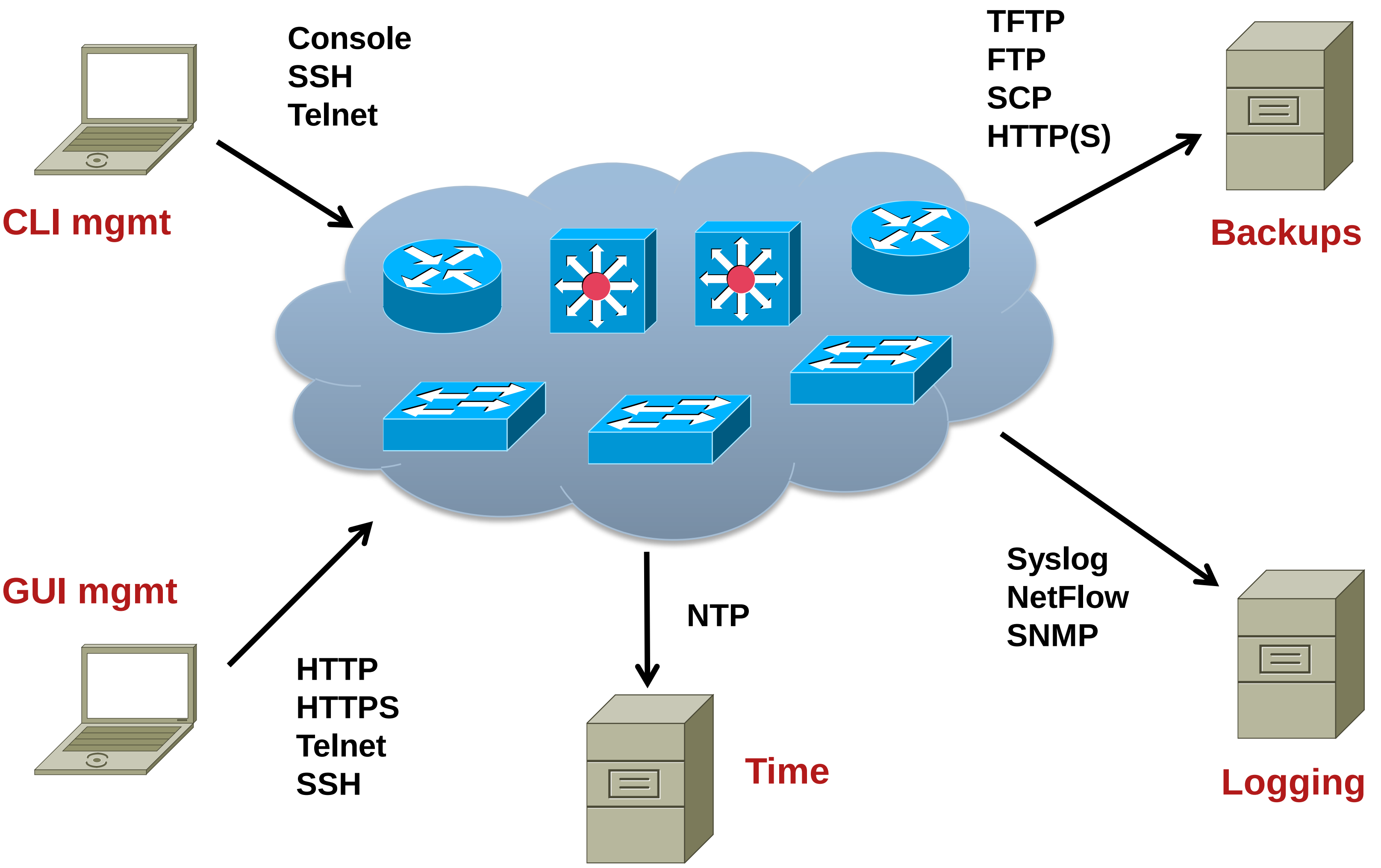
Maintenance and Troubleshooting Tools

CCNP TSHOOT: Module 5

Agenda

- NTP
- Syslog
- SNMP
- NetFlow
- (R)SPAN
- EEM

Fundamental Maintenance Tools



NTP

Network Time Protocol

- NTP specified in the RFC 5905, used to synchronize computer clocks in the Internet
- NTP uses hierarchy of servers. Accuracy of each server is defined by a number called the stratum
  - **Stratum 0**: Reference clock, e.g. atomic (cesium, rubidium) clocks, GPS clocks etc.
  - **Stratum 1**: NTP server whose system clocks are synchronized to within a few microseconds of their attached stratum 0 device
  - **Stratum N**: NTP server synchronized with NTP stratum N-1 server
- NTP is necessary for several reasons:
  - Key-chains - key expiration
  - Certificates – expiration
  - Logs – correlation logs from several devices

NTP Configuration

- NTP **client** configuration

```
Router(config)# ntp server IP [prefer]
```

- NTP **server** configuration

```
Router(config)# ntp master [1-15] ! stratum: 8 by default
```

- Time zone configuration

```
Router(config)# clock timezone CET 1

Router(config)# clock summer-time CEST recurring
  last Sun Mar 2:00 last Sun Oct 3:00
```

NTP Configuration and Verification

- Service timestamps add timestamp to debug and log messages

```
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime localtime show-timezone
!
clock timezone CET 1
clock summer-time CEST recurring last Sun Mar 2:00 last Sun Oct 3:00
!
ntp server 10.1.220.3 prefer
```

```
Router# show ntp status
Clock is synchronized, stratum 12, reference is 158.193.48.7
nominal freq is 119.2092 Hz, actual freq is 119.2078 Hz, precision is 2**18
reference time is D2054E5B.686C9787 (01:31:39.407 CEST Mon Aug 29 2011)
clock offset is -0.0317 msec, root delay is 2.15 msec
root dispersion is 12.08 msec, peer dispersion is 0.23 msec
Router# show ntp associations

      address           ref clock      st  when  poll reach  delay  offset    disp
*~158.193.48.7      127.127.1.0        11    37   512  377     2.2   -0.03     0.2
 * master (synced), # master (unsynced), + selected, - candidate, ~ configured
```

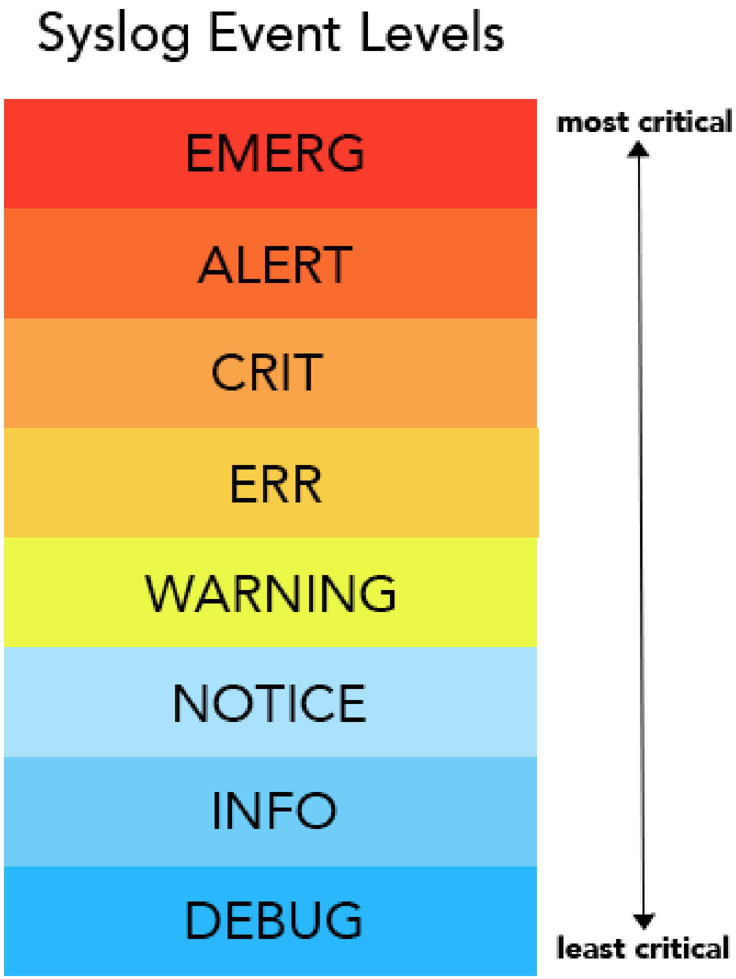
Syslog

Syslog

- Allows a device to report error and notification messages, either locally or to a remote logging server
- Using UDP port 514 (servers sometimes use TCP 514)
- Every syslog message contains a severity level and a facility
- Widely supported on many devices, including routers, switches, application servers, firewalls, and other network appliances

Syslog Levels

- Logging severity levels on Cisco devices:
  0) Emergencies
  1) Alerts
  2) Critical
  3) Errors
  4) Warnings
  5) Notifications
  6) Informational
  7) Debugging
- Enabling logging for a lower level (from importance point of view) will enable logging for all the above levels.



Syslog Event Levels

EMERG

ALERT

CRIT

ERR

WARNING

NOTICE

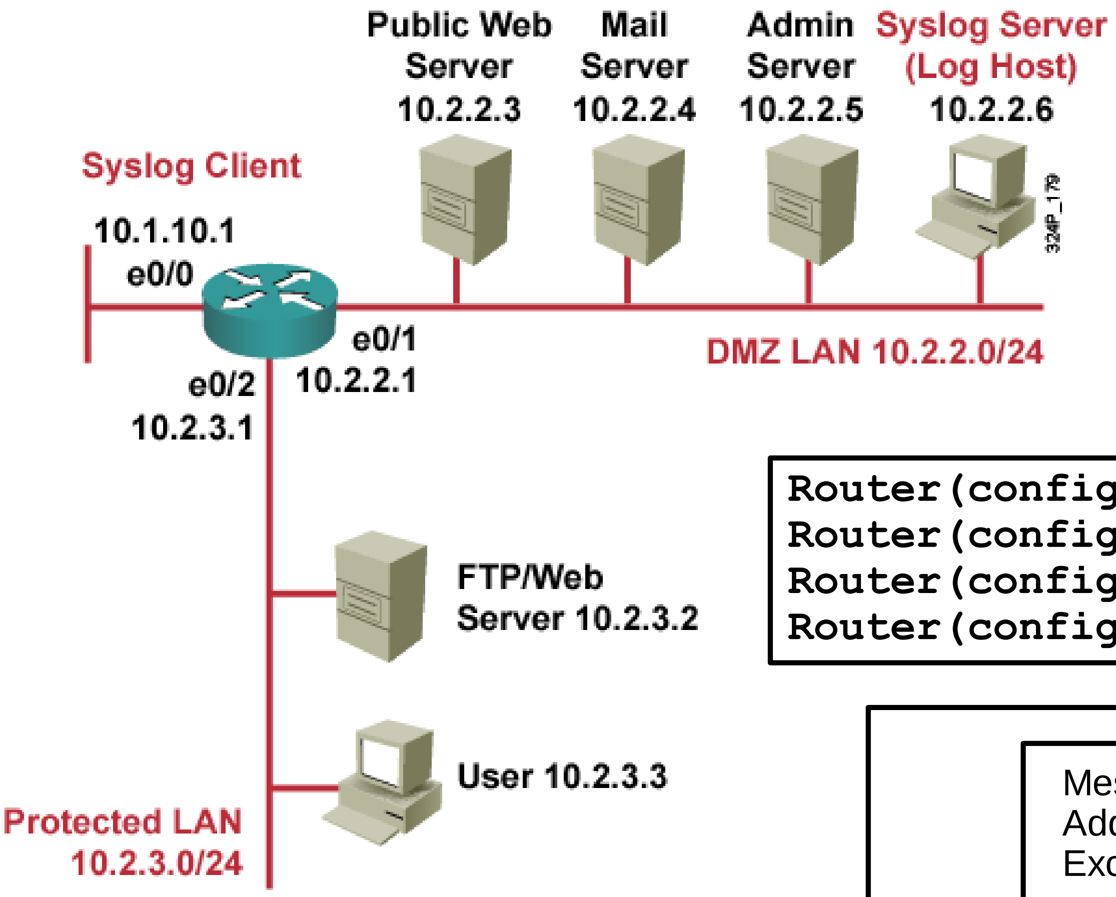
INFO

DEBUG

most critical

least critical

Logging to a Server



Messages are logged to a circular log buffer in RAM that is limited to 16384 Bytes.

```
Router(config)# logging buffered 16348
Router(config)# logging console warnings
Router(config)# logging trap alerts
Router(config)# logging 10.1.152.1
```

Messages are logged to a syslog server at IP Address 10.1.152.1. By default all messages Except level 7 are sent.

Logging messages on the console are limited to severity level 4 and lower. By default all messages from severity level 0 (emergencies) to severity level 7 (debugging) are logged.

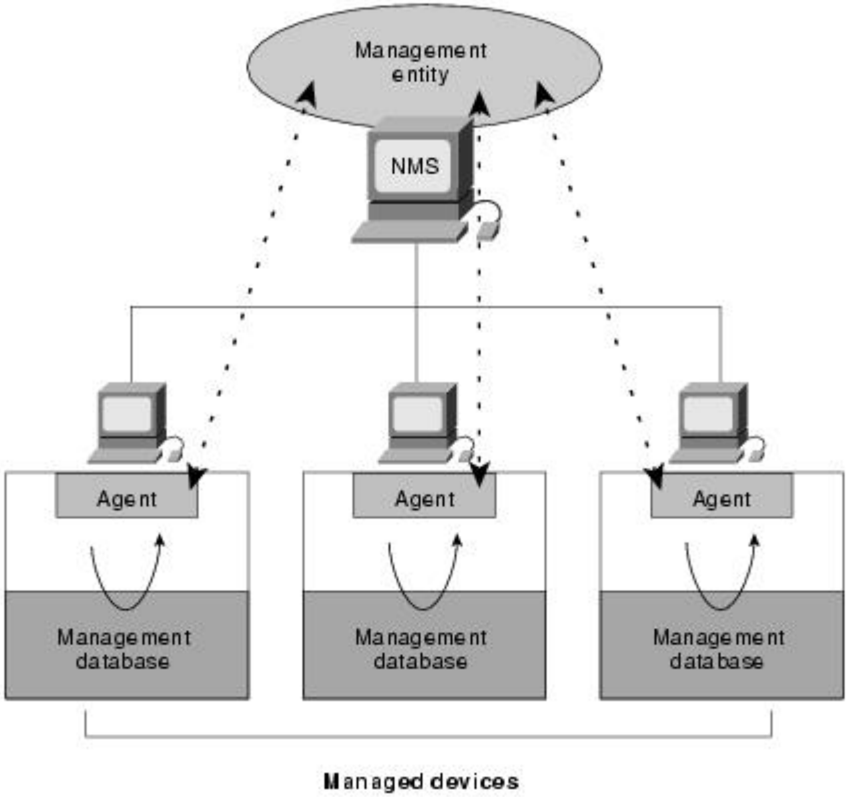
Logging to a Server

```
Router# show logging
Syslog logging: enabled (11 messages dropped, 0 messages rate-limited,
                0 flushes, 0 overruns, xml disabled, filtering disabled)
    Console logging: level warnings, 29 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging: level debugging, 2 messages logged, xml disabled,
                    filtering disabled
    Logging Exception size (4096 bytes)
    Count and timestamp logging messages: disabled

No active filter modules.

    Trap logging: level informational, 35 message lines logged
        Logging to 10.1.152.1 (udp port 514, audit disabled, link up), 2
message lines logged, xml disabled,
               filtering disabled

Log Buffer (16384 bytes):

*Mar  2 02:26:08.909: %SYS-5-CONFIG_I: Configured from console by console
*Mar  2 02:26:09.909: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host
10.1.152.1 started - CLI initiated
```

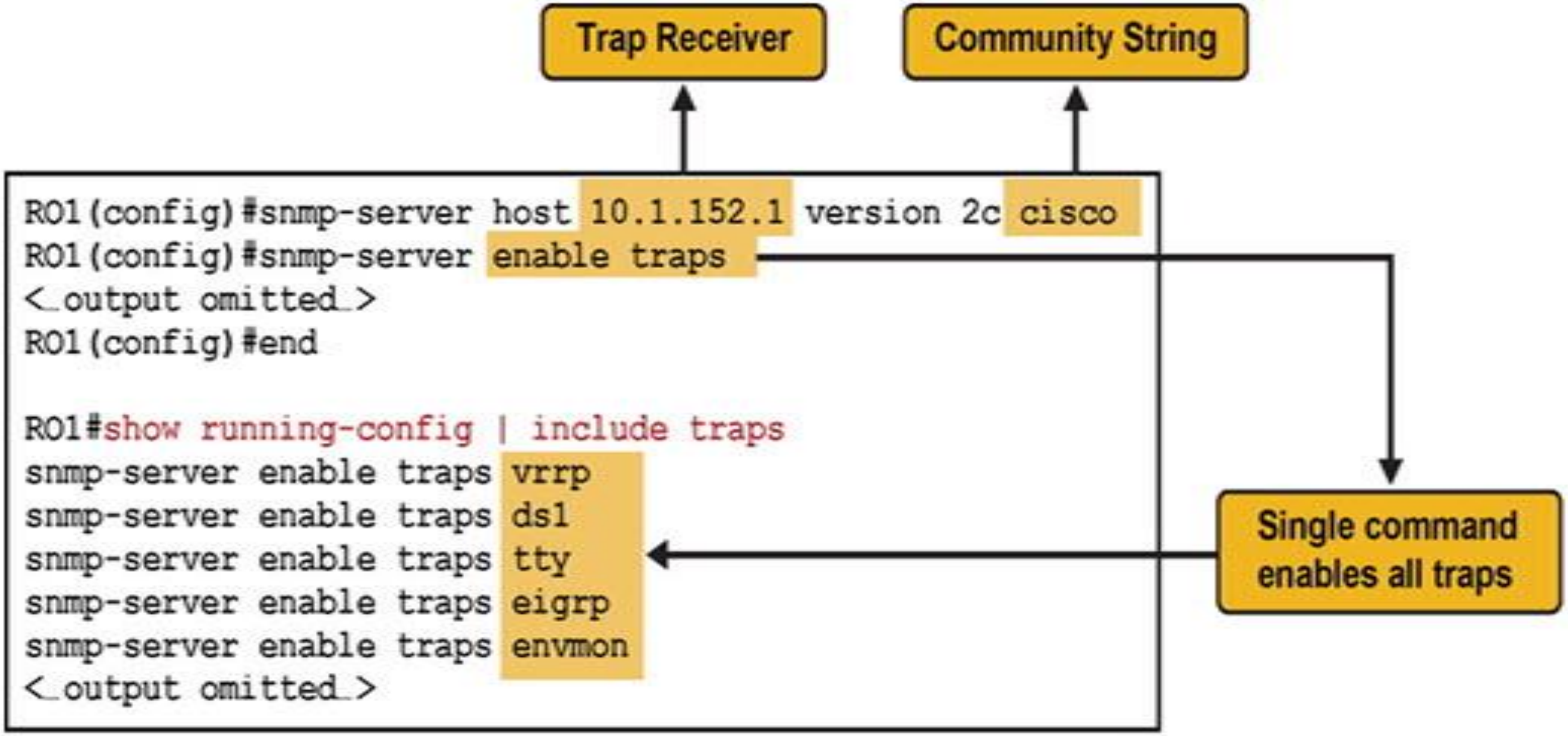
SNMP

SNMP

- Standard for managing devices and collect statistics
- Widely supported on many networking devices, including routers, switches, application servers, firewalls, and other network appliances
- Three key components:
  - NMS – network management system
  - Managed Device
  - Agent
- Polling - NMS query agent (UDP port 161)
- Trap - Agent inform NMS (UDP port 162)
- OID – Object identifier

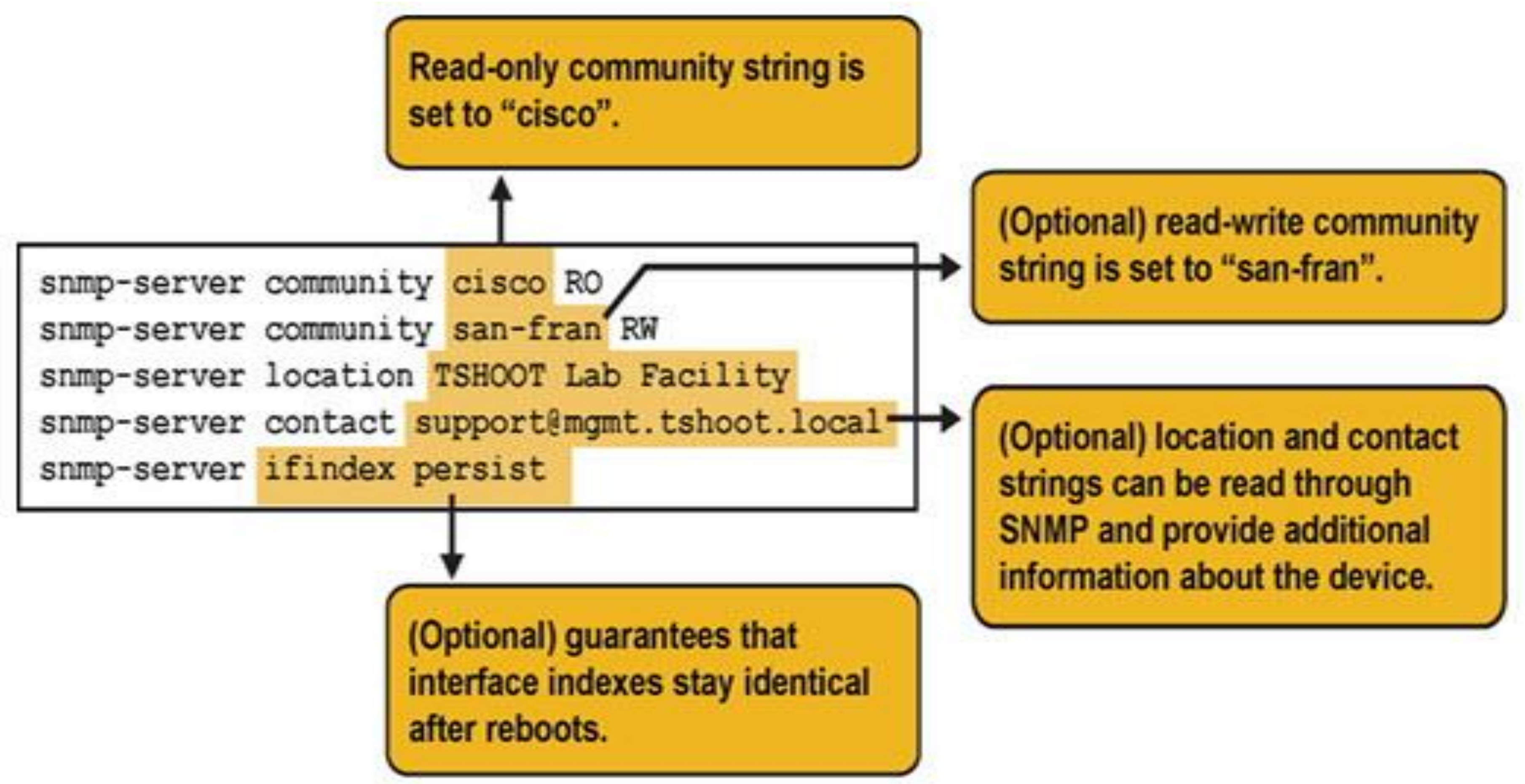


SNMP Configuration



```
R01(config)#snmp-server host 10.1.152.1 version 2c cisco
R01(config)#snmp-server enable traps
<output omitted>
R01(config)#end

R01#show running-config | include traps
snmp-server enable traps vrrp
snmp-server enable traps ds1
snmp-server enable traps tty
snmp-server enable traps eigrp
snmp-server enable traps envmon
<output omitted>
```

SNMP Configuration



```
snmp-server community cisco RO
snmp-server community san-fran RW
snmp-server location TSHOOT Lab Facility
snmp-server contact support@mgmt.tshoot.local
snmp-server ifindex persist
```

Read-only community string is set to “cisco”.

(Optional) read-write community string is set to “san-fran”.

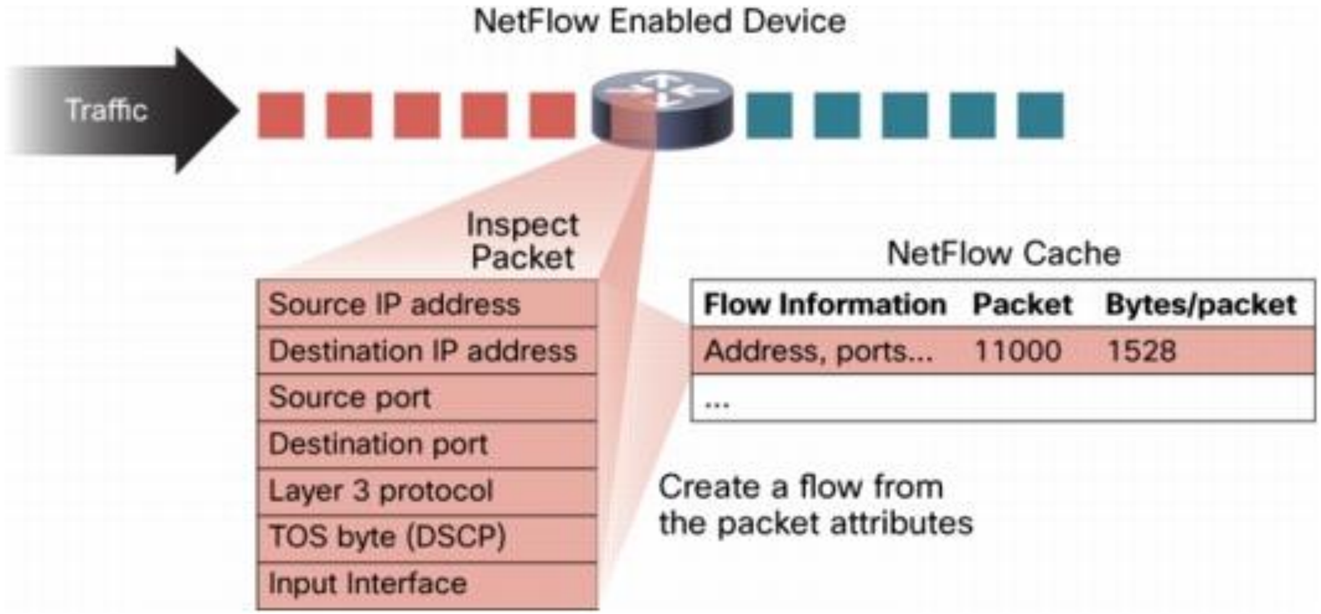
(Optional) location and contact strings can be read through SNMP and provide additional information about the device.

(Optional) guarantees that interface indexes stay identical after reboots.

NetFlow

NetFlow

- Defined in RFC 3954 (NetFlow v9) RFC 7011 (IPFIX)
- Standard for collection information about flows
- Two main components
  - exporter
  - collector

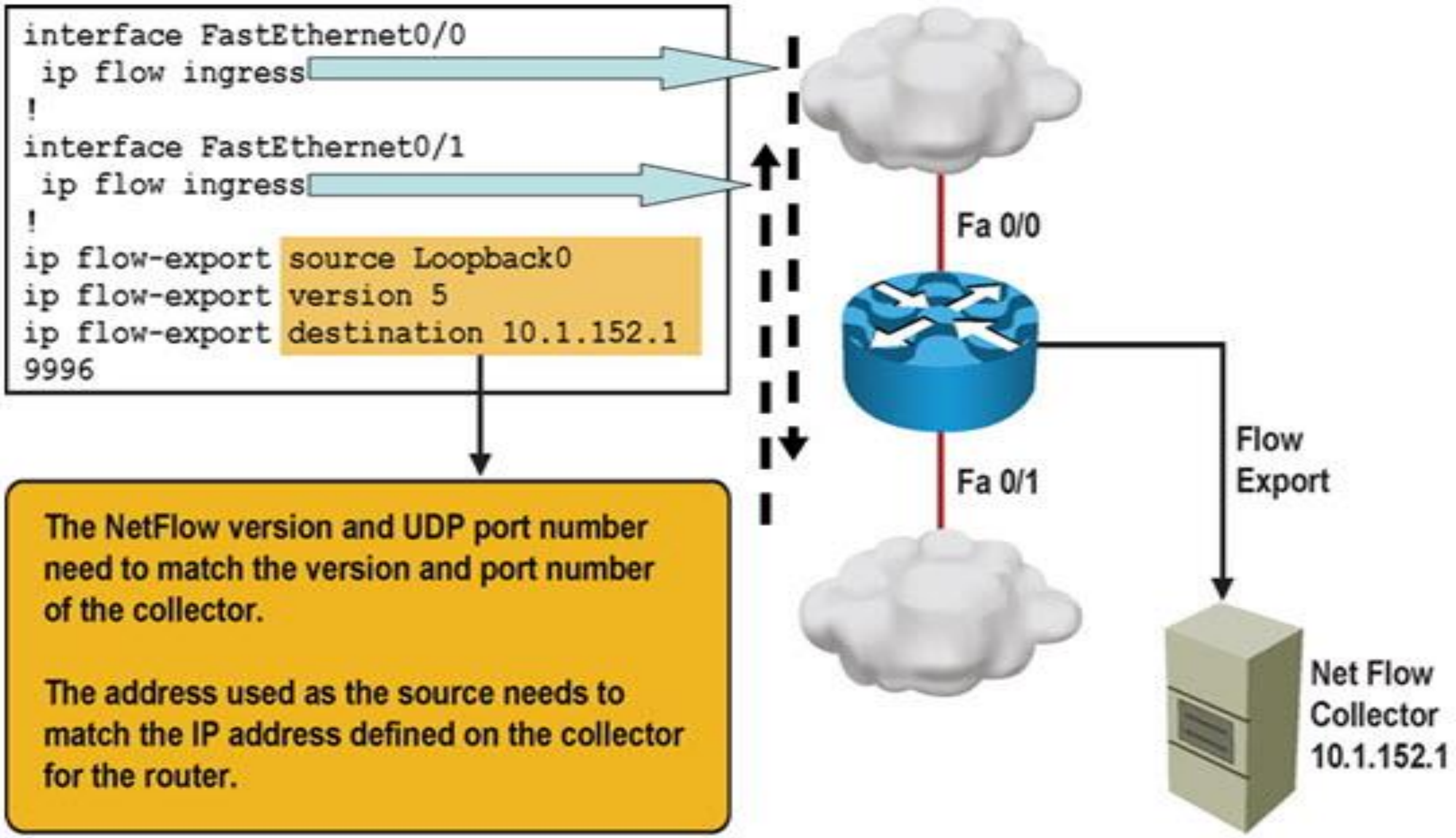


Source IP address
Destination IP address
Source port
Destination port
Layer 3 protocol
TOS byte (DSCP)
Input Interface

Flow Information	Packet	Bytes/packet
Address, ports...	11000	1528
...		

Gathering Information with NetFlow

A Simple NetFlow Configuration Example



```
interface FastEthernet0/0
 ip flow ingress
!
interface FastEthernet0/1
 ip flow ingress
!
ip flow-export source Loopback0
ip flow-export version 5
ip flow-export destination 10.1.152.1
9996
```

The NetFlow version and UDP port number need to match the version and port number of the collector.

The address used as the source needs to match the IP address defined on the collector for the router.

SNMP and NetFlow Comparison

- Both are used to gather statistics from Cisco switches and routers.
- SNMP’s focus is primarily on the collection of various statistics from components within network devices.
- A NetFlow enabled device collects information about the IP traffic flowing through the device.
- NetFlow uses a “push” based model – devices send data to a collector.
- SNMP is considered pull-based – the NMS queries SNMP Agents.
- NetFlow only gathers traffic statistics.
- SNMP can also collect many other performance indicators such as interface errors, CPU usage, and memory usage.
- Statistics collected using NetFlow have more granularity.
- NetFlow is currently supported on most Cisco IOS routers but only the 4500 and 6500 series switches

Gathering Information with NetFlow

- You can display the NetFlow cache content by issuing the **show ip cache flow** command

```
R1# show ip cache flow
<output omitted>
SrcIf          SrcIPaddress    DstIF     DstIPaddress     Pr  SrcP  DstP  Pkts
Se0/0/0.121    10.1.194.10     Null      224.0.0.10       58  0000  0000  27
Se0/0/0.121    10.1.194.14     Null      224.0.0.10       58  0000  0000  28
Fa0/0          10.1.192.5      Null      224.0.0.10       58  0000  0000  28
Fa0/1          10.1.192.13     Null      224.0.0.10       58  0000  0000  27
Fa0/1          10.1.152.1      Local     10.1.220.2       01  0000  0303   1
Se0/0/1        10.1.193.6      Null      224.0.0.10       58  0000  0000   28
Fa0/1          10.1.152.1      Se0/0/1   10.1.163.193     11  0666  E75E   1906
Se0/0/1        10.1.163.193    Fa0/0     10.1.152.1       11  E75E  0666   1905
```

EEM

Embedded Event Manager (EEM)

- Enables custom policies that trigger actions based on events:
  - syslog messages
  - Cisco IOS counter changes
  - SNMP MIB object changes
  - SNMP traps
  - CLI command execution
  - Timers and many other options
- Actions can consist of:
  - Sending SNMP traps or syslog messages
  - Executing CLI commands
  - Sending email
  - Running tool command language (TCL) scripts

Sample EEM

- The `occurs 1` option forces the event to be triggered on a single occurrence of the CLI pattern
- For more information, visit http://cisco.com/go/instrumentation

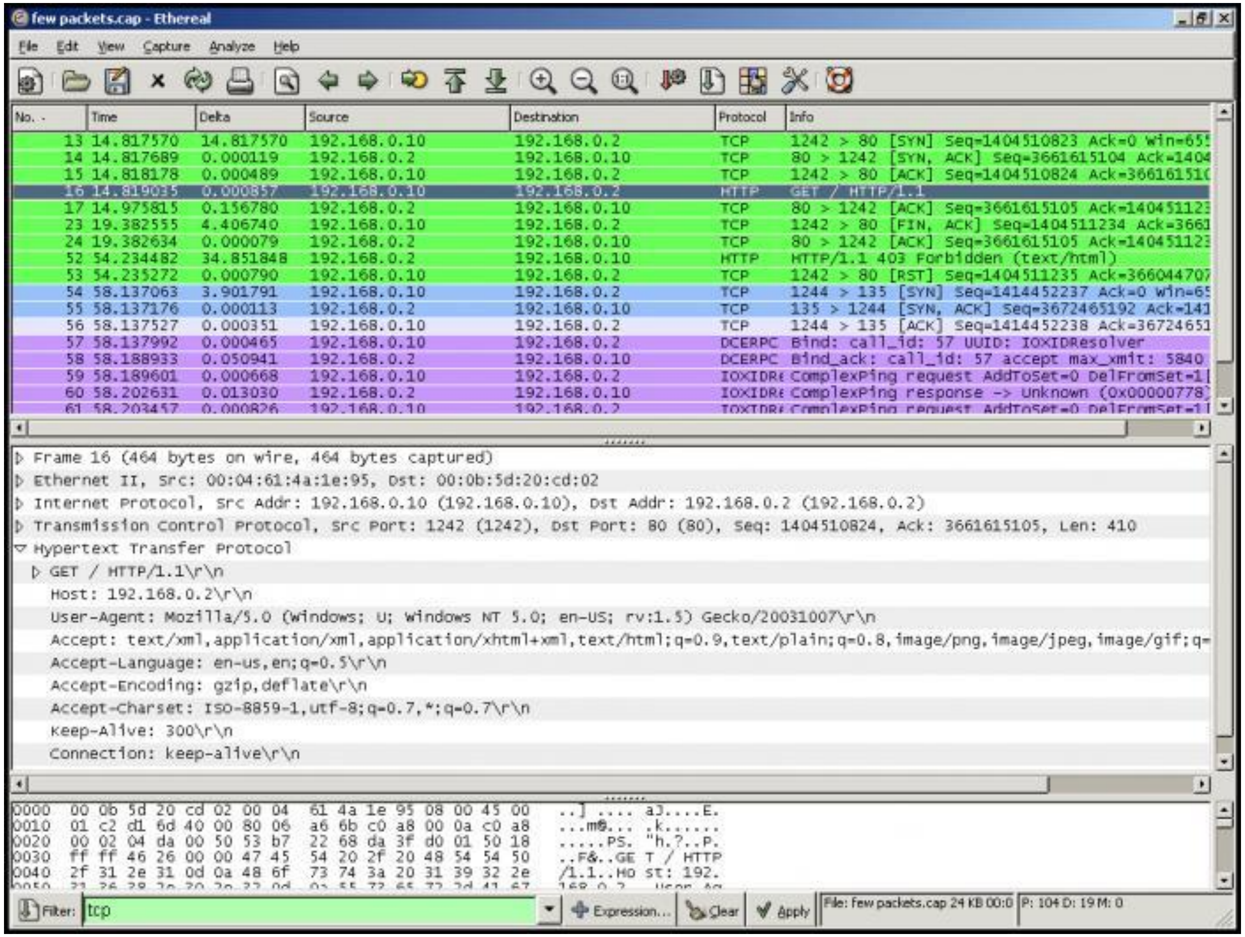
```
R1(config)# event manager applet CONFIG-STARTED

R1(config-applet)# event cli pattern "configure terminal" sync no skip no
occurs 1

R1(config-applet)# action 1.0 syslog priority critical msg "Configuration mode
was entered"

R1(config-applet)# action 2.0 syslog priority informational msg "Change
control policies apply. Authorized access only."
```

```
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.

R1(config)#
Jul 13 03:24:41.473 PDT: %HA_EM-2-LOG: CONFIG-STARTED: Configuration mode was
entered
Jul 13 03:24:41.473 PDT: %HA_EM-6-LOG: CONFIG-STARTED: Change control policies
apply. Authorized access only
```

RSPAN

Using Traffic Capturing Tools

- PCAP, PCAPng, MNM
- http://www.fit.vutbr.cz/~ivesely/pubs.php?id=10183



```
No. .  Time       Delta      Source         Destination    Protocol Info
    13 14.817570  14.817570  192.168.0.10   192.168.0.2    TCP      1242 > 80 [SYN] Seq=1404510823 Ack=0 Win=65
    14 14.817689  0.000119   192.168.0.2    192.168.0.10   TCP      80 > 1242 [SYN, ACK] Seq=3661615104 Ack=1404
    15 14.818178  0.000489   192.168.0.10   192.168.0.2    TCP      1242 > 80 [ACK] Seq=1404510824 Ack=36616151(
    16 14.819035  0.000857   192.168.0.10   192.168.0.2    HTTP     GET / HTTP/1.1
    17 14.975815  0.156780   192.168.0.2    192.168.0.10   TCP      80 > 1242 [ACK] Seq=3661615105 Ack=14045112
    23 19.382555  4.406740   192.168.0.10   192.168.0.2    TCP      1242 > 80 [FIN, ACK] Seq=1404511234 Ack=3661
    24 19.382634  0.000079   192.168.0.2    192.168.0.10   TCP      80 > 1242 [ACK] Seq=3661615105 Ack=14045112
    52 54.234482  34.851848  192.168.0.2    192.168.0.10   HTTP     HTTP/1.1 403 Forbidden (text/html)
    53 54.235272  0.000790   192.168.0.10   192.168.0.2    TCP      1242 > 80 [RST] Seq=1404511235 Ack=366044707
    54 58.137063  3.901791   192.168.0.10   192.168.0.2    TCP      1244 > 135 [SYN] Seq=1414452237 Ack=0 Win=65
    55 58.137176  0.000113   192.168.0.2    192.168.0.10   TCP      135 > 1244 [SYN, ACK] Seq=3672465192 Ack=141
    56 58.137527  0.000351   192.168.0.10   192.168.0.2    TCP      1244 > 135 [ACK] Seq=1414452238 Ack=36724651
    57 58.137992  0.000465   192.168.0.10   192.168.0.2    DCERPC   Bind: call_id: 57 UUID: IOXIDResolver
    58 58.188933  0.050941   192.168.0.2    192.168.0.10   DCERPC   Bind_ack: call_id: 57 accept max_xmit: 5840
    59 58.189601  0.000668   192.168.0.10   192.168.0.2    IOXIDRE  ComplexPing request AddToSet=0 DelFromSet=1[
    60 58.202631  0.013030   192.168.0.2    192.168.0.10   IOXIDRE  ComplexPing response -> Unknown (0x00000778)
    61 58.203457  0.000826   192.168.0.10   192.168.0.2    IOXIDRE  ComplexPing request AddToSet=0 DelFromSet=1[
```

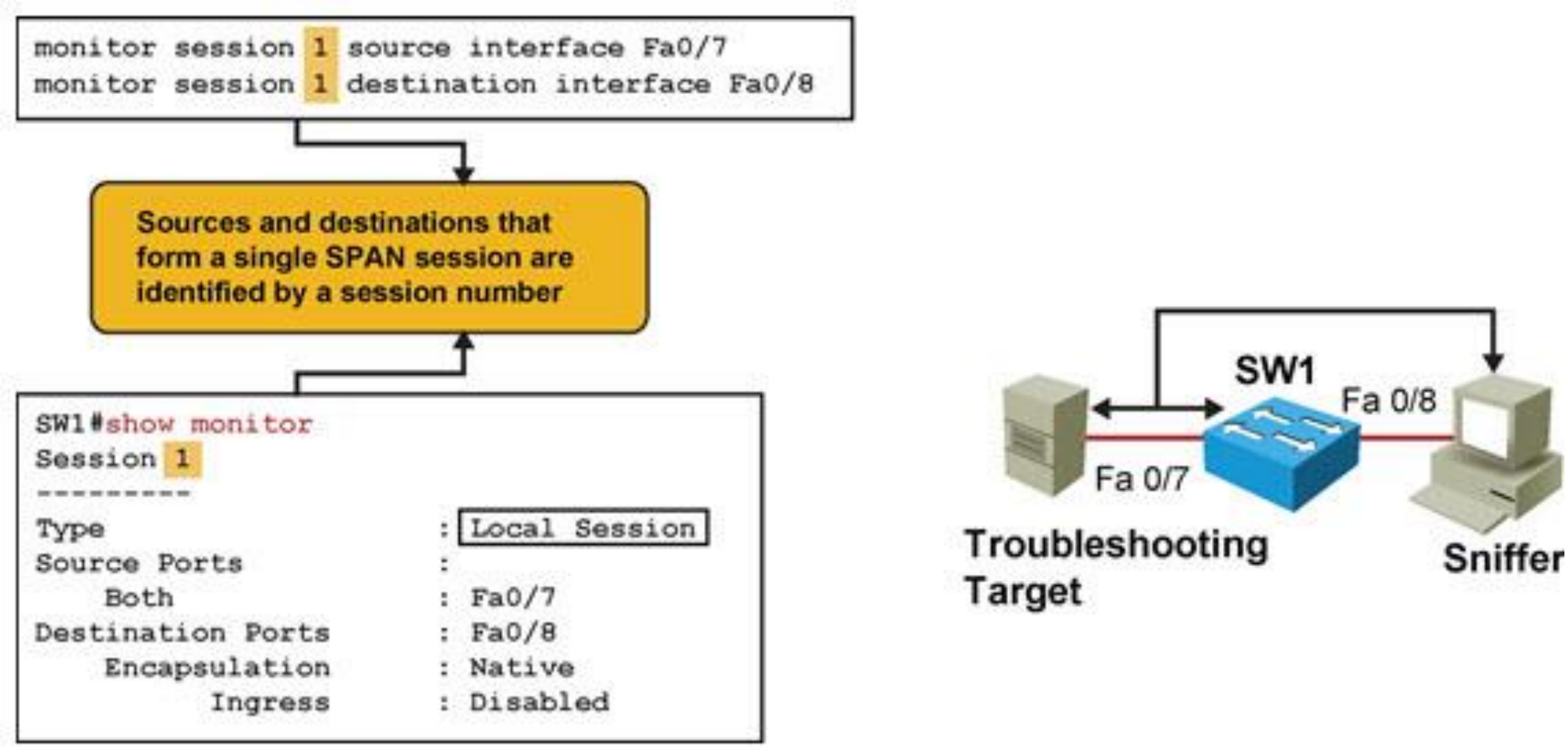
```
▷ Frame 16 (464 bytes on wire, 464 bytes captured)
▷ Ethernet II, Src: 00:04:61:4a:1e:95, Dst: 00:0b:5d:20:cd:02
▷ Internet Protocol, Src Addr: 192.168.0.10 (192.168.0.10), Dst Addr: 192.168.0.2 (192.168.0.2)
▷ Transmission Control Protocol, Src Port: 1242 (1242), Dst Port: 80 (80), Seq: 1404510824, Ack: 3661615105, Len: 410
▽ Hypertext Transfer Protocol
  ▷ GET / HTTP/1.1\r\n
    Host: 192.168.0.2\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.0; en-US; rv:1.5) Gecko/20031007\r\n
    Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,image/jpeg,image/gif;q=
    Accept-Language: en-us,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
    Keep-Alive: 300\r\n
    Connection: keep-alive\r\n
```

```
0000  00 0b 5d 20 cd 02 00 04  61 4a 1e 95 08 00 45 00   ..] .... aJ....E.
0010  01 c2 d1 6d 40 00 80 06  a6 6b c0 a8 00 0a c0 a8   ...m@... .k......
0020  00 02 04 da 00 50 53 b7  22 68 da 3f d0 01 50 18   .....PS. "h.?..P.
0030  ff ff 46 26 00 00 47 45  54 20 2f 20 48 54 54 50   ..F&..GE T / HTTP
0040  2f 31 2e 31 0d 0a 48 6f  73 74 3a 20 31 39 32 2e   /1.1..Ho st: 192.
0050  31 36 38 2e 30 2e 32 0d  0a 55 73 65 72 2d 41 67   168.0.2. .User-Ag
```

tcp

File: few packets.cap 24 KB 00:0 | P: 104 D: 19 M: 0

Switched Port Analyzer (SPAN)

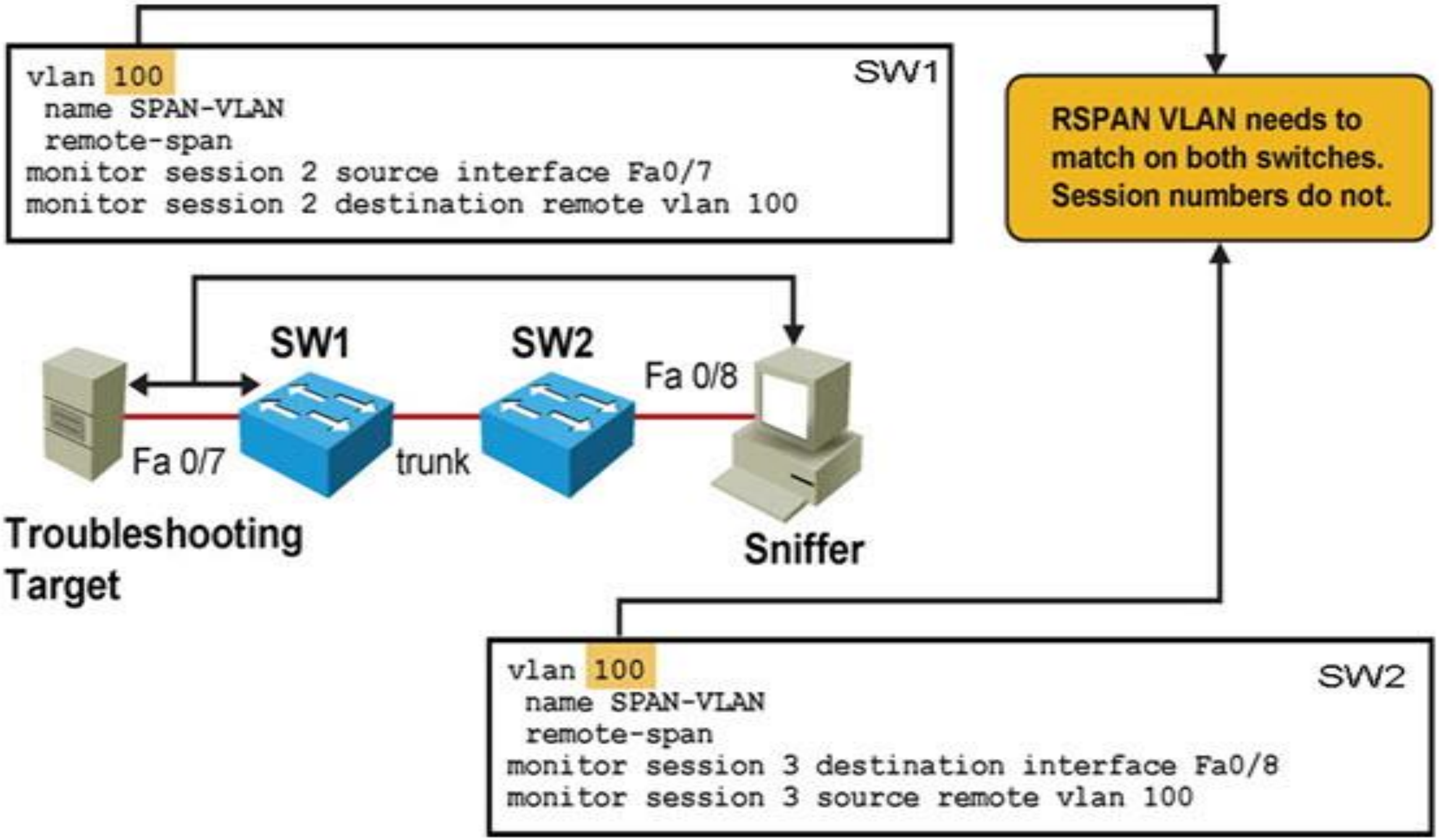


```
monitor session 1 source interface Fa0/7
monitor session 1 destination interface Fa0/8
```

Sources and destinations that form a single SPAN session are identified by a session number

```
SW1#show monitor
Session 1
---------
Type                    : Local Session
Source Ports            :
    Both                : Fa0/7
Destination Ports       : Fa0/8
    Encapsulation       : Native
          Ingress       : Disabled
```

Remote Switched Port Analyzer (RSPAN) ②



SW1

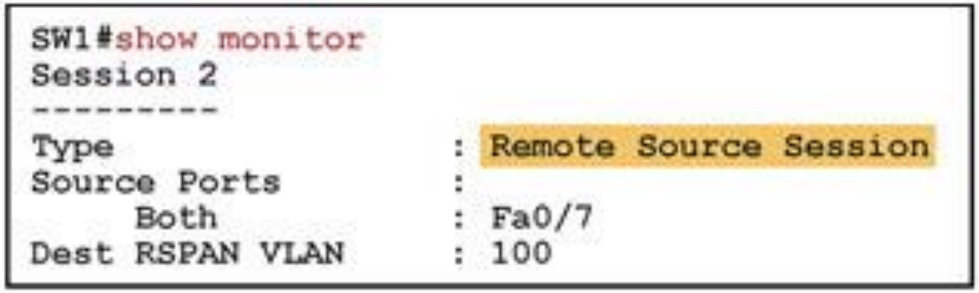
```
vlan 100
 name SPAN-VLAN
 remote-span
monitor session 2 source interface Fa0/7
monitor session 2 destination remote vlan 100
```

RSPAN VLAN needs to match on both switches. Session numbers do not.

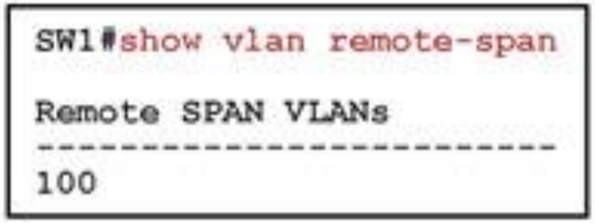
SW2

```
vlan 100
 name SPAN-VLAN
 remote-span
monitor session 3 destination interface Fa0/8
monitor session 3 source remote vlan 100
```

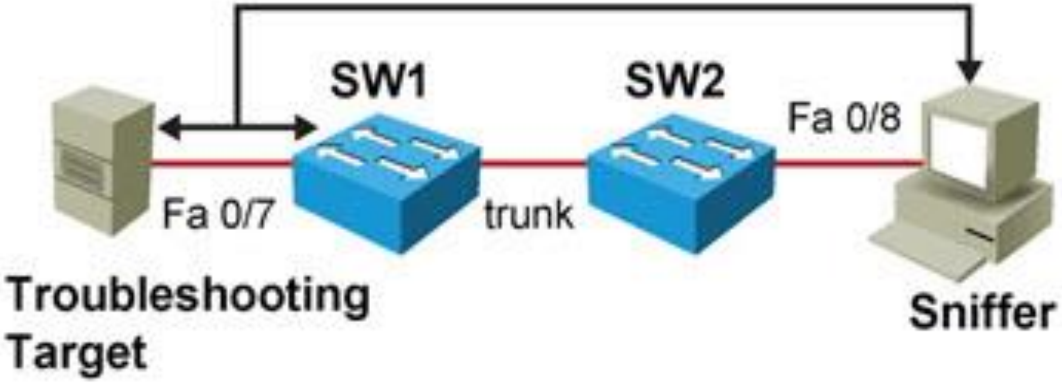
Remote Switched Port Analyzer (RSPAN) ①

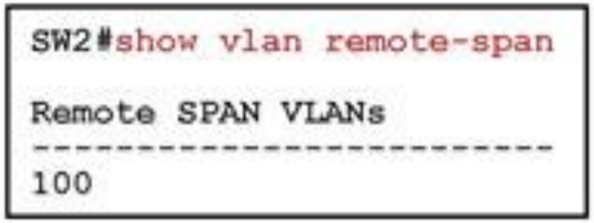


```
SW1#show monitor
Session 2
---------
Type                     : Remote Source Session
Source Ports             :
     Both                : Fa0/7
Dest RSPAN VLAN          : 100
```

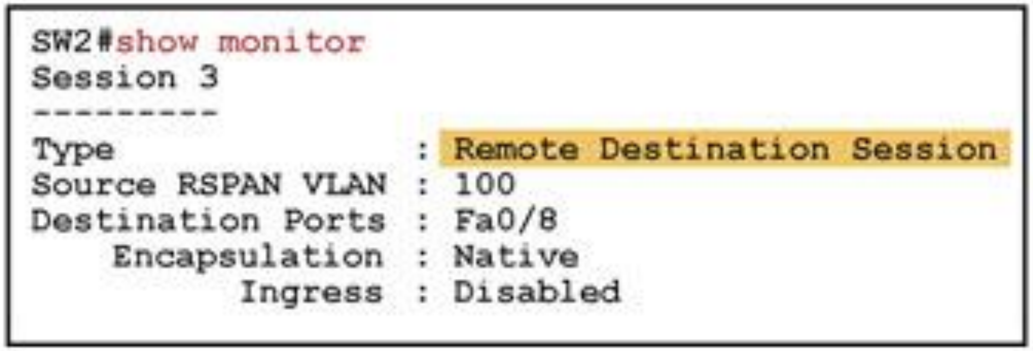


```
SW1#show vlan remote-span

Remote SPAN VLANs
-------------------------
100
```





```
SW2#show vlan remote-span

Remote SPAN VLANs
-------------------------
100
```



```
SW2#show monitor
Session 3
---------
Type                : Remote Destination Session
Source RSPAN VLAN   : 100
Destination Ports   : Fa0/8
    Encapsulation   : Native
          Ingress   : Disabled
```

Slides adapted by Vladimír Veselý and Matěj Grégr
partially from official course materials
but the most of the credit goes to CCIE#23527 Ing. Peter Palúch, Ph.D.

Last update: 2017-03-06