Technical white paper

HPN SDN Controller Link Discovery

Understanding and configuring controlled networks for link discovery

Table of contents

HPN SDN Controller Link Discovery

This document is intended for sales support and networking professionals who wish to configure their networks for successful HPN SDN Controller link discovery. This document will give a deeper understanding of how link discovery is performed and some recommended configurations.

How OpenFlow Link Discovery works

The HPN SDN Controller injects and observes packets in the controlled network to discover links between OpenFlow instances, Discovered links may be either one of the following types:

- Direct

  Links which span from one OpenFlow instance port to another, with no intermediate switches.

- Multi-hop

  Links which span from one OpenFlow instance port to another, traversing intermediate uncontrolled switches. An uncontrolled switch is a switch which does not have OpenFlow instances defined or connected to the controller (or team) which is attempting to discover the link.

The HPN SDN Controller discovers links and distinguishes link type by injecting two packets to each port in an OpenFlow instance. These packets have the same Ethernet type (0x8999), but are sent to different destination MAC addresses. The content of these packets is proprietary, but generally includes identification of the OpenFlow instance and port where the packet originated. The packets are injected immediately after the switch connects to the controller and periodically thereafter.

The HPN SDN Controller does not have access to information regarding vlan or port configuration of the switches that are hosting the OpenFlow instances it controls. Therefore, these generated packets do not contain an 802.1Q header, since port and OpenFlow instance vlan configuration would be required to construct the correct vlan tag.

The following are sample packets injected for link discovery:

Direct discovery packet sample:

```
▶ Frame 2: 75 bytes on wire (600 bits), 75 bytes captured (600 bits)
▼ Ethernet II, Src: cc:3e:5f:0d:bc:bf (cc:3e:5f:0d:bc:bf), Dst: LLDP_Multicast (01:80:c2:00:00:0e)
  ▶ Destination: LLDP_Multicast (01:80:c2:00:00:0e)
  ▶ Source: cc:3e:5f:0d:bc:bf (cc:3e:5f:0d:bc:bf)
    Type: Unknown (0x8999)
▶ Data (61 bytes)
```

Multi-hop discovery packet sample:

```
▶ Frame 1: 75 bytes on wire (600 bits), 75 bytes captured (600 bits)
▼ Ethernet II, Src: cc:3e:5f:0d:bc:bf (cc:3e:5f:0d:bc:bf), Dst: Hewlett-_e9:7b:cd (01:1b:78:e9:7b:cd)
  ▶ Destination: Hewlett-_e9:7b:cd (01:1b:78:e9:7b:cd)
  ▶ Source: cc:3e:5f:0d:bc:bf (cc:3e:5f:0d:bc:bf)
    Type: Unknown (0x8999)
▶ Data (61 bytes)
```

The controller OpenFlow Link Discovery application listens for these packets from the destination datapaths.

When the controller is configured for hybrid.mode=true, the controller installs a flow rule on every OpenFlow instance to steal these packets:

Table ID	Priority	Packets	Bytes	Match	Actions/Instructions	Flow Class ID
0	60000	1500	n/a	eth_type: bddp	apply_actions: output: CONTROLLER	com.hp.sdn.bddp.steal

Packets which match this flow rule are forwarded to the controller from the OpenFlow instance and port where they were received. Using the origin information contained within the received packet, the controller derives the source and destination of the link that this packet traversed and records a link between the OpenFlow instances. The link type is derived from the destination MAC address of the packet (direct or multi-hop). If a link is direct, it will be discovered as both “direct” and “multi-hop” from the reporting OpenFlow instance, but the type of “direct” has precedence over the type of “multi-hop”, so the link is recorded as “direct”.

Keys to Correct Link Discovery

In order to ensure that the desired link topology is correctly discovered, configure the switches such that all OpenFlow instances are configured for the same vlan mapping throughout the topology AND one of the following conditions are met:

- The link discovery packet is correctly tagged by the OpenFlow instance when it egresses the port to which it is injected.
- The untagged link discovery packet is stolen to the controller by the correct OpenFlow instance agent on a switch when it is received.

Though the HPN SDN Controller link discovery method works well in many network configurations, there are some situations where links are not discovered as intended. This may include missing links, links with an incorrect type, or discovering a link where one should not exist. If the untagged discovery packets can’t be associated with the correct OpenFlow instance on the destination switch, then the discovery packet will be dropped or associated with the wrong OpenFlow instance. This can occur for reasons explained in the following sections, including ambiguous vlan association across all OpenFlow instances in the topology.

The consequences of link discovery problems will be observed as impacts to other functionality, such as node discovery or topology calculation. For instance, a missing link in the middle of the network may cause a node’s location to bounce from the edge of the network to the middle of the network.

Known issues

All known link discovery problems relate to one of the following situations:

- The received link discovery packet is dropped by a switch in the network, because the packet is untagged and received on a tagged-only port.
- The received link discovery packet is associated with the wrong OpenFlow instance, because the packet is untagged and received on a port which associates untagged packets with a vlan not in the OpenFlow instance. Another manifestation of this situation occurs when the OpenFlow instance configurations across all switches are not configured with the same vlan set and a discovery packet gets associated with only one OpenFlow instance (see switch C in the example below).

Vendors follow various interpretations in their association of vlans to OpenFlow instances. Various implementations allow a single vlan per OpenFlow instance, multiple vlans per OpenFlow instance, or all vlans in a single OpenFlow instance. Thus when the controller injects an untagged packet into an OpenFlow instance, the vlan association for the packet is sometime ambiguous in the OpenFlow instance.

Please see the tables below for details about whether your network configuration will meet either of the above conditions:

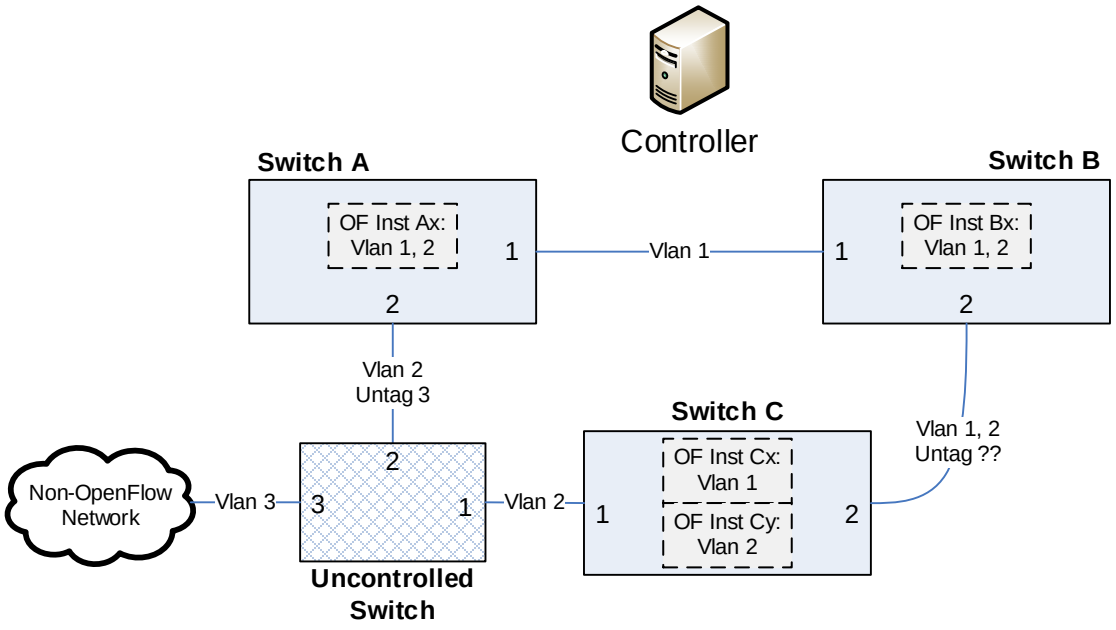
Discovery packet injection

Product	Tagging of Controller-Injected Packets
ProVision	In *aggregation* mode, the switch Openflow agent does not apply a tag to injected link discovery packets. In *virtualization* mode, the switch Openflow agent adds a tag for the vlan of the instance to controller-injected discovery packets when the packet egresses a tagged port.
Comware	The switch Openflow agent does not apply a tag to injected link discovery packets.
Other Vendors	See vendor-specific documentation regarding 802.1Q vlan tagging of controller-injected packets.

Discovery packet reception

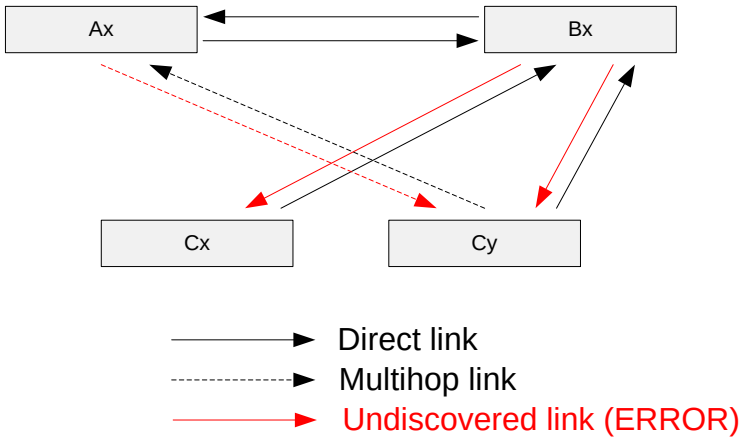
Product	Reception of Untagged Packets
ProVision	Untagged *direct* discovery packets are properly stolen to the controller. Untagged *multi-hop* discovery packets are associated with the untagged vlan on the port where they are received and properly stolen to the controller.
Comware	Untagged *direct* discovery packets are properly stolen to the controller if the Comware device is running a revision of firmware which has resolved issue 201402110424 (ex: 5503L01 for 5500HI) and the pvid on each inter-switch port is configured with the correct OpenFlow instance. Untagged *multi-hop* discovery packets are properly stolen to the controller if the ingress ports have the correct pvid configured that associates untagged packets with the correct OpenFlow instance vlan. NOTE: The inter-switch ports in an OpenFlow instance MUST NOT be configured with "bpdu-drop any" since the direct link discovery packet MAC is in the BPDU MAC range.
Other Vendors	See vendor-specific documentation regarding handling of untagged *direct* and *multi-hop* discovery packets shown above.

Non-conforming configuration example



Using the example topology above, we will illustrate a few of the issues that may arise with link discovery. Assume that switches A, B, C OpenFlow instances are configured to connect to the same controller and that each inter-switch physical connection is tagged with the vlans shown. The notation “OF Inst” indicates an OpenFlow instance running on the given switch.

The following will be the discovered topology:



All of the undiscovered links have switch C as their destination, so we may be tempted to assume that this is a misconfiguration on switch C. Actually, these links would be discovered with configuration changes on either switch C or the other controlled switches (A, B).

- The multi-hop link from Ax to Cy is not discovered because the controller injects an untagged link discovery packet from Ax port 1. The untagged packet is associated with vlan 3 by the uncontrolled switch, so the link discovery packet is never received by switch C.
- The direct links from Bx to Cx and Cy are not discovered because the controller injects an untagged link discovery packet from Bx on port 2. The untagged packet is received by switch C and associated with the untagged vlan on switch C port 2.
  - If the untagged vlan on switch C port 2 is vlan 1, only the Bx to Cx link will be discovered.
  - If the untagged vlan on switch C port 2 is vlan 2, only the Bx to Cy link will be discovered.
  - If the untagged vlan on switch C port 2 is neither vlan 1 or 2, no Bx to Cx/y links will be discovered.

Device-specific recommendations

In general, HP recommends that OpenFlow instance-to-vlan mappings remain consistent throughout the controlled network topology. If an OpenFlow instance contains a set of vlans on one switch, then neighboring switches should also have an OpenFlow instance with the same set of vlans.

ProVision: Virtualization Mode

There are no known issues when using ProVision products in virtualization mode. Since virtualization mode limits each OpenFlow instance to a single vlan, the vlan can be derived from the OpenFlow instance. Any packets which the controller injects into an OpenFlow instance in virtualization mode are tagged correctly by the OpenFlow instance - an 802.1Q header is added, if the egress port is configured to be tagged. No known issues exist under these circumstances.

Comware: PVID in OpenFlow instance

There are no known issues with link discovery (limited testing) when using Comware devices when the port’s pvid is set to an OpenFlow vlan. If any link discovery packets arrive untagged on a port, they will be assigned to the pvid. If the pvid is set to an OpenFlow vlan, then all link discovery packets will be received by the controller from the OpenFlow instance which contains that vlan.

If multiple OpenFlow instances are configured on a single Comware device, then at least one port must be used per OpenFlow instance per neighboring switch. This is required because only one pvid can be assigned per port, and no single vlan can be associated with multiple OpenFlow instances.

Previous discovery methods

LLDP Ethertype

Rather than using an Ethernet type of **0x8999**, previous versions of the controller injected packets with an Ethernet type of LLDP (**0x88cc**). By using this Ethernet type for direct link discovery packets, all direct links would be correctly discovered because LLDP is defined as being below the 802.1Q vlan tagging layer. This approach is no longer considered an option because LLDP is also used by end-hosts for other purposes, such as PoE. If the controller injects LLDP packets which reach end hosts, those LLDP packets will cause issues with technologies that depend on LLDP for configuration data, such as VoIP phones and wireless access points (among others). In testing, it was discovered that injection of LLDP packets from the controller for discovery purposes caused IP phones and PoE devices to malfunction.

Learn more at
hp.com/go/sdn/infolib

Sign up for updates
hp.com/go/getupdated

5998-6990, October 2014