthetic test traffic
- Experience and Adoption across markets and technology domains
- Vast range of Cisco and 3rd Party NMS tool support

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Latency</th>
<th>Jitter</th>
<th>Packet Loss</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains</td>
<td>IP</td>
<td>Ethernet</td>
<td>MPLS</td>
<td>VoIP</td>
</tr>
<tr>
<td>Operations</td>
<td>ICMP Echo</td>
<td>ICMP Jitter</td>
<td>UDP PathEcho</td>
<td>TCP Connect</td>
</tr>
<tr>
<td></td>
<td>ICMP PathEcho</td>
<td>UDP Echo</td>
<td>UDP Jitter</td>
<td>802.1ag Echo</td>
</tr>
</tbody>
</table>

See: [www.cisco.com/go/ipsla](http://www.cisco.com/go/ipsla)
IP SLA – ICMP and UDP Jitter Examples

RouterA(config)#

```
ip sla 1
  icmp-echo RouterC
  timeout 500
  frequency 10
ip sla schedule 1 start-time now
```

```
ip sla 10
  udp-jitter RouterD 16384 num-packets 1000 interval 20
  request-data-size 172
  tos 20
  frequency 60
ip sla schedule 10 start-time now
```
**IP SLA – ICMP Echo Operation**

Router#`show ip sla sta mon 1`
Round trip time (RTT)   Index 1
   Latest RTT: 1 ms
Latest operation start time: *05:26:00.226 UTC Fri Jan 4 2008
Latest operation return code: OK
Number of successes: 1
   Number of failures: 0
Operation time to live: 188 sec

Router#`sh ip sla sta 1 detail`
Round trip time (RTT)   Index 1
   Latest RTT: 1 ms
Latest operation start time: *05:26:30.224 UTC Fri Jan 4 2008
Latest operation return code: OK
Over thresholds occurred: FALSE
Number of successes: 2
   Number of failures: 0
Operation time to live: 155 sec
Operational state of entry: Active
Last time this entry was reset: Never
Router#sh ip sla statistics 10
Round trip time (RTT)  Index 10
  Latest RTT: 1 ms
Latest operation start time: *05:43:28.720 UTC Fri Jan 4 2008
Latest operation return code: OK RTT Values
  Number Of RTT: 10
  RTT Min/Avg/Max: 1/1/1 ms
Latency one-way time milliseconds
  Number of one-way Samples: 0
  Source to Destination one way Min/Avg/Max: 0/0/0 ms
  Destination to source one way Min/Avg/Max: 0/0/0 ms
Jitter time milliseconds
  Number of Jitter Samples: 9
  Source to Destination Jitter Min/Avg/Max: 20/20/23 ms
  Destination to Source Jitter Min/Avg/Max: 22/21/24 ms
Packet Loss Values
  Loss Source to Destination: 0    Loss Destination to Source: 0
  Out Of Sequence: 0    Tail Drop: 0    Packet Late Arrival: 0
Number of successes: 1
Number of failures: 0
Operation time to live: 3567 sec
Problem

- Need to monitor IP SLA
- Trigger actions upon violation of SLA

Solutions

- IP SLA Reaction Thresholds
- Using EEM and the EOT Event Detector
- Using EEM 3.x and the IP SLA Event Detector
Solution 1:
IP SLA Reaction Thresholds

RouterA(config) #
ip sla 10
  icmp-echo 3.3.3.3
  frequency 10
ip sla reaction-configuration 10 react timeout threshold-type consecutive 3 action-type trapAndTrigger
ip sla schedule 10 life forever start-time now
ip sla reaction-trigger 10 20

logging on
ip sla logging trap
snmp-server host nms_server version 2c public
snmp-server enable traps syslog

Send an SNMP trap after 3 consecutive timeouts and trigger IP SLA operation 20
Solution 2:
Enhanced Object Tracking and EEM

IP SLA
ip sla 10
icmp-echo 3.3.3.3
timeout 500
frequency 3
ip sla schedule 10 life forever start-time now

Enhanced Object Tracking (EOT)
track 10 rtr 10 reachability
delay down 10 up 20

Environment Variables
($_* variables to be defined)

EEM Applet
event manager applet email_server_unreachable
event track 10 state down
action 1.0 syslog msg "Ping has failed, server unreachable!"
action 1.1 cli command "enable"
action 1.2 cli command "del /force flash:server_unreachable"
action 1.3 cli command "show clock | append server_unreachable"
action 1.4 cli command "show ip route | append server_unreachable"
action 1.5 cli command "more flash:server_unreachable"
action 1.6 mail server "$_email_server" to "$_email_to" from "$_email_from" subject "Server Unreachable: ICMP-Echos Failed" body "$_cli_result"
action 1.7 syslog msg "Server unreachable alert has been sent to email server!"
Solution 3:
IP SLA Event Detector in EEM 3.0

Router(config)# ip sla 10
Router(config-ip-sla)# icmp-echo 3.3.3.3

Router(config)# ip sla enable reaction-alerts

Router(config)#ip sla reaction-config 10 react timeout threshold-type consecutive 3 action-type none

Router(config)# ip sla schedule 10 start now

Router(config)# event manager applet test
router(config-applet)# event ipsla operation-id 10 reaction-type timeout
router(config-applet)# action 1.0 syslog priorities emergencies
    msg “IP SLA operation $_ipsla_oper_id to server XYZ has timed out”

Trigger an Embedded Event Manager Applet after 3 consecutive timeouts of the IP SLA operation
Auto IP SLA – Don’t touch your Hub

Some IP SLA Topologies …

§ … are naturally Hub and Spoke

§ … have a large number of Spokes with similar IP SLA requirements

§ … consist of dynamically joining / disappearing Spokes

```bash
ip sla auto template type ip udp-jitter my-ipsla-template
parameters
request-data-size 64
num-packets 1000
ip sla auto schedule my-ipsla-schedule
frequency 45
start-time now
ip sla auto endpoint-list type ip my-ipsla-endpoints
discover
ageout 36000
ip sla auto group type ip my-ipsla-group
schedule my-ipsla-schedule
template udp-jitter my-ipsla-template
destination my-ipsla-endpoints
```

```bash
ip sla responder auto-register 10.10.10.2 endpoint-list my-ipsla-endpoints
```
EASy Package: Custom High-Availability

**Problem:** We need a failover from primary to secondary link – but with flexibility and custom notification beyond what a simple routing protocol based solution provides

**Solution:** Automate based on IP SLA, EOT and Embedded Event Manager

IP SLA Support in LMS 4.0
IP SLA Support in Unified Operations Manager 8.0

See: www.cisco.com/go/ucmanagement
Who is doing What on the Network
What is NetFlow?

- Developed and patented at Cisco® Systems in 1996
- NetFlow is the defacto standard for acquiring IP operational data
- Provides network and security monitoring, network planning, traffic analysis, and IP accounting
- NetFlow v9 (RFC3954) serves as the basis for IETF IPFIX Standard (RFC5101 & RFC5102)

Network World article – NetFlow Adoption on the Rise:
Flexible NetFlow (FNF)

Traditional NetFlow with the v5, v7, or v8 NetFlow export

NetFlow Version 9 (RFC3954)

Advantages: extensibility
- Integrate new technologies/data types quicker (MPLS, IPv6, BGP next hop, etc.)
- Integrate new aggregations quicker

Basis for IETF IPFIX Standard (RFC5101 & RFC5102)

Flexible NetFlow

Advantages: cache and export content flexibility
- User selection of flow keys
- User definition of the records

Flexible NetFlow
Multiple Monitors with Unique Key Fields

Traffic Analysis Cache

<table>
<thead>
<tr>
<th>Source IP</th>
<th>Dest. IP</th>
<th>Source Port</th>
<th>Dest. Port</th>
<th>Protocol</th>
<th>TOS</th>
<th>Input I/F</th>
<th>...</th>
<th>Pkts</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.3.3</td>
<td>2.2.2.2</td>
<td>23</td>
<td>22078</td>
<td>6</td>
<td>0</td>
<td>E0</td>
<td></td>
<td>1100</td>
</tr>
</tbody>
</table>

Security Analysis Cache

<table>
<thead>
<tr>
<th>Source IP</th>
<th>Dest. IP</th>
<th>Input I/F</th>
<th>Flag</th>
<th>...</th>
<th>Pkts</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.3.3</td>
<td>2.2.2.2</td>
<td>E0</td>
<td>0</td>
<td></td>
<td>1100</td>
</tr>
</tbody>
</table>
Flexible NetFlow
Configuration – Example

1. Configure the Exporter

Router(config)# flow exporter my-exporter
Router(config-flow-exporter)# destination 1.1.1.1

2. Configure the Flow Record

Router(config)# flow record my-record
Router(config-flow-record)# match ipv4 destination address
Router(config-flow-record)# match ipv4 source address
Router(config-flow-record)# collect counter bytes

3. Configure the Flow Monitor

Router(config)# flow monitor my-monitor
Router(config-flow-monitor)# exporter my-exporter
Router(config-flow-monitor)# record my-record

4. Apply to an Interface

Router(config)# interface s3/0
Router(config-if)# ip flow monitor my-monitor input
## Flexible Flow Record: Key Fields

<table>
<thead>
<tr>
<th>Flow</th>
<th>IPv4</th>
<th>IPv6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sampler ID</td>
<td>IP (Source or Destination)</td>
<td>IP (Source or Destination)</td>
</tr>
<tr>
<td>Direction</td>
<td>Payload Size</td>
<td>Payload Size</td>
</tr>
<tr>
<td><strong>Interface</strong></td>
<td><strong>Prefix (Source or Destination)</strong></td>
<td><strong>Prefix (Source or Destination)</strong></td>
</tr>
<tr>
<td>Input</td>
<td><strong>Packet Section (Header)</strong></td>
<td><strong>Packet Section (Header)</strong></td>
</tr>
<tr>
<td>Output</td>
<td><strong>Packet Section (Payload)</strong></td>
<td><strong>Packet Section (Payload)</strong></td>
</tr>
<tr>
<td><strong>Layer 2</strong></td>
<td><strong>Mask (Source or Destination)</strong></td>
<td><strong>Mask (Source or Destination)</strong></td>
</tr>
<tr>
<td>Source VLAN</td>
<td><strong>Packet Section (Payload)</strong></td>
<td><strong>Minimum-Mask (Source or Destination)</strong></td>
</tr>
<tr>
<td>Dest VLAN</td>
<td><strong>TTL</strong></td>
<td><strong>DSCP</strong></td>
</tr>
<tr>
<td>Dot1q VLAN</td>
<td><strong>Options bitmap</strong></td>
<td><strong>Protocol</strong></td>
</tr>
<tr>
<td>Dot1q priority</td>
<td><strong>Version</strong></td>
<td><strong>Extension Headers</strong></td>
</tr>
<tr>
<td>Source MAC address</td>
<td><strong>Precedence</strong></td>
<td><strong>Hop-Limit</strong></td>
</tr>
<tr>
<td>Destination MAC address</td>
<td><strong>DSCP</strong></td>
<td><strong>Flow Label</strong></td>
</tr>
<tr>
<td><strong>NEW</strong></td>
<td><strong>Header Length</strong></td>
<td><strong>Next-header</strong></td>
</tr>
<tr>
<td></td>
<td><strong>TOS</strong></td>
<td><strong>Header Length</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Payload Length</strong></td>
<td><strong>Version</strong></td>
</tr>
</tbody>
</table>
## Flexible Flow Record: Key Fields

<table>
<thead>
<tr>
<th>Routing</th>
<th>Transport</th>
<th>Application</th>
<th>Multicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>src or dest AS</td>
<td>Destination Port</td>
<td>TCP Flag: ACK</td>
<td>Replication Factor*</td>
</tr>
<tr>
<td>Peer AS</td>
<td>Source Port</td>
<td>TCP Flag: CWR</td>
<td>RPF Check Drop*</td>
</tr>
<tr>
<td>Traffic Index</td>
<td>ICMP Code</td>
<td>TCP Flag: ECE</td>
<td>Is-Multicast</td>
</tr>
<tr>
<td>Forwarding Status</td>
<td>ICMP Type</td>
<td>TCP Flag: FIN</td>
<td></td>
</tr>
<tr>
<td>IGP Next Hop</td>
<td>IGMP Type*</td>
<td>TCP Flag: PSH</td>
<td></td>
</tr>
<tr>
<td>BGP Next Hop</td>
<td>TCP ACK Number</td>
<td>TCP Flag: RST</td>
<td></td>
</tr>
<tr>
<td>Input VRF Name</td>
<td>TCP Header Length</td>
<td>TCP Flag: SYN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP Sequence Number</td>
<td>TCP Flag: URG</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP Window-Size</td>
<td>UDP Message Length</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP Source Port</td>
<td>UDP Source Port</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP Destination Port</td>
<td>UDP Destination Port</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP Urgent Pointer</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* IPv4 Flow only
### Flexible Flow Record: Non-Key Fields

#### Counters
- Bytes
- Bytes Long
- Bytes Square Sum
- Bytes Square Sum Long
- Packets
- Packets Long

#### Timestamp
- `sysUpTime First Packet`

#### IPv4
- Total Length Minimum ("
- Total Length Maximum ("
- TTL Minimum
- TTL Maximum

#### IPv4 and IPv6
- Total Length Minimum ("
- Total Length Maximum ("

Plus any of the potential “key” fields: will be the value from the first packet in the flow

(*) IPV4_TOTAL_LEN_MIN, IPV4_TOTAL_LEN_MAX  
(**)IP_LENGTH_TOTAL_MIN, IP_LENGTH_TOTAL_MAX
Service Planning
Flexible NetFlow Top Talkers - Examples

 orbsb Top ten IP addresses that are sending the most packets

Router# show flow monitor <monitor> cache
    aggregate ipv4 source address
    sort highest counter bytes top 10

 orbsb Top five destination addresses to which we're routing most traffic from the 10.10.10.0/24 prefix

Router# show flow monitor <monitor> cache
    filter ipv4 destination address 10.10.10.0/24
    aggregate ipv4 destination address
    sort highest counter bytes top 5

 orbsb 5 VLAN's that we're sending the least bytes to:

Router# show flow monitor <monitor> cache
    aggregate datalink dot1q vlan output
    sort lowest counter bytes top 5

 orbsb Top 20 sources of 1-packet flows:

Router# show flow monitor <monitor> cache
    filter counter packet 1
    aggregate ipv4 source address
    sort highest flow packet top 20
Service Planning
Flexible NetFlow Top Talkers – Example

Router# show flow monitor <monitor> cache
filter ipv4 destination address 10.10.10.0/24
counter packet regex[1-2]
aggregate ipv4 source address
ipv4 destination address
sort highest flow top 100

The top 100 pairs of IP addresses with one or two packet(s) that are destined for my servers' network
Example: Monitor low-TTL Traffic

**Problem**: We want to know about low-TTL traffic

**Solution**: Use Flexible Netflow and Embedded Event Manager 3.0 to detect traffic flows with TTL < 5

1. Configure flexible Netflow to match on TTL, Source- and Destination Address

```plaintext
flow record my-ttl-record
  match ipv4 ttl
  match ipv4 source address
  match ipv4 destination address
:
flow monitor my-ttl-monitor
  record my-record
:
```

2. Configure the Netflow Event Detector in EEM to notify upon a new flow record

```plaintext
event manager applet my-ttl-applet
  event nf monitor-name my-ttl-monitor event-type create event1
    entry-value "5" field ipv4 ttl entry-op lt
  action 1.0 syslog msg "Low-TTL flow from $_nf_source_address"
```

3. Syslog message and/or use show flow monitor my-ttl-monitor cache command

```
```
NAM 5.0 Interactive Reports
Analyze Performance/Usage Trends and Patterns

- Analyze data over last month or more
- Define custom time interval for analysis
- Export data in raw format for consumption by external management application
- Drill-down to analyze related trends to support planning decisions

See: www.cisco.com/go/nam
What if I need a Packet Capture?
Embedded Packet Capture (EPC)

**Problem:** Sometimes a Packet Capture would be useful for Troubleshooting, Security or Application Analysis, Baselining, etc.
BUT: deploying Packet Sniffers are slow, expensive and require local skills and equipment ...

**Solution:** Make use of IOS Embedded Packet Capture to capture PCAP format data and/or analyze on the device

1. Defining a capture buffer on the device
   ```
   Router# monitor capture buffer ...
   ```

2. Defining a capture point
   ```
   Router# monitor capture point ...
   ```

3. Associate capture point to buffer
   ```
   Router# monitor capture point associate ...
   ```

4. Start / Stop capture points
   ```
   Router# monitor capture point start ...
   ```

5. Show and/or Export the content of the buffer
   ```
   Router# monitor capture buffer <tracename> export
   ```

See: [http://www.cisco.com/go/epc](http://www.cisco.com/go/epc)
Available from: IOS 12.4(20)T
Platforms: 8xx, 18xx, 28xx, 38xx ISRs, 72xx
Example: Analyze process-switched traffic

We want to capture process-switched traffic:

1-3. Define a capture buffer, capture point and associate the two

```bash
Router# monitor capture buffer my-buffer size 100 max-size 1000 circular
Router# monitor capture point ip process-switched my-capture in
Router# monitor capture point associate my-capture my-buffer
```

4. Start capturing traffic

```bash
Router# monitor capture point start all
*Nov 25 10:00:58.990: %BUFCAP-6-ENABLE: Capture Point my-capture enabled.
```

5. Show / Analyze on the router …

```bash
Router# show monitor capture buffer my-buffer all parameters
  Capture buffer my-buffer (circular buffer)
  Buffer Size : 102400 bytes, Max Element Size : 1000 bytes, Packets : 28
  Allow-nth-pak : 0, Duration : 0 (seconds), Max packets : 0, pps : 0
  Associated Capture Points:
  Name : my-capture, Status : Active
  Configuration:
  monitor capture buffer my-buffer size 100 max-size 1000 circular
  monitor capture point associate my-capture my-buffer
```

We have some traffic
Off-line Analysis

5. ... or export as PCAP file and analyze externally

Router# monitor capture buffer my-buffer export tftp://10.10.10.10/mypcap
EPC – Additional Considerations

- Capture stop criteria:
  - manual stop
  - after a specified time interval
  - after given number of packets

- Capture point:
  - IPv4 or IPv6
  - CEF (drop, punt) or process switching
  - interface specific or all interfaces
  - Direction: in, out, both, from-us (process-switched specific)
  - multicast: only ingress packets are captured, not the replicated egress packets
  - MPLS: does not capture MPLS encapsulated frames today

- Buffer can be defined as linear or circular

- Buffer filter based on an access-list

  ```bash
  Router# monitor capture buffer my-buffer filter access-list 10
  ```

- Buffer export options: FTP, HTTP, HTTPS, RCP, SCP, or TFTP

**Note:** exec mode commands only, nothing in the configuration
What if I need a Packet Capture?
Diagnosing Transient Problems

Problem: you are seeing VPN tunnel drops on your VPN head-end router at 3:00 am every day. The tunnels continue to flap until the physical interface is reset. You want to analyze the traffic on the wire at that time.
EPC – EASy Package

Embedded Automation Systems (EASy)

EPC EASy Package Supports:
- Interactive Installation
- Timed or manual capture start
- Linear or circular buffer
- Buffer Export

To use the Package:
1. Browse and Download EPC EASy Package
   www.cisco.com/go/easy
2. Make Sure to also download EASy Installer
3. Watch VOD and/or read documentation
   www.cisco.com/go/easy
4. Customize and tailor to your needs
5. Install and Use
What if I need a Packet Capture?
NAM 5.0: Smart Capture Analysis
Highlights observed anomalies in packet traces

NAM enables:
- Packet trace analysis highlighting observed protocol/packet level anomalies
- One-click targeted packet captures
- Combined application visibility, traffic analysis and smart packet capture analysis

NAM benefits:
- Improves operational efficiency with on-demand captures
- Smart analysis pinpoints root-cause much faster than manually analyzing or scanning the packet traces

See: www.cisco.com/go/nam
NAM 5.0: Troubleshooting Workflow
Isolate Source of Application Performance Degradation

1. Analyze application performance over time
2. Zoom to investigate specific performance issues
3. Identify the Top N clients affected by the degradation
4. Isolate the servers with high response time
5. Drill-down to select server to analyze activity
NAM 5.0: WAN Optimization Analysis
Monitor Client Experience and Optimization Improvements

Select Branch Site, Server Site/Server, Application, and Reporting Interval

Analyze performance application traffic (Optimized vs. Passthru)

Examine Traffic Volume (Client, WAN) and achieved Compression Ratio

Examine number of Concurrent Connections (Optimized vs. Passthru)
Summary
You have many tools at your disposal!

- The embedded instrumentation in Cisco devices is an invaluable partner in helping to monitor and troubleshoot the network
- Features such as SNMP, NetFlow, IP-SLA and EPC provide many valuable monitoring and troubleshooting capabilities
- Combining these features with EEM unlocks the power of network automation
- There are many online resources such as EASy and CiscoBeyond to help you get started

- And, … Cisco NMS products such as LMS, NAM and Unified Operations Manager bring these instrumentation features to life
References – Instrumentation

Device Manageability Instrumentation (DMI) [www.cisco.com/go/instrumentation]

- Embedded Event Manager (EEM): [www.cisco.com/go/eem]
- Embedded Packet Capture (EPC): [www.cisco.com/go/epc]
- Flexible NetFlow: [www.cisco.com/go/netflow] and [www.cisco.com/go/fnf]
- IPSLA (formerly SAA, formerly RTR): [www.cisco.com/go/ipsla]
- Network Analysis Module: [http://www.cisco.com/go/nam]
- Unified Operations Manager: [http://www.cisco.com/go/ucmanagement]

- Feature Navigator: [www.cisco.com/go/fn]
- MIB Locator: [www.cisco.com/go/mibs]
Help is just a click away ...

www.cisco.com/go/easy

www.cisco.com/go/instrumentation

www.cisco.com/go/ciscobeyond

www.cisco.com/go/instrumentation

supportforums.cisco.com
See NMS Product Demos at the NMS Booth

Cisco Prime – A Strategy for Innovative Management

LAN Management Solution (LMS)
- Simplified management of borderless networks

Network Analysis Module (NAM)
- Consistent performance visibility across borderless networks

Collaboration Manager
- Manage and troubleshoot video collaboration services

Network Control System
- Converged wired/wireless access management
Q & A
For conference presentations visit:

www.networkerssolutionsforum.com

Please take a moment to complete the Networkers Conference Event Evaluation Form
Thank you.