Understanding VPLS Label Blocks Operation
This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1995, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. GateD is based on Kirton’s EGP, UC Berkeley’s routing daemon (routed), and DCN’s HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, the Juniper Networks logo, JUNOS, NetScreen, ScreenOS, and Steel-Belted Radius are registered trademarks of Juniper Networks, Inc. in the United States and other countries. JUNOS is a trademark of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Technology Overview Understanding VPLS Label Blocks Operation
Copyright © 2010, Juniper Networks, Inc.
All rights reserved. Printed in USA.

Writing: Sridhar Selvarangam
Editing: Nancy Kurahashi, Katie Smith, Marilyn Kerr, Roy Spencer
Illustration: Faith Bradford
Cover Design: Edmonds Design

Revision History
December 2009—Revision 1

The information in this document is current as of the date listed in the revision history.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. The JUNOS Software has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.
END USER LICENSE AGREEMENT

READ THIS END USER LICENSE AGREEMENT (“AGREEMENT”) BEFORE DOWNLOADING, INSTALLING, OR USING THE SOFTWARE. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE OR OTHERWISE EXPRESSING YOUR AGREEMENT TO THE TERMS CONTAINED HEREIN, YOU (AS CUSTOMER OR IF YOU ARE NOT THE CUSTOMER, AS A REPRESENTATIVE/AGENT AUTHORIZED TO BIND THE CUSTOMER) CONSENT TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT OR CANNOT AGREE TO THE TERMS CONTAINED HEREIN, THEN (A) DO NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND (B) YOU MAY CONTACT JUNIPER NETWORKS REGARDING LICENSE TERMS.

1. The Parties. The parties to this Agreement are (i) Juniper Networks, Inc. (if the Customer’s principal office is located in the Americas) or Juniper Networks (Cayman) Limited (if the Customer’s principal office is located outside the Americas) (such applicable entity being referred to herein as “Juniper”), and (ii) the person or organization that originally purchased from Juniper or an authorized Juniper reseller the applicable license(s) for use of the Software (“Customer”) (collectively, the “Parties”).

2. The Software. In this Agreement, “Software” means the program modules and features of the Juniper or Juniper-supplied software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller, or which was embedded by Juniper in equipment which Customer purchased from Juniper or an authorized Juniper reseller. “Software” also includes updates, upgrades and new releases of such software. “Embedded Software” means Software which Juniper has embedded in or loaded onto the Juniper equipment and any updates, upgrades, additions or replacements which are subsequently embedded in or loaded onto the equipment.

3. License Grant. Subject to payment of the applicable fees and the limitations and restrictions set forth herein, Juniper grants to Customer a non-exclusive and non-transferable license, without right to sublicense, to use the Software, in executable form only, subject to the following use restrictions:

   a. Customer shall use Embedded Software solely as embedded in, and for execution on, Juniper equipment originally purchased by Customer from Juniper or an authorized Juniper reseller.

   b. Customer shall use the Software on a single hardware chassis having a single processing unit, or as many chassis or processing units for which Customer has paid the applicable license fees, provided, however, with respect to the Steel-Belted Radius or Odyssey Access Client software only, Customer shall use such Software on a single computer containing a single physical random access memory space and containing any number of processors. Use of the Steel-Belted Radius or IMS AAA software on multiple computers or virtual machines (e.g., Solaris zones) requires multiple licenses, regardless of whether such computers or virtualizations are physically contained on a single chassis.

   c. Product purchase documents, paper or electronic user documentation, and/or the particular licenses purchased by Customer may specify limits to Customer’s use of the Software. Such limits may restrict use to a maximum number of seats, registered endpoints, concurrent users, sessions, calls, connections, subscribers, clusters, nodes, realms, devices, links, ports or transactions, or require the purchase of separate licenses to use particular features, functionalities, services, applications, operations, or capabilities, or provide throughput, performance, configuration, bandwidth, interface, processing, temporal, or geographical limits. In addition, such limits may restrict the use of the Software to managing certain kinds of networks or require the Software to be used only in conjunction with other specific Software. Customer’s use of the Software shall be subject to all such limitations and purchase of all applicable licenses.

   d. For any trial copy of the Software, Customer’s right to use the Software expires 30 days after download, installation or use of the Software. Customer may operate the Software after the 30-day trial period only if Customer pays for a license to do so. Customer may not extend or create an additional trial period by re-installing the Software after the 30-day trial period.

   e. The Global Enterprise Edition of the Steel-Belted Radius software may be used by Customer only to manage access to Customer’s enterprise network. Specifically, service provider customers are expressly prohibited from using the Global Enterprise Edition of the Steel-Belted Radius software to support any commercial network access services.

The foregoing license is not transferable or assignable by Customer. No license is granted herein to any user who did not originally purchase the applicable license(s) for the Software from Juniper or an authorized Juniper reseller.

4. Use Prohibitions. Notwithstanding the foregoing, the license provided herein does not permit the Customer to, and Customer agrees not to and shall not: (a) modify, unbundle, reverse engineer, or create derivative works based on the Software; (b) make unauthorized copies of the Software (except as necessary for backup purposes); (c) rent, sell, transfer, or grant any rights in and to any copy of the Software, in any form, to any third party; (d) remove any proprietary notices, labels, or marks on or in any copy of the Software or any product in which the Software is embedded; (e) distribute any copy of the Software to any third party, including as may be embedded in Juniper equipment sold in the secondhand market; (f) use any “locked” or key-restricted feature, function, service, application, operation, or capability without first purchasing the applicable license(s) and obtaining a valid key from Juniper, even if such feature, function, service, application, operation, or capability is enabled without a key; (g) distribute any key for the Software provided by Juniper to any third party; (h) use the Software in any manner that extends or is broader than the uses purchased by Customer from Juniper or an authorized Juniper reseller; (i) use Embedded Software on non-Juniper equipment; (j) use Embedded Software (or make it available for use) on Juniper equipment that the Customer did not originally purchase from Juniper or an authorized Juniper reseller; (k) disclose the results of testing or benchmarking of the Software to any third party without the prior written consent of Juniper; or (l) use the Software in any manner other than as expressly provided herein.

5. Audit. Customer shall maintain accurate records as necessary to verify compliance with this Agreement. Upon request by Juniper, Customer shall furnish such records to Juniper and certify its compliance with this Agreement.
6. **Confidentiality.** The Parties agree that aspects of the Software and associated documentation are the confidential property of Juniper. As such, Customer shall exercise all reasonable commercial efforts to maintain the Software and associated documentation in confidence, which at a minimum includes restricting access to the Software to Customer employees and contractors having a need to use the Software for Customer’s internal business purposes.

7. **Ownership.** Juniper and Juniper’s licensors, respectively, retain ownership of all right, title, and interest (including copyright) in and to the Software, associated documentation, and all copies of the Software. Nothing in this Agreement constitutes a transfer or conveyance of any right, title, or interest in the Software or associated documentation, or a sale of the Software, associated documentation, or copies of the Software.

8. **Warranty, Limitation of Liability, Disclaimer of Warranty.** The warranty applicable to the Software shall be as set forth in the warranty statement that accompanies the Software (the “Warranty Statement”). Nothing in this Agreement shall give rise to any obligation to support the Software. Support services may be purchased separately. Any such support shall be governed by a separate, written support services agreement.

9. **Termination.** Any breach of this Agreement or failure by Customer to pay any applicable fees due shall result in automatic termination of the license granted hereunder. Upon such termination, Customer shall destroy or return to Juniper all copies of the Software and related documentation in Customer’s possession or control.

10. **Taxes.** All license fees payable under this agreement are exclusive of tax. Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software. If applicable, valid exemption documentation for each taxing jurisdiction shall be provided to Juniper prior to invoicing, and Customer shall promptly notify Juniper if their exemption is revoked or modified. All payments made by Customer shall be net of any applicable withholding tax. Customer will provide reasonable assistance to Juniper in connection with such withholding taxes by promptly: providing Juniper with valid tax receipts and other required documentation showing Customer’s payment of any withholding taxes; completing appropriate applications that would reduce the amount of withholding tax to be paid; and notifying and assisting Juniper in any audit or tax proceeding related to transactions hereunder.

11. **Export.** Customer agrees to comply with all applicable export laws and restrictions and regulations of any United States and any applicable foreign agency or authority, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals.

12. **Commercial Computer Software.** The Software is “commercial computer software” and is provided with restricted rights. Use, duplication, or disclosure by the United States government is subject to restrictions set forth in this Agreement and as provided in DFARS 227.7201 through 227.7202-4, FAR 12.212, FAR 27 405(b)(2), FAR 52.227-19, or FAR 52.227-14(ALT II) as applicable.

13. **Interface Information.** To the extent required by applicable law, and at Customer’s written request, Juniper shall provide Customer with the interface information needed to achieve interoperability between the Software and another independently created program, on payment of applicable fee, if any.

14. **Third Party Software.** Any licensor of Juniper whose software is embedded in the Software and any supplier of Juniper whose products or technology are embedded in (or services are accessed by) the Software shall be a third party beneficiary with respect to this Agreement, and such licensor or vendor shall have the right to enforce this Agreement in its own name as if it were Juniper. In addition, certain third party software may be provided with the Software and is subject to the accompanying license(s), if any, of its respective owner(s). To the extent portions of the Software are distributed under and subject to open source licenses obligating Juniper to make the source code for such portions publicly available (such as the GNU General Public License (“GPL”) or the GNU Library General Public License (“LGPL”)), Juniper will make such source code portions (including Juniper modifications, as appropriate) available upon request for a period of up to three years from the date of distribution. Such request can be made in writing to Juniper Networks, Inc., 1194 N. Mathilda Ave., Sunnyvale, CA 94089, ATTN: General Counsel. You may obtain a copy of the GPL at http://www.gnu.org/licenses/gpl.html, and a copy of the LGPL at http://www.gnu.org/licenses/lgpl.html.

15. **Miscellaneous.** This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous
agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présentés confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s’y rattache, soient rédigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).
Table of Contents

Overview .........................................................................................................1
   Elements of Network Layer Reachability Information ................................1
   Requirements for NLRI Elements .............................................................2
   How Labels are Used in Label Blocks .....................................................2
   Label Block Composition .......................................................................2
   Label Blocks in JUNOS ..........................................................................3
   VPLS Label Block Structure ..................................................................3
Example: Building a VPLS From Router 1 to Router 3 .................................5
Overview

This document describes the details of how virtual private LAN service (VPLS) labels are defined and exchanged in the Border Gateway Protocol (BGP) control plane. This document also describes how label blocks are allocated and used in the VPLS control plane for autodiscovery and signaling in the JUNOS Software implementation.

A VPLS is a Layer 2 (L2) service that emulates a local area network (LAN) across a wide area network (WAN).

There are two primary functions of the VPLS control plane: autodiscovery and signaling.

- **Autodiscovery** – A method for automatically recognizing each provider edge (PE) router in a particular VPLS domain, using BGP update messages.

- **Signaling** – Each pair of PE routers in a VPLS domain sends and withdraws VPN labels to each other. The labels are used to establish and dismantle pseudowires between the routers. Signaling is also used to transmit certain characteristics of a pseudowire.

The PE router uses BGP extended communities to identify the members of its VPLS. Once the PE discovers its members, it is able to establish and tear down pseudowires between members by exchanging and withdrawing labels and transmitting certain characteristics of the pseudowires.

The PE sends common update messages to all remote PEs, using a distinct BGP update message, thereby reducing the control plane load. This is achieved by using label blocks.

**Elements of Network Layer Reachability Information**

VPLS BGP network layer reachability information (NLRI) is used to exchange VPLS membership and parameters. A VPLS BGP NLRI has the elements defined in Table 1 on page 1.

<table>
<thead>
<tr>
<th>Element</th>
<th>Acronym</th>
<th>Description</th>
<th>Default Size (Octets)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td></td>
<td>Total length of the NLRI size represented in bytes.</td>
<td>2</td>
</tr>
<tr>
<td>Route Distinguisher</td>
<td>RD</td>
<td>A unique identifier for each routing instance configured on a PE.</td>
<td>8</td>
</tr>
<tr>
<td>VPLS Edge ID</td>
<td>VE ID</td>
<td>A unique number to identify the edge site.</td>
<td>2</td>
</tr>
<tr>
<td>VE Block Offset</td>
<td>VBO</td>
<td>A value used to identify a label block from which a label value will be selected to set up pseudowires for a remote site</td>
<td>2</td>
</tr>
</tbody>
</table>
Table 1: Elements of the NLRI (continued)

<table>
<thead>
<tr>
<th>Element</th>
<th>Acronym</th>
<th>Description</th>
<th>Default Size (Octets)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VE Block Size</td>
<td>VBS</td>
<td>Indicates the number of pseudowires that peers can have in a single block.</td>
<td>2</td>
</tr>
<tr>
<td>Label Base</td>
<td>LB</td>
<td>The starting value of the label in the advertised label block.</td>
<td>3</td>
</tr>
</tbody>
</table>

Requirements for NLRI Elements

JUNOS Software requires a unique route distinguisher (RD) for each routing instance configured on a PE. A PE router might use the same RD across a VPLS (or VPN) domain or it might use different RDs. Using different RDs helps identify the originator of the VPLS NLRI.

The VPLS edge (VE) ID can be a unique VE ID, site ID, or customer edge (CE) ID. The VE ID is used by a VPLS PE router to index into label blocks used to derive the transmit and receive VPN labels needed for transport of VPLS traffic. The VE ID identifies a particular site, so it needs to be unique within the VPLS domain, except for some scenarios such as multihoming.

All PE routers have full mesh connectivity with each other to exchange labels and set up pseudowires. The VE block size (VBS) is a configurable value that represents the number of label blocks required to cover all the pseudowires for the remote peer.

A single label block contains 8 labels (1 octet) by default. The default VBS in JUNOS Software is 2 blocks (2 octets) for a total of 16 labels.

How Labels are Used in Label Blocks

Each PE router creates a mapping of the labels in the label block to the sites in a VPLS domain. A PE router advertising a label block with a block offset indicates which sites can use the labels to reach it. When a PE router is ready to advertise its membership to a VPLS domain, it allocates a label block and advertises the VPLS NLRI. In this way, other PE routers in the same VPLS domain can learn of the existence of the VPLS and set up pseudowires to it if needed. The VPLS NLRI advertised for this purpose is referred to as the default VPLS NLRI. The label block in the default VPLS NLRI is referred to as the default label block.

Label Block Composition

A label block (set of labels) is used to reach a given site ID. A single label block contains 8 labels (1 octet) by default. The VBS is 2 octets by default in JUNOS Software.

The label block advertised is defined as a label base (LB) and a VE block size (VBS). It is a contiguous set of labels (LB, LB + 1,...LB + VBS-1). For example, when Router PE-A sends a VPLS update, it sends the same label block information to all other PE routers. Each PE router that receives the LB advertisement infers the label intended for Router PE-A by adding its own site ID to the label base.
In this manner, each receiving PE gets a unique label for PE-A for that VPLS. This simple method is enhanced by using a VE block offset (VBO).

A label block is defined as: \(<\text{Label Base (LB)}, \text{VE block offset (VBO)}, \text{VE block size (VBS)}>\) is the set \(\{\text{LB} + \text{VBO}, \text{LB} + \text{VBO} + 1, \ldots, \text{LB} + \text{VBO} + \text{VBS}-1\}\).

**Label Blocks in JUNOS**

Instead of a single large label block to cover all VE IDs in a VPLS, the JUNOS Software implementation contains several label blocks, each with a different label base. This makes label block management easier, and also allows Router PE-A to cater gracefully to a PE router joining a VPLS with a site ID not covered by the set of label blocks that Router PE-A has already advertised.

**VPLS Label Block Structure**

This section illustrates how a label block is uniquely identified.

A VPLS BGP NLRI with site ID V, VE block offset VBO, VE block size VBS, and label base LB communicates the following to its peers:

- Label block for V: Labels from LB to \((\text{LB} + \text{VBS} -1)\).
- Remote VE set for V: from VBO to \((\text{VBO} + \text{VBS} -1)\).

The label block advertised is a set of labels used to reach a given site ID. If there are several label blocks, the remote VE set helps to identify which label block to use. The example in Figure 1 on page 3 illustrates label blocks. There are 2 blocks and each block has eight labels. In this example, the label values are 64 to 71 and 80 to 87.

**Figure 1: VPLS Label Block Structure**

To create a one-to-one mapping of these sixteen labels to sixteen sites, assume the site IDs are the numbers 1 to 16, as shown in the illustration. The site block indicates which site ID can use which label in the label block. So, in the first block, site ID 1 will use 64, site ID 2 will use 65, and so forth. Finally, site ID 8 will use 71. The 9th site ID will use the second block instead of the first block.
The labels are calculated by comparing the values of VBO <= Local site ID < (VBO + VBS). Consequently, site ID 9 will use 80, site ID 10 will use 81, and so on.

To further illustrate the one-to-one mapping of labels to sites, assume a label block with site offset of 1 and a label base of 10. The combination of label base and block offset contained in the VPLS NLRI provides the mapping of labels to site IDs. The block offset is the starting site ID that can use the label block as advertised in the VPLS NLRI.

To advertise the default VPLS NLRI, a PE router picks a starting block offset that fits its own site ID and is such that the end block offset is a multiple of a single label block, in JUNOS Software a single label block is eight labels by default.

The end block offset is the last site ID that maps to the last label in the label block. The end offset for the first block is 8 which maps to label 17 and the second block is 16. For example, a site with ID 3 picks a block offset of 1 and advertises a label block of size 8 to cover sites with IDs 1 to 8. A site with ID 10 picks a block offset of 9 to cover sites with IDs 9 to 16.

The VPLS NLRI shown in Figure 2 on page 4 is for site ID 18. The label base contains value 262145. The block offset contains value 17. The illustration shows which site IDs correspond to which labels.

If a PE router configured with site ID 17 is in the same VPLS domain as a PE router configured with site ID 18, it receives the VPLS NLRI as shown in Figure 3. So it uses label 262145 to send traffic to site 18. Similarly, a PE router configured with site ID 19 uses label 262147 to send traffic to a PE router configured with site ID 18. However, only PE routers configured with site IDs 17 to 24 can use the label block shown to set up pseudowires.
Example: Building a VPLS From Router 1 to Router 3

This example illustrates how VPLS label blocks are allocated for a specific configuration. It is organized in the following sections:

- Requirements on page 5
- Overview and Topology on page 5
- Configuration on page 6

Requirements

This configuration example requires three Juniper Networks routers.

Overview and Topology

In the network shown in Figure 3 on page 5 Router 1 is establishing a pseudowire to Router 3

![Figure 3: Router 1 to Router 3 Topology](image)

Each PE filters the VPLS NLRI contained in the BGP update messages based on route target communities. Those VPLS NLRI instances that match the route target (in this case 8717:2000:2:1) are imported for further processing. The NLRI for Router 1 and Router 3 is shown in Table 2 on page 5.

Table 2: NLRI Exchange Between for Router 1 and Router 3

<table>
<thead>
<tr>
<th>Router 1 NLRI Advertisement to Router 3</th>
<th>Router 3 NLRI Advertisement to Router 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>RD - 8717:1000</td>
<td>RD - 8717:1000</td>
</tr>
<tr>
<td>VE ID - 1</td>
<td>VE ID - 2</td>
</tr>
<tr>
<td>VE Block Offset - 1</td>
<td>VE Block Offset - 1</td>
</tr>
<tr>
<td>VE Block Size - 8</td>
<td>VE Block Size - 8</td>
</tr>
<tr>
<td>Label Base - 262161</td>
<td>Label Base - 262153</td>
</tr>
</tbody>
</table>
To set up a pseudowire to Router 3, Router 1 must select a label to use to send traffic to Router 3 and also select a label that it expects Router 3 to use to send traffic to itself. The site ID contained in the VPLS NLRI from Router 3 is 2.

Router 1 learns of the existence of site ID 2 in the same VPLS domain. Using the equation \( VBO \leq \text{Local Site ID} < (VBO + VBS) \), Router 1 checks if the route advertised by site ID 2 fits in the label block and block offset that it previously advertised to Router 3. In this example it does fit, so the site ID 2 is mapped by the VPLS NLRI advertised by Router 1, and Router 1 is ready to set up a pseudowire to Router 3.

To select the label to reach Router 3, Router 1 looks at the label block advertised by Router 3 and performs a calculation. The calculation a PE router uses to check if its site ID is mapped in the label block from the remote peer is \( VBO \leq \text{Local Site ID} < (VBO + VBS) \). So, Router 1 selects label \( 262153 + (1 - 1) = 262153 \) to send traffic to Router 3. Using the same equation, Router 1 looks at its own label block that it advertised and selects label \( 262161 + (2 - 1) = 262162 \) to receive traffic from Router 3. Router 1 programs its forwarding state such that any traffic destined to Router 3 carries the pseudowire label 262153 and any traffic coming from Router 3 is expected to have the pseudowire label 262162. This completes the operations on the VPLS NLRI received from Router 3. Router 1 now has a pseudowire set up to Router 3.

Router 3 operation is very similar to the Router 1 operation. Since the Router 3 site ID of 2 fits in the label block and block offset advertised by Router 1, Router 3 selects label \( 262161 + (2 - 1) = 262162 \) to send traffic to Router 1. Router 3 looks at its own label block that it advertised and selects label \( 262153 + (1 - 1) = 262153 \) to receive traffic from Router 1. This completes the creation of a pseudowire to Router 1.

By default, for VPLS operation JUNOS Software uses a virtual tunnel (VT) loopback interface to represent a pseudowire. This example uses a label-switched interface (LSI) instead of a VT interface because there is no change in the VPLS control plane operation. Thus, for an MX platform, if there is a tunnel physical interface card (PIC) configured, it is mandatory to include the `no-tunnel-services` statement at the `[edit routing-instances routing-instance-name protocols vpls]` hierarchy level.

## Configuration

The following sections present the steps to configure and verify the example.

- Configuring Router 1 on page 6
- Configuring Router 3 on page 7
- Verifying the VPLS Label Allocations on page 8

### Configuring Router 1

#### Step-by-Step Procedure

1. Configure Router 1. Create the `edut` routing instance. Specify the `vpls` instance type. Configure the route distinguisher and specify the value `8717:1000`. Configure the route target and specify the value `8717:100` Configure the VPLS
protocol. Specify 10 as the site range. Specify 1 as the site ID. Include the no-tunnel-services statement.

[edit routing-instances]
edut {
    instance-type vpls;
    interface ge-5/0/2.0;
    route-distinguisher 8717:1000;
    vrf-target target:8717:100;
    protocols {
        vpls {
            site-range 10;
            no-tunnel-services;
            site router-1 {
                site-identifier 1;
            }
        }
    }
}

Configuring Router 3

Step-by-Step Procedure

1. Configure Router 3. Create the edut routing instance. Specify the vpls instance type. Configure the route distinguisher and specify the value 8717:2000. Configure the route target and specify the value 8717:200. Configure the VPLS protocol. Specify 10 as the site range. Specify 2 as the site ID. Include the no-tunnel-services statement.

[edit routing-instances]
edut {
    instance-type vpls;
    interface ge-4/0/2.0;
    route-distinguisher 8717:2000;
    vrf-target target:8717:100;
    protocols {
        vpls {
            site-range 10;
            no-tunnel-services;
            site router-3 {
                site-identifier 2;
            }
        }
    }
}
**Verifying the VPLS Label Allocations**

**Step-by-Step Procedure**

1. As shown in the figure and the configuration, Site A is attached to Router 1. Site A is assigned a site ID of 1. Before Router 1 can announce its membership to VPLS edut using a BGP update message, Router 1 needs to allocate a default label block. In this example, the label base of the label block allocated by Router 1 is 262161. Since Router 1’s site ID is 1, Router 1 associates the assigned label block with block offset of 1. The following are the messages sent from Router 1 to Router 3 as displayed using the `monitor traffic interface interface-name` command.

   ```
   user@Router1> monitor traffic interface ge-5/3/2
   Jun 14 12:26:31.280818 BGP SEND 10.10.10.1+179 -> 10.10.10.3+53950
   Jun 14 12:26:31.280824 BGP SEND message type 2 (Update) length 88
   Jun 14 12:26:31.280828 BGP SEND flags 0x40 code Origin(1): IGP
   Jun 14 12:26:31.280833 BGP SEND flags 0x40 code ASPath(2) length 0: <null>
   Jun 14 12:26:31.280837 BGP SEND flags 0x40 code LocalPref(5): 100
   Jun 14 12:26:31.280844 BGP SEND flags 0xc0 code Extended Communities(16): 2:8717:100 800a:19:0:0
   Jun 14 12:26:31.280848 BGP SEND flags 0x90 code MP_reach(14): AFI/SAFI 25/65
   Jun 14 12:26:31.280853 BGP SEND nhop 10.10.10.1 len 4
   Jun 14 12:26:31.280862 BGP SEND 8717:1000:1:1 (label base : 262161 range : 8, ce id: 1, offset: 1)
   Jun 14 12:26:31.405067 BGP RECV 10.10.10.3+53950 -> 10.10.10.1+179
   Jun 14 12:26:31.405074 BGP RECV message type 2 (Update) length 88
   Jun 14 12:26:31.405082 BGP RECV flags 0x40 code Origin(1): IGP
   Jun 14 12:26:31.405086 BGP RECV flags 0x40 code ASPath(2) length 0: <null>
   Jun 14 12:26:31.405093 BGP RECV flags 0x40 code LocalPref(5): 100
   Jun 14 12:26:31.405098 BGP RECV flags 0xc0 code Extended Communities(16): 2:8717:100 800a:19:0:0
   Jun 14 12:26:31.405103 BGP RECV nhop 10.10.10.3 len 4
   ```

2. As shown in the figure and the configuration, Site B is attached to Router 3. Site B is assigned a site ID of 2. Before Router 3 can announce its membership to VPLS edut using a BGP update message, Router 3 assigns a default label block with the label base of 262153. The block offset for this label block is 1 because its own site ID of 2 fits in the block being advertised. The following are the messages sent from Router 3 to Router 1 as displayed using the `monitor traffic interface interface-name` command.

   ```
   user@Router3> monitor traffic interface ge-2/0/1
   Jun 14 12:26:31.282008 BGP SEND 10.10.10.3+53950 -> 10.10.10.1+179
   Jun 14 12:26:31.282018 BGP SEND message type 2 (Update) length 88
   Jun 14 12:26:31.282026 BGP SEND flags 0x40 code Origin(1): IGP
   Jun 14 12:26:31.282034 BGP SEND flags 0x40 code ASPath(2) length 0: <null>
   Jun 14 12:26:31.282041 BGP SEND flags 0x40 code LocalPref(5): 100
   Jun 14 12:26:31.282052 BGP SEND flags 0xc0 code Extended Communities(16): 2:8717:100 800a:19:0:0
   ```
3. Verify the connection status messages for Router 1 using the `show vpls connections` command. Notice the base label is 262161, the incoming label from Router 3 is 262162, and the outgoing label to Router 3 is 262153.

```bash
user@Router1> show vpls connections instance edut extensive

Instance: edut
Local site: router-1 (1)
  Number of local interfaces: 1
  Number of local interfaces up: 1
  IRB interface present: no
  ge-5/0/2.0 lsi.1049600
  Intf - vpls edut local site 1 remote site 2
  Label-base Offset Range Preference
    262161 1 8 100
  connection-site Type St Time last up # Up
  trans 2 rmt Up Jun 14 12:26:31 2009 1
  Remote PE: 10.10.10.3, Negotiated control-word: No
  Incoming label: 262162, Outgoing label: 262153
  Local interface: lsi.1049600, Status: Up, Encapsulation: VPLS
  Description: Intf - vpls edut local site 1 remote site 2
  Connection History:
    Jun 14 12:26:31 2009 status update timer
    Jun 14 12:26:31 2009 loc intf up
    Jun 14 12:26:31 2009 PE route changed
    Jun 14 12:26:31 2009 Out lbl Update 262153
    Jun 14 12:26:31 2009 In lbl Update 262162
    Jun 14 12:26:31 2009 loc intf down

Layer-2 VPN connections:

Legend for connection status (St)
  EI -- encapsulation invalid
  CCC/TCC/VPLS
  EM -- encapsulation mismatch
  VC-Dn -- Virtual circuit down
  WE -- interface and instance encaps not same
  NC -- interface encapsulation not
  NP -- interface hardware not present
```
CM -- control-word mismatch      <- only outbound connection is up
CN -- circuit not provisioned    --> only inbound connection is up
OR -- out of range               Up -- operational
OL -- no outgoing label          Dn -- down
LD -- local site signaled down   CF -- call admission control failure
RD -- remote site signaled down  SC -- local and remote site ID collision
LN -- local site not designated  LM -- local site ID not minimum designated
RN -- remote site not designated RM -- remote site ID not minimum designated
XX -- unknown connection status  IL -- no incoming label
MM -- MTU mismatch               MI -- Mesh-Group ID not available
BK -- Backup connection         ST -- Standby connection
PF -- Profile parse failure      PB -- Profile busy
Legend for interface status
Up -- operational
Dn -- down

4. Verify the connection status messages for Router 3 using the `show vpls connections` command. Notice the base label is 262153, the incoming label from Router 1 is 262153, and the outgoing label to Router 1 is 262162.

    user@Router3> show vpls connections instance edut extensive

Instance: edut
Local site: router-3 (2)
Number of local interfaces: 1
Number of local interfaces up: 1
IRB interface present: no
ge-4/0/2.0
lsi.1050368         1         Intf - vpls edut local site 2 remote site 1
Label-base        Offset     Range     Preference
                   262153     1          8         100
connection-site           Type  St     Time last up          # Up
trans                      rmt   Up     Jun 14 12:26:31 2009 1
1                         rmt   Up     Jun 14 12:26:31 2009
Remote PE: 10.10.10.1, Negotiated control-word: No
Incoming label: 262153, Outgoing label: 262162
Local interface: lsi.1050368, Status: Up, Encapsulation: VPLS
Description: Intf - vpls edut local site 2 remote site 1
Connection History:
Jun 14 12:26:31 2009  status update timer
Jun 14 12:26:31 2009  loc intf up lsi.1050368
Jun 14 12:26:31 2009  PE route changed
Jun 14 12:26:31 2009  Out lbl Update 262162
Jun 14 12:26:31 2009  In lbl Update 262153
Jun 14 12:26:31 2009  loc intf down
Layer-2 VPN connections:

Legend for connection status (St)
EI -- encapsulation invalid   NC -- interface encapsulation not
CCC/TCC/VPLS                   same
EM -- encapsulation mismatch   WE -- interface and instance encap not
same
VC-Dn -- Virtual circuit down  NP -- interface hardware not present
CM -- control-word mismatch  -< -- only outbound connection is up
CN -- circuit not provisioned  >- -- only inbound connection is up
OR -- out of range Up -- operational
OL -- no outgoing label Dn -- down
LD -- local site signaled down CF -- call admission control failure
RD -- remote site signaled down SC -- local and remote site ID collision
LN -- local site not designated LM -- local site ID not minimum designated
RN -- remote site not designated RM -- remote site ID not minimum designated
XX -- unknown connection status IL -- no incoming label
MM -- MTU mismatch MI -- Mesh-Group ID not available
BK -- Backup connection ST -- Standby connection
PF -- Profile parse failure PB -- Profile busy

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

Example: Building a VPLS From Router 1 to Router 3