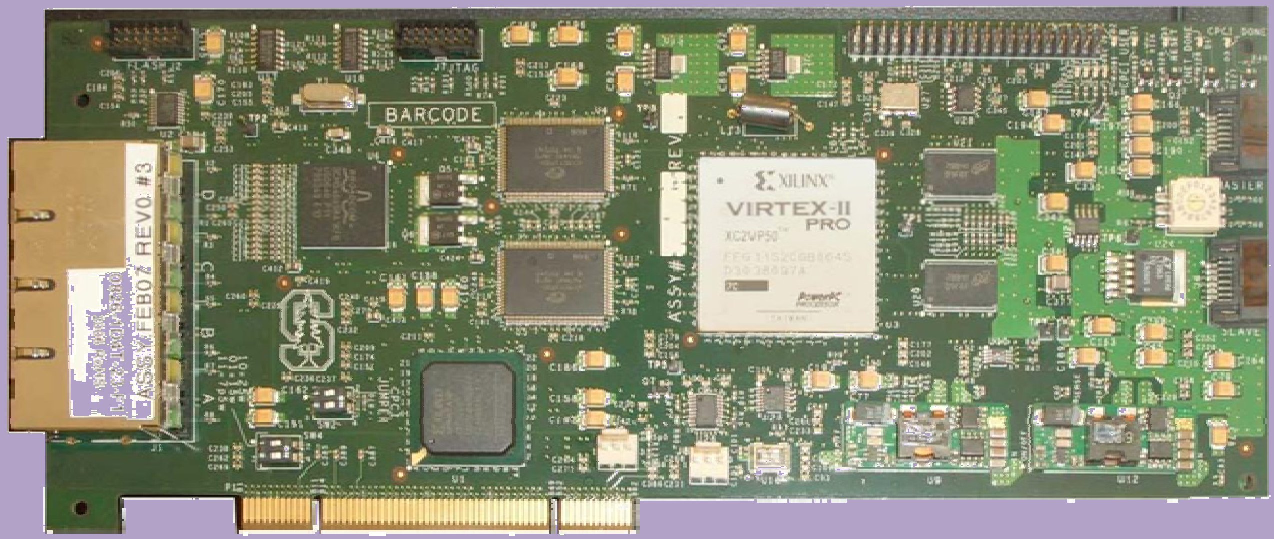


Addition of Virtual Interfaces in NetFlow Probe for the NetFPGA

Muhammad Shahbaz
Zaheer Ahmed
Habibullah Jamal
Asrar Ashraf
Nadeem Yousaf
Raania Naeem Khan

Presentation Organization

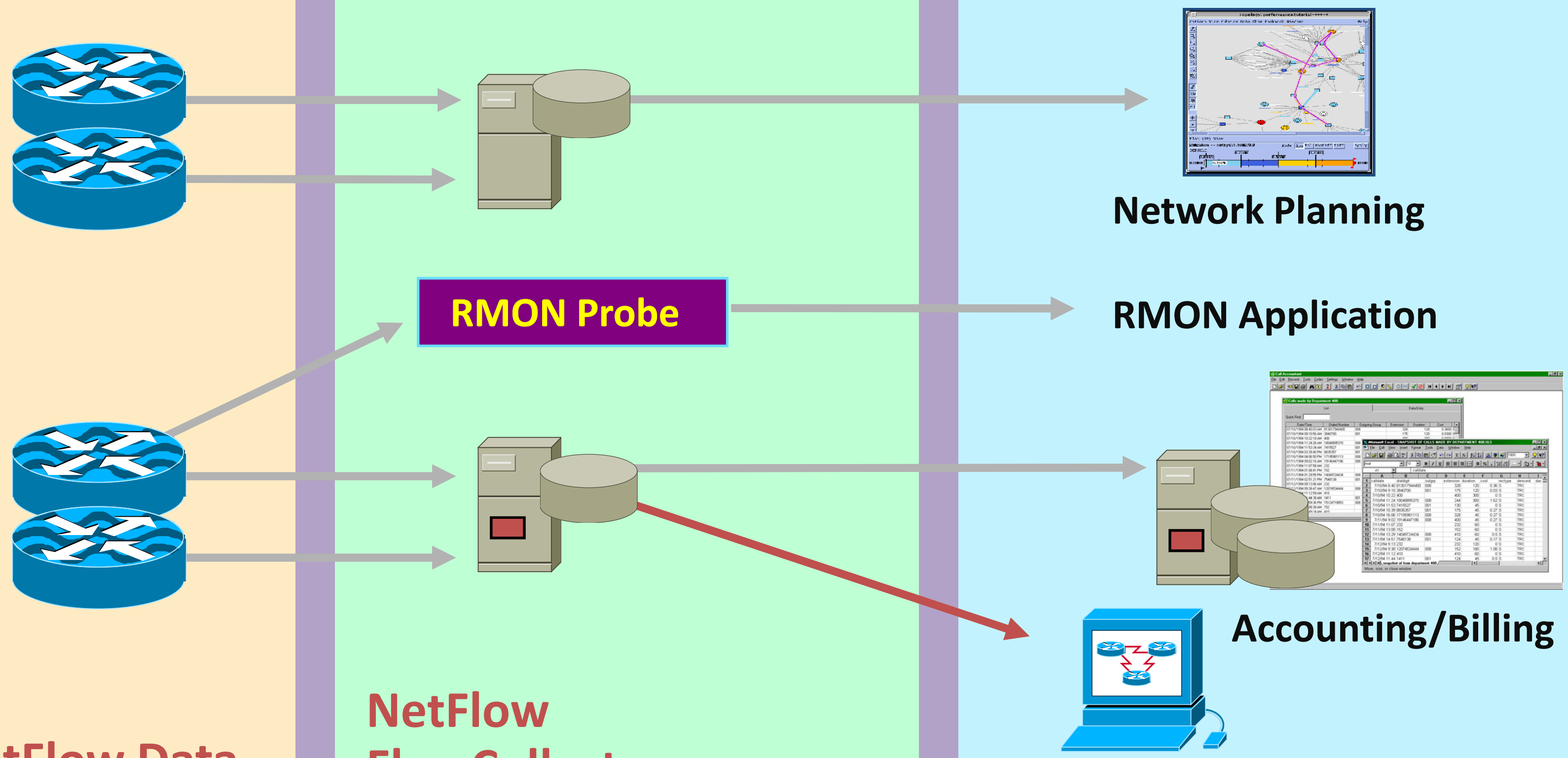
- NetFlow Overview
- Virtual Interfaces in NetFlow
- Hardware Architecture of NetFlow probe
- Software Architecture of NetFlow probe
- Sample Netflow Record
- Extended Applications
- Conclusion
- Demonstration Setup
- Questions / Answers

NetFlow Overview

- Network Protocol developed by Cisco for Collecting IP traffic information
- Cisco proprietary but supported by other platforms like Juniper, Linux etc.
- Netflow enabled routers/probes generate netflow **records**
- Exported via UDP or SCTP to data-collectors
- Netflow record identified traditionally by **7-Tuple keys** formed by combining
  - Source IP
  - Destination IP
  - Source port for UDP or TCP and 0 for other protocols
  - Destination port for UDP or TCP and 0 for other protocols
  - IP protocol
  - Ingress interface
  - IP Type of Service(TOS)
- **Netflow Records** contain extensive information regarding traffic flow including Version, Sequence number, ingress interface, timestamp and other data statistics of particular data flow.

NetFlow Overview (contd.)

Net Flow Applications



RMON Probe

Network Planning

RMON Application

Accounting/Billing

NetFlow Data Export:

- Data Switching
- Data Aggregation
- Data Export

NetFlow FlowCollector:

- Data Collection
- Data Filtering
- Data Aggregation
- Data Storage

Network Data Analyzer:

- Data Presentation
- Flow Control and Configuration

Partner Applications

Image From NetFlow PPT by Michael Lin, Cisco Systems

NetFlow Overview (contd.)
Standalone NetFlow Architecture

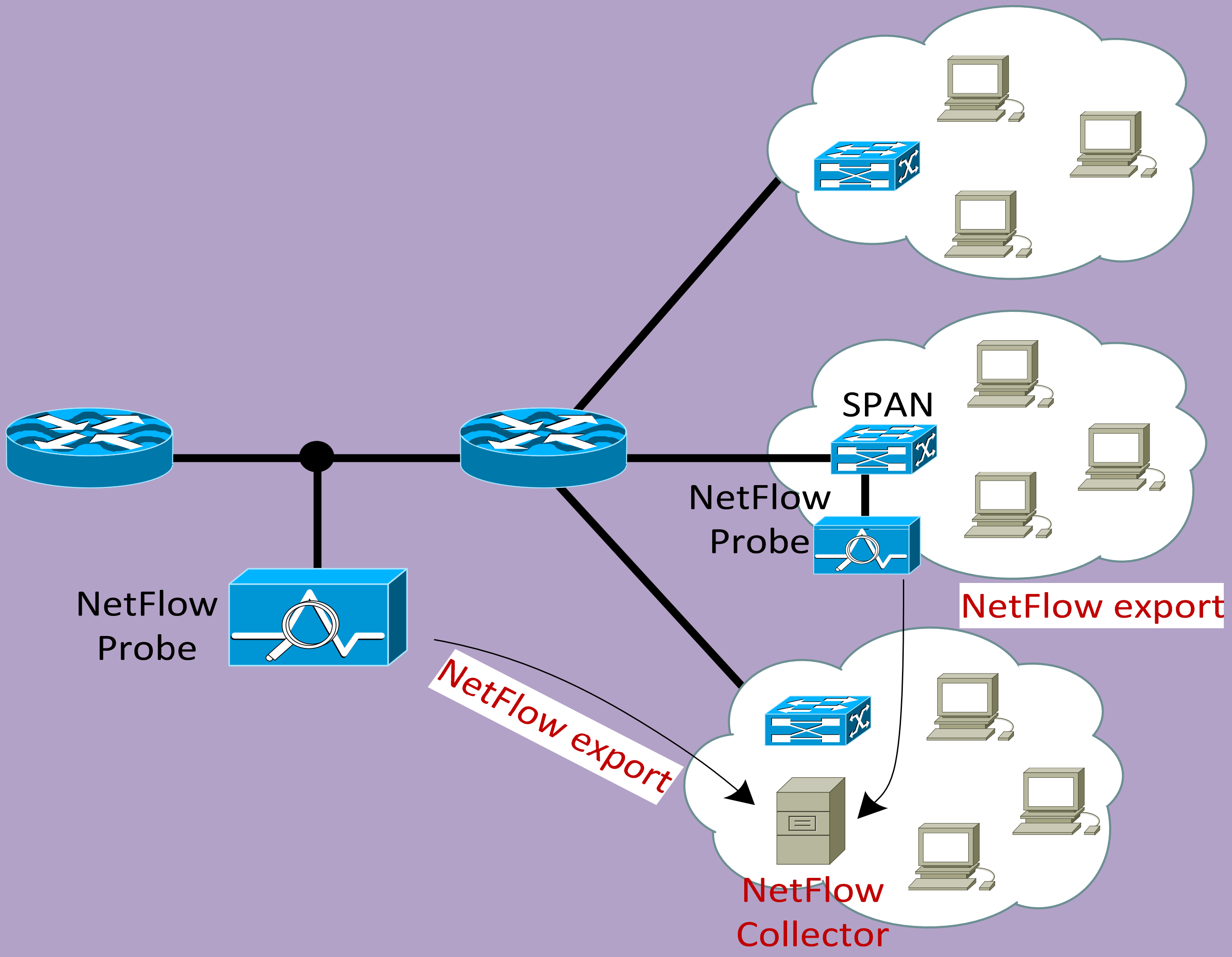
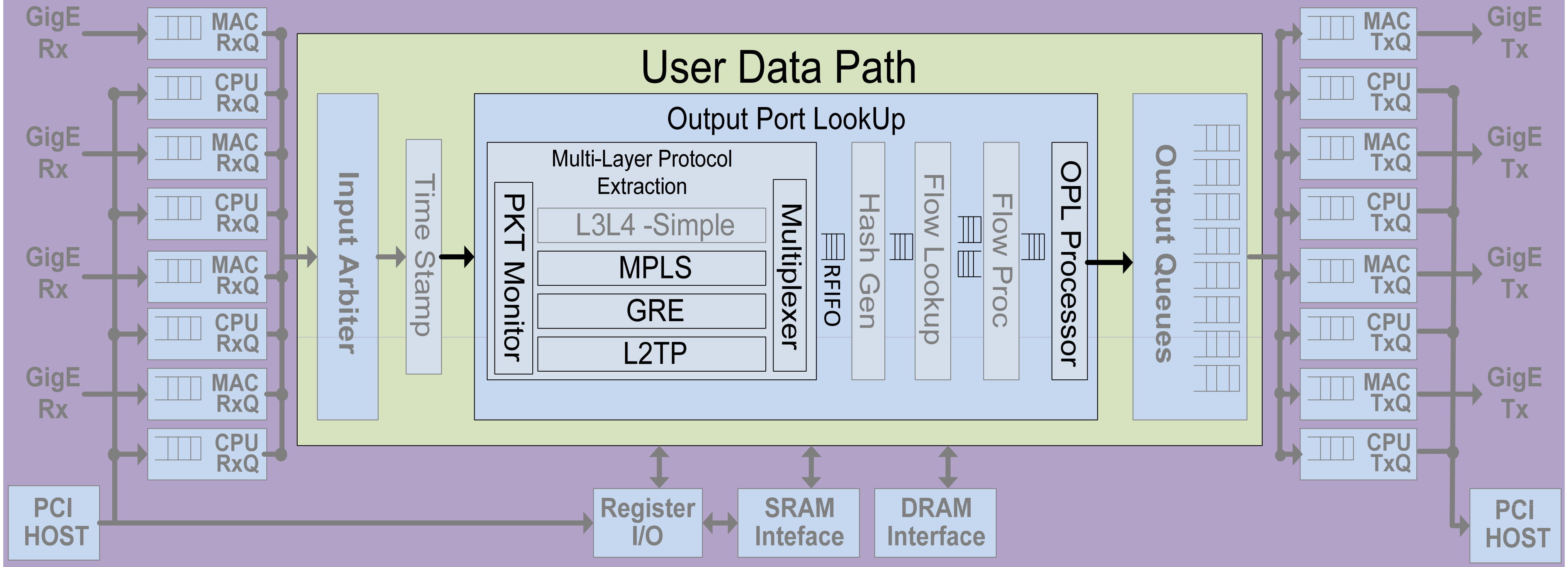


Image From Wikipedia, NetFlow

Virtual Interfaces in NetFlow

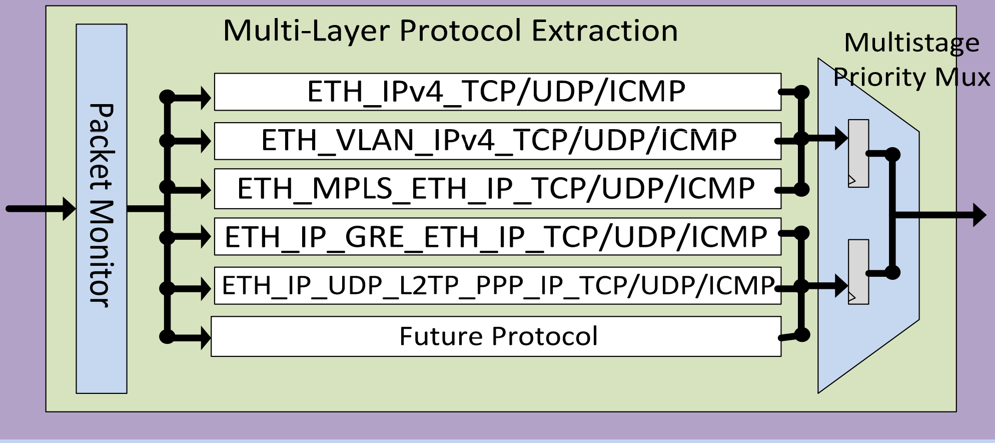
- Virtual interfaces are usually found in technologies like
  - Layer 2 Tunneling Protocol (L2TP)
  - Generic Routing Encapsulation (GRE) tunnels
  - Multiprotocol Label Switching over Virtual Private Network (MPLS-VPN)
- Collect network flow information from L2TP, GRE and MPLS enabled networks

Hardware Architecture of NetFlow Probe



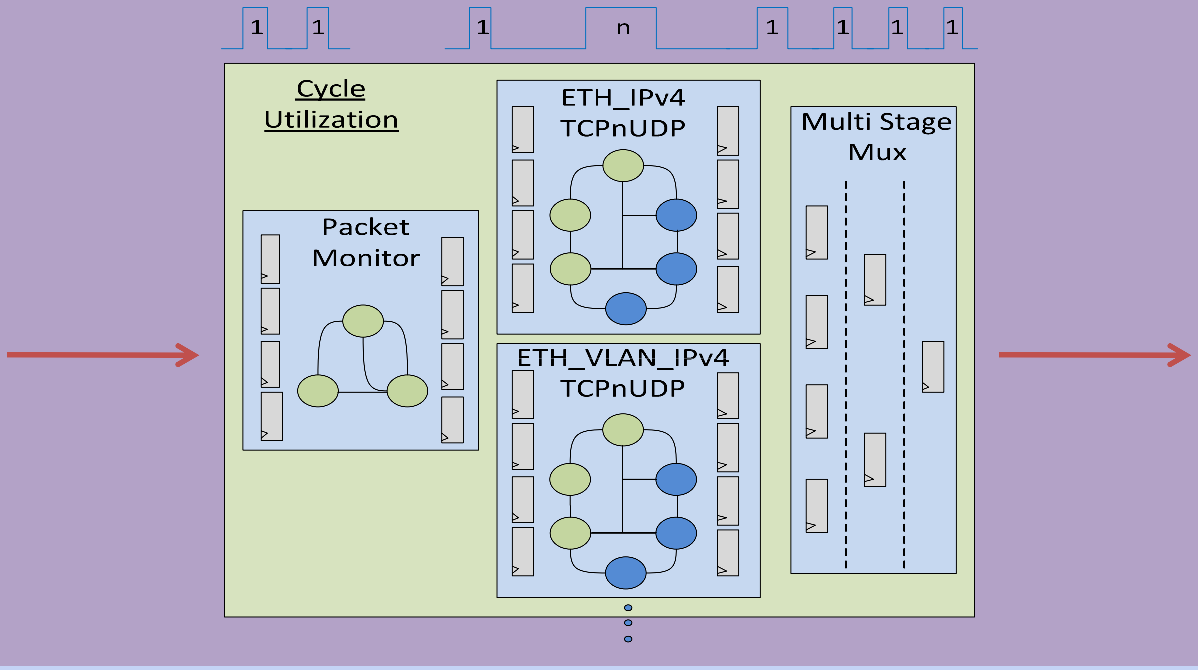
Multi-Layer Protocol Extraction Block

- Composed of following protocols:
  - original L3/L4 block provided with the reference NetFlow design
  - MPLS block for the extraction of multi protocol label switched packets with support for only two labels,
  - GRE block for the parsing of GRE encapsulated protocol packets
  - L2TP block for mining layer 2 tunneled PPP packets
  - Future protocols.
- Architecture of Multi-Layer Protocol Extraction consists of
  - Packet Monitor that tracks the state of the packet during the extraction process
  - a configurable stack of protocol combinations
  - a Multi-stage Priority Multiplexer.


Multi-Layer Protocol Extraction
Packet Monitor
ETH_IPv4_TCP/UDP/ICMP
ETH_VLAN_IPv4_TCP/UDP/ICMP
ETH_MPLS_ETH_IP_TCP/UDP/ICMP
ETH_IP_GRE_ETH_IP_TCP/UDP/ICMP
ETH_IP_UDP_L2TP_PPP_IP_TCP/UDP/ICMP
Future Protocol
Multistage Priority Mux

Multi-Layer Protocol Extraction Block (contd.)

- Multi-Stage Protocol Extraction Pipeline
  - Packet Monitor broadcasts packet words to all components with a latency of 2 cycles
  - Header Information extracted after *n+2* cycles delay
  - *n* is either total number of words taken by the protocol combination with largest header **or** size of incoming packet which ever is smaller.
  - Total latency for the example shown below is be **n+7** cycles

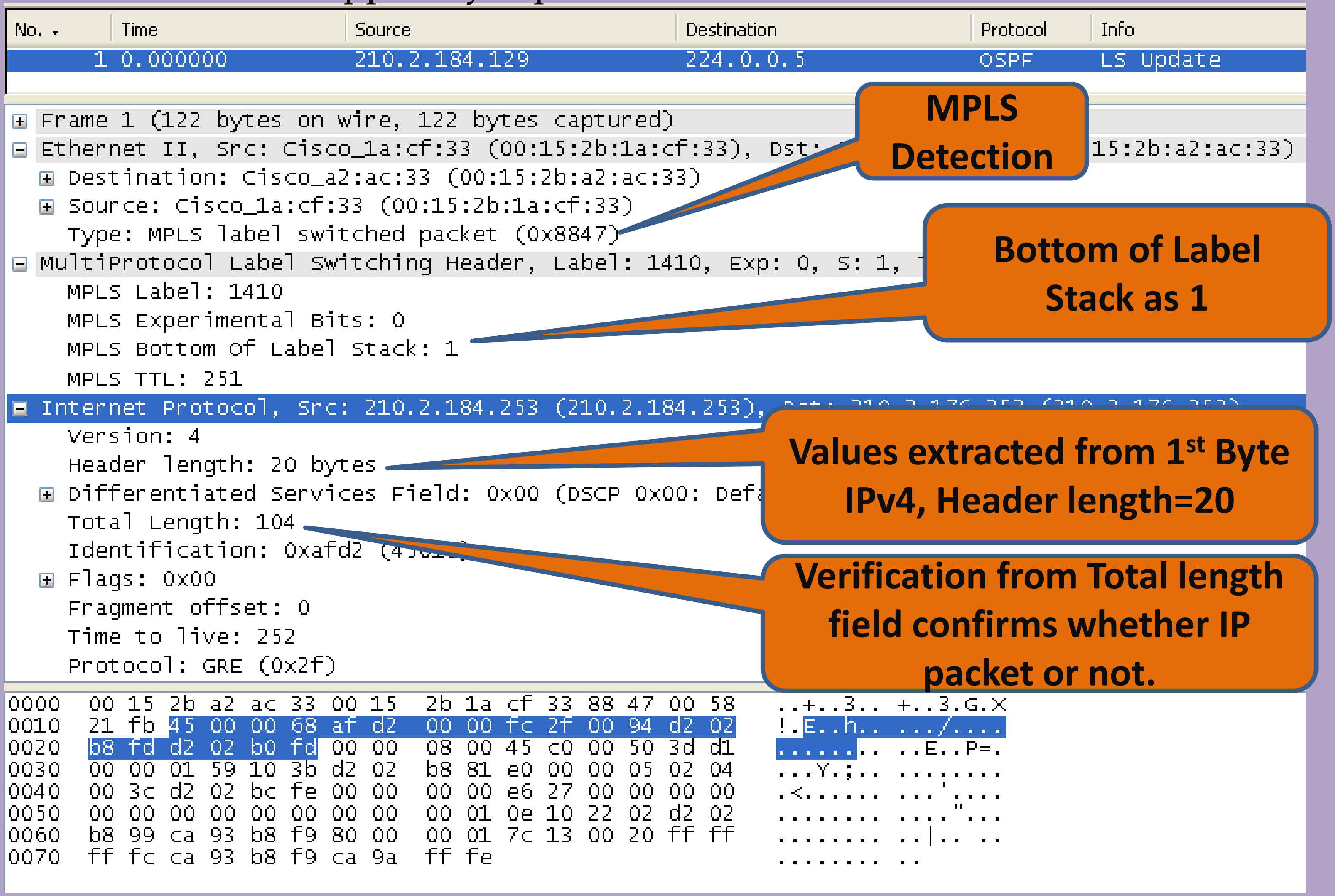


Multi-Layer Protocol Extraction Block (contd.)

- MPLS Decoding and Extraction
  - Multiple Protocol Label Switching (MPLS) tunnels are detected based on lower layer protocol type field as 0x8847
  - Detection of Upper Layer Protocol is not defined in MPLS Standard Documents
  - Upper Layer Protocols are detected based on byte pattern detection and verification.
  - Currently IP Protocol Detection is supported as MPLS upper layer protocol.
  - Flow for IP Detection
    - Check for final MPLS header from 'Bottom of Label Stack' field
    - Check top nibble of first byte after MPLS header (0x4 for IPv4 and 0x6 for IPv6).
    - Check Lower Nibble as Header Length
    - Treat it as IPv4 or IPv6 Packet and verify Length of remaining packet from expected Total Length field of IP header
    - If verified, Upper Layer Protocol is IPv4 or IPv6
    - Else It is treated as Ethernet packet
  - Support for Any other Protocol above MPLS can be very easily added due to the scalable architecture of the design.

Multi-Layer Protocol Extraction Block (contd.)

- MPLS with IP as upper layer protocol



```
No. -   Time       Source            Destination       Protocol  Info
    1 0.000000     210.2.184.129     224.0.0.5         OSPF      LS Update

⊞ Frame 1 (122 bytes on wire, 122 bytes captured)
⊟ Ethernet II, Src: Cisco_1a:cf:33 (00:15:2b:1a:cf:33), Dst:          15:2b:a2:ac:33)
  ⊞ Destination: Cisco_a2:ac:33 (00:15:2b:a2:ac:33)
  ⊞ Source: Cisco_1a:cf:33 (00:15:2b:1a:cf:33)
    Type: MPLS label switched packet (0x8847)
⊟ MultiProtocol Label Switching Header, Label: 1410, Exp: 0, S: 1, T
    MPLS Label: 1410
    MPLS Experimental Bits: 0
    MPLS Bottom Of Label Stack: 1
    MPLS TTL: 251
⊟ Internet Protocol, Src: 210.2.184.253 (210.2.184.253), Dst: 210.2.176.253 (210.2.176.253)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Def
    Total Length: 104
    Identification: 0xafd2 (45010)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 252
    Protocol: GRE (0x2f)

0000  00 15 2b a2 ac 33 00 15  2b 1a cf 33 88 47 00 58   ..+..3.. +..3.G.X
0010  21 fb 45 00 00 68 af d2  00 00 fc 2f 00 94 d2 02   !.E..h.. .../....
0020  b8 fd d2 02 b0 fd 00 00  08 00 45 c0 00 50 3d d1   ........ ..E..P=.
0030  00 00 01 59 10 3b d2 02  b8 81 e0 00 00 05 02 04   ...Y.;.. ........
0040  00 3c d2 02 bc fe 00 00  00 00 e6 27 00 00 00 00   .<...... ...'....
0050  00 00 00 00 00 00 00 00  00 01 0e 10 22 02 d2 02   ........ ...."...
0060  b8 99 ca 93 b8 f9 80 00  00 01 7c 13 00 20 ff ff   ........ ..|.. ..
0070  ff fc ca 93 b8 f9 ca 9a  ff fe                     ........ ..
```

MPLS Detection

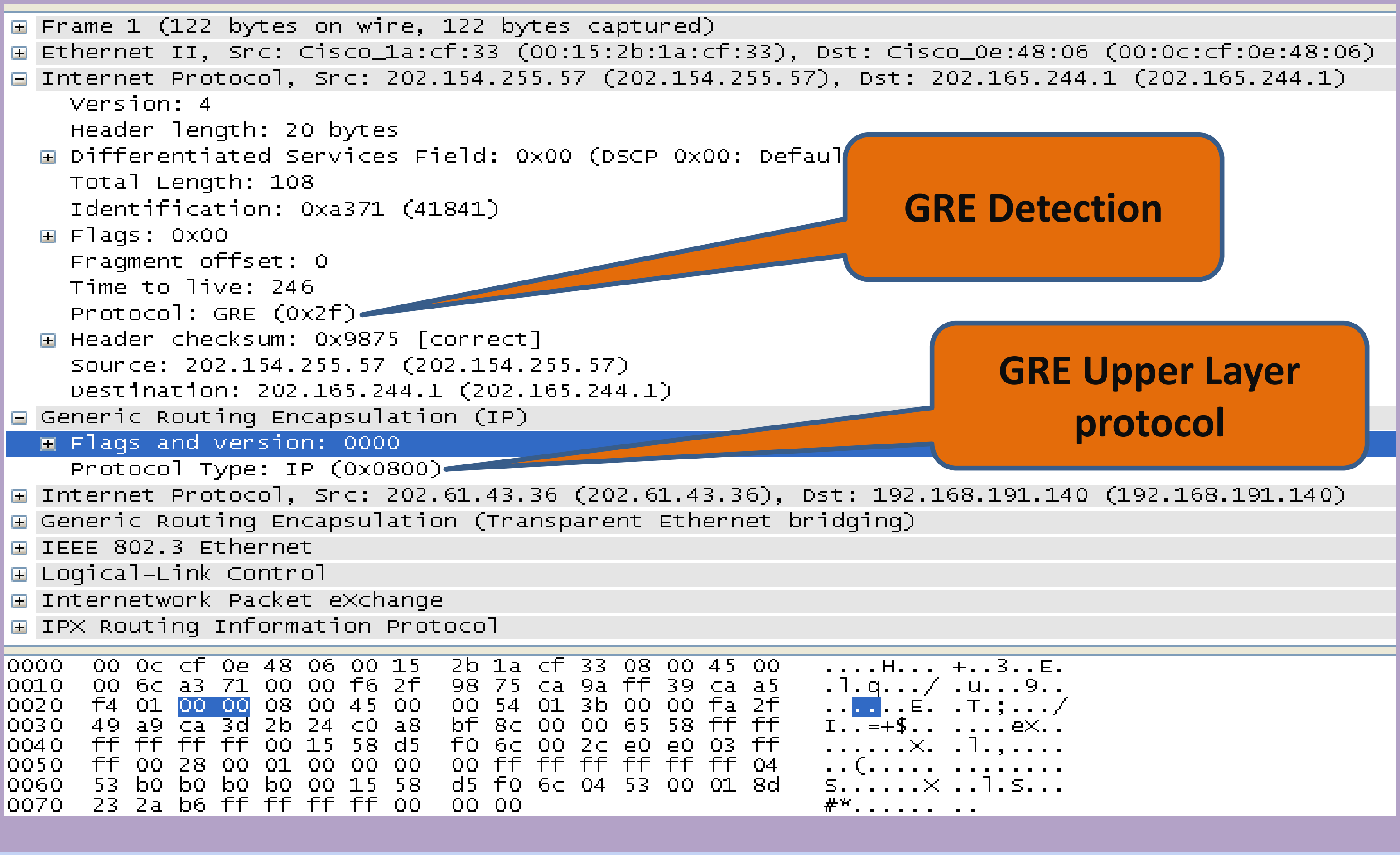
Bottom of Label Stack as 1

Values extracted from 1st Byte IPv4, Header length=20

Verification from Total length field confirms whether IP packet or not.

Multi-Layer Protocol Extraction Block (contd.)

- GRE Decoding and Extraction
  - Generic Routing Encapsulation (GRE) tunnels detected based on lower layer protocol type field as 0x2f
  - Sample GRE packet

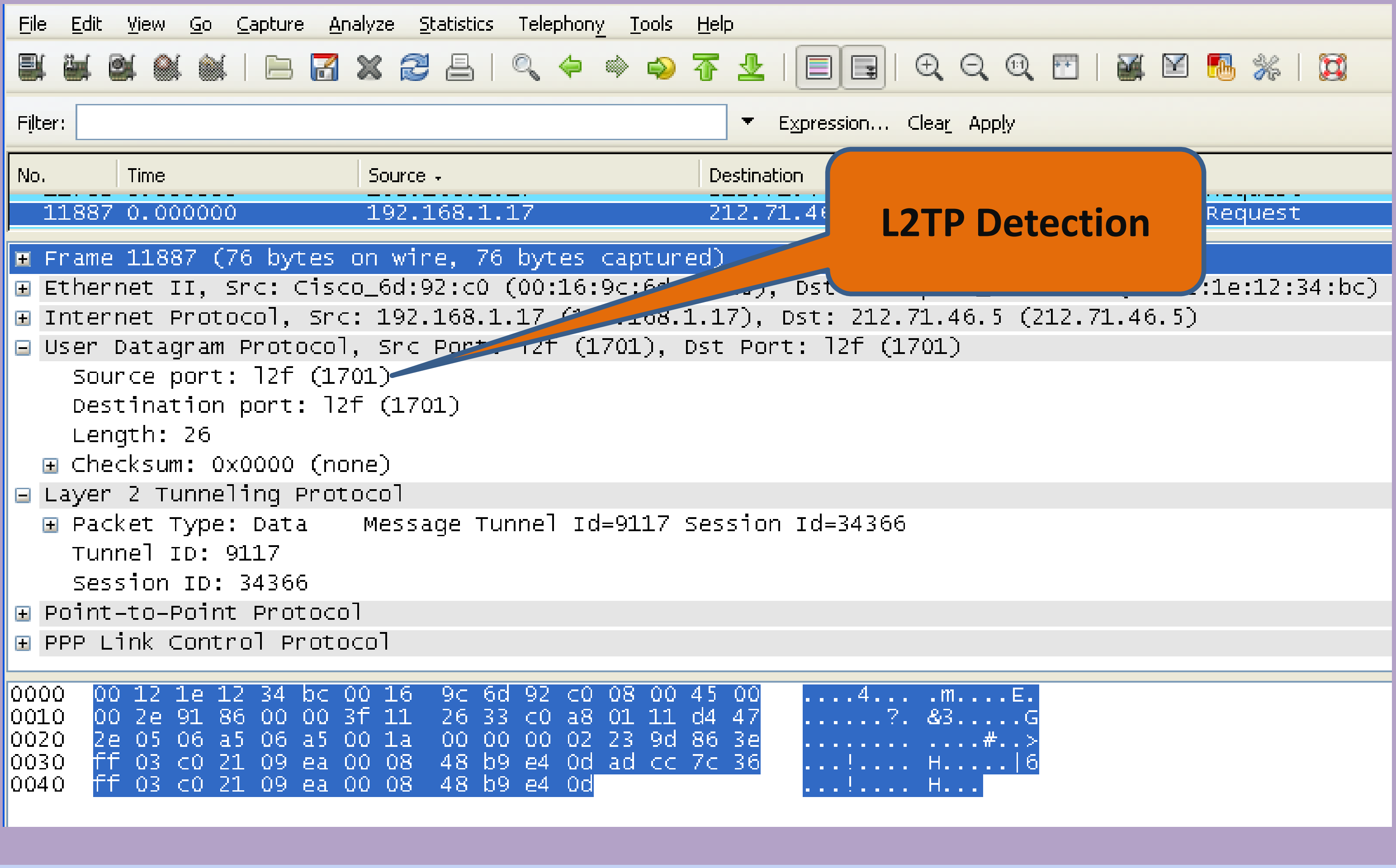


```
⊞ Frame 1 (122 bytes on wire, 122 bytes captured)
⊞ Ethernet II, Src: Cisco_1a:cf:33 (00:15:2b:1a:cf:33), Dst: Cisco_0e:48:06 (00:0c:cf:0e:48:06)
⊟ Internet Protocol, Src: 202.154.255.57 (202.154.255.57), Dst: 202.165.244.1 (202.165.244.1)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Defaul
    Total Length: 108
    Identification: 0xa371 (41841)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 246
    Protocol: GRE (0x2f)
  ⊞ Header checksum: 0x9875 [correct]
    Source: 202.154.255.57 (202.154.255.57)
    Destination: 202.165.244.1 (202.165.244.1)
⊟ Generic Routing Encapsulation (IP)
  ⊞ Flags and version: 0000
    Protocol Type: IP (0x0800)
⊞ Internet Protocol, Src: 202.61.43.36 (202.61.43.36), Dst: 192.168.191.140 (192.168.191.140)
⊞ Generic Routing Encapsulation (Transparent Ethernet bridging)
⊞ IEEE 802.3 Ethernet
⊞ Logical-Link Control
⊞ Internetwork Packet eXchange
⊞ IPX Routing Information Protocol

0000  00 0c cf 0e 48 06 00 15  2b 1a cf 33 08 00 45 00   ....H... +..3..E.
0010  00 6c a3 71 00 00 f6 2f  98 75 ca 9a ff 39 ca a5   .l.q.../ .u...9..
0020  f4 01 00 00 08 00 45 00  00 54 01 3b 00 00 fa 2f   ......E. .T.;.../
0030  49 a9 ca 3d 2b 24 c0 a8  bf 8c 00 00 65 58 ff ff   I..=+$.. ....eX..
0040  ff ff ff ff 00 15 58 d5  f0 6c 00 2c e0 e0 03 ff   ......X. .l.,....
0050  ff 00 28 00 01 00 00 00  00 ff ff ff ff ff ff 04   ..(..... ........
0060  53 b0 b0 b0 b0 00 15 58  d5 f0 6c 04 53 00 01 8d   S......X ..l.S...
0070  23 2a b6 ff ff ff ff 00  00 00                     #*...... ..
```

Multi-Layer Protocol Extraction Block (contd.)

- L2TP Decoding and Extraction
  - L2TP Decoding Performed only on UDP Port Number 1701
  - Sample L2TP Packet



File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply
No.
Time
Source
Destination

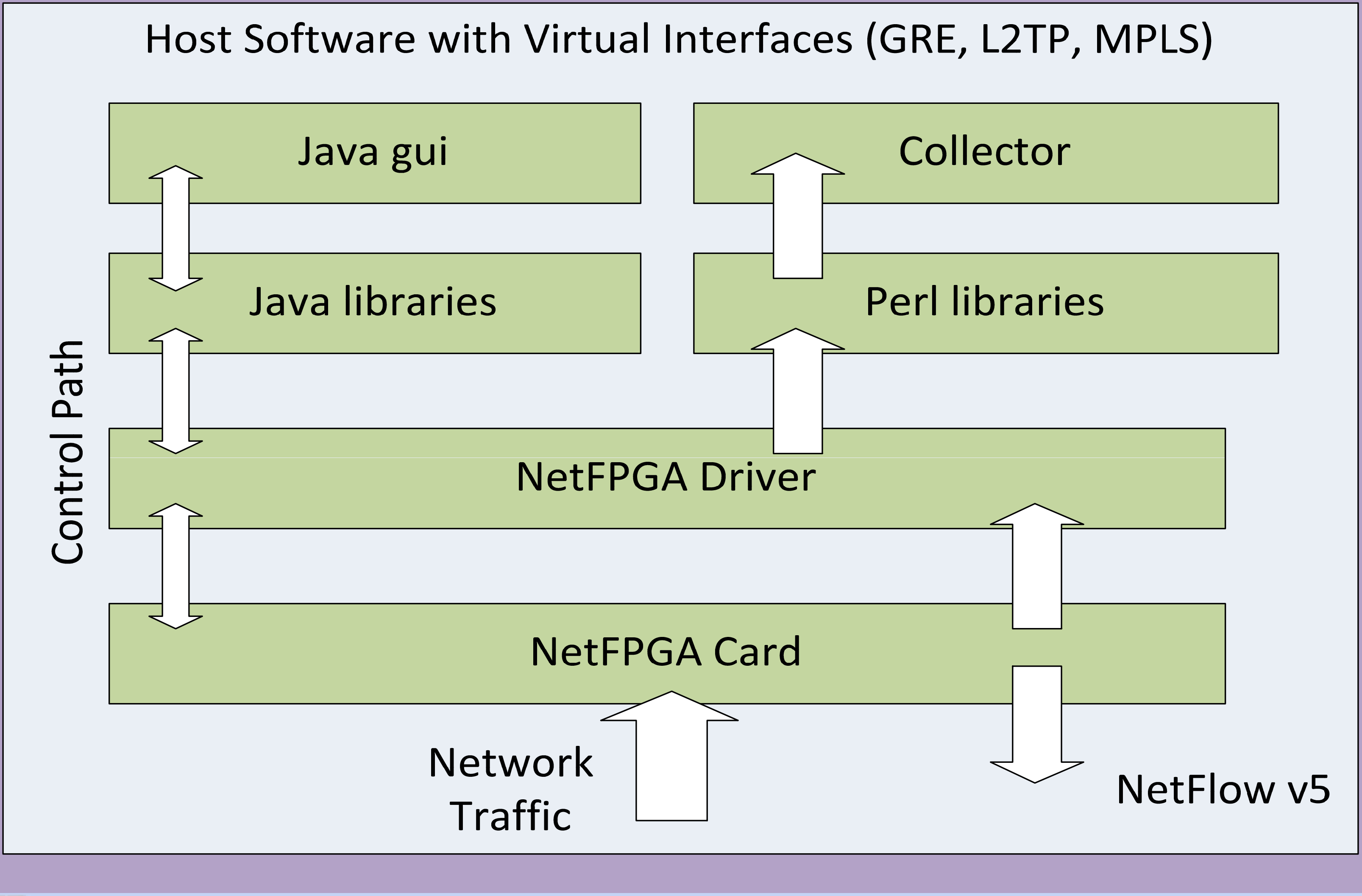
```
11887 0.000000      192.168.1.17      212.71.4                                Request
```

L2TP Detection

```
⊞ Frame 11887 (76 bytes on wire, 76 bytes captured)
⊞ Ethernet II, Src: Cisco_6d:92:c0 (00:16:9c:6d        ), Dst                        :1e:12:34:bc)
⊞ Internet Protocol, Src: 192.168.1.17 (1     168.1.17), Dst: 212.71.46.5 (212.71.46.5)
⊟ User Datagram Protocol, Src Port      l2f (1701), Dst Port: l2f (1701)
    Source port: l2f (1701)
    Destination port: l2f (1701)
    Length: 26
  ⊞ Checksum: 0x0000 (none)
⊟ Layer 2 Tunneling Protocol
  ⊞ Packet Type: Data    Message Tunnel Id=9117 Session Id=34366
    Tunnel ID: 9117
    Session ID: 34366
⊞ Point-to-Point Protocol
⊞ PPP Link Control Protocol
```

```
0000  00 12 1e 12 34 bc 00 16  9c 6d 92 c0 08 00 45 00   ....4... .m....E.
0010  00 2e 91 86 00 00 3f 11  26 33 c0 a8 01 11 d4 47   ......?. &3.....G
0020  2e 05 06 a5 06 a5 00 1a  00 00 00 02 23 9d 86 3e   ........ ....#..>
0030  ff 03 c0 21 09 ea 00 08  48 b9 e4 0d ad cc 7c 36   ...!.... H.....|6
0040  ff 03 c0 21 09 ea 00 08  48 b9 e4 0d               ...!.... H...
```

Software Architecture of NetFlow probe



Resource Utilization

- Only about 3% extra resources were used to incorporate the support for GRE, MPLS and L2TP Protocols

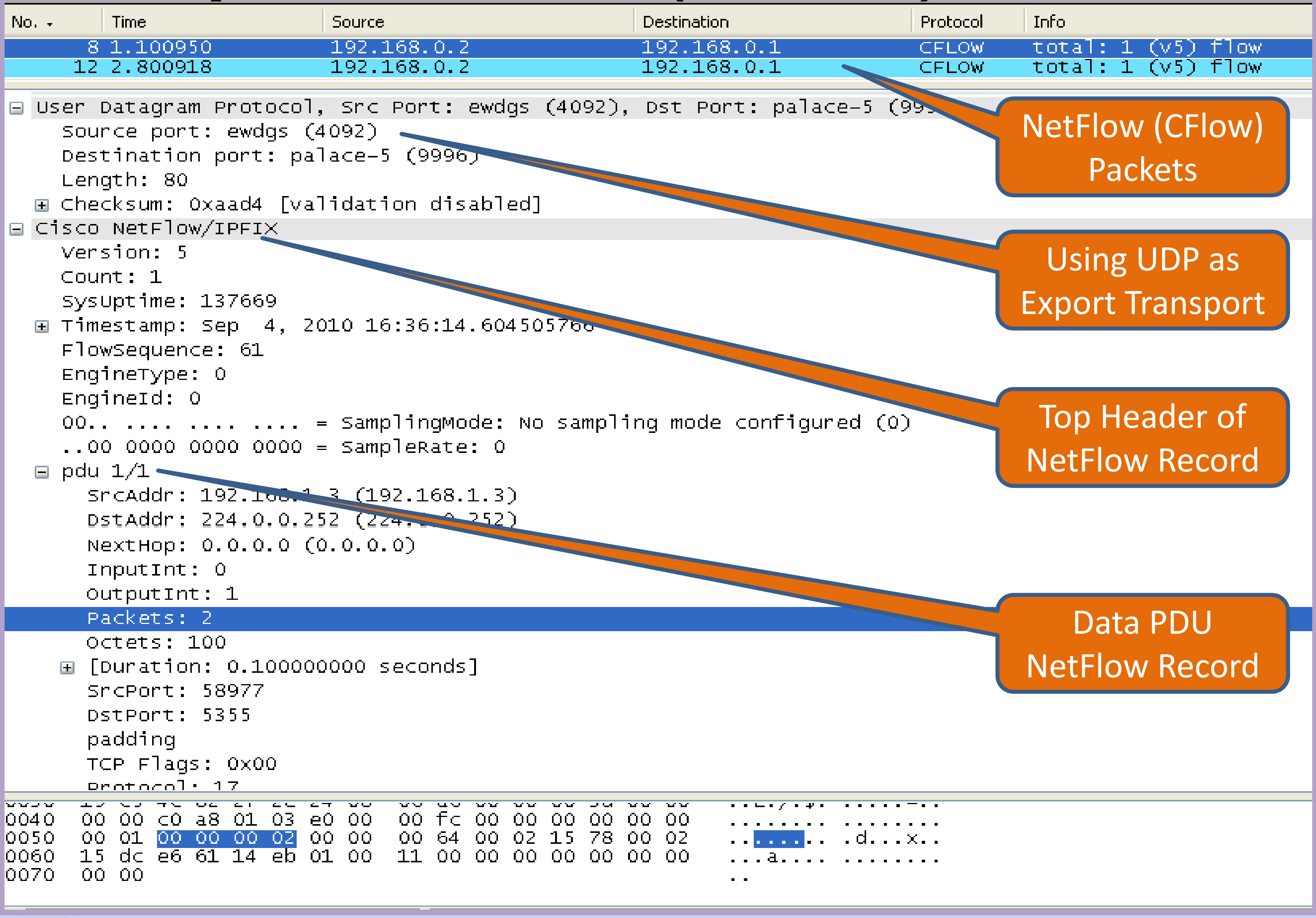
Table 1: Resource utilization of the Current Architecture with Virtual Interfaces (MPLS, L2TP, and GRE) plus l3l4 protocol

Resources	XC2VP50 Utilization	Utilization Percentage
Slices	18276 out of 23616	77%
4 - Input LUTS	25165 out of 47232	53%
Flip Flops	21244 out of 47232	44%
Block RAMs	200 out of 232	86%

Table 2: Resource utilization of the Original NetFlow probe Architecture with only l3l4 protocol

Resources	XC2VP50 Utilization	Utilization Percentage
Slices	17617 out of 23616	74%
4 - Input LUTS	23319 out of 47232	49%
Flip Flops	19504 out of 47232	41%
Block RAMs	200 out of 232	86%

Sample Netflow (Cflow) Record



```
No. -   Time        Source          Destination     Protocol  Info
      8 1.100950    192.168.0.2     192.168.0.1     CFLOW     total: 1 (v5) flow
     12 2.800918    192.168.0.2     192.168.0.1     CFLOW     total: 1 (v5) flow

⊟ User Datagram Protocol, Src Port: ewdgs (4092), Dst Port: palace-5 (99
    Source port: ewdgs (4092)
    Destination port: palace-5 (9996)
    Length: 80
  ⊞ Checksum: 0xaad4 [validation disabled]
⊟ Cisco NetFlow/IPFIX
    Version: 5
    Count: 1
    SysUptime: 137669
  ⊞ Timestamp: Sep  4, 2010 16:36:14.604505760
    FlowSequence: 61
    EngineType: 0
    EngineId: 0
    00.. .... .... .... = SamplingMode: No sampling mode configured (0)
    ..00 0000 0000 0000 = SampleRate: 0
  ⊟ pdu 1/1
      SrcAddr: 192.168.1.3 (192.168.1.3)
      DstAddr: 224.0.0.252 (224.0.0.252)
      NextHop: 0.0.0.0 (0.0.0.0)
      InputInt: 0
      OutputInt: 1
      Packets: 2
      Octets: 100
    ⊞ [Duration: 0.100000000 seconds]
      SrcPort: 58977
      DstPort: 5355
      padding
      TCP Flags: 0x00
      Protocol: 17

0040  00 00 c0 a8 01 03 e0 00  00 fc 00 00 00 00 00 00   ........ ........
0050  00 01 00 00 00 02 00 00  00 64 00 02 15 78 00 02   ........ .d...x..
0060  15 dc e6 61 14 eb 01 00  11 00 00 00 00 00 00 00   ...a.... ........
0070  00 00                                              ..
```

NetFlow (CFlow) Packets

Using UDP as Export Transport

Top Header of NetFlow Record

Data PDU NetFlow Record

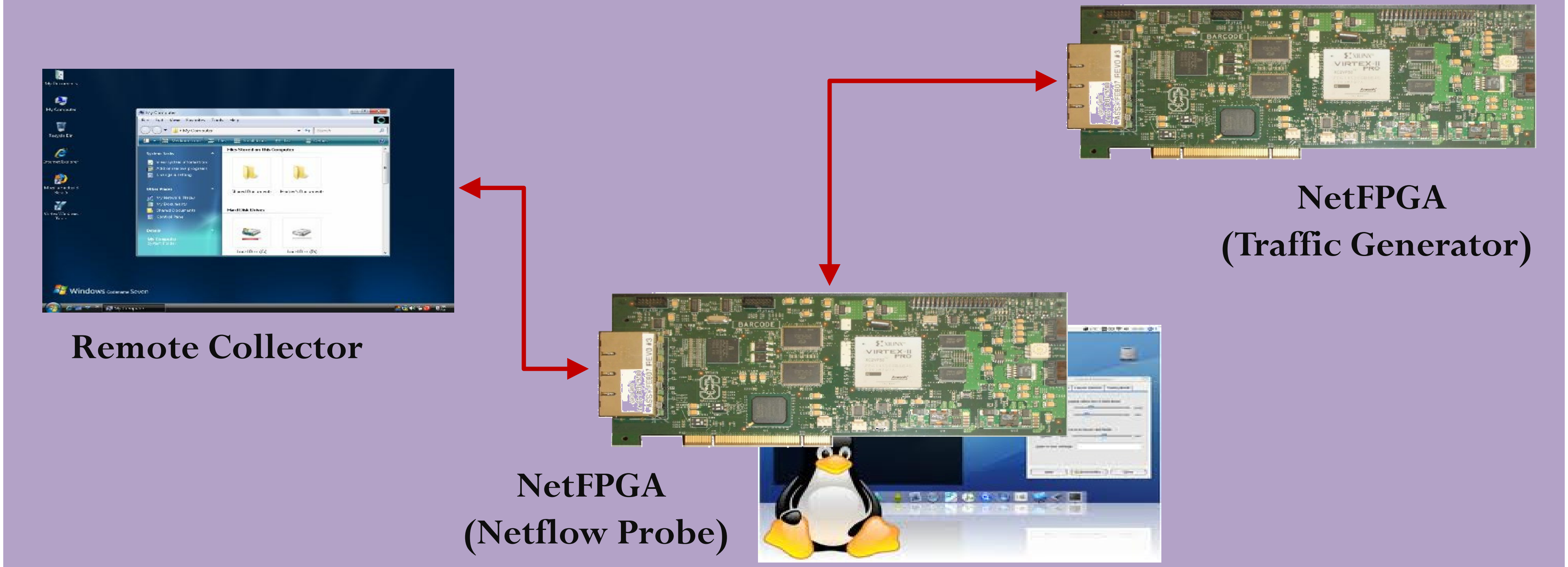
Extended Applications

- Deep Packet Inspection (DPI) based VoIP Monitoring
  - Telecom Regulatory Authorities Perspective
    - VoIP Header and RTP Monitoring for illegalV oIP Identification and Mitigation
  - Scalability Tested for upto 40G Data Rates using HighTech Global Cards
  - Flexible Protocol addition

Conclusion

- Presented a generic protocol extraction layer for Netflow probe architecture
- Primary focus on extraction mechanism for technologies supporting virtual interfaces i.e. MPLS, L2TP and GRE
- The architecture finds applications in
  - Deep Packet Inspection (DPI)
  - Voice over IP (VoIP) monitoring
  - Accounting /Billing

System Demonstration Setup for Multi-Gigabit networks

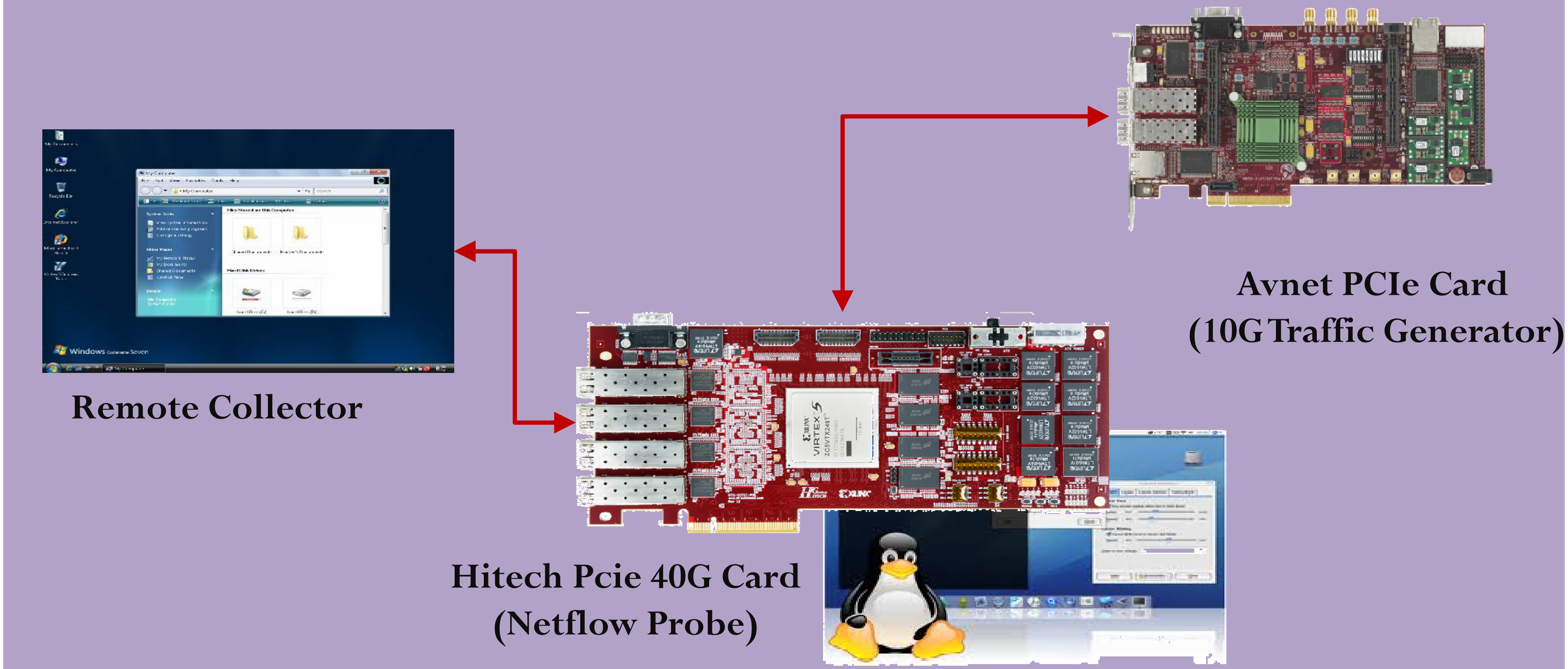


Remote Collector

NetFPGA
(Netflow Probe)

NetFPGA
(Traffic Generator)

System Demonstration Setup for Multi-10Gigabit networks



Avnet PCIe Card (10G Traffic Generator)

Remote Collector

Hitech Pcie 40G Card (Netflow Probe)

Questions / Answers