TOMORROW starts here.

Cisco Connect Praha, Česká republika
24.–25. 3. 2015
Nástroje pro monitoring, řízení a optimalizaci aplikačního provozu v Intelligent WAN architektuře

TECH-WAN3

Radek Boch
Systems Engineer Cisco
rboch@cisco.com
CCIE #7095
Intelligent WAN Solution Components

Application Optimization

- **Transport Independent**
  - Consistent operational model
  - Simple provider migrations
  - Scalable and modular design
  - IPsec routing overlay design

- **Intelligent Path Control**
  - Dynamic Application best path based on policy
  - Load balancing for full utilization of bandwidth
  - Improved network availability

- **Application Optimization**
  - Application visibility with performance monitoring
  - Application acceleration and bandwidth optimization

- **Secure Connectivity**
  - Certified strong encryption
  - Comprehensive threat defense
  - Cloud Web Security for secure direct Internet access
Add Application Visibility
Add Unified Performance Monitor (Cisco AVC)

No Probes
- Deep Packet Inspection
- Passive Monitoring for Voice, Video, Critical apps and best effort apps
- No additional hardware (and included in AX license)

Smart Capacity Planning
- Better use of costly bandwidth
- Per-branch and per-application level reporting

Business Aligned Privacy Enforcement
- No need for complex IP and port ACLs
- See inside HTTP flows to identify specific Cloud applications (HTTP is the new TCP!)

60% of IT Professionals Cite Performance as Key Challenge for Cloud
What is Application Visibility and Control (AVC) Enabled Technologies

Application Recognition
- NBAR2
- Metadata

Perf. Collection & Exporting
- Unified Monitoring
  - Traffic Statistics
  - Response Time
  - Voice/Video Monitoring
  - URL Collection

NFv9/IPFIX

Reporting Tools

Control
- QoS (w/ NBAR2)
- PfR
- WAAS

Management Tool
- Cisco Prime Infrastructure
- 3rd Party Tools

App Visibility & User Experience Report

<table>
<thead>
<tr>
<th>App</th>
<th>BW</th>
<th>Transaction Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP</td>
<td>3M</td>
<td>150 ms</td>
</tr>
<tr>
<td>Sharepoint</td>
<td>10M</td>
<td>500 ms</td>
</tr>
</tbody>
</table>
What do we Want to Monitor?

<table>
<thead>
<tr>
<th>Application Traffic Stats</th>
<th>Conversation Traffic Stats</th>
<th>URL Visibility</th>
<th>Application Response Time</th>
<th>Media Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filters</td>
<td>Filters</td>
<td>Filters</td>
<td>Filters</td>
<td>Filters</td>
</tr>
<tr>
<td>▪ DNS/DHT</td>
<td>▪ Remaining traffic not included in other filters</td>
<td>▪ HTTP Traffic</td>
<td>▪ Selected TCP Applications</td>
<td>▪ RTP Applications</td>
</tr>
<tr>
<td>▪ Traffic statistics per application</td>
<td>▪ Traffic statistics per application, client and server</td>
<td>▪ Sample traffic statistics, TCP performance and host/URL data per connection</td>
<td>▪ Traffic statistics and TCP performance metrics per application, client and server</td>
<td>▪ Traffic statistics and media performance metrics per application, client and server</td>
</tr>
</tbody>
</table>
Collecting Performance Metrics
NetFlow and NBAR Integration

NetFlow
- Monitors data in Layers 2 thru 4
- Determines applications by combination of Port or Port/IP Addressed
- Flow information who, what, when, where

NBAR
- Examines data from Layers 3 thru 7
- Utilizes Layers 3 and 4 plus packet inspection for classification
- Stateful inspection of dynamic-port traffic
- Packet and byte counts
Foundation: Flexible NetFlow (FNF)  
Build Performance Monitoring

Devices

Metering Process  
(Flexible NetFlow Performance Monitor)

Export Process  
(NetFlow v9, IPFIX)

IETF Scope

NMS

Capacity Planning  
Security  
Performance Analysis  
Visibility
Flexible NetFlow (FNF) Record

NetFlow Key Fields vs Non-key Fields

- IPv4 and IPv6 support
- Key fields are unique per flow record (match statement)
- Non-key fields are attributes or characteristics of a flow (collect statement)
- If packet key fields are unique, new entry in flow record is created
  - First packet of a flow will create the Flow entry using the Key Fields™
  - Remaining packets of this flow will only update statistics (bytes, counters, timestamps)
- Otherwise, update the non-key fields, i.e. packet count

### Key Fields for Packet 1

<table>
<thead>
<tr>
<th>Key Fields</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source IP</td>
<td>1.1.1.1</td>
</tr>
<tr>
<td>Destination IP</td>
<td>2.2.2.2</td>
</tr>
<tr>
<td>Source port</td>
<td>23</td>
</tr>
<tr>
<td>Destination port</td>
<td>22078</td>
</tr>
<tr>
<td>Layer 3 Protocol</td>
<td>TCP - 6</td>
</tr>
<tr>
<td>TOS Byte</td>
<td>0</td>
</tr>
</tbody>
</table>

### Non-key Fields for Packet 1

<table>
<thead>
<tr>
<th>Non-key Fields</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>1250</td>
</tr>
</tbody>
</table>

### Key Fields for Packet 2

<table>
<thead>
<tr>
<th>Key Fields</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source IP</td>
<td>3.3.3.3</td>
</tr>
<tr>
<td>Destination IP</td>
<td>4.4.4.4</td>
</tr>
<tr>
<td>Source port</td>
<td>80</td>
</tr>
<tr>
<td>Destination port</td>
<td>22079</td>
</tr>
<tr>
<td>Layer 3 Protocol</td>
<td>TCP - 6</td>
</tr>
<tr>
<td>TOS Byte</td>
<td>0</td>
</tr>
</tbody>
</table>

### Non-key Fields for Packet 2

<table>
<thead>
<tr>
<th>Non-key Fields</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>519</td>
</tr>
</tbody>
</table>
### Traffic Analysis Cache

<table>
<thead>
<tr>
<th>Key Fields</th>
<th>Packet 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source IP</td>
<td>3.3.3.3</td>
</tr>
<tr>
<td>Destination IP</td>
<td>2.2.2.2</td>
</tr>
<tr>
<td>Source Port</td>
<td>23</td>
</tr>
<tr>
<td>Destination Port</td>
<td>22078</td>
</tr>
<tr>
<td>Layer 3 Protocol</td>
<td>TCP - 6</td>
</tr>
<tr>
<td>TOS Byte</td>
<td>0</td>
</tr>
<tr>
<td>Input Interface</td>
<td>Ethernet 0</td>
</tr>
</tbody>
</table>

### Non-Key Fields

<table>
<thead>
<tr>
<th>Non-Key Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packets</td>
</tr>
<tr>
<td>Bytes</td>
</tr>
<tr>
<td>Timestamps</td>
</tr>
<tr>
<td>Next Hop Address</td>
</tr>
</tbody>
</table>

### Security Analysis Cache

<table>
<thead>
<tr>
<th>Key Fields</th>
<th>Packet 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source IP</td>
<td>3.3.3.3</td>
</tr>
<tr>
<td>Destination IP</td>
<td>2.2.2.2</td>
</tr>
<tr>
<td>Input Interface</td>
<td>Gi0/1</td>
</tr>
<tr>
<td>SYN Flag</td>
<td>0</td>
</tr>
</tbody>
</table>

### Non-Key Fields

<table>
<thead>
<tr>
<th>Non-Key Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packets</td>
</tr>
<tr>
<td>Timestamps</td>
</tr>
</tbody>
</table>
Flexible NetFlow Interface Attachment

- Each flow monitor can export to multiple collectors (flow export)
- Each flow monitor can be associated to one set of records (flow record)
- Flow monitor attachment to interface is uni-directional, i.e. to monitor bi-directional traffic, need to apply for both ‘in’ and ‘out’ directions
Flexible NetFlow – Configuration

**Configure the Exporter**

```
flow exporter my-exporter
destination 1.1.1.1
```

**Configure the Flow Record**

```
flow record my-record
match ipv4 destination address
match ipv4 source address
collect counter bytes
```

**Configure the Flow Monitor**

```
flow monitor my-monitor
exporter my-exporter
record my-record
```

**Configure the interface**

```
int s3/0
ip flow monitor my-monitor input
```

**match:** Specify a key field

**collect:** Specify a non-key field
What Does a DoS Attack Look Like?

- Typical DoS attacks have the same (or similar) entries:
  - Input interface, destination IP, one packet per flow, constant bytes per packet (B/Pk)
  - Export to a security-oriented collector: Lancope, Arbor, etc.

```
Router# show ip cache flow
...
SrcIf  SrcIPaddress  SrcP  SrcAS  DstIf  DstIPaddress  DstP  DstAS  Pr  Pkts  B/Pk
29    192.1.6.69     77    aaa   49     194.20.2.2   1308  bbb   6    1    40
29    192.1.6.222   1243   aaa   49     194.20.2.2   1774  bbb   6    1    40
29    192.1.6.108   1076   aaa   49     194.20.2.2   1869  bbb   6    1    40
29    192.1.6.159   903    aaa   49     194.20.2.2   1050  bbb   6    1    40
29    192.1.6.54    730    aaa   49     194.20.2.2   2018  bbb   6    1    40
29    192.1.6.136   559    aaa   49     194.20.2.2   1821  bbb   6    1    40
29    192.1.6.216   383    aaa   49     194.20.2.2   1516  bbb   6    1    40
29    192.1.6.111   45     aaa   49     194.20.2.2   1894  bbb   6    1    40
29    192.1.6.29   1209    aaa   49     194.20.2.2   1600  bbb   6    1    40
```
Useful Fields for Security Monitoring

Attacks that Use Consistent Packet Size or Worms that Use Consistent Packet Size

Several Flows with the Same Fragment Offset: Same Packet Sent Over and Over

Flow Issued From the Same Origin

Very Large Packets or Attacks that Might Always Have The Same Generated Identification
Flexible NetFlow: Key Fields for Security

<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>Class ID</td>
<td>Packet Section (Header)</td>
<td>Packet Section (Header)</td>
</tr>
<tr>
<td></td>
<td>Prefix (Source or Destination)</td>
<td>Prefix (Source or Destination)</td>
</tr>
<tr>
<td></td>
<td>Mask (Source or Destination)</td>
<td>Mask (Source or Destination)</td>
</tr>
<tr>
<td></td>
<td>Packet Section (Payload)</td>
<td>Packet Section (Payload)</td>
</tr>
<tr>
<td></td>
<td>Minimum-Mask (Source or Destination)</td>
<td>Minimum-Mask (Source or Destination)</td>
</tr>
<tr>
<td></td>
<td>TTL</td>
<td>TTL</td>
</tr>
<tr>
<td></td>
<td>Protocol</td>
<td>Protocol</td>
</tr>
<tr>
<td></td>
<td>Options bitmap</td>
<td>Options bitmap</td>
</tr>
<tr>
<td></td>
<td>Version</td>
<td>Version</td>
</tr>
<tr>
<td></td>
<td>Fragmentation Flags</td>
<td>Precedence</td>
</tr>
<tr>
<td></td>
<td>Fragmentation Offset</td>
<td>Identification</td>
</tr>
<tr>
<td></td>
<td>Identification</td>
<td>Header Length</td>
</tr>
<tr>
<td></td>
<td>Header Length</td>
<td>Total Length</td>
</tr>
<tr>
<td></td>
<td>DSCP</td>
<td>DSCP</td>
</tr>
<tr>
<td>Interface</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Layer 2

- Source VLAN
- Dest VLAN
- Dot1q VLAN
- Dot1q priority
- Source MAC address
- Post Source MAC address
- Destination MAC address
- Post Destination MAC address

For your reference
# Flexible NetFlow: Key Fields for Security

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

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

<table>
<thead>
<tr>
<th>Counters</th>
<th>Timestamp</th>
<th>IPv4</th>
<th>IPv4 and IPv6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bytes</td>
<td>sysUpTime First Packet</td>
<td>Total Length Minimum (*)</td>
<td>Total Length Minimum (**)</td>
</tr>
<tr>
<td>Bytes Long</td>
<td>sysUpTime First Packet</td>
<td>Total Length Maximum (*)</td>
<td>Total Length Maximum (**)</td>
</tr>
<tr>
<td>Bytes Square Sum</td>
<td>Absolute first packet</td>
<td>TTL Minimum</td>
<td></td>
</tr>
<tr>
<td>Bytes Square Sum Long</td>
<td>Absolute last packet</td>
<td>TTL Maximum</td>
<td></td>
</tr>
<tr>
<td>Packets</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Packets Long</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bytes replicated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bytes replicated Long</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Packets replicated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Packets Replicated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Packets Replicated Long</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- 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
Cache Types

- **Normal**
  - Similar to today’s NetFlow
  - More flexible active and inactive timers: one second minimum
  - Flow accounts for a single packet
  -Desirable for real-time traffic monitoring, DDoS detection, logging
  -Desirable when only very small flows are expected (ex: sampling)
  -Caution: may result in a large amount of export data
  -To track a set of flows without expiring the flows from the cache
  -Entire cache is periodically exported (update timer)
  -After the cache is full (size configurable), new flows will not be monitored
  -Uses update counters rather than delta counters
  -Exports on a regular basis
  -Used by ART and Media

- **Immediate**
  -Generates the record in the NetFlow cache at the end of a transaction.

- **Permanent**
  -To track a set of flows without expiring the flows from the cache
  -Entire cache is periodically exported (update timer)
  -After the cache is full (size configurable), new flows will not be monitored
  -Uses update counters rather than delta counters

- **Synchronized**
  -Exports on a regular basis
  -Used by ART and Media

- **Transaction-End**
  -A transaction is a set of logical exchanges between endpoints
  -Generates the record in the NetFlow cache at the end of a transaction.
Evolving to Unified Monitoring

- Certain metrics available for certain features.
  - Multiple features to configure
  - Separate provisioning
  - This was the current model for IOS

- All metrics are available within single feature
  - Single provisioning
  - This is the current model for IOS XE
  - This is new in IOS – 15.4(1)T

© 2014 Cisco and/or its affiliates. All rights reserved.
Unified Monitoring with Performance Monitor Provisioning

- Flexible, single monitoring policy for voice/video, application, traffic discovery
- Match traffic to monitor using L3, L4, or L7 information
- IPv4 and IPv6 supported
- Collect only relevant information for each traffic type
- Per traffic type sampling
What do we Want to monitor?

<table>
<thead>
<tr>
<th>Application Traffic Stats</th>
<th>Conversation Traffic Stats</th>
<th>URL Visibility</th>
<th>Application Response Time</th>
<th>Media Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Filters</strong></td>
<td><strong>Filters</strong></td>
<td><strong>Filters</strong></td>
<td><strong>Filters</strong></td>
<td><strong>Filters</strong></td>
</tr>
<tr>
<td>▪ DNS/DHT</td>
<td>▪ Remaining traffic not included in other filters</td>
<td>▪ HTTP Traffic</td>
<td>▪ Selected TCP Applications</td>
<td>▪ RTP Applications</td>
</tr>
<tr>
<td>▪ Traffic statistics per application</td>
<td>▪ Traffic statistics per application, client and server</td>
<td>▪ Sample traffic statistics, TCP performance and host/URL data per connection</td>
<td>▪ Traffic statistics and TCP performance metrics per application, client and server</td>
<td>▪ Traffic statistics and media performance metrics per application, client and server</td>
</tr>
</tbody>
</table>
Traffic Statistics
Application Usage

- Define what is a flow
  - Usual 5 tuples?
  - Aggregation per application?
  - Aggregation per DSCP?
- 2 monitoring processes available
  - Flexible NetFlow
  - Unified Performance Monitor
Choice 1 – Flexible NetFlow
Configure a Flow Record

```
flow record RECORD-FNF
match ipv4 tos
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
match interface input
match flow direction
collect interface output
collect counter bytes long
collect counter packets
!
```

- **Match** = Key Field
- **Collect** = Non Key Field
Configure a Flow Monitor
Details

```
flow monitor <monitor-name>
record <record-name>
exporter <exporter-name>
cache type {normal | immediate | permanent}
cache entries <number-of-entries>
cache timeout {active | inactive | update} <value-in-sec>
statistics packet protocol
statistics packet size
```

- One record only
- Potentially multiple exporters
- 3 types of cache (see next slide)
- Collect Protocol Distribution Statistics
- Collect Size Distribution Statistics
Configure a Flow Exporter

Details

`flow exporter <exporter-name>`

`destination <ipv4-address | ipv6-address> [vrf <vrf-name>]`
`dscp <value>`
`export-protocol [netflow-v5 | netflow-v9 | ipfix ]`
`option { exporter-stats | interface-table | sampler-table | vrf-table | application-table | application-attributes | c3pl-class-table | c3pl-policy-table } timeout <value in sec>`
`source <interface-name>`
`template resend timeout <value in sec>`
`transport udp <destination-port>`
`ttl <value>`
### Flexible NetFlow - NBAR Integration

**Key Fields**

<table>
<thead>
<tr>
<th>Packet #1</th>
<th>Packet #2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Source IP</strong></td>
<td>10.1.1.1</td>
</tr>
<tr>
<td><strong>Destination IP</strong></td>
<td>173.194.34.134</td>
</tr>
<tr>
<td><strong>Source Port</strong></td>
<td>20457</td>
</tr>
<tr>
<td><strong>Destination Port</strong></td>
<td>23</td>
</tr>
<tr>
<td><strong>Layer 3 protocol</strong></td>
<td>6</td>
</tr>
<tr>
<td><strong>TOS byte</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Ingress Interface</strong></td>
<td>Ethernet 0</td>
</tr>
</tbody>
</table>

**Flow record app_record**

- match ipv4 source address
- match ipv4 destination address
- match application name

**NetFlow cache**

<table>
<thead>
<tr>
<th>Src. IP</th>
<th>Dest. IP</th>
<th>Src. Port</th>
<th>Dest. Port</th>
<th>Layer 3 Prot.</th>
<th>TOS Byte</th>
<th>Ingress Intf.</th>
<th>App Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.1.1</td>
<td>173.194.34.134</td>
<td>20457</td>
<td>80</td>
<td>6</td>
<td>0</td>
<td>Ethernet 0</td>
<td>HTTP</td>
</tr>
<tr>
<td>10.1.1.1</td>
<td>72.163.4.161</td>
<td>30307</td>
<td>80</td>
<td>6</td>
<td>0</td>
<td>Ethernet 0</td>
<td>Youtube</td>
</tr>
</tbody>
</table>

First packet of a flow will create the Flow entry using the Key Fields.
Remaining packets of this flow will only update statistics (bytes, counters, timestamps).
Example 2 - Collecting Traffic Statistics
FNF Example – Adding Application Name [1/2]

flow record RECORD-FNF-NBAR
match ipv4 dscp
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match interface input
match flow direction
match application name account-on-resolution
collect interface output
collect counter bytes long
collect counter packets
(..)

flow monitor MONITOR-FNF-NBAR
record RECORD-FNF-NBAR
exporter EXPORTER-CPI

flow exporter EXPORTER-CPI
destination 10.151.1.131
source loopback0
transport udp 9991
option application-table timeout
option application-attributes

- “match application name”: calls NBAR
- “account-on-resolution” (ASR1000): accurate accounting until classification

interface GigabitEthernet0/0/0
ip flow monitor MONITOR-FNF-NBAR input
ip flow monitor MONITOR-FNF-NBAR output

Export the full list of supported applications
Export the full list of attributes per application

For your reference
Example 2 - Collecting Traffic Statistics

FNF Example – Adding Application Name  [2/2]

ASR1#sh flow monitor MONITOR-FNF-NBAR cache format table

- Cache type: Normal (Platform cache)
- Cache size: 200000
- Current entries: 122

- Flows added: 188447
- Flows aged: 188325

- Active timeout (60 secs) 188325

<table>
<thead>
<tr>
<th>IPV4 SRC ADDR</th>
<th>IPV4 DST ADDR</th>
<th>INTF INPUT</th>
<th>FLOW DIRN</th>
<th>IP DSCP</th>
<th>IP PROT</th>
<th>APP NAME</th>
<th>intf output</th>
<th>bytes long</th>
<th>pkts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.1.1</td>
<td>2.1.1.22</td>
<td>Gi0/0/0</td>
<td>Input</td>
<td>0x00</td>
<td>6</td>
<td>cisco facebook</td>
<td>Gi0/0/2</td>
<td>4624</td>
<td>24</td>
</tr>
<tr>
<td>1.1.1.1</td>
<td>2.1.1.24</td>
<td>Gi0/0/0</td>
<td>Input</td>
<td>0x00</td>
<td>6</td>
<td>cisco webex-meeting</td>
<td>Gi0/0/2</td>
<td>10902</td>
<td>48</td>
</tr>
<tr>
<td>1.1.1.34</td>
<td>2.1.1.1</td>
<td>Gi0/0/0</td>
<td>Input</td>
<td>0x00</td>
<td>17</td>
<td>cisco rtp</td>
<td>Gi0/0/2</td>
<td>4560</td>
<td>76</td>
</tr>
<tr>
<td>1.1.1.1</td>
<td>2.1.1.27</td>
<td>Gi0/0/0</td>
<td>Input</td>
<td>0x00</td>
<td>17</td>
<td>port dns</td>
<td>Gi0/0/2</td>
<td>512</td>
<td>8</td>
</tr>
<tr>
<td>2.1.1.1</td>
<td>1.1.1.1</td>
<td>Gi0/0/2</td>
<td>Output</td>
<td>0x00</td>
<td>17</td>
<td>cisco gtalk</td>
<td>Gi0/0/0</td>
<td>302035</td>
<td>2607</td>
</tr>
<tr>
<td>1.1.1.10</td>
<td>2.1.1.1</td>
<td>Gi0/0/0</td>
<td>Input</td>
<td>0x00</td>
<td>6</td>
<td>cisco ssl</td>
<td>Gi0/0/2</td>
<td>1885774</td>
<td>1797</td>
</tr>
<tr>
<td>2.1.1.1</td>
<td>1.1.1.12</td>
<td>Gi0/0/2</td>
<td>Output</td>
<td>0x00</td>
<td>17</td>
<td>cisco active-directory</td>
<td>Gi0/0/0</td>
<td>378</td>
<td>2</td>
</tr>
<tr>
<td>2.1.1.10</td>
<td>1.1.1.11</td>
<td>Gi0/0/2</td>
<td>Output</td>
<td>0x00</td>
<td>6</td>
<td>cisco hulu</td>
<td>Gi0/0/0</td>
<td>32680</td>
<td>52</td>
</tr>
</tbody>
</table>
Choice 2 – Unified Monitor

- Database for historical stats for clients
- Class and flow correlation for all features
- Data Manager supports stats aggregation and complex query
- Threshold monitoring and alert
- Common stats export (V9, SNMP, ConnectedApp API)
- Integration with other infra components
  - ConnectedApps
  - EEM
  - eEdge

Unified Monitoring Infrastructure (performance monitor)

Producer1
- Stats metrics

Producer2
- ART metrics

Producer3
- RTP metrics

Unified Provisioning

Client2

© 2014 Cisco and/or its affiliates. All rights reserved.
Configuration Options

- Configuring records, monitors, class-maps and policy-maps manually (Performance Monitor - Full-featured)

- Configuring records, monitors, class-maps and policy-maps with a Network Management Platforms - ie Cisco Prime Infrastructure

- Configuring with ezPM (Easy Performance Monitor (ezPM) - Simplified method)
Example 4 - Collecting Traffic Statistics
Unified Monitoring Example [1/4]

```plaintext
flow record type performance-monitor conv_ts_ipv4
  match routing vrf input
  match ipv4 protocol
  match application name account-on-resolution
  match connection client ipv4 address
  match connection server ipv4 address
  match connection server transport port
  collect ipv4 dscp
  collect ipv4 ttl
  collect interface input
  collect interface output
  collect timestamp sysuptime first
  collect timestamp sysuptime last
  collect connection new-connections
  collect connection sum-duration
  collect connection server counter bytes long
  collect connection server counter packets long
  collect connection client counter bytes long
  collect connection client counter packets long
```

Similar record definition for IPv6
Example 4 - Collecting Traffic Statistics
Unified Monitoring Example [2/4]

```bash
flow monitor type performance-monitor conv_ts_ipv4
  record conv_ts_ipv4
  history size 1
  cache type synchronized
  cache timeout synchronized 60 export-spread 15
  cache entries 12500!
```

```bash
flow monitor type performance-monitor conv_ts_ipv6
  record conv_ts_ipv6
  history size 1
  cache type synchronized
  cache timeout synchronized 60 export-spread 15
  cache entries 12500
```
Example 4 - Collecting Traffic Statistics
Unified Monitoring Example [3/4]

Define the Traffic you are interested in.
The Performance Monitors will be applied appropriately

```plaintext
class-map match-all conv_ts_ipv4
match access-group name conv_ipv4_tcp
match access-group name conv_ipv4_udp
match protocol ip
!
class-map match-all conv_ts_ipv6
match access-group name conv_ipv6_tcp
match access-group name conv_ipv6_udp
match protocol ipv6

ip access-list extended conv_ipv4_tcp
permit tcp any any
!
ipv6 access-list conv_ipv6_tcp
permit tcp any any
!
ip access-list extended conv_ipv4_udp
permit udp any any
!
ipv6 access-list conv_ipv6_udp
permit udp any any
```

For your reference
Example 4 - Collecting Traffic Statistics
Unified Monitoring Example [4/4]

```plaintext
policy-map type performance-monitor policy-in
  class my-visibility-conv_ts_ipv4
  flow monitor my-visibility-conv_ts_ipv4
  class my-visibility-conv_ts_ipv6
  flow monitor my-visibility-conv_ts_ipv6

Apply the Performance Monitors to the appropriate class and then apply the policy on the interface

Interface GigabitEthernet0/0/0
  service-policy type performance-monitor input policy-in
```

For your reference
AVC Configuration
Prime Infrastructure

- Enable AVC with just ON/OFF button
- With Cisco Prime Infrastructure 2.0
AVC Configuration
Prime AVC One-Click

- Enable AVC in one-click
  - One device at a time
- Two simple steps
  1. Select interface(s)
  2. Enable
Example 5 – Collecting Traffic Statistics

ezPM Fine-grained Monitoring – Configuration

• Records/Monitors/Class-maps/Policy-map pre-defined
• Equivalent to ~650 lines of configuration
Fine-grained Monitoring

Application-Experience Profile (ezPM)

**Application Traffic Stats**
- Filters: DNS/DHT
- Traffic statistics per application

**Conversation Traffic Stats**
- Filters: Remaining traffic not included in other filters
- Traffic statistics per application, client and server

**URL Visibility**
- Filters: HTTP Traffic
- Sample traffic statistics, TCP performance and host/URL data per connection

**Application Response Time**
- Filters: Selected TCP Applications
- Traffic statistics and TCP performance metrics per application, client and server

**Media Performance**
- Filters: RTP Applications
- Traffic statistics and media performance metrics per application, client and server

© 2014 Cisco and/or its affiliates. All rights reserved.
Coarse-grained Monitoring

Application Statistics Profile (ezPM)

- **Application Stats**
  - Traffic statistics per application
  - Per Application and interface statistics

- **Application Client Server stats**
  - Traffic statistics per application, client and server
  - Per Application, interface and client/server statistics (superset of Application Stats)

- Filters
  - Protocol IP

- **Filters**
  - Protocol IP

- Provides the most common functionalities with good performance
  - Optimized for most common use cases
  - capacity planning, media apps etc.

- Supports all applications

- Ease of use
  - Similar statistics available with Flexible NetFlow but much easier with ezPM

- Coarse-grained – when performance is required
  - (high granularity is not)

- Fine-grained – when high granularity is required

© 2014 Cisco and/or its affiliates. All rights reserved.
Coarse-grained vs. Fine-grained Reporting

Configuration Overview – ezPM

Fined-grained Profiles

- application-experience

Traffic Context

- traffic monitor application-response-time
- traffic monitor application-traffic-stats
- traffic monitor conversation-traffic-stats
- traffic monitor media
- traffic monitor url

Coarse-grained Profiles

- application-stats

Attach to the interface

Interface

For your reference
Example 6 – Collecting Traffic Statistics

ezPM example – Coarse-grained Monitoring – Config

• Records/Monitors/Class-maps/Policy-map pre-defined

! User defined ezPM context
performance monitor context my-stats profile application-statistics
exporter destination 10.10.10.10 source GigabitEthernet0/0/1
traffic-monitor application-stats
traffic-monitor application-client-server-stats

! Attach the context to the interface
interface GigabitEthernet0/0/2
performance monitor context my-traffic-stats

• Define the ezPM profile you want to use

• Then choose one of the monitors to apply (application-client-server-stats is a superset of application-stats)
# IOS Implementation

## Recommendation Summary

### Use Case

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Option1 Before IOS 15.4(1)T</th>
<th>Option2 With IOS 15.4(1)T</th>
<th>Option3 15.4(3)T and newer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Usage (per App, per Interface)</td>
<td>Performance Agent</td>
<td>Flexible NetFlow</td>
<td>ezPM Coarse-grained</td>
</tr>
<tr>
<td>Client/Server stats Per App</td>
<td>Performance Agent</td>
<td>Flexible NetFlow</td>
<td>ezPM Coarse-grained</td>
</tr>
<tr>
<td>Application Response Time</td>
<td>Performance Agent</td>
<td>ezPM Fine-grained</td>
<td>ezPM Fine-grained</td>
</tr>
<tr>
<td>Voice and Video Performance</td>
<td>Performance Monitor</td>
<td>ezPM Fine-grained</td>
<td>ezPM Fine-grained</td>
</tr>
</tbody>
</table>
# IOS-XE Implementation

## Recommendation Summary

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Option1</th>
<th>Option2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Usage (per App, per Interface)</td>
<td>Flexible NetFlow</td>
<td>ezPM Coarse-grained</td>
</tr>
<tr>
<td>Client/Server stats Per App</td>
<td>Flexible NetFlow</td>
<td>ezPM Coarse-grained</td>
</tr>
<tr>
<td>Application Response Time</td>
<td>ezPM Fine-grained</td>
<td>ezPM Fine-grained</td>
</tr>
<tr>
<td>Voice and Video Performance</td>
<td>ezPM Fine-grained</td>
<td>ezPM Fine-grained</td>
</tr>
</tbody>
</table>
Application Response Time (ART)
What do we Want to Monitor?

<table>
<thead>
<tr>
<th>Category</th>
<th>Filters</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Traffic Stats</td>
<td>Filters: DNS/DHT</td>
<td>Traffic statistics per application</td>
</tr>
<tr>
<td></td>
<td>Filters: Remaining traffic not included in other filters</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▪ Traffic statistics per application, client and server</td>
<td></td>
</tr>
<tr>
<td>Conversation Traffic Stats</td>
<td>Filters: DNS/DHT</td>
<td>Traffic statistics per application, client and server</td>
</tr>
<tr>
<td></td>
<td>Filters: Remaining traffic not included in other filters</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▪ Traffic statistics per application, client and server</td>
<td></td>
</tr>
<tr>
<td>URL Visibility</td>
<td>Filters: HTTP Traffic</td>
<td>Sample traffic statistics, TCP performance and host/URL data per connection</td>
</tr>
<tr>
<td>Application Response Time</td>
<td>Filters: HTTP Traffic</td>
<td>▪ Traffic statistics and TCP performance metrics per application, client and server</td>
</tr>
<tr>
<td></td>
<td>▪ Selected TCP Applications</td>
<td></td>
</tr>
<tr>
<td>Media Performance</td>
<td>Filters: RTP Applications</td>
<td>▪ Traffic statistics and media performance metrics per application, client and server</td>
</tr>
</tbody>
</table>
Application response time provides insight into application behavior (network vs server bottleneck) to accelerate problem isolation.

- Separate application delivery path into multiple segments.
- Server Network Delay (SND) approximates WAN Delay.
- Latency per application.
Understand IOS ART Metrics Calculation

Network Delay (ND)
ND = CND + SND

Response Time (RT)
t(First response pkt) – t(Last request pkt)

Transaction Time (TT)
t(Last response pkt) – t(First request pkt)

Application Delay (AD)
AD = RT – SND

For your reference

Quantify User Experience

Identify Server Performance Issue
Media Monitoring
### What do we Want to Monitor?

<table>
<thead>
<tr>
<th>Category</th>
<th>Filters</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Traffic Stats</td>
<td>DNS/DHT</td>
<td>Traffic statistics per application</td>
</tr>
<tr>
<td>Conversation Traffic Stats</td>
<td>Remaining traffic not included in other filters</td>
<td>Traffic statistics per application, client and server</td>
</tr>
<tr>
<td>URL Visibility</td>
<td>HTTP Traffic</td>
<td>Sample traffic statistics, TCP performance and host/URL data per connection</td>
</tr>
<tr>
<td>Application Response Time</td>
<td>Selected TCP Applications</td>
<td>Traffic statistics and TCP performance metrics per application, client and server</td>
</tr>
<tr>
<td>Media Performance</td>
<td>RTP Applications</td>
<td>Traffic statistics and media performance metrics per application, client and server</td>
</tr>
</tbody>
</table>
Media Monitoring / Performance Monitor
Active versus Passive Monitoring

Active Monitoring

Router 1
IPSLA Sender

Active Probing

Router 2
IPSLA Responder

Passive Monitoring

Flexible Netflow

Flow Record

Enhanced RTP and TCP metrics reporting
Filtering and classification (based on existing C3PL model)

Flow Record
Exporting with NetFlow v9 or IPFIX
Foundation: Flexible NetFlow (FNF)
Exporting Process: NetFlow v9 and IPFIX

Static Flow Export Format

- **NetFlow Version 5**
  - Fixed number of fields (18 fields)
    - e.g. source/destination IP & port, input/output interfaces, packet/byte count, ToS

Flexible & Extensible Flow Export Format

- **NetFlow v9 / IPFIX**
  - Users define flow record format
  - Flow format is communicated to collector
NetFlow v9 Export Packet

- FlowSet
  - ID = 0 reserved for template flowset
  - IS = 1 reserved for data flowsets
- Matching ID numbers are the way to associate template to the data records
- The header follows the same format as prior NetFlow versions so collectors will be backward compatible
- Each data record represents one flow
- If exported flows have different fields, they cannot be contained in the same template record (i.e., BGP next hop cannot be combined with MPLS-aware, NetFlow records)
IPFIX Export Packet

- Set
  - ID = 2 reserved for template flowset
  - IS = 3 reserved for data flowsets
- Matching ID numbers are the way to associate template to the data records
- The header follows the same format as prior NetFlow versions so collectors will be backward compatible
- Each data record represents one flow
- If exported flows have different fields, they cannot be contained in the same template record (i.e., BGP next hop cannot be combined with MPLS-aware, NetFlow records)
Option Templates

- Use for exporting non-traffic related information to NetFlow collector or reporting tools.
- Available only with Flexible NetFlow

```plaintext
flow exporter MYEXPORTER
destination 10.35.89.59
source GigabitEthernet0/0/1
transport udp 2055
option interface-table timeout 300
option sampler-table timeout 300
option application-table timeout 300
```

```
router#show flow exporter MYEXPORTER templates
Flow Exporter insight:
  Client: Option options interface-table
  Exporter Format: NetFlow Version 9
Template ID    : 256
Source ID      : 6
Record Size    : 104
Template layout
---------------------------------------------------
|       Field           |  Type | Offset |  Size  |
---------------------------------------------------
| v9-scope system       |     1 |     0  |     4  |
| interface input snmp  |    10 |     4  |     4  |
| interface name        |    82 |     8  |    32  |
| interface description |    83 |    40  |    64  |
```

© 2014 Cisco and/or its affiliates. All rights reserved.
# Available Option Templates

<table>
<thead>
<tr>
<th>Option Template</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>application-table</td>
<td>NBAR Application ID to name mapping</td>
</tr>
<tr>
<td>application-attributes</td>
<td>Application attributes definition per application</td>
</tr>
<tr>
<td>c3pl-class-table</td>
<td>QoS class-map ID to name mapping</td>
</tr>
<tr>
<td>c3pl-policy-table</td>
<td>QoS policy-map ID to name mapping</td>
</tr>
<tr>
<td>exporter-stats</td>
<td>Exporter Statistics Option</td>
</tr>
<tr>
<td>interface-table</td>
<td>Interface SNMP ifIndex to name mapping</td>
</tr>
<tr>
<td>Sampler-table</td>
<td>Export Sampler Option</td>
</tr>
<tr>
<td>sub-application-table</td>
<td>NBAR Sub-application ID to name mapping</td>
</tr>
<tr>
<td>vrf-table</td>
<td>VRF ID to name mapping</td>
</tr>
<tr>
<td>queue-id (hidden)</td>
<td>Queue index and queue drop information</td>
</tr>
</tbody>
</table>

Note: Check the IOS release for exact support
Understanding AVC - reference:
Hierarchical Quality of Service (HQoS)
Intelligent WAN Solution Components
Application Optimization - HQoS

- **Transport Independent**
  - Consistent operational model
  - Simple provider migrations
  - Scalable and modular design
  - IPsec routing overlay design

- **Intelligent Path Control**
  - Dynamic Application best path based on policy
  - Load balancing for full utilization of bandwidth
  - Improved network availability

- **Application Optimization**
  - Application visibility with performance monitoring
  - Application acceleration and bandwidth optimization

- **Secure Connectivity**
  - Certified strong encryption
  - Comprehensive threat defense
  - Cloud Web Security for secure direct Internet access

© 2014 Cisco and/or its affiliates. All rights reserved.
WAN Circuits

- Slow speed (≤ 768 kbps)
  - Service Providers no longer offer these

- Medium speed (≥ 1 Mbps to < 100 Mbps)
  - Use hierarchical policies for sub-line-rate Ethernet connections to provide shaping and CBWFQ/LLQ
  - Use software based routers

- High speed (100 Mbps to 10 Gbps)
  - Use hardware queuing via ASR1000 and ISR4451-X
# WAN Queue Mapping

<table>
<thead>
<tr>
<th>Applications</th>
<th>Classes</th>
<th>DSCP</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF, BGP, HSRP</td>
<td>Network-Control</td>
<td>CS6, CS7</td>
<td>5-10%</td>
</tr>
<tr>
<td>Voice</td>
<td>Voice</td>
<td>EF</td>
<td>PQ (Police 20%)</td>
</tr>
<tr>
<td>HD Video Conference</td>
<td>HD-Video</td>
<td>CS4</td>
<td>PQ (Police 20%)</td>
</tr>
<tr>
<td>SD Video Conference</td>
<td>SD-Video</td>
<td>AF41</td>
<td>15%</td>
</tr>
<tr>
<td>Voice/Video Signaling</td>
<td>Signaling</td>
<td>CS3</td>
<td>5-10%</td>
</tr>
<tr>
<td>SSH, Telnet</td>
<td>Management</td>
<td>AF32</td>
<td>3-5%</td>
</tr>
<tr>
<td>DLSW, TACACs, SYSLOG, Radius</td>
<td>Business-CTRL</td>
<td>CS2</td>
<td>10-15%</td>
</tr>
<tr>
<td>Oracle, Citrix, VDI</td>
<td>Business</td>
<td>AF21</td>
<td>10-15%</td>
</tr>
<tr>
<td>TFTP, FTP</td>
<td>Bulk</td>
<td>AF11</td>
<td>5%</td>
</tr>
<tr>
<td>Default</td>
<td>Class-default</td>
<td>All Others</td>
<td>15-25%</td>
</tr>
</tbody>
</table>
WAN Queuing Configuration
Classification

class-map match-all Voice-Bearer
  match dscp ef
class-map match-all Voice-Signaling
  match dscp cs3 af31
class-map match-all Network-control
  match dscp cs6
class-map match-all HD-Video
  match dscp cs4
class-map match-all SD-Video
  match dscp af41 af42 af43
class-map match-all Business
  match dscp cs2 af21 af22 af23
class-map match-all Bulk
  match dscp af11 af12 af13

This is an 8-Class Model which would be remarked to match the MPLS Providers 4 or 6 Classes of Service
WAN Queuing Configuration
150Mbps Hierarchical Policy-Map (ASR1k)

```
policy-map wan_remaining%
class Voice-Bearer
  priority percent 25
class HD-Video
  priority percent 20
class Network-control
  bandwidth remaining percent 15
  queue-limit 100
class Voice-Signaling
  bandwidth remaining percent 15
  queue-limit 100
class SD-Video
  bandwidth remaining percent 20
  queue-limit 200
class Business
  bandwidth remaining percent 15
  queue-limit 250
class Bulk
  bandwidth remaining percent 10
  queue-limit 200
class class-default
  bandwidth remaining percent 25
  queue-limit 400
  random-detect dscp-based
  random-detect dscp 0 200 275
```

```
policy-map Shape_150M
class class-default
  shape average 150000000 600000 0
  service-policy wan_remaining%
```

```
interface Gig x/y
description **** CIR = 150Mbps ****
bandwidth 150000
service-policy output Shape_150M
```

Default queue-limits tend to be too shallow and need to be increased, 64 packets on ISR, and 50ms on ASR1000

Default WRED thresholds may need to be increased, they will not adjust automatically

The Bc (bits) needs to be tuned to minimize the queuing delay and jitter for realtime traffic. Use the following formula to tune the Bc based on a Tc = 4ms.

\[ Bc = \text{Shape Rate (bps)} \times 0.004 \text{ (sec)} \]

\[ Bc = 0 \]
Per-Tunnel Quality of Service
The Need for QoS

- QoS is crucial on DMVPN for:
  - Sharing network bandwidth
  - Marshaling bandwidth usage of applications
  - Meeting application latency & speed requirements
- The “greedy spoke” problem:

  - Packets are lost, AND other spokes are starved

  - Most common problem
Per-Tunnel QoS Groups

- Spokes register as part of a specific group during NHRP registration
- Each spoke tunnel inherits the QoS policy for the corresponding group

```
interface Tunnel0
    nhrp map group gold service-policy output gold
    nhrp map group silver service-policy output silver

policy-map gold
    class class-default
    ! offer 5Mbps to each spoke in the group
    shape average 5000000

policy-map silver
    class class-default
    ! offer 1Mbps to each spoke in the group
    shape average 1000000
```

```
hub# sh ip nhrp group-map
Interface: Tunnel0
NHRP group: gold
QoS policy: gold
Tunnels using the QoS policy:
Tunnel destination overlay/transport address
10.0.0.1/172.16.1.1

NHRP group: silver
QoS policy: silver
Tunnels using the QoS policy:
Tunnel destination overlay/transport address
10.0.0.2/172.16.2.1
10.0.0.3/172.16.3.1
```
Hierarchical Shaper

- Tunnel bandwidth parent policy
  - Each tunnel is allocated a maximum bandwidth
  - A shaper provides the backpressure mechanism
- Protected packets processed by the client policy
- Reserved bandwidth, LLQ, etc.

---

Use “set dscp tunnel” to remark outer header to match MPLS provider Class of Service without remarking inner packet (end to end Campus QoS intact).

```plaintext
class-map control
  match precedence 6
! class-map voice
  match precedence 5
! policy-map sub-policy
  class control
    bandwidth 20
    set dscp tunnel af41
  class voice
    priority percent 60
    set dscp tunnel ef
! policy-map gold
  class class-default
    shape average 5000000
    service-policy sub-policy
! policy-map silver
  class class-default
    shape average 1000000
    service-policy sub-policy
```
Cisco WAN optimization solution
Intelligent WAN Solution Components

Application Optimization

- **Transport Independent**
  - Consistent operational model
  - Simple provider migrations
  - Scalable and modular design
  - IPsec routing overlay design

- **Intelligent Path Control**
  - Dynamic Application best path based on policy
  - Load balancing for full utilization of bandwidth
  - Improved network availability

- **Application Optimization**
  - Application visibility with performance monitoring
  - Application acceleration and bandwidth optimization

- **Secure Connectivity**
  - Certified strong encryption
  - Comprehensive threat defense
  - Cloud Web Security for secure direct Internet access

© 2014 Cisco and/or its affiliates. All rights reserved.
Cisco WAAS: WAN Optimization Deployment

- **Branch Office**: WAAS Express
- **Branch Office**: WAAS Service Module/UCSe
- **Regional Office**: WAAS Appliance
- **Regional Office**: WAAS-XE on 4451

**Virtual Private Cloud**
- vWAAS WAE
- Nexus 1000v VSM
- UCS/x86 Server
- FC SAN

**Data Center or Private Cloud**
- AppNav + WAAS
- VMware ESXi Server
- Nexus 1000v VSM
- vPATH
- FC SAN

© 2014 Cisco and/or its affiliates. All rights reserved.
WAAS Enables High Performance for Centralized Applications Delivery

Email
- 24X Faster

File
- 17X Faster

MS Sharepoint
- 30X Faster

VDI (Citrix)
- 3-8X Faster

© 2014 Cisco and/or its affiliates. All rights reserved.
How WAAS works
Building Blocks of WAAS
DRE Pattern Matching

DRE Signature Database

DRE Byte Cache

Original Message

Encoded Message

© 2014 Cisco and/or its affiliates. All rights reserved.
as latency increases, throughput drops (a lot)
TCP Flow Optimization

TFO throughput improvement over standard TCP

slow start
Congestion avoidance
Application Optimizers

- CIFS/SMB
  - (includes print services)
- Exchange Optimization
- HTTP
- HTTPS
- NFS
- Citrix ICA
- Akamai
Cisco and Akamai Solution
How the Akamai Platform Works

- Distributed cloud platform, on-demand scale
- 750 Cities, 80 Countries, 2K locations, 130k Servers, 1100 Networks
- A single network hop from 90% of internet users
- Delivering 15-30% of all daily web traffic
- 19M hits per sec, 2 trillion cloud interactions daily
- 150M mobile apps delivered daily
- Defending against attacks over 500Gbps

Web-enabled Applications
Mobile Applications
IP Applications

Edge Region close to Origin Server
High Performance Global Overlay Network
Edge Region close to End User

“SureRoute” and Akamai Protocol optimize route and reduce round trips

Security embedded into Akamai Edge Servers
Extending Akamai to the Branch with Akamai Connect
Akamai Intelligence Inside Cisco ISR-AX

COMPLETING THE LAST MILE

Branch

ISR-AX

Akamai Connect

Data Center

WAN/MPLS

Internet

Akamai Intelligent Platform

Optimal Experience Regardless of Device, Connectivity or Cloud

All HTTP Traffic in Private, Public, Akamai Cloud

Prepositioning | Dynamic HTTP Caching (YouTube) | Any Transport

© 2014 Cisco and/or its affiliates. All rights reserved.
## Akamai Caching Technology

### Transparent Caching: with four (4) different mode settings

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
</table>
| **BASIC**  | - Follows IETF HTTP 1.1 guidelines for standard object caching  
- Only caches responses marked explicitly as cacheable                                          |
| **STANDARD** | - Default mode  
- Also caches objects with no explicit cache marker and with a last-modified date. It ignores “reload” headers from clients |
| **ADVANCED** | - Caches media files more aggressively, and all object types for longer times (when there is no explicit expiration time)                 |
| **BYPASS**  | - Turns off caching for a configured site(s)                                                                                               |

### Connected Cache (CC): Retrieves content from Akamai’s Intelligent Platform

- **Connected Cache (CC):** Retrieves content from Akamai’s Intelligent Platform
  - **Over-the-Top Caching (OTT):** Caching content of 3rd party Web sites using a predefined configuration
  - **Cache Warming or Prepositioning:** Scheduled fetch and cache of content from a Web site

© 2014 Cisco and/or its affiliates. All rights reserved.
Akamai Connected Cache

Edge Caching — Cloud Platform

- Akamaized Content has special caching rules
- Greatly improves page loading performance
- General Internet content is not cached in cloud platform
- Benefit today ends at nearest edge server to datacenter or directly connected user
Akamai Connected Cache

HTTP Akamaized Web Content

Edge Caching — Akamai Connect

- With Akamai Connect caching no longer ends at nearest edge server and extends to the Branch/DC
- Akamai Connect uses an authenticated Akamai service to automatically get up to date caching rules
Seamless Management—Immediate Time to Value
Akamai Connect Delivered Through WAAS Central Manager

One Click Enablement

Easy Scheduling
for content pre-positioning

Visualization and Reporting
using WAAS Central Manager
Deploying WAAS as part of IWAN solution
In-Path, Out of Path
ISR 4xxx Branch Use-Case
Integrated Branch

- Needs: High performance branch with Integrated WAAS
- Solution: ISR 4xxx with fully integrated WAAS
- Benefit: Simple deployment. No WCCP, no external components
**Challenges/Needs:** Dual router, asymmetrical in the branch, router redundancy

**Solution:** ISR4xxx integrated AppNav and WAAS, form single cluster

**Benefit:** Eliminate complex configuration with integrated AppNav
Cisco AppNav Virtualization Technology

Abstraction

✓ Mix form factors
✓ Decouple from topology

Partitioning

✓ Traffic classification
✓ Logical groupings
✓ App, Branch, server based

Elasticity

✓ Dynamic resource creation
✓ Load-based response
✓ Cloudy! oooh!
A Day In The Life Of A Flow

To Branches

SC-1  SC-2

SN-1  SN-2

To servers

For your reference
• SC receives a TCP SYN packet, called green flow, from one of the branches containing a WAAS device.

• The SE classifies green flow and a pending entry is made into the flow database selecting SN-1.
A Day In The Life Of A Flow

To Branches

• SC receives a TCP SYN packet, called green flow, from one of the branches containing a WAAS device.

• The SE classifies green flow and a pending entry is made into the flow database selecting SN-1.

• The frame is encapsulated and is transmitted to SN-1. SN-1 processes the frame and returns an indication the green flow should continue to be intercepted.
• SC receives a TCP SYN packet, called green flow, from one of the branches containing a WAAS device.

• The SE classifies green flow and a pending entry is made into the flow database selecting SN-1.

• The frame is encapsulated and is transmitted to SN-1. SN-1 processes the frame and returns an indication the green flow should continue to be intercepted.

• The other SC(s) are updated with the flow information and the frame is transmitted to destination.
A Day In The Life Of A Flow

SC receives a TCP SYN packet, called green flow, from one of the branches containing a WAAS device.

The SE classifies green flow and a pending entry is made into the flow database selecting SN-1.

The frame is encapsulated and is transmitted to SN-1. SN-1 processes the frame and returns an indication the green flow should continue to be intercepted.

The other SC(s) are updated with the flow information and the frame is transmitted to destination.

A TCP SYN-ACK frame is returned from the destination device and in this example goes to SC-1. SC-1 checks database, finds flow and sends the response frame to SN-1.
A Day In The Life Of A Flow

SC receives a TCP SYN packet, called green flow, from one of the branches containing a WAAS device.

The SE classifies green flow and a pending entry is made into the flow database selecting SN-1.

The frame is encapsulated and is transmitted to SN-1. SN-1 processes the frame and returns an indication the green flow should continue to be intercepted.

The other SC(s) are updated with the flow information and the frame is transmitted to destination.

A TCP SYN-ACK frame is returned from the destination device and in this example goes to SC-1. SC-1 checks database, finds flow and sends the response frame to SN-1.

SN-1 processes the frame and returns it to SN-1 which in turn forwards the frame to the destination device.
IWAN Flows with WAAS

Overview

Before WAAS
- Ingress FNF on all interfaces is sufficient
- LAN in traffic = WAN out traffic
- WAN In traffic = LAN out traffic

After WAAS (with offpath redirection)
- Ingress FNF on all interfaces will give wrong results
  - LAN in traffic > WAN out traffic
  - LAN out traffic > WAN in traffic
- WAAS requires FNF on both ingress and egress of the same interfaces
WAAS Segments

- **Client-side Un-optimized (Segment 1)**
- **Server-side Optimized (Segment 2)**
- **Client-side Optimized (Segment 4)**
- **Servers-side Un-optimized (Segment 8)**

Connection types:
- **Optimized Connection**
- **Pass-thru & Non-optimized Connection**
- **Pass-through (Segment 16)**
- **No WAAS (Segment 0)**
PA Monitoring and Export with WAAS

- With WAAS, ART monitors both Un-optimized and Optimized segments.
- Each device (branch and headend) exports two records per TCP connection.

### ART Unoptimized

<table>
<thead>
<tr>
<th>Segment ID</th>
<th>Src IP</th>
<th>Dst IP</th>
<th>Dst Port</th>
<th>Protocol Type</th>
<th>Resp Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.1.1.1</td>
<td>2.2.2.2</td>
<td>80</td>
<td>6 (TCP)</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>1.1.1.1</td>
<td>2.2.2.2</td>
<td>80</td>
<td>6 (TCP)</td>
<td>100</td>
</tr>
</tbody>
</table>

### ART Optimized

- With WAAS, a TCP connection between client and server is split into 3 TCP connections.
Need to decide where is the best place to run NBAR
Running NBAR on the WAN side is not desirable because NBAR will see compressed traffic
Where should I run NBAR if I want application-aware QoS when WAAS is present?
ANS = On the LAN Side
How AppNav Addresses NBAR2 & WAAS Interop

Overview

If NBAR is enabled on WAN interface, and WAAS is enabled, automatically run NBAR on Uncompress Virtual Interface

© 2014 Cisco and/or its affiliates. All rights reserved.
Summary
IWAN building blocks

• Application Visibility
  • Traffic
  • Response time
  • Multimedia quality

• HQoS

• WAN optimization
  • Latency
  • Cisco WAAS
  • Akamai integration

• Easy Monitoring
Jak využít NetFlow?
Řešení FlowMon, ukázky použití

Petr Špringl
springl@invea.com
Hlavní využití NetFlow

• Viditelnost do sítě, objem a struktura provozu, reporting
  ▪ Flow Monitoring

• Automatická detekce bezpečnostních a provozních incidentů
  ▪ Network Behavior Analysis

• Sledování výkonnostních ukazatelů
  ▪ Network Performance Monitoring

• Ochrana proti DoS/DDoS útokům
Mám vyřešený dohled dostupnosti serverů, přes SNMP získávám informace o komunikaci na síti.

Mám nasazený firewall/IDS, blokuji nežádoucí komunikaci a umím detekovat malware.

Mám definovanou bezpečnostní politiku a podle ní nastavené prostředí.

Tak na co potřebuji další nástroj?
Flow vs. SNMP

Flow Monitoring

- Kdo, s kým, co, jak dlouho komunikoval, kolik přeneseno dat
- Kompletní přehled o dění v síti včetně struktury provozu
- Možnost rozkrýt až na úroveň jednotlivých komunikací
- Efektivní troubleshooting - umožní vyřešit většinu síťových problémů
- Navíc:
  - Informace pro optimalizaci sítě
  - Bezpečnostní monitoring
  - Performance monitoring
  - Sledování trendů
  - SLA

SNMP

- Kolik paketů, kolik bajtů proteklo přes daný port
- Pouze základní informace o množství provozu
- Nevím o jaký provoz se jedná, vím jen jeho množství
- Základní troubleshooting – teče/neteče/kolik
• Open-source i komerční nástroje pro získávání informací ze SNMP
  ▪ Objem přenesených dat
  ▪ Počet paketů
  ▪ A tím to končí ...
NetFlow – Flow Monitoring

• Na první pohled stejné výstupy
  ▪ Bits/s, packets/s, flows/s
  ▪ Kde je ten rozdíl?
NetFlow – Flow Monitoring

• Co je tohle za špičku?
NetFlow – Flow Monitoring

• Drill-down na základě libovolných atributů
• Ukaž mi to 10 IP konverzací dle objemu dat
NetFlow – Flow Monitoring

• Jaká se tam vyskytovala doménová jména?
• Ukaž mi, kam se přistupovalo v rámci play.cz
NetFlow – Flow Monitoring

- Jde o poslech rádia impuls v kvalitě 128kbps
- Mám všechny detaily vč. URL
- A můžu mít detailů ještě víc
NetFlow – Flow Monitoring

- Kompletně L3 a L4
- MAC adresy z L2, vybrané informace L7

<table>
<thead>
<tr>
<th>Start Time - first seen</th>
<th>Duration</th>
<th>IP Protocol</th>
<th>Source IP address</th>
<th>Source Port</th>
<th>Destination IP address</th>
<th>Destination Port</th>
<th>TCP Flags</th>
<th>Input Src Mac Addr</th>
<th>Output Dst Mac Addr</th>
<th>Src VLAN label</th>
<th>TOS</th>
<th>Packets</th>
<th>Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014-06-18 08:39:33.843</td>
<td>299.978</td>
<td>TCP</td>
<td>.cz</td>
<td>50177</td>
<td>89.233.146.155</td>
<td>80</td>
<td>A....</td>
<td>ZyWALL USG 100</td>
<td>00:0c:42:14:50:f2</td>
<td>5</td>
<td>0</td>
<td>3318</td>
<td>132864</td>
</tr>
<tr>
<td>2014-06-18 08:44:33.945</td>
<td>299.971</td>
<td>TCP</td>
<td>.cz</td>
<td>50177</td>
<td>89.233.146.155</td>
<td>80</td>
<td>A....</td>
<td>ZyWALL USG 100</td>
<td>00:0c:42:14:50:f2</td>
<td>5</td>
<td>0</td>
<td>3339</td>
<td>133560</td>
</tr>
<tr>
<td>2014-06-18 08:49:34.046</td>
<td>299.976</td>
<td>TCP</td>
<td>.cz</td>
<td>50177</td>
<td>89.233.146.155</td>
<td>80</td>
<td>A....</td>
<td>ZyWALL USG 100</td>
<td>00:0c:42:14:50:f2</td>
<td>5</td>
<td>0</td>
<td>3369</td>
<td>135012</td>
</tr>
<tr>
<td>2014-06-18 08:54:34.100</td>
<td>299.928</td>
<td>TCP</td>
<td>.cz</td>
<td>50177</td>
<td>89.233.146.155</td>
<td>80</td>
<td>A....</td>
<td>ZyWALL USG 100</td>
<td>00:0c:42:14:50:f2</td>
<td>5</td>
<td>0</td>
<td>3355</td>
<td>134380</td>
</tr>
<tr>
<td>2014-06-18 08:59:34.226</td>
<td>299.981</td>
<td>TCP</td>
<td>.cz</td>
<td>50177</td>
<td>89.233.146.155</td>
<td>80</td>
<td>A....</td>
<td>ZyWALL USG 100</td>
<td>00:0c:42:14:50:f2</td>
<td>5</td>
<td>0</td>
<td>3340</td>
<td>133732</td>
</tr>
<tr>
<td>2014-06-18 09:04:34.285</td>
<td>299.963</td>
<td>TCP</td>
<td>.cz</td>
<td>50177</td>
<td>89.233.146.155</td>
<td>80</td>
<td>A....</td>
<td>ZyWALL USG 100</td>
<td>00:0c:42:14:50:f2</td>
<td>5</td>
<td>0</td>
<td>3494</td>
<td>139868</td>
</tr>
</tbody>
</table>
NetFlow - bezpečnost (NBA)
Jak vypadal provoz naší sítě během jednoho dne? Je všechno v souladu s naším očekáváním?
Podívejme se na špičku. To bude v pořádku, IT oddělení provádí migraci dat, nic zvláštěního.

Opravdu?
Nebo bychom měli prozkoumat provoz z jiného úhlu pohledu?
Podívejme se na provoz z pohledu bezpečnostní analýzy, co je červená špička kolem jedné ráno?

**Traffic by flows**

<table>
<thead>
<tr>
<th>Time</th>
<th>Flows/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>1:00</td>
<td>1.5</td>
</tr>
<tr>
<td>3:00</td>
<td>1.0</td>
</tr>
<tr>
<td>5:00</td>
<td>0.5</td>
</tr>
<tr>
<td>7:00</td>
<td>0.0</td>
</tr>
<tr>
<td>9:00</td>
<td>0.0</td>
</tr>
<tr>
<td>11:00</td>
<td>0.0</td>
</tr>
<tr>
<td>13:00</td>
<td>0.0</td>
</tr>
<tr>
<td>15:00</td>
<td>0.0</td>
</tr>
<tr>
<td>17:00</td>
<td>0.0</td>
</tr>
</tbody>
</table>

**Traffic statistics** (2013-05-08 00:40 - 2013-05-08 01:50)

<table>
<thead>
<tr>
<th>Priority</th>
<th>Flows</th>
<th>Average flows/s</th>
<th>Byte/s</th>
<th>Average byte/s</th>
<th>Packets</th>
<th>Average packets/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>High priority</td>
<td>744.0</td>
<td>0.177</td>
<td>7.7 Gb</td>
<td>1.9 Mb/s</td>
<td>0.2 M</td>
<td>2.0 k/s</td>
</tr>
<tr>
<td>Medium priority</td>
<td>64.0</td>
<td>0.015</td>
<td>9.0 Kb</td>
<td>2.2 K/s</td>
<td>129.0 K</td>
<td>0.0 packets/s</td>
</tr>
<tr>
<td>Low priority</td>
<td>1.1 k</td>
<td>0.251</td>
<td>106.5 Kb</td>
<td>26.0 K/s</td>
<td>1.4 K</td>
<td>0.3 packets/s</td>
</tr>
<tr>
<td>Legitimate traffic</td>
<td>9.0 k</td>
<td>2.153</td>
<td>15.2 Mb</td>
<td>3.7 Mb/s</td>
<td>112.7 K</td>
<td>26.0 packets/s</td>
</tr>
<tr>
<td>Total traffic</td>
<td>10.9 k</td>
<td>2.596</td>
<td>7.7 Gb</td>
<td>1.9 Mb/s</td>
<td>8.4 M</td>
<td>2.0 k/s</td>
</tr>
</tbody>
</table>

FlowMon © INVEA-TECH 2015
Jde o útok z dvojice IP adres, které se nachází v Číně, navíc jde o známé útočníky (reputace IP).

<table>
<thead>
<tr>
<th>#</th>
<th>Source</th>
<th>Event type</th>
<th>Detail</th>
<th>Timestamp</th>
<th>NetFlow source</th>
<th>Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>221.12.174.69</td>
<td>UNSHOCT</td>
<td>Continuation of attack (unsuccessful), attempts: 6, targets: 1, total upload: 6.77 KB, maximal upload: 3.14 KB. Part of distributed attack.</td>
<td>2013-05-08 01:33:15</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
<tr>
<td>3</td>
<td>192.168.3.110</td>
<td>BLACKLIST</td>
<td>Known attackers, attempts: 2, uploaded: 4.52 KB, downloaded: 2.27 KB.</td>
<td>2012-05-06 01:33:17</td>
<td>FlowMon probe</td>
<td>221.12.174.69</td>
</tr>
<tr>
<td>4</td>
<td>121.8.154.28</td>
<td>UNSHOCT</td>
<td>End of attack (unsuccessful), summary: Total count of targets: 1, maximum transferred: 1.26 KB, total count of attempts: 75, duration of attack: 875.29 seconds. Part of distributed attack.</td>
<td>2013-05-08 01:32:26</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
<tr>
<td>5</td>
<td>192.168.3.110</td>
<td>BLACKLIST</td>
<td>Known attackers, attempts: 34, uploaded: 76.11 KB, downloaded: 35.13 KB.</td>
<td>2013-05-06 01:28:30</td>
<td>FlowMon probe</td>
<td>121.8.154.28, 221.12.174.69</td>
</tr>
<tr>
<td>6</td>
<td>221.12.174.69</td>
<td>UNSHOCT</td>
<td>Continuation of attack (unsuccessful), attempts: 59, targets: 1, total upload: 58.69 KB, maximal upload: 1.14 KB. Part of distributed attack.</td>
<td>2013-05-06 01:20:09</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
<tr>
<td>7</td>
<td>121.8.154.28</td>
<td>BLACKLIST</td>
<td>Continuation of attack (unsuccessful), attempts: 50, targets: 1, total upload: 55.36 KB, maximal upload: 1.26 KB. Part of distributed attack.</td>
<td>2013-05-08 01:28:02</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
<tr>
<td>10</td>
<td>221.12.174.69</td>
<td>UNSHOCT</td>
<td>Continuation of attack (unsuccessful), attempts: 50, targets: 1, total upload: 55.50 KB, maximal upload: 1.16 KB. Part of distributed attack.</td>
<td>2013-05-08 01:23:29</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
<tr>
<td>12</td>
<td>221.12.174.69</td>
<td>UNSHOCT</td>
<td>Continuation of attack (unsuccessful), attempts: 55, targets: 1, total upload: 60.96 KB, maximal upload: 1.17 KB. Part of distributed attack.</td>
<td>2013-05-06 01:13:42</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
<tr>
<td>14</td>
<td>221.12.174.69</td>
<td>UNSHOCT</td>
<td>Continuation of attack (unsuccessful), attempts: 52, targets: 1, total upload: 57.96 KB, maximal upload: 1.16 KB. Part of distributed attack.</td>
<td>2013-05-06 01:13:12</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
<tr>
<td>15</td>
<td>221.12.174.69</td>
<td>UNSHOCT</td>
<td>Start of attack (unsuccessful), attempts: 26, targets: 1, total upload: 31.09 KB, maximal upload: 1.17 KB. Part of distributed attack.</td>
<td>2013-05-08 01:11:10</td>
<td>FlowMon probe</td>
<td>192.168.3.110</td>
</tr>
</tbody>
</table>
NetFlow – NBA

- Oba útoky nebyly úspěšné, můžeme se podívat na detaily až na úroveň jednotlivých toků.
Vybrané incidenty
Infekce botnetem

- Finanční instituce
- Botnetem infikováno několik stanic
- Podvržené čínské adresy útočí do vietnamu

<table>
<thead>
<tr>
<th>#</th>
<th>Zdrojová IP</th>
<th>Typ události</th>
<th>Detail</th>
<th>Čas</th>
<th>Zdroj Netflow dat</th>
<th>Omluva</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>112.90.18.105</td>
<td>L3ANOMALY</td>
<td>The traffic not belonging to any internal network was detected (this may indicate spoofing). Transferred: 2.66 MiB, packets: 65 559.</td>
<td>2013-08-24 07:15:21</td>
<td>localhost</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>112.91.30.17</td>
<td>L3ANOMALY</td>
<td>The traffic not belonging to any internal network was detected (this may indicate spoofing). Transferred: 2.37 MiB, packets: 58 379.</td>
<td>2013-08-24 07:15:21</td>
<td>localhost</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>121.10.112.17</td>
<td>L3ANOMALY</td>
<td>The traffic not belonging to any internal network was detected (this may indicate spoofing). Transferred: 2.37 MiB, packets: 58 266.</td>
<td>2013-08-24 07:15:21</td>
<td>localhost</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>182.61.138.105</td>
<td>L3ANOMALY</td>
<td>The traffic not belonging to any internal network was detected (this may indicate spoofing). Transferred: 2.66 MiB, packets: 65 415.</td>
<td>2013-08-24 07:15:21</td>
<td>localhost</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>210.73.221.181</td>
<td>L3ANOMALY</td>
<td>The traffic not belonging to any internal network was detected (this may indicate spoofing). Transferred: 2.36 MiB, packets: 58 086.</td>
<td>2013-08-24 07:15:21</td>
<td>localhost</td>
<td></td>
</tr>
</tbody>
</table>
DDoS útok a opět botnet

- Informační technologie
- Stanice z lokální sítě pod kontrolou útočníka
- Provádí DDoS útok na povel C&C centra
- Detekováno jako DDoS útok odcházející z dané infrastruktury
Útok na HTTP autentizaci

- Zdravotnictví
- Vedeno z IP adresy v Indonésii
- Pokusy o uhodnutí hesla do phpMyAdmin
- Detaily ze sedmé vrstvy (hostname, URL)
Porušování politik

- Průmyslová výroba
- TOR (Onion router) klient na koncové stanici
- Uživatel obchází bezpečnostní opatření
  - Pro přístup k zablokovaným zdrojům
Manipulace s DNS

• Informační technologie
• Změna používaného DNS serveru na stanici
• Možnost manipulace s DNS záznamy a přístupem na webové servery
Chybná konfigurace

- Průmyslová výroba
- Chybně nakonfigurovaný řídící systém
- Pokusy o navazování komunikace do Internetu
- V systému od výroby bez možnosti změny

<table>
<thead>
<tr>
<th>#</th>
<th>Source</th>
<th>Event type</th>
<th>Detail</th>
<th>Timestamp</th>
<th>Netflow source</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:55:00</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>2</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:49:51</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>3</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:44:51</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>4</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:39:49</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>5</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:34:48</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>6</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:29:41</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>7</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:24:39</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>8</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:19:36</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>9</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:14:35</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>10</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:09:34</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>11</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 15:04:30</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
<tr>
<td>12</td>
<td>192.168.103.80</td>
<td>DNSANOMALY</td>
<td>Attempt to use of unauthorized DNS server (connections: 1).</td>
<td>2013-04-15 14:59:30</td>
<td>localhost</td>
<td>1.2.3.4</td>
</tr>
</tbody>
</table>

© INVEA-TECH 2014
Odposlech provozu

- Služby
- Pokročilý malware přesměroval provoz na infikovanou stanici prostřednictvím DHCP

![DHCP sniffing screenshot](image-url)
Řešení FlowMon
Monitorování provozu v síti (NetFlow/IPFIX)
- Kompletní viditelnost do dění v síti
- Real-time a historická data pro LAN & WAN & komunikaci do Internetu
- Optimalizace správy a provozu sítě
- Efektivní troubleshooting

Bezpečnost datové sítě (NBA, NBAD)
- Založeno na behaviorální analýze, nikoliv známých signaturách
- Detekce pokročilého malware, zero-day útoků, podezřelých přenosů dat, změn chování a dalších incidentů
Architektura

- **FlowMon Kolektor**
  - sběr a vizualizace síťových statistik
  - dlouhodobé uložení a reporting
- **FlowMon ADS**
  - detekce bezpečnostních a provozních událostí a anomálií
  - SW součást výbavy kolektoru
Nasazení řešení

• Centralizovaná architektura
  ▪ Sběr NetFlow z jednoho nebo několika core přepínačů
  ▪ Ukládání a vyhodnocování na centrálním kolektoru

• Sledovaný provoz
  ▪ Klienti – WAN
  ▪ Klienti – servery
  ▪ Servery – WAN

• Vhodné pro
  ▪ Datová centra
  ▪ Velké podniky
Nasazení řešení

• Decentralizovaná architektura
  - Sběr NetFlow z řady hraničních routerů
  - Ukládání a vyhodnocování na centrálním kolektoru

• Sledovaný provoz
  - Lokalita – WAN
  - Lokalita – MPLS
  - Lokalita – DC

• Vhodné pro
  - Pobočky
  - Velký počet lokalit
<table>
<thead>
<tr>
<th>Vlastnosti a funkce</th>
<th>Co je dobré vědět</th>
</tr>
</thead>
<tbody>
<tr>
<td>Podpora verzí NetFlow</td>
<td>NetFlow v5, v9, v10 (IPFIX) a hlavní rozdíly</td>
</tr>
<tr>
<td>Disková kapacita</td>
<td>Pro uložení provozu v plném rozsahu, agregovaná data při incidentu nestačí</td>
</tr>
<tr>
<td>Úroveň detailu</td>
<td>Individuální toky, i pro historická data</td>
</tr>
<tr>
<td>Reporting &amp; alerting</td>
<td>Na základě libovolných atributů, definovatelné</td>
</tr>
<tr>
<td>Behaviorální analýza</td>
<td>Thresholdy nejsou behaviorální analýza, požadujte automatickou detekci incidentů</td>
</tr>
<tr>
<td>Sledování aktivních zařízení</td>
<td>Flow na úrovni přístupové vrstvy, sledování MAC-IP a podpora konceptu BYOD</td>
</tr>
<tr>
<td>Technické řešení</td>
<td>Software vs. HW vs. Virtual</td>
</tr>
</tbody>
</table>
• Česká společnost, univerzitní spin-off, spolupráce CESNET a univerzity, projekty EU
• Založena 2007
• Oblasti působení:
  ▪ Flow Monitoring
  ▪ Network Behavior Analysis
  ▪ Network Performance Monitoring
• Přes 500 instalací řešení FlowMon
• Obraťte se na nás pro více informací
• Případové studie a reference
  ▪ www.invea.cz/spolecnost/reference
  ▪ www.invea.cz/produkty-sluzby/flowmon/flowmon-pripadove-studie
• Pilotní projekt řešení FlowMon ve Vaší datové síti
Petr Špringl
springl@invea.com
+420 724 899 760

INVEA-TECH a.s.
U Vodárny 2965/2
616 00 Brno
www.invea.com
Summary
IWAN building blocks

- Application Visibility
  - Traffic
  - Response time
  - Multimedia quality

- HQoS

- WAN optimization
  - Latency
  - Cisco WAAS
  - Akamai integration

- Easy Monitoring