

Deployment of a Virtual Private LAN Service using Ethernet over MPLS technology

Actual Date: 28/06/06
Dissemination Level: Public
Document Code: GARR-06-01
Authors: Laura Leone (GARR) Laura.leone@garr.it
Stefano Suin (SERRA) stefano@unipi.it

Abstract

The document describes testing of a Virtual Private LAN Service realized using the Ethernet over MPLS framework of the MAN of the University of Pisa in cooperation with GARR. The testbed setup have been operational in the production network for about two months. The production network and related configured services have not been affected in the functionalities. Particular applications in the MAN framework, such as community separation and centralized authentication service per community have been improved in the technicalities. VPLS is a way to provide on a general purpose IP network the equivalent of geographical distributed, standalone, multipoint broadcast domains. The case study has been performed in a single domain environment and using an implementation of the technology on the equipment from a single vendor.

Table of Contents

1	Executive Summary	2
2	Introduction to VPLS technology	3
3	VPLS technology overview	4
4	VPLS BGP-based provisioning	6
5	VPLS implementation on IP/MPLS SERRA backbone	7
6	VPLS scalability (max number of storable MAC addresses)	8
7	Filter VPLS based	9
8	Example of VPLS production services deployment	10
	8.1 Communities segregation	10
	8.2 MAN traffic video	11
9	Conclusions	12
10	Acknowledgements	12
11	References	13
12	Acronyms	14

Table of Figures

Figure 1: VPLS Broadcast domain	5
Figure 2: SeRRA Setup	8
Figure 3: Hub& Spoke Topology	11

1 Executive Summary

The document describes testing of a Virtual Private LAN Service realized using the Ethernet over MPLS framework of the MAN of the University of Pisa in cooperation with GARR.

In the last few years, Internet evolved from the traditional client-server model, to new communication sharing models, making available many different services which requirements are based on multipoint to multipoint connectivity (e.g. peer to peer applications, service location protocols). Moreover, the rapid migration to IP-based services, a ubiquitous enterprise access need to both public & private services and Internet as the new communication media, require a technology able to offer performance similar to traditional ATM/Frame Relay services: overlay network, predictable performance with Service Level Agreements, CoS/QoS, Traffic Engineering, traffic protection.

Traditional WAN circuit oriented technologies (e.g. F.R., ATM), based on Hub&Spoke topology, put forward the requirements of the new model. This new technology requires support for native IP, a simple User to Network multiservice Interface (UNI), simple Network-Network Interface (NNI) transport interface, great bandwidth availability at cheap cost, multipoint to multipoint connectivity.

A first, basic, answer is the simple, multipoint, architecture of the Ethernet technology, transposed in the context of a Wide Area Network.

Many benefits can be found using Ethernet based technologies, and between the others, the availability of logical interfaces (802.1q), flexible bandwidth provisioning up to 10Gb at affordable prices when compared to traditional transport technology, usage of an extremely wide used LAN protocol and therefore homogeneous and completely integrated with the LAN, which leads to a "very common" way of handling it. All these benefits, and others such the general availability of high speed Ethernet at cheaper costs, even inside small/medium networks, have brought this technology to become largely used also in the telco market.

Metro-Ethernet become the first step for Ethernet out of the LAN environment, providing an answer many requirements. The old solid and well-known Ethernet protocol has been integrated with extremely new technologies such as hierarchical QoS, traffic shaping, mirroring, sampling, point-to-point and multi-point to multi-point circuits etc.

The great diffusion of Ethernet technology as WAN transport was strongly supported by the possibility for any kind of customer to build their own private metropolitan/geographical layer 2 networks.

Anyway there are some known limits to this technology. The most important are:

- the number of VLANs of the single ethernet domain is constrained in 12 bits, giving a small number of available VLANs (4096 or 2^{12}); also the usage of Q-in-Q technology for

distinguishing not only the single community but also different traffics on the backbone for any of those, is only a partial answer to this needs.

- Scalability of the Spanning Tree protocol (IEEE 802.1d) towards TE and redundancy problems. Even carrier class protection, today is based on proprietary technology and available on "ring topology" only) (eg. no traffic engineering, no benefit from meshed paths, preemption strategies with independent paths, redundancy, protection, etc.).
- Learning rate of the Ethernet MAC address, which becomes fundamental to minimize traffic dependant from unknown MAC addresses

All that said, the idea is to virtualize the very same broadcast domain to have multi-point to multi-point networks, but built on L3 topology. This framework has to be scalable, adaptable and redundable to a geographical network topology: in one word VPLS on IP/MPLS.

To get first hands-on experince with these recent developments, SERRA and GARR created the testbed setup described in this document, which has been operational in the SERRA production network for about two months.

2 Introduction to VPLS technology

The primary objective of the emergent standard for VPLS (Virtual private LAN Service) is the multipoint interconnection of users located in geographically distributed sites as if they were linked by a single (Ethernet) LAN. It is then possible to define logically separated virtual private LANs for different user communities, which share the same physical infrastructure and apply to each different security and management policies.

Until now, there are few cases of implementation and test of such emulation of layer 2 links.. This document describes configuration and testing of VPLS technology in the production environment of the IP/MPLS MAN of Pisa University, done in collaboration with GARR. The Pisa MAN is based on a private optical infrastructure connecting all the city areas and its suburbs on an area of 30 km of diameter. The Pisa MAN is used to connect all local research and education entities to GARR. The University of Pisa is connected to the GARR backbone through a Gigabit Ethernet link. In this scenario a VPLS service has been setup and tested using a Juniper M7i. The Pisa MAN is fully served by Juniper equipment, so that the test environment is based on a single vendor.

Of the two existing IETF drafts specification for VPLS creation, the one produced by Kompella et al [ref...] has been implemented. This draft uses the traffic engineering capabilities of the MPLS protocol and MP-BGP protocol to perform signalling and topology autodiscovery. MP-BGP is usually the protocol used to exchange VPN-IPv4 routes across a network based on IP and MPLS. Even if this VPLS implementation technology requires IP/MPLS and the BGP protocol the configuration effort is small. In this case study we show that VPLS is not complex to implement and it provides stable and robust configurations. Future tests of this technology will investigate the implementation of traffic classification and guarantees and interdomain communications. The implementation examined in this document is a single domain configuration.

The document has two sections: the first describes the setup and testing results,, the second, in form of Appendixes reports the equipment configuration and troubleshooting commands on the JunOS platform.

3 VPLS technology overview

The Virtual Private LAN Service Network (VPLS) or Transparent LAN Service (TLS) allows a layer 2 multipoint Ethernet connectivity among LAN geographically distributed over a shared physical IP/MPLS backbone. The technology is based on the Layer 2 VPN framework, which transports MPLS frames through a full-mesh of Label Switched Path (LSP) interconnecting Provider Edge (PE) routers of the backbone on which are enabled VPLS instances. The Customer Edge (CE) routers of the users through the Ethernet connection to the router PE in “hub and spoke” mode, become part of the layer 2 VPLS domain. A VPLS is at level 2 what a BGP MPLS VPN (rfc2547bis) realizes at layer 3, that is a layer 2 Private Virtual Network (VPN), any to any. A PE router behaves as “learning bridge” for the router MAC addresses of the users so that the backbone acts like a VPN Ethernet multipoint network.

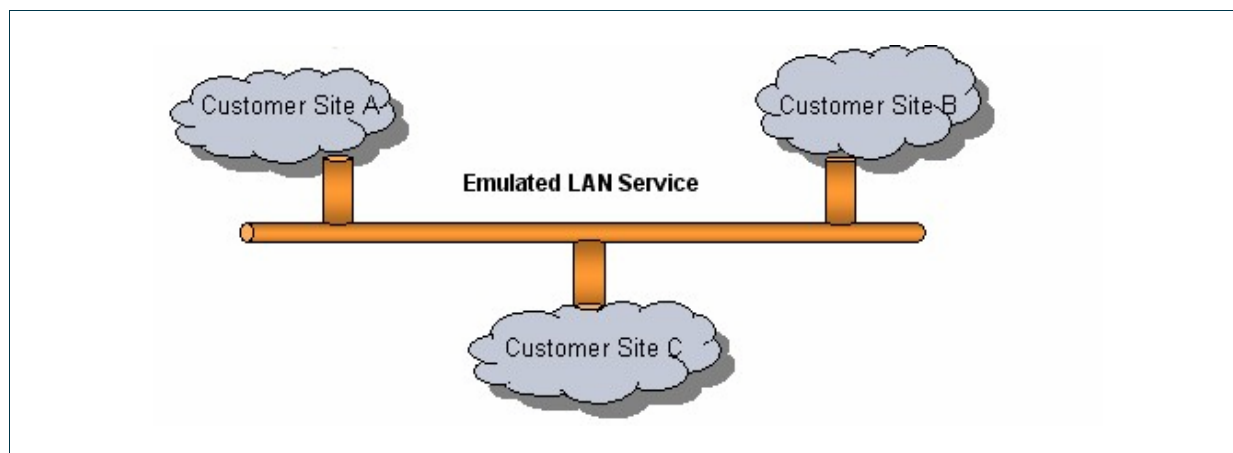


Figure 1: VPLS Broadcast domain

Layer 2 information are typically managed by switches in LAN domains. In a VPLS framework a PE act as switches, storing MAC/port information in VPLS table. The key difference between a physical and a virtual LAN is the creation of “Virtual Ports” associated to routers, which store also outgoing and incoming labels to find the path towards remote sites and to identify the received traffic. The traffic is forwarded across the backbone using the appropriate LSP. The VPLS provisioning specifications are defined in IETF drafts draft-kompella-ppvnp-vpls-01 [<http://www.ietf.org/internet-drafts/draft-Kompella-ppvnp-vpls-02.txt>] and in draft-lasserre-vkompella-ppvnp-vpls-02 [<http://www.ietf.org/internet-drafts/draft-lasserre-vkompella-ppvnp-vpls-01.txt>]. The two drafts differs in the definition of autodiscovery and signalling mechanisms. Auto-Discovery is the method used to enable PEs, participating in a VPLS domain, to discover one each other. Signalling is the protocol used to set up MPLS tunnels and distribute labels between PEs for packet demultiplexing purpose. The two drafts mentioned above differ in the definition of this two mechanisms. The Draft Kompella proposes BGP for both mechanisms. Draft Lassere-VKompella proposes instead to do not use the auto discovery mechanisms and to use the LDP protocol for signalling purpose. The testbed used in this document implements the Kompella one.

To make MAC address known to the remote sites, PE routers have to exchange labeled VPN-IPv4 routes stored on each PEs, for each VPLS instance, in the VPN Connection Table (VCT). MP-BGP is the protocol used to do perform the data exchange. In the Appendix A there is the summary of functionalities and the implementation details and nodes configurations commands.

In a IP/MPLS network with BGP protocol already configured, it is simpler to implement the Draft Kompella. Using BGP for both *autodiscovery* and *signalling* ensures synchronization between the discovery and signalling actions. Moreover the route-client BGP configuration helps in configuring Hierarchical VPLS model.

VPLS Technology pros :

- Cost-Effective: VPLS benefits of the low costs and simplicity of Ethernet technology using the MPLS scalability, reliability and functionalities.
- Multi-Protocol: VPLS transparently transports multiprotocol traffic

- Private Routing Domain: VPLS gives a transparent interface that allows private network customer routing
- Scalable bandwidth
- Capability of supporting high capacity bandwidth applications
- Competitive local access costs
- Quality of Service and Guarantee configurations per VPLS instance (VRF)

VPLS Technology cons :

- Scalability : it's not possible to summarize MAC address, as it's done with IP addresses therefore in a VPLS environment the maximum number of user's hosts participating in each instance has to be controlled. Each PE floods broadcast, multicast and unicast unrecognized Ethernet traffic to all the other PE participating in the service instance creating a possible overload and high bandwidth consumption on the equipment.. For this is the reason it's fundamental to reduce the number of VPLS-enabled PE.
- VPLS is not a Standard: there are two drafts describing the service, differing in the full-mesh setup model of the VC Tunnel among PE/VE in the IP/MPLS backbone : draft-ietf-l2vpn-vpls-ldp-01 and draft-ietf-l2vpn-vpls-bgp-01. The first one, supported by Cisco, implements LDP, the second, supported by Juniper, implements BGP for label signalling and distribution functionalities.

4 VPLS BGP-based provisioning

The following definitions are used in VPLS BGP-based provisioning:

- VPLS Edge ID (VE ID): Identifies the VE (VPLS Edge) involved in the VPLS instance (independently of the number of logical or physical involved interfaces)
- Route Distinguisher (RD): 6 Byte value inserted in every exchanged NLRI among PEs that uniquely identifies a VPLS instance. The RD (to be configured on every VE) is in the format ASNumber:Number where
 - ASNumber (2 Byte) identifies the AS

- Number (4 Byte) progressive ID associated to VPLS instance
- Instance Type: Identifies the particular routing-instance. In this case it is set to vpls (It could be l2vpn in MPLS L2/L2,5 VPN or vrf in IP/MPLS L3 VPN)
- Number of Sites: total number of VE involved in the VPLS instance considered.
- Route Target Import/Export: in VPN MPLS/IP framework generally determinates the VPN logical topology. In the VPLS case, RT Import/Export must have the same value and must be the same among all the participating PEs to create the full-mesh topology. In this way split-horizon is assured : traffic entering a VE is not back-propagate. Generally, for working purpose, it's preferred that the RT have the same value of RD.

5 VPLS implementation on IP/MPLS SeRRA backbone

MPLS is implemented on the University of Pisa MAN (SeRRA is the TLC operator) Traffic Engineering, Fast Reroute and the possibility to implement layer 2 and layer 3 VPN services have been achieved based on this existing MPLS framework.

University of Pisa and GARR collaborated in realizing a VPLS testbed using the production IP/MPLS MAN infrastructure. The VPLS testing phase has been carried out using a router Juniper M7i (jlab.unipi.it) connected to the production backbone. All the Pisa backbone is equipped with Juniper platforms. The created topology is reported in the following figure.

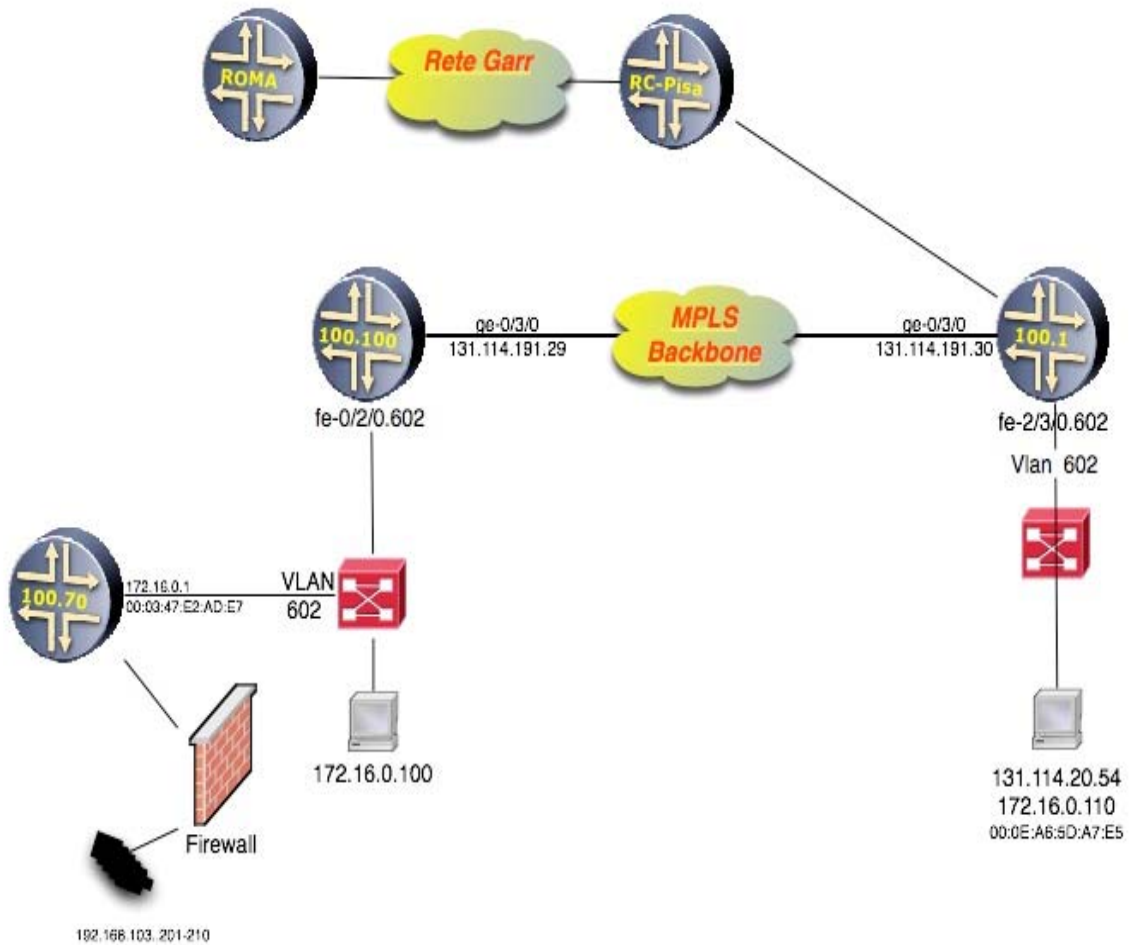


Figure 2: SeRRA Setup

6 VPLS scalability (max number of storable MAC addresses)

A serious issue related to the VPLS technology is the upper limit in the number of MAC address per VPLS instance, as a function of the Juniper equipment used. Exceeding the maximum allowed number of MAC-address causes the PE involved interface to go in "blocking" state. The PE router behaves as a "learning

bridge” for the CE router and creates a MAC table or Forwarding Information Base (FIB) storing the addresses received from each VPLS instance. Being the VPLS domain a geographical LAN, the number of addresses could increase disrupting the interface operation. In the case of SerRA setup it has been measured the maximum number supported of MAC addresses by the M7i Juniper used. The host 172.16.0.110 (00:0E:A6:5D:A7:E5) generated UDP traffic to the address 172.16.0.1 (00:03:47:E2:AD:E7) using the packETH (<http://packeth.sourceforge.net/>) traffic generator. The maximum measured number of MAC address is 64000 for each VPLS instance, with a maximum number of 2000 instances.

During the test, the interface, processed a large number of frames using all the 4 available hardware queue. The q0 queue manages the best effort and TCP-based protocols traffic. Injecting a lots of frame in the interface, the q0 went in “starvation” mode (saturation mode), and it was not able to process packets anymore. Two solution are applicable to this issue, one is limiting the maximum MAC-address number, the other is prioritizing the traffic using QoS mechanisms for each VPLS instance. The QoS implementation will be discussed in a different document. This document presentes the filter based solution.

7 Filter VPLS based

VPLS-based filters can be configured per VPLS instance or interface, the filters can be applied simultaneously.

A VPLS-based filter (Forwarding Table Filter or FTF) gives the possibility to limit the number of flooded MAC address (layer 2 broadcast, layer 2 multicast, layer 2 unicast with unknown destination MAC address), but only for each VPLS instance. The filter is applied to the Destination MAC (DMAC) Forwarding Table (or VFT – VPLS Forwarding Table) and is not related to the specific ingress as a physical/logical interface.

It is also possible to apply per interface filtering, but in the test performed we noticed the known MAC-address filtering works correctly, instead the multicast and broadcast flows are not supported (the policer counters give 0 values). Anti-flooding filters (Broadcast, Multicast, Unknown) have to be configured per instance.

8 Example of VPLS production services deployment

Different users communities are connected to the Pisa MAN and each of them has specific needs and associated policies. VPLS is applied as the solution to achieve segregation of communities giving to each of them a specific, dedicated broadcast domain : a specific VPLS instance.

Two meaningful example of this implementation are reported as case studies.

8.1 Communities segregation

Different communities, associated to a specific broadcast domain, have to be logically separated from each other and have different issues to be considered for the traffic management. In VPLS technology, PE routers are connected in full-mesh mode, but it's possible to create a hierarchical model to solve the issue related to the scalability. Using MP-BGP adjacencies is possible to create a "hub and spoke" setup in a single VPLS broadcast domain. In such a setup there are some "central" PEs (hub) maintaining all the information and other PEs (spokes) directly connected to them. All the "spokes" are not directly connected one to each other, but communicate through the "Hub". All the PEs participating in the same VPLS community instance continue to share the same Route Target identifying the specific community. The hub and spoke topology is depicted in Figure 3.

Jlab router (131.114.100.100) and Jfib router (131.114.100.30) are in the same broadcast domain while Jing router (131.114.100.50) is visible to them only through the router Jser (131.114.100.1) in Hub and Spoke mode. Such topology is created using BGP peerings and associated LSPs, between each PE node and the Jser router. Moreover there is an adjacency between router 131.114.100.30 and router 131.114.100.100..

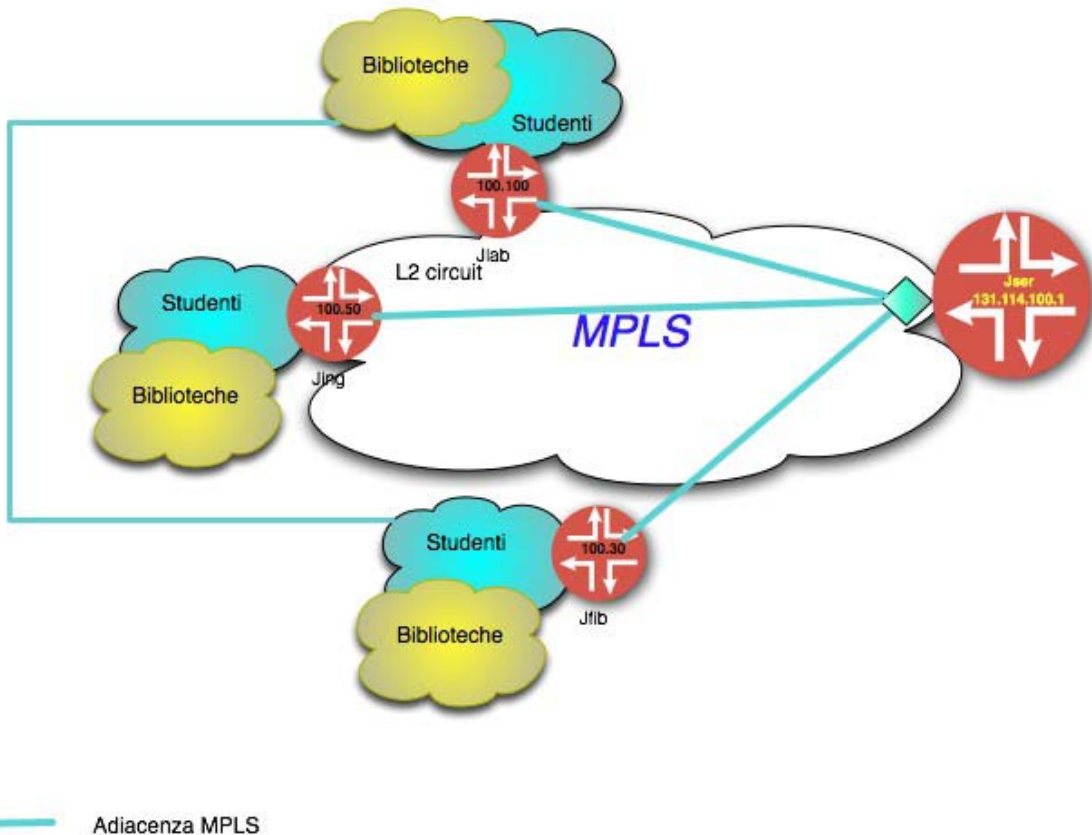


Figure 3: Hub& Spoke Topology

In the different VPLS instances, hosts have been included to capture the traffic flows and it has been verified the correct VPLS forwarding and traffic segregation.

8.2 MAN traffic video

In Pisa 8 digital videocameras have been installed to monitor city areas for the Civil Protection Agency. This traffic has been transported through the MAN backbone using a VPLS instance interconnecting the 8 video source. Television traffic flows are in PAL format and are distributed to a restricted critical community (police forces) geographically distributed on the MAN and included in the same VPLS instance of the Video source. Video traffic has strict requirements in terms of path delay and reliability. LSP Traffic Protection capabilities have been also tested, causing links to go down and rerouting the traffic on the backbone using MPLS Traffic Engineering features.

9 Conclusions

The main objective of technology VPLS (Virtual Private LAN Service) is the multipoint interconnection of customers geographically distributed using a level 2 Virtual Private Network (VPN) framework. The proposed technology couples the benefits of the high capacity on Ethernet technology with the scalability and the reliability features of MPLS technology. VPLS allows to supply to the customer with an Ethernet interface to use band not tied to the physical interface. It is possible to create logically separated users communities assuring traffic segregation among them as reported in the case study of this document. Logically segregated user communities sharing the same physical infrastructure benefit of security policy totally configured or imported from technology VPN of level 2. MP-BGP improves the scalability of the solution giving the possibility to implement a hierarchical architecture. Generalized MPLS (GMPLS) extends the MPLS technology to realize a control plan common to packets and circuits. As reported in the executive summary this technology could be evaluated in order to provide a simple User-Network-Interface (UNI) multiservice interface, simple Network-Network Interface (NNI) transport interface, great bandwidth availability at cheap cost, multipoint to multipoint connectivity.

Issues like traffic classification and guarantee and interdomain deployment will be covered in a future study.

10 Acknowledgements

Thanks to Mauro Campanella from INFN- GARR for the document review. Thanks to Paolo Caturegli from Centro SerRa for technical support. Thanks to Juniper Italia for the provisioning of M7i router for the experimentation, and for the technical support of Fabio Palozza and Alessandro Salesi.

11 References

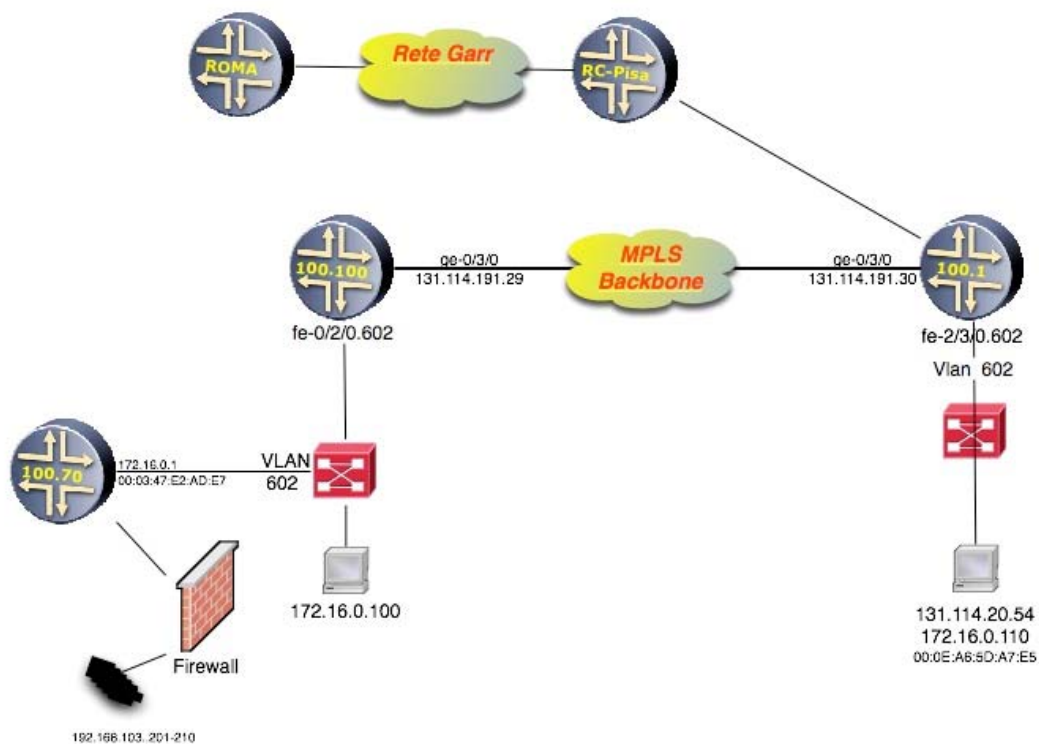
- [1] http://www.juniper.net/solutions/literature/white_papers/200045.pdf
- [2] <ftp://ftp.ietf.org/internet-drafts/draft-martini-l2circuit-trans-mpls-13.txt>
- [3] http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a008016102a.html#1130490
- [4] Building Core Networks with OSPF , BGP and MPLS Boot Camp. by Cisco System
- [5] MPLS Training course documentation by Fabio Palozza (Juniper Networks - Italy)
- [6] http://vpls.org/vpls_technical_overview.shtml

12 Acronyms

[PE]	[Provider Edge Router]
[CE]	[Customer Edge Router]
[LSP]	[Label Switched Path]
[VRF]	[Virtual Route Forwarding]
[TE]	[Traffic Engineering]

Appendix A Model Configuration and validation

A.1 Setup



The router used for the VPLS experimentation is the, Juniper M7i jlab.unipi.pi.it with loopback IP address 131.114.100.100

A.2 M7i Hardware configuration

```
stefano@jlab.unipi.it> show chassis hardware
Hardware inventory:
Item          Version Part number Serial number  Description
Chassis              39362      M7i
Midplane    REV 04  710-008761 CM7180      M7i Midplane
Power Supply 0 Rev 06  740-008537 5384422     AC Power Supply
Routing Engine REV 09  740-009459 1000586008 RE-5.0
CFEB             REV 07  750-010464 CM1598      Internet Processor II
FPC 0
  PIC 2          REV 11  750-002992 CM6749      4x F/E, 100 BASE-TX
  PIC 3          REV 09  750-007641 CM8939      1x G/E IQ, 1000 BASE
  SFP 0          REV 01  740-011782 P6S13AH     SFP-SX
FPC 1
  PIC 2          REV 07  750-009487 CM1909      ASP - Integrated
  PIC 3          REV 08  750-009099 CM6167      1x G/E, 1000 BASE
```

To deploy VPLS services a Service Tunnel PIC is required.

A.3 Software configuration

A.3.1 MP-iBGP configuration for L2VPN routes exchanging

```
local-as 64861;
group ibgp {
  type internal;
  family inet-vpn {
    unicast;
  }
  family l2vpn {
    unicast;
  }
  neighbor 131.114.100.1;
}
```

A.3.2 Interface configuration

On the existing IP/MPLS framework VPLS implies the configuration of tagged logical interface included in the VPLS family.

```

interfaces {
  fe-1/3/1 {
    vlan-tagging;
    encapsulation vlan-vpls;
    unit 89 {
      vlan-id 89;
      family inet {
        address 172.16.16.1/24;
      }
    }
    unit 90 {
      vlan-id 90;
      family inet {
        address 172.30.30.1/24;
      }
    }
    unit 602 {
      encapsulation vlan-vpls;
      vlan-id 602;
      family vpls;
    }
  }
}

```

A.3.3 VPLS routing instance configuration

The VPLS routing instance is accomplished defining besides the involved logical interface also Route Target and Route Distinguisher values. There is also the VPLS-based filter configuration.

```

routing-instances {
  VPN-VPLS-A {
    instance-type vpls;
    interface fe-1/3/1.602;
    route-distinguisher 64861:1;
    vrf-target target:64861:1;
  }
}
forwarding-options {
  family vpls {
    flood {
      inut Broadcast-Multicast-limit:

```

A.3.4 VPLS protocol configuration

VPLS protocol configuration with definition of MAC table size and aging-time

```
protocols {
  vpls {
    site-range 5;
    mac-table-size 65535;
    mac-table-aging-time 300;
    site VPLS-A {
      site-identifier 1;
    }
  }
}
```

A.3.5 VPLS user communities configuration

```
routing-instances {
  VPLS-BIBLIO {
    instance-type vpls;
    interface fe-2/3/3.1000;
    route-distinguisher 64861:10;
    vrf-target target:64861:10;
    protocols {
      vpls {
        site-range 10;
        site VPLS-BIBLIO {
          site-identifier 1;
        }
      }
    }
  }
}

VPLS-STUDENTI {
  instance-type vpls;
  interface fe-2/3/3.1001;
  route-distinguisher 64861:11;
  vrf-target target:64861:11;
  protocols {
    vpls {
      site-range 10;
      site VPLS-STUDENTI {
        site-identifier 1;
      }
    }
  }
}
```

A.3.6 VPLS-based multicast and broadcast filter per VPLS instance

The policer unknown regards “unknown destination MAC” flows that will be regarded as Broadcast traffic

```

firewall {
  family vpls {
    filter Broadcast-Multicast-limit {
      term Broadcast-limit {
        from {
          destination-mac-address {
            ff:ff:ff:ff:ff:ff/48;
          }
        }
        then policer Broadcast-limit;
        count Broadcast-packet
      }
      term Multicast-limit {
        from {
          destination-mac-address {
            01:00:5e:00:00:00/25;
          }
        }
        then policer Multicast-limit;
        count Multicast-packet
      }
      term default {
        then policer Unknown;
      }
    }
  }
}

```

Filter policer definition

The burst-size-limit value is obtained from the multiplying of interface bandwidth value by the time duration in which a burst equal to the interface rate is accepted. Burst size should be at least 10 times the I/F MTU (at a maximum 100Mbps). In the specific case the Burst-Size is 1522 Byte * 10 = 15kB

```

firewall {
  policer Broadcast-limit {
    if-exceeding {
      bandwidth-limit 10m;
      burst-size-limit 15k;
    }
    then discard;
  }
  policer Multicast-limit {
    if-exceeding {
      bandwidth-limit 30m;
      burst-size-limit 15k;
    }
    then discard;
  }
  policer Unknown
  if-exceeding {
    bandwidth-limit 2m;
    burst-size-limit 15k;
  }
  then discard;
}

```

A.3.7 VPLS-based interface multicast and broadcast filter definition

Interface configuration on 131.114.100.100 router

```
unit 602 {
  description "VPLS client interface";
  encapsulation vlan-vpls;
  vlan-id 602;
  family vpls {
    filter {
      output Broadcast-Multicast-limit;
    }
  }
}
```

A.4 Functionalities monitoring and validation

Remote PE reachability check by using Service Tunnel PIC (virtual interface) on jlab and jser router

```
stefano@jlab.unipi.it> run show route table mpls.0

mpls.0: 7 destinations, 7 routes (7 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0          *[MPLS/0] 1w3d 21:11:59, metric 1
           Receive
1          *[MPLS/0] 1w3d 21:11:59, metric 1
           Receive
2          *[MPLS/0] 1w3d 21:11:59, metric 1
           Receive
800001     *[VPLS/7] 20:23:48 // input MPLS frame tagged with label 800001
           via vt-1/2/0.32768, Pop // MPLS label POP

vt-1/2/0.32768 *[VPLS/7] 20:23:48, metric2 10
              > to 131.114.191.30 via ge-0/3/0.0,
```

```
stefano@jser.unipi.it> run show route table mpls.0

vt-1/0/0.32768 (VPLS)
  user 0          indr 262174 7
        131.114.191.29 Push 800001 ge-0/3/0.0
```

A.4.1 Remote CE reachability check

Host 172.16.0.110 and host 172.16.0.100 are included in the same VPLS, in fact they share the same broadcast domain.

```
ping -d 172.16.0.1
PING 172.16.0.1 (172.16.0.1) 56(84) bytes of data.
64 bytes from 172.16.0.1: icmp_seq=1 ttl=64 time=0.255 ms
64 bytes from 172.16.0.1: icmp_seq=2 ttl=64 time=0.258 ms
64 bytes from 172.16.0.1: icmp_seq=3 ttl=64 time=0.260 ms

--- 172.16.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2036ms
rtt min/avg/max/mdev = 0.255/0.257/0.260/0.018 ms

<maccapc:root [~]# arp -a
? (172.16.0.2) at 00:30:05:65:CD:AF [ether] on eth1
sun-paolo.unipi.it (131.114.20.24) at 00:30:4F:18:11:23 [ether] on eth0
maccarone.unipi.it (131.114.20.40) at 00:03:93:AC:BA:4A [ether] on eth0
? (172.16.0.1) at 00:03:47:E2:AD:E7 [ether] on eth1
lan-serra-gw.unipi.it (131.114.20.1) at 00:04:96:18:4A:2A [ether] on eth0
```

A.4.2 Hosts included in the VPLS instance

```
[edit]
stefano@jlab.unipi.it# run show route forwarding-table family vpls | match ^00
00:03:47:e2:ad:e7/48
00:0e:a6:5d:a7:e5/48
00:30:05:65:cd:aa/48
00:30:05:65:cd:af/48
00:e0:fc:59:5f:40/48
00:e0:fc:5b:39:bc/48

[edit]
stefano@jlab.unipi.it# run show route forwarding-table family vpls | match ^00 | count
Count: 6 lines
```

A.4.3 MP-BGP peering validation

A.4.3.1 BGP summary

A.4.3.2 VPN route exchanging and announcements

```
stefano@jlab.unipi.it> show route receive-protocol bgp 131.114.100.1
VPN-VPLS-A.l2vpn.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
Prefix      Nexthop      MED  Lclpref  AS path
64861:1:2:1/96  131.114.100.1    100   I
```

```
VPN-VPLS-A.l2vpn.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

64861:1:1:1/96 // RD: Site-D of local PE
*[L2VPN/7] 1w5d 18:14:10
  Discard
64861:1:2:1/96 // RD: Site-ID of remote PE
*[BGP/170] 2d 17:23:56, localpref 100, from 131.114.100.1
  AS path: I
  > to 131.114.191.30 via ge-0/3/0.0, label-switched-path 2jser
```

```
stefano@jlab.unipi.it> show bgp summary
Groups: 2 Peers: 4 Down peers: 1
Table      Tot Paths  Act Paths  Suppressed  History  Damp State  Pending
bgp.l3vpn.0      4      4      0      0      0      0
bgp.l2vpn.0      1      1      0      0      0      0 // Tabella BGP relative alle rotte
inet.0          0      0      0      0      0      0
Peer          AS  InPkt  OutPkt  OutQ  Flaps  Last Up/Dwn State|#Active/Received/Damped...
131.114.100.1 64861  6171   6169   0     0     2d 3:22:04 Establ
  bgp.l3vpn.0: 4/4/0
  bgp.l2vpn.0: 1/1/0
  VPN-MPLS-A.inet.0: 2/2/0
  VPN-MPLS-B.inet.0: 2/2/0
VPN-VPLS-A.l2vpn.0: 1/1/0
```

A.4.4 VPLS instance summary

A.4.4.1 VPLS summary

```

stefano@jlab.unipi.it# run show vpls connections
Layer-2 VPN Connections:

Legend for connection status (St)
OR -- out of range      WE -- intf encaps != instance encaps
EI -- encapsulation invalid  Dn -- down
EM -- encapsulation mismatch  VC-Dn -- Virtual circuit down
CM -- control-word mismatch  -> -- only outbound conn is up
CN -- circuit not provisioned  <- -- only inbound conn is up
OL -- no outgoing label      Up -- operational
NC -- intf encaps not CCC/TCC  XX -- unknown
NP -- intf h/w not present

Legend for interface status
Up -- operational
Dn -- down

Instance: VPN-VPLS-A
Local site: VPLS-A (1)
  connection-site      Type St   Time last up      # Up trans
  2                    rmt  Up    Feb 9 14:39:50 2006      1
  Local interface: vt-1/2/0.32768, Status: Up, Encapsulation: VPLS
  Remote PE: 131.114.100.1, Negotiated control-word: No
  Incoming label: 800001, Outgoing label: 800000

[edit]
stefano@jlab.unipi.it#

```

A.4.5 VPLS forwarding table (MAC address storing)

On jlab router (131.114.100.100) host 172.16.0.110 MAC-address - **00:0e:a6:5d:a7:e5/48** is received on network interface ge-0/3/0.0 and encapsulated in the LSP with label 800000. Host MAC-address 172.16.0.1 - **00:30:05:65:cd:af/48** is received from the PE-CE fe-0/2/0.602 interface.

```

Stefano@jlab.unipi.it# run show route forwarding-table family vpls
Routing table: VPN-VPLS-A.vpls

VPLS:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          dynm  0      flood 411  1
default          perm  0      rjct 397  1
00:03:47:e2:ad:e7/48
00:0e:a6:5d:a7:e5/48      dynm  0      indir 262144  4 131.114.191.30  Push
800000          ge-0/3/0.0
00:30:05:65:cd:aa/48      dynm  0      ucst 412  6 fe-0/2/0.602
00:03:47:e2:ad:e7/48      dynm  0      ucst 412  6 fe-0/2/0.602
00:30:05:65:cd:af/48      dynm  0      ucst 412  6 fe-0/2/0.602
00:e0:fc:59:5f:40/48      dynm  0      ucst 412  6 fe-0/2/0.602
00:e0:fc:5b:39:bc/48      dynm  0      ucst 412  6 fe-0/2/0.602

```

```

run show route forwarding-table family vpls
Routing table: VPN-VPLS-B.vpls
VPLS:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          dymn  0      flood 469  1
default          perm  0      rjct 375  1
00:0e:a6:5d:a7:e5/48
800001          ge-0/3/0.0      dymn  0      ucst 470  2      fe-2/3/0.602
00:30:05:65:cd:aa/48
800001          ge-0/3/0.0      dymn  0      indr 262174  7      131.114.191.29  Push
00:30:05:65:cd:af/48
800001          ge-0/3/0.0      dymn  0      indr 262174  7      131.114.191.29  Push
00:e0:fc:59:5f:40/48
800001          ge-0/3/0.0      dymn  0      indr 262174  7      131.114.191.29  Push
00:e0:fc:5b:39:bc/48
800001          ge-0/3/0.0      dymn  0      indr 262174  7      131.114.191.29  Push

```

On jser router (131.114.100.1) there is the inverse situation

A.4.6 Community segregation and hub&spoke model visibility

On Jfib router (131.114.100.30), VPLS site 3, there is visibility of Jser router (131.114.100.1) and Jlng router (131.114.100.100)

```

Instance: VPLS-BIBLIO
Local site: VPLS-BIBLIO (3)
connection-site      Type St   Time last up      # Up trans
1                    rmt  Up    Mar 6 13:53:32 2006      1
  Local interface: vt-0/3/0.32768, Status: Up, Encapsulation: VPLS
  Remote PE: 131.114.100.1, Negotiated control-word: No
  Incoming label: 800016, Outgoing label: 800026
10                   rmt  Up    Mar 13 12:58:01 2006      1
  Local interface: vt-0/3/0.32771, Status: Up, Encapsulation: VPLS
  Remote PE: 131.114.100.100, Negotiated control-word: No
  Incoming label: 800025, Outgoing label: 800026

Instance: VPLS-STUDENTI
Local site: VPLS-STUDENTI (3)
connection-site      Type St   Time last up      # Up trans
1                    rmt  Up    Mar 6 13:53:31 2006      1
  Local interface: vt-0/3/0.32769, Status: Up, Encapsulation: VPLS
  Remote PE: 131.114.100.1, Negotiated control-word: No
  Incoming label: 800000, Outgoing label: 800034
10                   rmt  Up    Mar 10 11:46:17 2006      1
  Local interface: vt-0/3/0.32770, Status: Up, Encapsulation: VPLS
  Remote PE: 131.114.100.100, Negotiated control-word: No
  Incoming label: 800009, Outgoing label: 800010

```

On Jing router (131.114.100.50), site identifier 5, there is PE HUB visibility

```

Instance: VPLS-BIBLIO
Local site: VPLS-BIBLIO (5)
connection-site      Type St   Time last up      # Up trans
1                    rmt Up    Mar 7 17:50:25 2006      3
Local interface: vt-0/3/0.32768, Status: Up, Encapsulation: VPLS
Remote PE: 131.114.100.1, Negotiated control-word: No
Incoming label: 800016, Outgoing label: 800028

Instance: VPLS-STUDENTI
Local site: VPLS-STUDENTI (5)
connection-site      Type St   Time last up      # Up trans
1                    rmt Up    Mar 6 13:47:21 2006      1
Local interface: vt-0/3/0.32769, Status: Up, Encapsulation: VPLS
Remote PE: 131.114.100.1, Negotiated control-word: No
Incoming label: 800000, Outgoing label: 800036
    
```

A.4.7 MAC Cache aging and filtering

MAC-address 00:aa:bb:cc:dd:ee/48 acquisition with a 60 seconds aging time.

```

stefano@jser.unipi.it-re1> show system uptime
Current time: 2006-02-12 13:52:46 CET
System booted: 2005-11-24 11:49:28 CET (11w3d 02:03 ago)
Protocols started: 2005-11-24 11:50:53 CET (11w3d 02:01 ago)
Last configured: 2006-02-12 13:49:57 CET (00:02:49 ago) by stefano
1:52PM up 80 days, 2:03, 2 users, load averages: 0.13, 0.09, 0.08

stefano@jser.unipi.it-re1> show route forwarding-table family vpls
Routing table: VPN-VPLS-B.vpls
VPLS:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          dynm  0      flood 469  1
default          perm  0      rjct  375  1
00:0e:a6:5d:a7:e5/48      dynm  0      ucst  470  3 fe-2/3/0.602
00:30:05:65:cd:aa/48      dynm  0      indr  262174  7
                    131.114.191.29 Push 800001 ge-0/3/0.0
00:30:05:65:cd:af/48      dynm  0      indr  262174  7
                    131.114.191.29 Push 800001 ge-0/3/0.0
fe-2/3/0.602      dynm  0      flood 547  1
00:aa:bb:cc:dd:ee/48      dynm  0      ucst  470  3 fe-2/3/0.602
00:e0:fc:59:5f:40/48      dynm  0      indr  262174  7
                    131.114.191.29 Push 800001 ge-0/3/0.0
00:e0:fc:5b:39:bc/48      dynm  0      indr  262174  7
                    131.114.191.29 Push 800001 ge-0/3/0.0
    
```

After 60 seconds the MAC entry expires

```

stefano@jser.unipi.it-re1> show system uptime

Current time: 2006-02-12 13:53:01 CET
System booted: 2005-11-24 11:49:28 CET (11w3d 02:03 ago)
Protocols started: 2005-11-24 11:50:53 CET (11w3d 02:02 ago)
Last configured: 2006-02-12 13:49:57 CET (00:03:04 ago) by stefano
1:53PM up 80 days, 2:04, 2 users, load averages: 0.10, 0.08, 0.08

stefano@jser.unipi.it-re1> show route forwarding-table family vpls
Routing table: VPN-VPLS-B.vpls
VPLS:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          dnm  0          flood 469  1
default          perm 0          rjct  375  1
00:30:05:65:cd:aa/48
                  dnm  0          indr 262174 6
                  131.114.191.29 Push 800001 ge-0/3/0.0
fe-2/3/0.602     dnm  0          flood 547  1
00:e0:fc:59:5f:40/48
                  dnm  0          indr 262174 6
                  131.114.191.29 Push 800001 ge-0/3/0.0
00:e0:fc:5b:39:hc/48

```

A.4.7.1 *Generation of UDP flow with random MAC source address with 20ms acquisition rate. Beside this rate the VCT loose the informations.*

SDRAM FPC state

```

CSBR0(jlab.unipi.it vty)# show jtree 0 mem
Memory Statistics:
8388608 bytes total (2 banks)
7379440 bytes used
1009168 bytes free
8128 pages total
7256 pages used
872 pages free
31 max freelist size

```

A.4.7.2 PEs VCT check after generating 64k MAC source addresses

On Jlab router

```
stefano@jlab.unipi.it> show route forwarding-table family vpls | match ^00 | count
Count: 6 lines

stefano@jlab.unipi.it> show route forwarding-table family vpls | match ^00 | count
Count: 150 lines

stefano@jlab.unipi.it> show route forwarding-table family vpls | match ^00 | count
Count: 65535 lines

stefano@jlab.unipi.it> show system uptime
Current time: 2006-02-12 11:11:30 UTC
System booted: 2006-01-23 13:54:02 UTC (2w5d 21:17 ago)
Protocols started: 2006-01-23 13:55:01 UTC (2w5d 21:16 ago)
Last configured: 2006-02-09 14:43:54 UTC (2d 20:27 ago) by stefano
11:11AM up 19 days, 21:17, 5 users, load averages: 0.07, 0.20, 0.14
```

On Jser router

```
stefano@jser.unipi.it-re1> show route forwarding-table family vpls | match ^00 | count
Count: 65536 lines
```

State of FPC SDRAM

```
CSBR0(jlab.unipi.it vty)# show jtree 0 mem
Memory Statistics:
 8388608 bytes total (2 banks)
 7379440 bytes used
1009168 bytes free
 8128 pages total
 7256 pages used
 872 pages free
 31 max freelist size
```

Using the sniffer on the 172.16.0.1 is possible to consider that frames arriving on destination host are not counted by the filter applied at the interface if the destination is a unknown MAC-address

```
18:22:35.965107 00:0e:a6:5d:a7:e5 > ff:ff:ff:ff:ff:ff, ethertype Unknown (0x9001), length 60:
18:22:36.638036 00:0e:a6:5d:a7:e5 > ff:ff:ff:ff:ff:ff, ethertype Unknown (0x9001), length 60:
18:22:37.406141 00:0e:a6:5d:a7:e5 > ff:ff:ff:ff:ff:ff, ethertype Unknown (0x9001), length 60:
18:22:44.981405 00:e0:fc:5b:39:bc > ff:ff:ff:ff:ff:ff, ethertype Unknown (0x9001), length 60:
18:22:55.323554 00:e0:fc:59:5f:40 > ff:ff:ff:ff:ff:ff, ethertype Unknown (0x9001), length 6
```

Counters:		
Name	Bytes	Packets
VPLS-counter	0	0
test	0	0
Multicast-limit	0	0
Broadcast-limit	0	0

If we send frames with MAC-address 00:e0:fc:59:5f:40 counters work.

Counters:		
Name	Bytes	Packets
VPLS-counter	6400	100
test	6400	100
Multicast-limit	0	0
Broadcast-limit	0	0
Policers:		
Name	Packets	
Multicast-limit-Multicast-limit	0	
Filter: __default_bpdu_filter__		

After having generated Broadcast traffic (saturation of broadcast policer) from host 172.16.0.100 to host 172.16.0.1 it's visible the filter per instance running.

```

stefano@jlab.unipi.it# run show firewall
Filter: mf-classifier
Filter: Broadcast-Multicast-limit
Counters:
Name                Bytes      Packets
Multicast-packet    0          0
Broadcast-packets   2687956521 41999002
Policers:
Name                Packets
Multicast-limit-Multicast-limit 0
Broadcast-limit-Broadcast-limit 90077343
Filter: __default_bpdu_filter__
Filter: vpls-filter
Counters:
Name                Bytes      Packets
VPLS-counter        32843     229
test                0         0

```

