JUNIPER NETWORKS

Junos® OS

Dynamic Profiles for VPLS Pseudowires Feature Guide for MX Series Routers

Release
14.1

Published: 2014-05-03

Copyright © 2014, Juniper Networks, Inc.

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks 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.

Junos® OS Dynamic Profiles for VPLS Pseudowires Feature Guide for MX Series Routers
14.1
Copyright © 2014, Juniper Networks, Inc.
All rights reserved.

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS 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

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at http://www.juniper.net/support/eula.html. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

Copyright © 2014, Juniper Networks, Inc.

Copyright © 2014, Juniper Networks, Inc.

List of Tables

Copyright © 2014, Juniper Networks, Inc.

Copyright © 2014, Juniper Networks, Inc.

About the Documentation

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at http://www.juniper.net/techpubs/.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at http://www.juniper.net/books.

Supported Platforms

For the features described in this document, the following platforms are supported:

- MX Series

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

Copyright © 2014, Juniper Networks, Inc.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   system {
     scripts {
       commit {
         file ex-script.xsl;
       }
     }
   }
   interfaces {
     fxp0 {
       disable;
       unit 0 {
         family inet {
           address 10.0.0.1/24;
         }
       }
     }
   }
   ```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

   ```
   [edit]
   user@host# load merge /var/tmp/ex-script.conf
   load complete
   ```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   commit {
     file ex-script-snippet.xsl; }
   ```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

Copyright © 2014, Juniper Networks, Inc.

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the *CLI User Guide*.

Documentation Conventions

Table 1 on page ix defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page ix defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the **configure** command: user@host> **configure**

Copyright © 2014, Juniper Networks, Inc.

Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Encloses optional keywords or variables.	**stub <default-metric** ***metric*****>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	**community name members [** ***community-ids*** **]**

Indention and braces ({ })	Identifies a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.

Copyright © 2014, Juniper Networks, Inc.

Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to **techpubs-comments@juniper.net**, or fill out the documentation feedback form at https://www.juniper.net/cgi-bin/docbugreport/. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit http://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/

Copyright © 2014, Juniper Networks, Inc.

- Search technical bulletins for relevant hardware and software notifications: http://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://tools.juniper.net/SerialNumberEntitlementSearch/

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at http://www.juniper.net/cm/.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see http://www.juniper.net/support/requesting-support.html.

Copyright © 2014, Juniper Networks, Inc.

PART 1

Overview

Copyright © 2014, Juniper Networks, Inc.

 Copyright © 2014, Juniper Networks, Inc.

CHAPTER 1

Dynamic Profiles for VLAN Interfaces and Protocols

Dynamic Profiles for VPLS Pseudowires

A router often has two types of interfaces:

- Static interfaces, which are configured before the router is booted
- Dynamic interfaces, which are created after the router is booted and while it is running

A virtual private LAN service (VPLS) pseudowire interface (such as **lsi.1048576**) is dynamically created by the system. Therefore, the logical interface unit number for the VPLS pseudowire is not available in advance to configure characteristics such as virtual local area network (VLAN) identifiers and other parameters. As a result, certain VLAN manipulation features that are easily applied to static interfaces (such as **xe-**, **ge-**, and so on) are either not supported on dynamic interfaces or supported in a nonstandard method.

However, on MX Series routers, there is another configuration method that dynamic interfaces can use to determine their VLAN parameters when they are created by a running router: *dynamic profiles*. A dynamic profile is a conceptual container that includes parameters associated with a dynamic entity, parameters whose values are not known at the time the entity is configured.

A dynamic profile acts as a kind of template that enables you to create, update, or remove a configuration that includes client access (for example, interface or protocol) or service (for example, CoS) attributes. Using these profiles you can consolidate all of the common attributes of a client (and eventually a group of clients) and apply the attributes simultaneously. The router contains several predefined variables that enable dynamic association of interfaces and logical units to incoming subscriber requests. While configuring a dynamic profile, use the **$junos-interface-ifd-name** variable for a dynamic physical interface and the **$junos-underlying-unit-number** variable for a dynamic logical interface (unit). When a client accesses the router, the dynamic profile configuration replaces the predefined variable with the actual interface name or unit value for the interface the client is accessing.

Copyright © 2014, Juniper Networks, Inc.

Dynamic profiles for VPLS are supported only on MX Series routers. For more information about dynamic profiles, see the *Junos OS Subscriber Management and Services Library*.

Related Documentation

- *Ethernet Networking Feature Guide for MX Series Routers*
- *Dynamic Profiles Overview*
- Use Cases for Dynamic Profiles for VPLS Pseudowires on page 4
- Example: Configuring VPLS Pseudowires with Dynamic Profiles—Basic Solutions on page 7
- Example: Configuring VPLS Pseudowires with Dynamic Profiles—Complex Solutions on page 11

Use Cases for Dynamic Profiles for VPLS Pseudowires

A dynamic profile is a set of characteristics, defined in a type of template, that you can use to provide dynamic subscriber access and services for broadband applications. These services are assigned dynamically to interfaces. You can use dynamic profiles to configure the VLAN parameters of the dynamic interfaces on MX Series routers.

Two use cases for configuring VPLS pseudowires with dynamic profiles are:

- **Configuring an extra VLAN tag onto pseudowire traffic** — This is a common scenario where all the traffic received from a customer edge (CE) interface needs an additional VLAN tag toward the core. In such cases, you can use dynamic profiles at ingress and egress to control the pseudowire behavior. You can apply dynamic profiles to receive the desired frames, set the additional VLAN tag for these frames, and send these tagged frames with the desired VLAN identifier.
- **Configuring a VPLS pseudowire as a trunk interface** — This is another common scenario where the requirement is to accept traffic from a particular source and to route the traffic based on specific criteria. Dynamic profiles can be used to create multiple pseudowire trunk interfaces to accept the traffic based on specific VLAN identifiers, and to route the accepted traffic to the desired destination.

Related Documentation

- *Dynamic Profiles Overview*
- Dynamic Profiles for VPLS Pseudowires on page 3

Copyright © 2014, Juniper Networks, Inc.

PART 2

Configuration

Copyright © 2014, Juniper Networks, Inc.

 Copyright © 2014, Juniper Networks, Inc.

CHAPTER 2

Examples

Example: Configuring VPLS Pseudowires with Dynamic Profiles—Basic Solutions

The following limitations apply to dynamic profiles for VPLS on MX Series routers:

- The **native-vlan-id** statement is not supported.
- The **native-inner-vlan-id** statement is not supported.
- The **interface-mode access** statement option is not supported.
- The **vlan-id-range** statement is not supported.

In many cases, a configuration using dynamic profiles is more efficient than a static configuration, as shown by the examples in this topic.

VPLS Pseudowire Interfaces Without Dynamic Profiles

Consider the following configuration, which does not use dynamic profiles to manipulate VLAN identifiers:

```
[edit routing-instances]
green {
   instance-type vpls;
   interface ge-0/0/1.1;
   interface ge-0/0/2.1;
   interface ge-0/0/3.1;
   vlan-tags outer 200 inner 100;
   protocols vpls {
      vpls-id 10;
      neighbor 10.1.1.20;
```

Copyright © 2014, Juniper Networks, Inc.

```
  }
  {...more...}
}

[edit interfaces]
ge-0/0/1 {
  unit 0 {
    vlan-id 10;
  }
}
ge-0/0/2 {
  unit 0 {
    vlan-id 20;
  }
}
ge-0/0/3 {
  unit 0 {
    vlan-id 30;
  }
}
```

NOTE: This is not a complete router configuration.

With this configuration, broadcast packets inside frames arriving with VLAN identifier 10 on **ge-0/0/1** are normalized to a dual-tagged frame with an outer VLAN value of 200 and an inner VLAN value of 100. The broadcast packet and frames exiting **ge-0/0/2** or **ge-0/0/3** have the outer VLAN value stripped and the inner VLAN value swapped to 20 and 30 respectively, according to the interface configuration. However, this stripping of the outer VLAN tag and the swapping is extra work, because the frames will still egress the VPLS pseudowire in routing instance **green** with an outer VLAN tag value of 200 and an inner VLAN tag value of 100, also according to the configuration.

The same configuration can be accomplished more effectively using dynamic profiles.

VPLS Pseudowire Interfaces and Dynamic Profiles

Consider the following configuration, which uses dynamic profiles to manipulate VLAN identifiers:

```
[edit routing-instances]
green {
  instance-type vpls;
  interface ge-0/0/1.1;
  interface ge-0/0/2.1;
  interface ge-0/0/3.1;
  vlan-id 100; # Desired inner VLAN tag on the VPLS pseudowire
  protocols vpls {
    vpls-id 10;
    neighbor 10.1.1.20 {
      associate-profile green_vpls_pw_1; # The profile
    }
  }
  {...more...}
```


Copyright © 2014, Juniper Networks, Inc.

```
}

[edit interfaces]
ge-0/0/1 {
   unit 0 {
      vlan-id 10;
   }
}
ge-0/0/2 {
   unit 0 {
      vlan-id 20;
   }
}
ge-0/0/3 {
   unit 0 {
      vlan-id 30;
   }
}

[edit dynamic-profiles]
green_vpls_pw_1 interfaces $junos-interface-ifd-name {
   unit $junos-underlying-unit-number {
      vlan-tags outer 200 inner 100;
   }
}
```

NOTE: This is not a complete router configuration.

With this configuration, broadcast packets inside frames arriving with VLAN identifier 10 on **ge-0/0/1** are normalized to a frame with VLAN identifier 100. The broadcast packet and frames exiting **ge-0/0/2** or **ge-0/0/3** have this VLAN value swapped to 20 and 30 respectively, according to the interface configuration. Frames egress the VPLS pseudowire in routing instance **green** with an outer VLAN tag value of 200 pushed on top of the normalized value.

CE Routers Without Dynamic Profiles

You can apply a dynamic profile to an entire VPLS configuration, not just a neighbor.

Consider the following configuration, which does not use dynamic profiles to manipulate VLAN identifiers on a customer edge (CE) router with VLAN identifier 100:

```
[edit routing-instances]
green {
   instance-type vpls;
   interface ge-0/0/1.1;
   interface ge-0/0/2.1;
   interface ge-0/0/3.1;
   vlan-tags outer 200 inner 100;
   protocols vpls {
      vpls-id 10;
      neighbor 10.1.1.20;
   }
```

Copyright © 2014, Juniper Networks, Inc.

```
  {...more...}
}

[edit interfaces]
ge-0/0/1 {
  unit 0 {
    vlan-id 100;
  }
}
ge-0/0/2 {
  unit 0 {
    vlan-id 100;
  }
}
ge-0/0/3 {
  unit 0 {
    vlan-id 100;
  }
}
```

NOTE: This is not a complete router configuration.

With this configuration, broadcast packets inside frames arriving on **ge-0/0/1** are normalized to a dual-tagged frame with an outer VLAN value of 200 and an inner VLAN value of 100. The same configuration can be accomplished using dynamic profiles.

CE Routers and Dynamic Profiles

Consider the following configuration, which uses dynamic profiles at the **protocols** level:

```
[edit routing-instances]
green {
  instance-type vpls;
  interface ge-0/0/1.1;
  interface ge-0/0/2.1;
  interface ge-0/0/3.1;
  vlan-id 100; # Desired inner VLAN tag on the VPLS pseudowire
  protocols vpls {
    associate-profile green_vpls_pw_2; # The profile
    vpls-id 10;
    neighbor 10.1.1.20;
  }
  {...more...}
}

[edit interfaces]
ge-0/0/1 {
  unit 0 {
    vlan-id 100;
  }
}
ge-0/0/2 {
  unit 0 {
```


Copyright © 2014, Juniper Networks, Inc.

```
      vlan-id 100;
    }
  }
  ge-0/0/3 {
    unit 0 {
      vlan-id 100;
    }
  }

  [edit dynamic-profiles]
  green_vpls_pw_2 interfaces $junos-interface-ifd-name {
    unit $junos-underlying-unit-number {
      vlan-tags outer 200 inner 100;
    }
  }
```

NOTE: This is not a complete router configuration.

With this configuration, broadcast packets inside frames arriving with VLAN identifier 100 on **ge-0/0/1** are normalized to a frame with VLAN identifier 100 (in this case, they are unchanged). The broadcast packet and frames exiting **ge-0/0/2** or **ge-0/0/3** are unchanged as well, according to the interface configuration. Frames egress the VPLS pseudowire in routing instance **green** with an outer VLAN tag value of 200 pushed on top of the normalized value.

Related Documentation

- *Dynamic Profiles for VPLS Pseudowires Feature Guide for MX Series Routers*
- Example: Configuring VPLS Pseudowires with Dynamic Profiles—Complex Solutions on page 11

Example: Configuring VPLS Pseudowires with Dynamic Profiles—Complex Solutions

Dynamic profiles for VPLS pseudowires can be helpful in a variety of VLAN configurations. This section explores some of these situations through examples.

NOTE: These examples are not complete router configurations.

All of the examples in this section address the same basic topology. A routing instance **blue** uses a trunk bridge to connect different departments in an organization, each with their own VLANs, at two different sites. The organization uses a BGP-based VPLS with a virtual switch to accomplish this.

- Configuration of Routing Instance and Interfaces Without Dynamic Profiles on page 12
- Configuration of Routing Instance and Interfaces Using Dynamic Profiles on page 13
- Configuration of Tag Translation Using Dynamic Profiles on page 15

Copyright © 2014, Juniper Networks, Inc.

Configuration of Routing Instance and Interfaces Without Dynamic Profiles

The basic configuration of routing instance and interfaces without dynamic profiles follows:

```
[edit routing-instance blue]
instance-type virtual-switch;
route-distinguisher 10.1.1.10:1;
vrf-target target:1000:1;
interface ge-3/0/0; # The trunk interface
bridge-domains {
   sales {
      vlan-id 10;
      interface ge-0/0/0.1;
      ... # Other interfaces and statements for Sales
   }
   engineering {
      vlan-id 20;
      interface ge-1/0/2.0;
      ... # Other interfaces and statements for Engineering
   }
   accounting {
      vlan-id 30;
      interface ge-2/0/3.0;
      ... # Other interfaces and statements for Accounting
   }
   others {
      vlan-id—list [ 40 50 ]; # Other departments
   }
}
   protocols vpls {
      site-range 10;
      site sample-site-1 {
         site-identifier 1;
      }
   }
... # Other statements for instance Blue

[edit interfaces]
ge-0/0/1 {
   unit 0 {
      vlan-id 100;
   }
}
ge-3/0/0 {
   unit 0 {
      family bridge {
         interface-mode trunk; # This is the trunk
         vlan-id-list [ 10 20 30 40 50 ];
      }
   }
}
... # More interface statements
```


Copyright © 2014, Juniper Networks, Inc.

This configuration switches the departmental VLAN traffic (sales, engineering, etc.) bridge domains over the VPLS pseudowire trunk connecting to the other site.

Configuration of Routing Instance and Interfaces Using Dynamic Profiles

Here is how dynamic profiles can be applied to this basic configuration.

First, consider the requirement to push an outer VLAN tag value of 200 onto the VPLS pseudowire frames on egress. Dynamic profiles easily satisfy this requirement.

```
[edit routing-instance green]
instance-type virtual-switch;
... # Other routing instance statements
protocols vpls {
   site-range 10;
   site sample-site-1 {
      site-identifier 1;
   }
   associate-profile green_vpls_pw_1; # Apply profile here
}
... # Other routing instance statements

[edit dynamic-profiles]
green_vpls_pw_1 interfaces $junos-interface-ifd-name {
   unit $junos-underlying-unit-number {
      vlan-id 200; # This is the outer tag
      family bridge {
         interface-mode trunk;
         inner-vlan-id-list [ 10 20 30 40 50 ];
      }
   }
}
```

NOTE: This is not a complete router configuration.

With the dynamic profile, a packet in a frame arriving on an interface is classified as belonging to one of the bridge domains (VLANs 10–50). At the egress of the trunk VPLS pseudowire, the outer VLAN tag 200 is pushed onto the frame. At the ingress of the pseudowire at the remote location, the outer VLAN tag 200 is removed and the frame is delivered to the appropriate bridge domain.

But what if the packets associated with the Accounting VLAN are not to be forwarding to the remote site? Dynamic profiles are useful here as well.

This configuration keeps the Accounting frames from reaching the remote site.

```
[edit routing-instances green]
instance-type virtual-switch;
... # Other routing instance statements
protocols vpls {
   site-range 10;
   site sample-site-1 {
      site-identifier 1;
```

Copyright © 2014, Juniper Networks, Inc.

```
    }
    associate-profile green_vpls_pw_2; # Apply profile here
}
... # Other routing instance statements

[edit dynamic-profiles]
green_vpls_pw_2 interfaces $junos-interface-ifd-name {
    unit $junos-underlying-unit-number {
        family bridge {
            interface-mode trunk;
            inner-vlan-id-list [ 10 20 40 50 ]; # Removed Accounting VLAN 30
        }
    }
}
```

NOTE: This is not a complete router configuration.

In this case, frames arriving on the interfaces are classified according to their bridge domains and switched, if necessary, to the VPLS pseudowire trunk, except for Engineering frames. Engineering frames (VLAN 30) are only switched within the interfaces listed within bridge domain **accounting** and any statically configured trunk interfaces and are prevented from crossing the VPLS pseudowire due to the absence of VLAN 30 on the trunk.

We can combine the two examples and use dynamic profiles to forward the frames (other than **accounting** frames) to the remote site with an out tag of 200.

This configuration keeps the Accounting frames from reaching the remote site and pushes an outer tag of 200 on VPLS pseudowire traffic.

```
[edit routing-instances green]
instance-type virtual-switch;
... # Other routing instance statements
protocols vpls {
    site-range 10;
    site sample-site-1 {
        site-identifier 1;
    }
    associate-profile green_vpls_pw_3; # Apply profile here
}
... # Other routing instance statements

[edit dynamic-profiles]
green_vpls_pw_3 interfaces $junos-interface-ifd-name {
    unit $junos-underlying-unit-number {
        vlan-id 200; # This is the outer tag
        family bridge {
            interface-mode trunk;
            inner-vlan-id-list [ 10 20 40 50 ]; # Removed Accounting VLAN 30
        }
    }
}
```


Copyright © 2014, Juniper Networks, Inc.

NOTE: This is not a complete router configuration.

In this case, frames arriving on the interfaces are classified according to their bridge domains and switched, if necessary, to the VPLS pseudowire trunk with an outer VLAN tag of 200, except for Engineering frames. Engineering frames (VLAN 30) are only switched within the interfaces listed within bridge domain **accounting** and any statically configured trunk interfaces and are prevented from crossing the VPLS pseudowire due to the absence of VLAN 30 on the trunk.

Configuration of Tag Translation Using Dynamic Profiles

Consider a final case where the bridge domain VLANs need translation at the VPLS pseudowire trunk interface. In this case, **sales** (VLAN 10) is mapped to VLAN 110 and **engineering** (VLAN 20) is mapped to VLAN 120.

This configuration adds tag translation to the VPLS pseudowire traffic.

```
[edit routing-instances green]
instance-type virtual-switch;
... # Other routing instance statements
protocols vpls {
   site-range 10;
   site sample-site-1 {
      site-identifier 1;
   }
   associate-profile green_vpls_pw_4; # Apply profile here
}
... # Other routing instance statements


[edit dynamic-profiles]
green_vpls_pw_4 interfaces $junos-interface-ifd-name {
   unit $junos-underlying-unit-number {
      family bridge {
         interface-mode trunk;
         vlan-id-list [ 10 20 30 40 50 ]; # All VLANs
         vlan-rewrite translate 110 10; # Sales VLAN
         vlan-rewrite translate 120 20; # Engineering VLAN
      }
   }
}
```

NOTE: This is not a complete router configuration.

This translates the **sales** and **engineering** VLAN tags exiting the VPLS pseudowire accordingly. At the ingress of the VPLS pseudowire, VLANs 110 and 120 are translated back to 10 and 20, respectively.

Related Documentation

- *Dynamic Profiles for VPLS Pseudowires Feature Guide for MX Series Routers*
- Example: Configuring VPLS Pseudowires with Dynamic Profiles—Basic Solutions on page 7

Copyright © 2014, Juniper Networks, Inc.

Copyright © 2014, Juniper Networks, Inc.