

## Table of Contents

<b>1. Setup Requirements .....</b>	<b>2</b>
1.1. What Will Be Provided .....	2
1.2. Hardware Requirements .....	2
1.3. Firewall/NAT Requirements .....	2
<b>2. VeloCloud Gateway Installation.....</b>	<b>3</b>
2.1. Install Gateway Software .....	3
2.2. Disable Offload Features on Interface .....	3
2.3. VeloCloud Gateway Network Configuration.....	4
2.4. Remove the blocked subnets .....	6
<b>3. Configure VCG on the VCO .....</b>	<b>7</b>
3.1. Adding VeloCloud Gateway to the VCO .....	7
3.2. Enable Partner Gateway Mode.....	8
3.3. Configure Handoff Detail Per Customer .....	9
3.4. Per-Customer BGP Configuration .....	10
3.5. Gateway Assignment for VeloCloud Edge .....	11
<b>4. Verification .....</b>	<b>13</b>
4.1. Verify Customer VRF Configuration on Partner Gateway.....	13
4.2. Verify VCE Routes On Partner Gateway .....	13
<b>5. Notes and Considerations .....</b>	<b>15</b>
5.1. Special Consideration When Using 802.1ad encapsulation.....	15

This document describes the steps needed to install and deploy VeloCloud Gateway (VCG) as a Partner Gateway in the 2.2 and 2.3 software release. It also covers how to configure the VRF/VLAN and BGP configuration necessary on the VeloCloud Orchestrator (VCO).

## 1. Setup Requirements

### 1.1. What Will Be Provided

- Gateway OVA package or qcow2 image, e.g. `velocloud-vcg-2.3.0-R23-20161227-GA.ova`

### 1.2. Hardware Requirements

The VeloCloud Gateway is recommended to run on virtual hardware. Below is the minimum specs required for the gateway.

- 8 Intel vCPU's at 2.0 Ghz or higher. CPU must support AES-NI, SSSE3 and RDTSC instruction sets.
- 16 GB of memory
- 24 GB magnetic or SSD based, persistent disk volume
- 2 x 1 Gbps (or higher) network interface
- eth0 must be the interface connected to the public network and responsible for serving customer traffic
- Single public IP address must be provisioned on eth0 (Can be either applied directly or be NAT'd on an upstream device)
- eth1 is used for handing of customer traffic to the partner PE router.
- PE router must be able to support 802.1q VLAN tagging and QinQ support as needed

VeloCloud recommends using the following components for the underlying hardware:

- Intel Xeon E5 Family processors
- Intel dual 10 Gbps NIC with support for DPDK (<http://dpdk.org/doc/nics>)

### 1.3. Firewall/NAT Requirements

If the VeloCloud Gateway is deployed behind Firewall and/or NAT device, the following applies

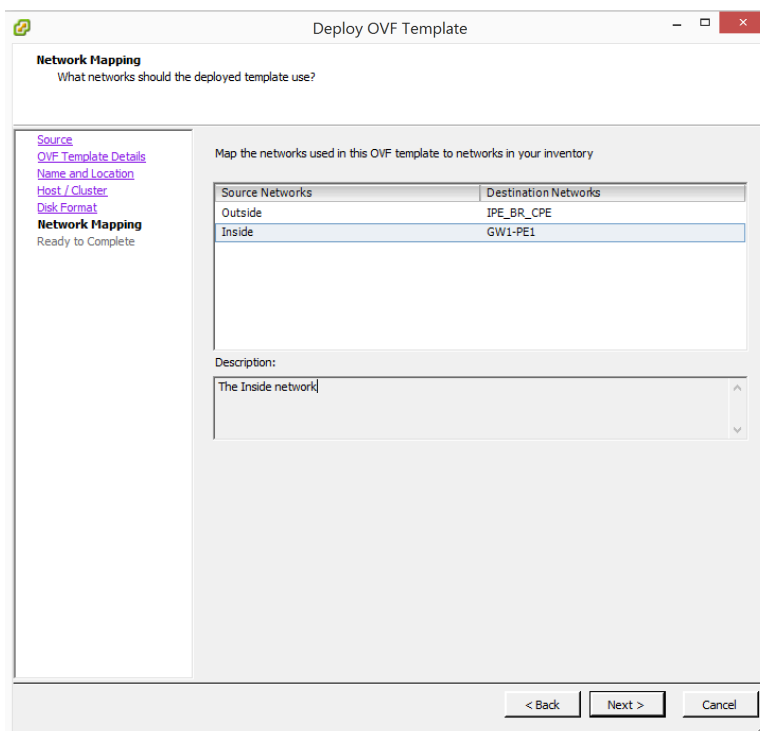
- Firewall needs to allow outbound traffic from the VeloCloud Gateway to TCP/443 (for communication with VeloCloud Orchestrator)
- Firewall needs to allow inbound traffic from the Internet to UDP/2426 (VCMP), UDP/4500, UDP/500. If NAT is not used, then firewall needs to also allow IP/50 (ESP)
- If NAT is used, the above ports must be translated to externally reachable IP address. Both the 1:1 NAT and port translation are supported.

## 2. VeloCloud Gateway Installation

### 2.1. Install Gateway Software

**Step 1:** Deploy the provided ova package in a VMware Hypervisor or the qcow2 image in KVM. Note that the required quagga version should already be included with the base image.

**Step 2:** Follow the respective VM installation process. Map the appropriate networks to the VM's interfaces. "Outside" is the interface associated with the public network and "Inside" is the interface connected to the PE router.



**Step 3:** After the boot up sequence, login to the VM with the username and password provided by VeloCloud.

### 2.2. Disable Offload Features on Interface

- Disable RX/TX checksum on physical interfaces

```
ethtool -K <interface> rx off tx off
```

- Disable GRO (Generic Receive Offload) on physical interfaces (to avoid unnecessary re-fragmentation in VCG)

```
ethtool -K <interface> gro lro tso off
```

- Disable CPU C-states (power states affect real-time performance). Typically this can be done as part of kernel boot options by appending **processor.max\_cstate=1** or just disable in BIOS. See

[https://docs.fedoraproject.org/en-US/Fedora/13/html/Virtualization\\_Guide/chap-Virtualization-KVM\\_guest\\_timing\\_management.html](https://docs.fedoraproject.org/en-US/Fedora/13/html/Virtualization_Guide/chap-Virtualization-KVM_guest_timing_management.html)

- For production deployment, vCPU's must be pinned to the instance. No oversubscription on the cores should be allowed to take place. See [https://docs.fedoraproject.org/en-US/Fedora/13/html/Virtualization\\_Guide/ch25s06.html](https://docs.fedoraproject.org/en-US/Fedora/13/html/Virtualization_Guide/ch25s06.html)

## 2.3. VeloCloud Gateway Network Configuration

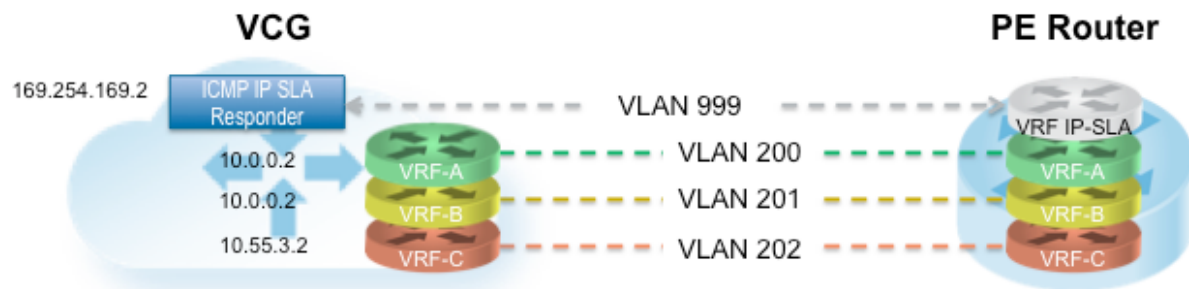


Figure 1 - VRF/VLAN Hand Off to PE

In this example, we assume eth0 is the interface facing the public network (Internet) and eth1 is the interface facing the internal network (customer VRF through the PE). BGP peering configuration is managed on the VCO on a per customer/VRF basis under "Configure > Customer". Note that the IP address of each VRF is configurable per customer. The IP address of the management VRF inherits the IP address configured on the VCG interface in Linux.

A **Management VRF** is created on the VCG and is used to send periodic ARP refresh to the default gateway IP to determine the next-hop MAC. It is recommended that a dedicated VRF is set up on the PE router for this purpose. The same management VRF can also be used by the PE router to send IP SLA probe to the VCG to check for VCG status (VCG has stateful ICMP responder that will respond to ping only when its service is up). BGP Peering is not required on the Management VRF.

If a Management VRF is not set up, then you can use one of the customer VRFs as Management VRF, although this is not recommended.

**Step 1:** Configure the network interfaces in `/etc/network/interfaces` file.

**IMPORTANT:** Note the metric for the public interface (eth0) must be lower than the Hand Off interface so eth0 is properly used as the default route for Internet traffic. Also, a default gateway **must** be configured on the Inside interface for the VCG to properly monitor the status of the interface.

```
# OUTSIDE OR PUBLIC INTERFACE
auto eth0
iface eth0 inet static
    metric "1"
    address <IP ADDRESS>
    netmask <NETMASK>
    network <NETWORK ADDRESS>
    broadcast <BROADCAST>
```

```

gateway <NEXT-HOP PUBLIC>
# dns-* options are implemented by the resolvconf package, if
installed
dns-nameservers 8.8.8.8 8.8.4.4
dns-search localdomain

#INSIDE OR HANDOFF INTERFACE
auto eth1
iface eth1 inet static
    metric "13"
    address 192.168.1.2
    netmask 255.255.255.0
    network 192.168.1.0
    broadcast 192.168.1.255
    gateway 192.168.1.1

```

**Step 2:** Edit the `/etc/config/gatewayd` and specify the correct VCMP and WAN interface. VCMP interface is the public interface that terminates the overlay tunnels. The WAN interface in this context is the handoff interface.

```

"vcmp.interfaces": [
    "eth0"
],

(..snip..)

"wan": [
    "eth1"
],

```

**Step 3:** Configure the **Management VRF**. This VRF is used by the VCG to ARP for next-hop MAC (PE router). The same next-hop MAC will be used by all the VRFs created by the VCG. You need to configure the **Management VRF** parameter in `/etc/config/gatewayd`.

The Management VRF is the same VRF used by the PE router to send IP SLA probe to. The VCG only responds to the ICMP probe if the service is up and if there are edges connected to it. Below table explains each parameter that needs to be defined. This example has Management VRF on the 802.1q VLAN ID of 1000.

<b>mode</b>	QinQ (0x8100), QinQ (0x9100), none, 802.1Q, 802.1ad
<b>c_tag</b>	C-Tag value for QinQ encapsulation or 802.1Q VLAN ID for 802.1Q encapsulation
<b>s_tag</b>	S-Tag value for QinQ encapsulation
<b>interface</b>	Handoff interface, typically eth1

```

"vrf_vlan": {
    "tag_info": [

```

```
{
  "resp_mode": 0,
  "proxy_arp": 0,
  "c_tag": 1000,
  "mode": "802.1Q",
  "interface": "eth1",
  "s_tag": 0
}
],
```

**Step 4:** Edit the `/etc/config/gatewayd-tunnel` to include both interfaces in the wan parameter. Save the change.

```
wan="eth0 eth1"
```

## 2.4. Remove the blocked subnets

By default, the VCG blocks traffic to 10.0.0.0/8 and 172.16.0.0/14. We will need to remove them before using this VCG because we expect VCG to be sending traffic to private subnets as well. If you do not edit this file, when you try to send traffic to blocked subnets, you will find the following messages in `/var/log/gwd.log`

```
2015-12-18T12:49:55.639  ERR      [NET] proto_ip_recv_handler:494 Dropping packet destined for
10.10.150.254, which is a blocked subnet.
2015-12-18T12:52:27.764  ERR      [NET] proto_ip_recv_handler:494 Dropping packet destined for
10.10.150.254, which is a blocked subnet. [message repeated 48 times]
2015-12-18T12:52:27.764  ERR      [NET] proto_ip_recv_handler:494 Dropping packet destined for
10.10.150.10, which is a blocked subnet.
```

**Step 1:** On VCG, edit `/opt/vc/etc/vc_blocked_subnets.json` file. You will find that this file first has the following.

```
[
  {
    "network_addr": "10.0.0.0",
    "subnet_mask": "255.0.0.0"
  },
  {
    "network_addr": "172.16.0.0",
    "subnet_mask": "255.255.0.0"
  }
]
```

**Step 2:** Remove the two networks. The file should look like below after editing. Save the change.

```
[
]
```

**Step 3:** Restart the VCG process by `sudo /opt/vc/bin/vc_procmon restart`.

## 3. Configure VCG on the VCO

### 3.1. Adding VeloCloud Gateway to the VCO

**Step 1:** Go to **Operator > Gateway Pool** and create a new VeloCloud Gateway pool. For running VeloCloud Gateway in the Service Provider network, select “Allow” or “Only Partner Gateways” under the drop down for **Partner Gateway Hand Off**. This will enable the option to include the partner gateway in this gateway pool.

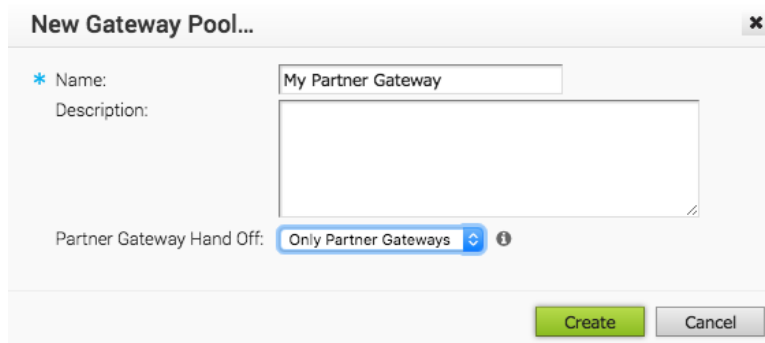


Figure 2 - Create Partner Gateway Pool

**Step 2:** Go to **Operator > Gateway** and create a new gateway and assign to the pool. The IP address of the gateway entered here has to match the **public IP address** of the gateway. If unsure, you can run **curl ipinfo.io/ip** from the VCG which will return the public IP of the VCG.

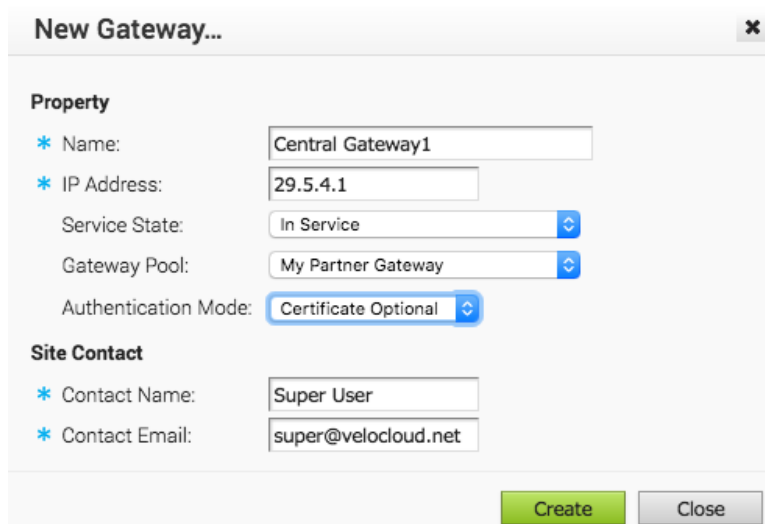


Figure 3 - Add new Partner Gateway

**Step 3:** Make note of the activation key

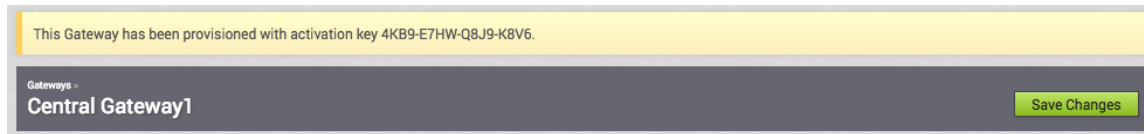


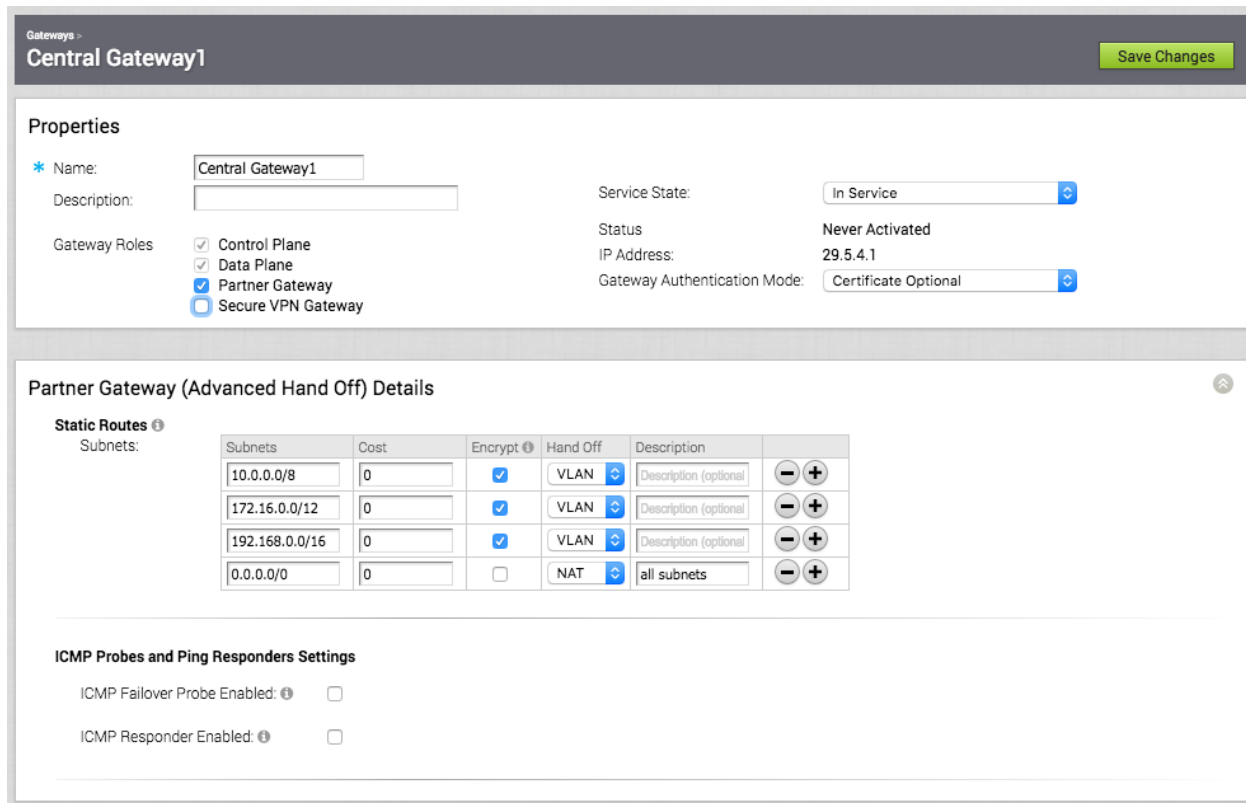
Figure 4 - Get the Gateway Activation Key

**Step 4:** Go to the VeloCloud Gateway via SSH access and activate the gateway

```
cd /opt/vc/bin
./activate.py -s <vco-fqdn-hostname-or-ip> -i <activation-key>
```

## 3.2. Enable Partner Gateway Mode

**Step 1:** In the same gateway page (**Operator > Gateways**), enable the Partner Gateway mode by checking the checkbox **Partner Gateway**. Uncheck the checkbox Secure VPN Gateway (this is needed only if you plan to use this VCG to establish IPsec tunnel to non-VeloCloud site)



Gateways > Central Gateway1 Save Changes

### Properties

\* Name: Central Gateway1

Description:

Gateway Roles:

- ☒ Control Plane
- ☒ Data Plane
- ☒ Partner Gateway
- ☐ Secure VPN Gateway

Service State: In Service

Status: Never Activated

IP Address: 29.5.4.1

Gateway Authentication Mode: Certificate Optional

### Partner Gateway (Advanced Hand Off) Details

**Static Routes**

Subnets	Cost	Encrypt	Hand Off	Description	
10.0.0.0/8	0	<input checked="" type="checkbox"/>	VLAN	Description (optional)	- +
172.16.0.0/12	0	<input checked="" type="checkbox"/>	VLAN	Description (optional)	- +
192.168.0.0/16	0	<input checked="" type="checkbox"/>	VLAN	Description (optional)	- +
0.0.0.0/0	0	<input type="checkbox"/>	NAT	all subnets	- +

**ICMP Probes and Ping Responders Settings**

ICMP Failover Probe Enabled: ☐

ICMP Responder Enabled: ☐

Figure 5 - Gateway Properties and Global Handoff Details

There are additional parameters that can be configured.

- **Static Routes:** Specify which subnets or routes that the VCG should advertise to the VeloCloud Edge, along with the handoff mode and whether or not to encrypt the traffic. This is global per VCG and



applies to **ALL** customers. With BGP, this section is typically used only if there is a shared subnet that all the customers need to access and if NAT handoff is required.

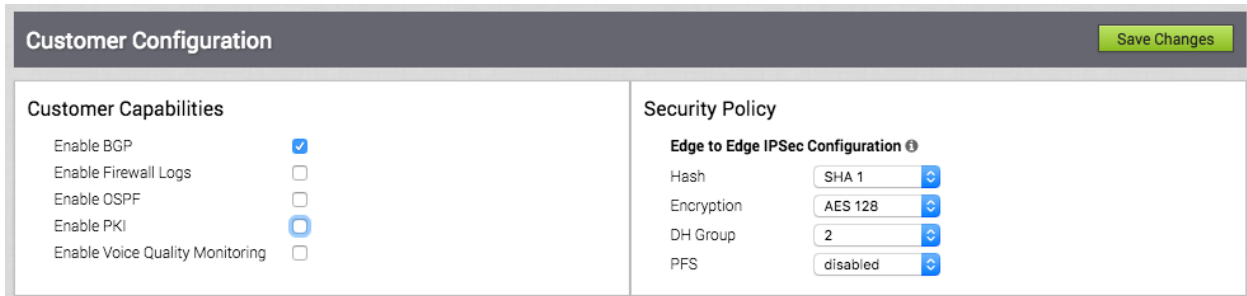
Remove the unused subnets from the Static Route list above if you do not have any subnets that you need to advertise to the VCE and have the handoff of type NAT.

The ICMP probe parameters are optional and recommended only if you want to use ICMP to check the health of the VCG. With BGP support on the Partner Gateway, using ICMP probe for failover and route convergence is no longer required.

- **ICMP Failover Probe:** The VeloCloud Gateway can use ICMP probe to check for the reachability of a particular IP and can notify the VeloCloud Edge to failover to the secondary gateway if the VeloCloud Gateway detects that the particular IP is not reachable.
- **ICMP Responder Enabled:** This will allow the VeloCloud Gateway to respond to the ICMP probe from the next hop router when its tunnels are up.
  - **Mode=Conditional:** The VCG will only respond to the ICMP request when its service is up and when at least one tunnel is up
  - **Mode=Always:** The VCG will always respond to the ICMP request from its peer

### 3.3. Configure Handoff Detail Per Customer

**Step 1:** Go to the customer level, under **Configure > Customer**, check the checkbox **Enable BGP**. Refresh the Web browser for this change to take effect.



The screenshot shows the 'Customer Configuration' page. On the left, under 'Customer Capabilities', the 'Enable BGP' checkbox is checked, while 'Enable Firewall Logs', 'Enable OSPF', 'Enable PKI', and 'Enable Voice Quality Monitoring' are unchecked. On the right, under 'Security Policy', the 'Edge to Edge IPsec Configuration' section shows 'Hash' set to 'SHA 1', 'Encryption' set to 'AES 128', 'DH Group' set to '2', and 'PFS' set to 'disabled'. A 'Save Changes' button is located in the top right corner.

Figure 6 - Customer Capabilities

**Step 2:** In the **Gateway Pool** section, select the gateway and click **Click here to configure** link which will pop up another window. Configure the handoff details per Gateway or for all the Gateways used by this customer, e.g. if QinQ is used, specify the S-Tag/C-Tag for mapping this customer traffic into the corresponding VRF.

For most deployment use cases, the configuration Per Gateway is required and is recommended.

Gateway Pool

SP Partner Gateway

	Gateway	IP Address	...	...
1	vcg-kansas	24.6.180.28	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	vcg-reston	24.6.180.27	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Central Gateway1	24.6.180.29	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Enable Partner HandOff ☒

Configure Hand Off

☐ All Gateways ⓘ

☒ Per Gateway ⓘ

Select Gateway Central Gateway1

Gateway "Central Gateway1" Hand Off

This gateway has not been configured for handoff for the current customer.

[Click here to configure](#)

Figure 7 - Per Customer Gateway Configuration

Hand Off Details: 'vcg-reston' Gateway

Hand Off Interface

Tag Type 802.1Q

C-Tag (Customer tag): 101

Local IP Address ⓘ 192.168.101.2/24

Use for Private Tunnels: ⓘ ☒

Advertise via BGP: ⓘ ☒

Static Routes

Subnets	Cost	Encrypt ⓘ	Hand Off	Description	
Ex: 10.0.2.0/24	Cost (1-100)	<input type="checkbox"/>	VLAN	Description (opti	[-] [+]

BGP

Enable BGP ☐

Update Cancel

Figure 8 - Gateway Handoff Details

### 3.4. Per-Customer BGP Configuration

**Step 1:** Under the Hand Off Details page per Figure 8 - Gateway Handoff Details, check the **Enable BGP** check box and add the BGP peering configuration to be applied to the corresponding Partner Gateway.

**Step 2:** Enable **Use for Private Tunnels** and **Advertise via BGP** if you plan to also build the VCMP Overlay across your MPLS network in 2-arm mode.

**IMPORTANT:** The 2-arm mode refers to the case when the Partner Gateway expects VCMP to come in through both the public and private (VRF side) interfaces. This is needed so a VCE WAN interface connected to a CE router can reach the Gateway over the private link to establish the Overlay. If you leak the Partner Gateway public IP into the customer VRF, then do not enable this checkbox

Note the VCG to PE peering only supports single-hop EBGP peering in the initial 2.2 Release.

Hand Off Details: 'vcg-reston' Gateway

Hand Off Interface

Tag Type

802.1Q

C-Tag (Customer tag):

101

Local IP Address

192.168.101.2/24

Use for Private Tunnels:

☒

Advertise via BGP:

☒

Static Routes

Subnets	Cost	Encrypt	Hand Off	Description	
Ex: 10.0.2.0/24	Cost (1-100)	<input type="checkbox"/>	VLAN	Description (opti	- +

BGP

Enable BGP

☒

Customer ASN

65100

Neighbor IP

192.168.101.1

Neighbor-ASN

65000

Secure BGP Routes

☐

BGP Inbound Filters

Match		Exact Match	Action	
Type	Value		Type	Set
Prefix	subnet	<input checked="" type="checkbox"/>	Permit	None

BGP OutBound Filters

Match		Exact Match	Action	
Type	Value		Type	Set
Prefix	subnet	<input checked="" type="checkbox"/>	Permit	None

Update

Cancel

Figure 9 - Per Gateway BGP Configuration

To support multiple customers on the Hand Off interface, a simple configuration would be to use 801.Q tag for the customer traffic to be handed off to the PE and configure sub-interfaces on the PE side for each customer VRF. BGP peering between the VCG and PE can then be configured per customer VRF.

The Hand Off Details page also provides the BGP filters to affect traffic flow based on BGP attributes and influence Gateway selection on the VCE's.

## 3.5. Gateway Assignment for VeloCloud Edge

Once the Partner Gateways are configured for a given customer, they can be assigned to the VCE's belonging to the Enterprise. In 2.2 Release, a VCE can connect to 2 Partner Gateways simultaneously per assignment under Edge Overview configuration. Gateway selection for a given destination prefix on the VCE is chosen based on the received BGP attributes from the Partner Gateways. If all attributes are equal for a prefix, the first Gateway in the assignment order is preferred.

**Step 1:** Go to the Edge configuration page, under **Configure > Edges > select VCE > Edge Overview, Partner Gateway Assignment**, assign the Gateways the VCE should connect to with the preferred Partner Gateway assigned as Gateway 1.

VeloCloud Edges >
Customer1 Silver2 VCE-1 (Connected)
Save Changes
?

Edge Overview
Device
Business Policy
Firewall

### Properties

\* Name:
Customer1 Silver2 VCE-1

Description:

Enable Pre-Notifications:
☒

Enable Alerts:
☒

Status:
Activated

Activated:
Wed Sep 21, 20:05

Software Version:
2.2.0 (build R22-20160914-GA-CANDIDATE)

Local Credentials:
\*\*\*\*\*
View...

### Profile

### Partner Gateway Assignment

Gateway 1:
vcg-reston

Gateway 2:
vcg-kansas

Figure 10 - Per Gateway BGP Configuration

## 4. Verification

### 4.1. Verify Customer VRF Configuration on Partner Gateway

This step displays the configurations in section 3.3 and 3.4 on the VCG through CLI. SSH access to the Partner Gateway is required.

**Step 1:** For customer hand off configuration in section 3.3, `/opt/vc/bin/debug.py --vrf` can be executed to display the existing Enterprises with Partner Gateway enabled and their corresponding VRF's. Output from `/opt/vc/bin/debug.py --bgp_view_summary` can be used to display all the BGP neighbors that the VCG peers with. The output below shows an example of a customer configuration using 802.1Q tag for hand off, and the VCG peering with a PE using BGP ASN of 65000.

```
ubuntu@vcg-reston:~$ /opt/vc/bin/debug.py --vrf
{
  "vrf_dump": [
    {
      "c_tag": 101,
      "enterprise_id": "9679aeba-dbd1-4d13-9dcf-8854c62d783d",
      "enterprise_name": "SP - Demo - Customer1",
      "mode": "802.1Q",
      "s_tag": 0
    }
  ]
}
```

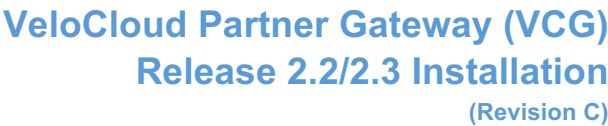
```
ubuntu@vcg-reston:~$ /opt/vc/bin/debug.py --bgp_view_summary
enterpriseLogicalId      neighborIp  neighborAS  msgRcvd  msgSent
upDownTime      state  pfxRcvd
9679aeba-dbd1-4d13-9dcf-8854c62d783d  192.168.101.1  65000  7898  7231
3d22h27m  Established  12
dispEntries 1 startEntryIdx 0 totalEntries 1
```

Note the significance of the **enterpriseLogicalId**. It's an identifier on the VCG for the corresponding customer VRF. It can be used to display outputs on the Partner Gateway for the specific customer VRF (see section 4.2).

### 4.2. Verify VCE Routes On Partner Gateway

After a VCE is assigned with the appropriate Partner Gateways, we can verify the advertisement of the VCE's prefixes on the Partner Gateways by looking at the output of `/opt/vc/bin/debug.py --routes`. From the example output below, **EnterpriseID** classifies the customer VRF and we can identify the prefixes originating from the VCE's such as 10.168.0.0/26 and 10.168.64.0/26 showing in the table as local routes with **Type** "edge2edge". These routes showing as local to the Gateways greatly simplifies large branch office deployments and centralizes layer 3 troubleshooting with the VeloCloud SD-WAN architecture.

```
ubuntu@vcg-reston:~$ /opt/vc/bin/debug.py --routes
EnterpriseID      Address      Netmask      Type
Destination  Reachable  Metric  Preference  Flags  C-Tag  S-Tag  Handoff  Mode
9679aeba-dbd1-4d13-9dcf-8854c62d783d  10.168.128.2  255.255.255.255  edge2edge  2a679377-
53a3-4585-978b-156cd1b27c1d  True  0  0  0x4090  1  0  N/A
```



VeloCloud Networks **Confidential** – For customers/partners with NDA Page 14 of 15

## 5. Notes and Considerations

### 5.1. Special Consideration When Using 802.1ad encapsulation

It seems certain 802.1ad devices do not populate the outer tag EtherType with 0x88A8. Special change is required in **/etc/config/gatewayd** to interoperate with these devices.

Assuming a Management VRF is configured with **S-Tag: 20** and **C-Tag: 100**, edit the **vrf\_vlan** section in **/etc/config/gatewayd** as follow. Also define **resp\_mode** to 1 so that the VCG will relax its check to allow Ethernet frames that have incorrect EtherType of 0x8100 in the outer header.

```
"vrf_vlan": {
  "tag_info": [
    {
      "resp_mode": 1,
      "proxy_arp": 0,
      "c_tag": 100,
      "mode": "802.1ad",
      "interface": "eth1",
      "s_tag": 20
    }
  ]
},
```