

Midonet Cross-site Router Peering

[MN OSS Community Document]

[Revision History](#)

[References](#)

[Assumptions](#)

[Terms](#)

[Router Peering User Stories](#)

[Router-Peering Using VTEP-Router](#)

[Option 1 - Provider router for Internet and WAN traffic](#)

[Option 2 - VTEP-Router is an edge-router to the WAN](#)

[Networks Diagrams](#)

[Option 1 - Provider router for Internet and WAN traffic](#)

[Option 2 - VTEP-Router is an edge-router to the WAN](#)

[Forwarding Tables](#)

[Operations to Neutron API mapping Table](#)

[Workflow](#)

[VTEP device management API](#)

Revision History

Date	Author	Notes
21 Oct. 2015	Alon irena	Initial proposal
25 Oct. 2015	Alon	Added Workflow
2 Nov. 2015	Alon irena	1. Added Terms table 2. Changed GW to be a router with peering-enable 3. Added clarifications to existing workflow
17 Nov. 2015	Alon irena	1. PEERING_VTEP_NETWORK removed 2. Workflow reworked

26 Nov. 2015	Alon irena	1. Added the option of the VTEP=Router act as an edge router 2. Added 'VTEP device management API'
17 Dec. 2015	Alon Pino	1. Added ARP entries setting API 2. Updated - no support for BC by VTEP router for 1st phase

References

1	Demo slides - VPC Router Peering
2	Router Peering Operational Guide

Assumptions

- User interacts with orchestration layer only
- There is an admin owned VTEP-router per AZ
- There is a peering tenant router per AZ which is implicitly created by orchestration layer (to connect to the VTEP-router)
- VTEP-router does not support broadcast
- As long as peering tenant router does not support Dynamic Routing - there is a need to update local peering tenant router with routes to tenant subnets in other AZs
- VTEP's [mac →VTEP IP] table is populated by orchestration

Terms

Term	Description
VTEP_ROUTER	Admin Router that is used as a GW device for peering Tenants across sites
TENANT_ROUTER	Tenant router that is used for peering
TENANT_INTER_AZ_NET	The logical L2 network on which all specific

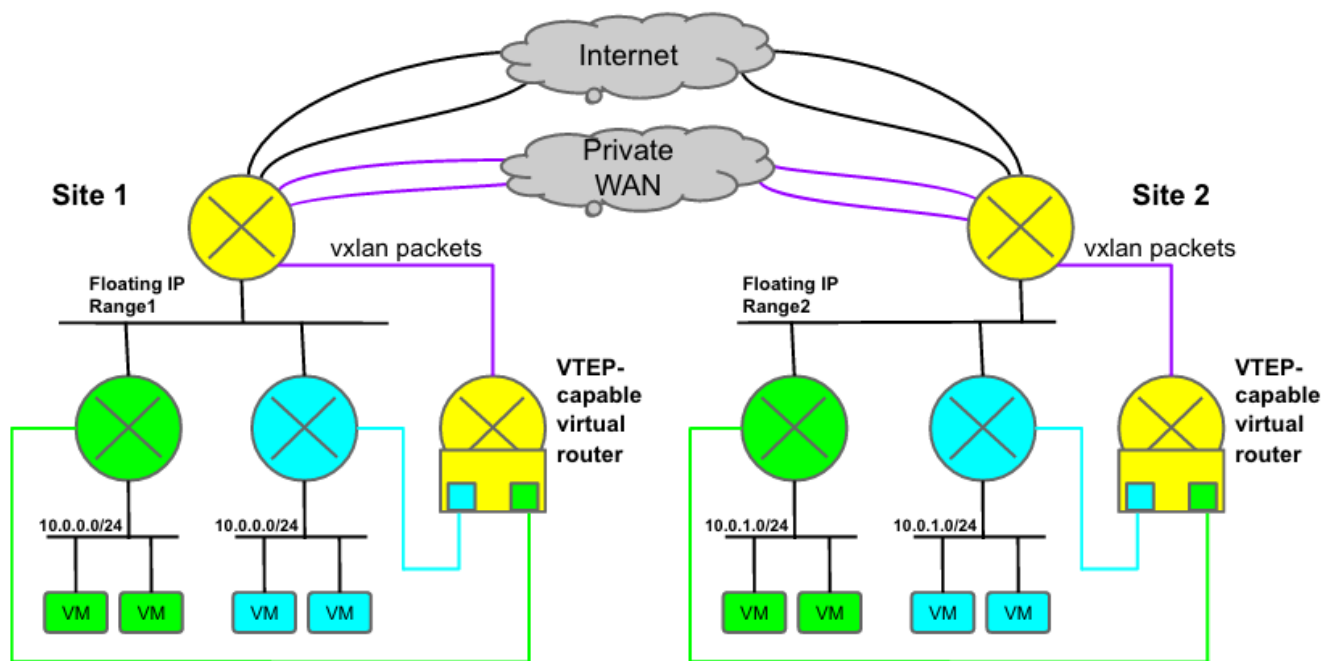
	tenant's peering-routers are connected
TENANT_INTER_AZ_IP_ADDRESS	The ip-address of the local peering-router on the subnet used for TENANT_INTER_AZ_NET
WAN_IP_ADDRESS	The address of the VTEP-router or the edge router on the WAN
PROVIDER_VTEP_NETWORK	Network between Provider router and VTEP router

Router Peering User Stories

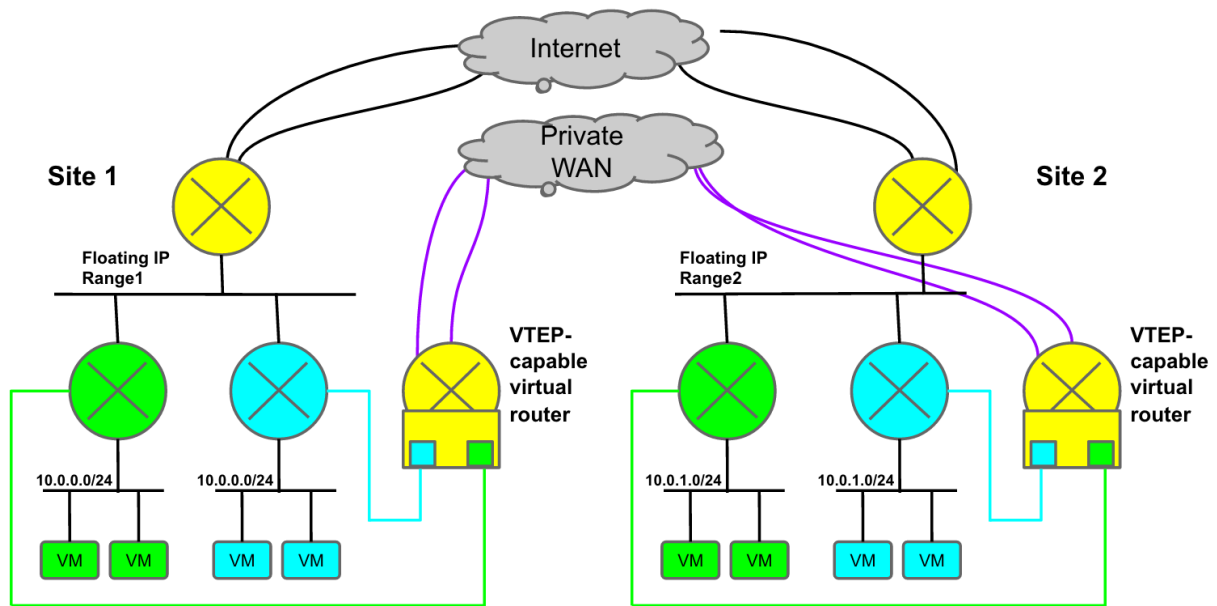
1. As a cloud admin , I want to create a VTEP router in one site so it can be set as the gw-device accessible from outside of the cloud platform
2. As a tenant admin I want to create a border-gw associated with the VTEP router so that I have connectivity between peering-routers in different sites
3. As a tenant admin I want to connect subnets to my peering-router so that I can route between private subnets on different sites
4. As a cloud admin I want to be able to allow overlapping tenants subnets to coexist so that each can route between its own private subnets on different sites
5. As a tenant admin I want to be able to add/delete subnets to/from my peering-router
6. As a tenant admin I want to be able to delete my border-gw associated with the VTEP router
7. As a cloud admin I want to be able to connect a new site so that existing sites can communicate with it
8. As a cloud admin ,I want to create a **edge** VTEP router so it can be connected directly to the WAN

Router-Peering Using VTEP-Router

Option 1 - Provider router for Internet and WAN traffic

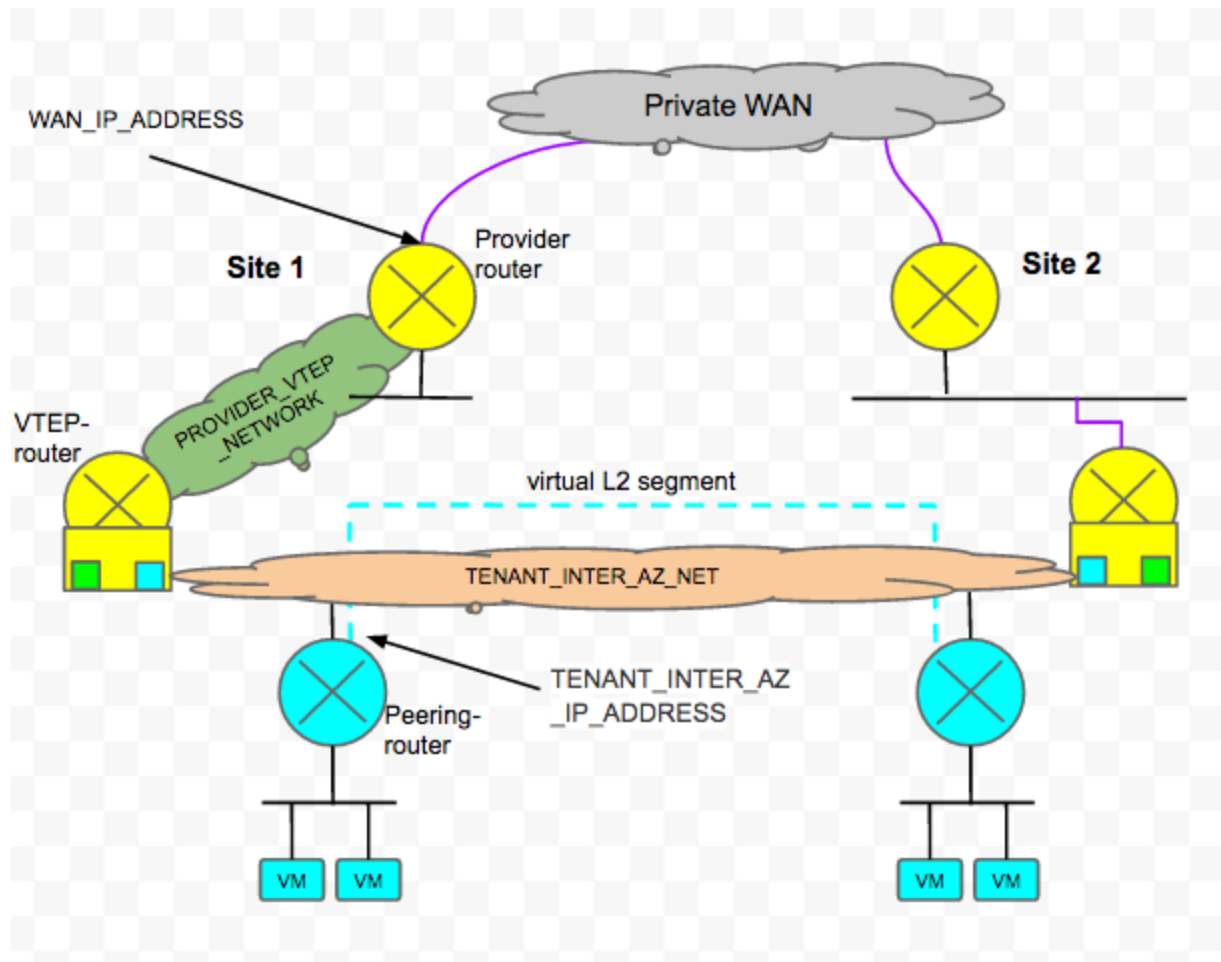


Option 2 - VTEP-Router is an edge-router to the WAN

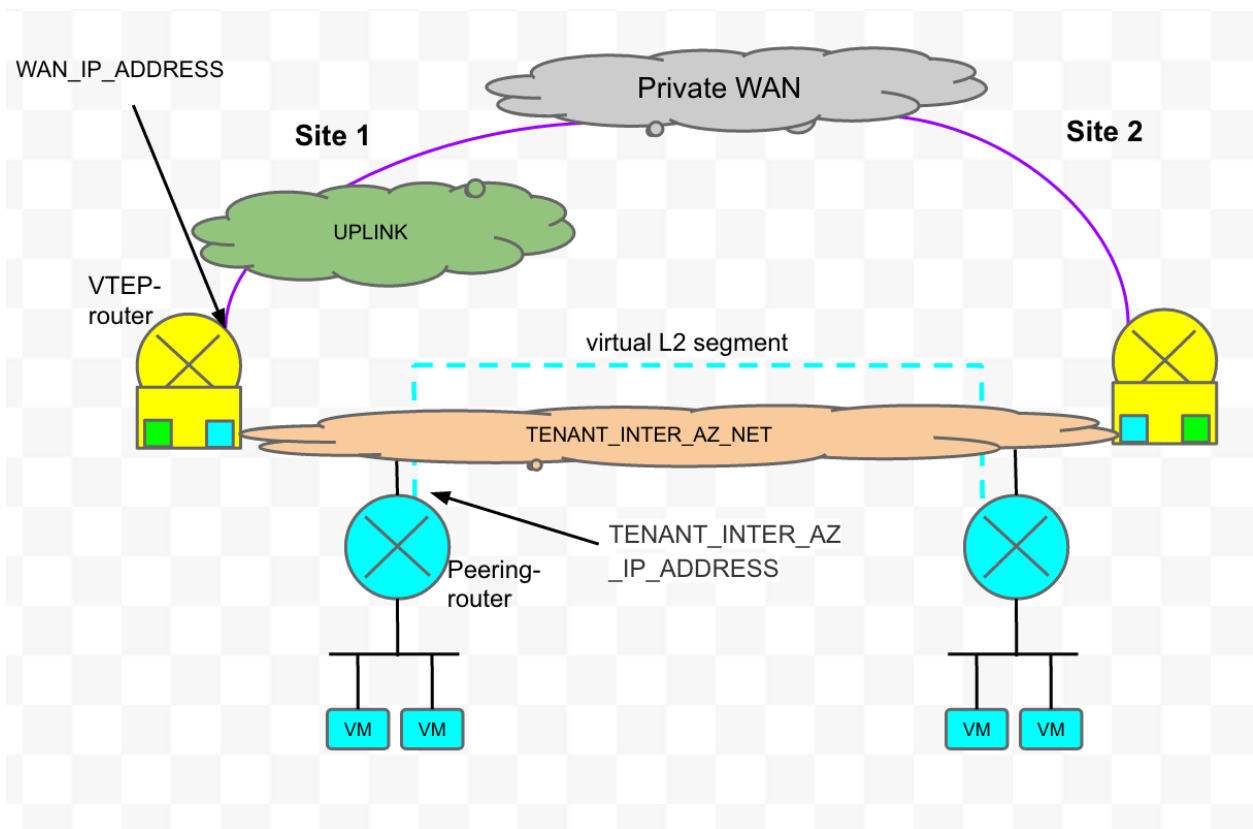


Networks Diagrams

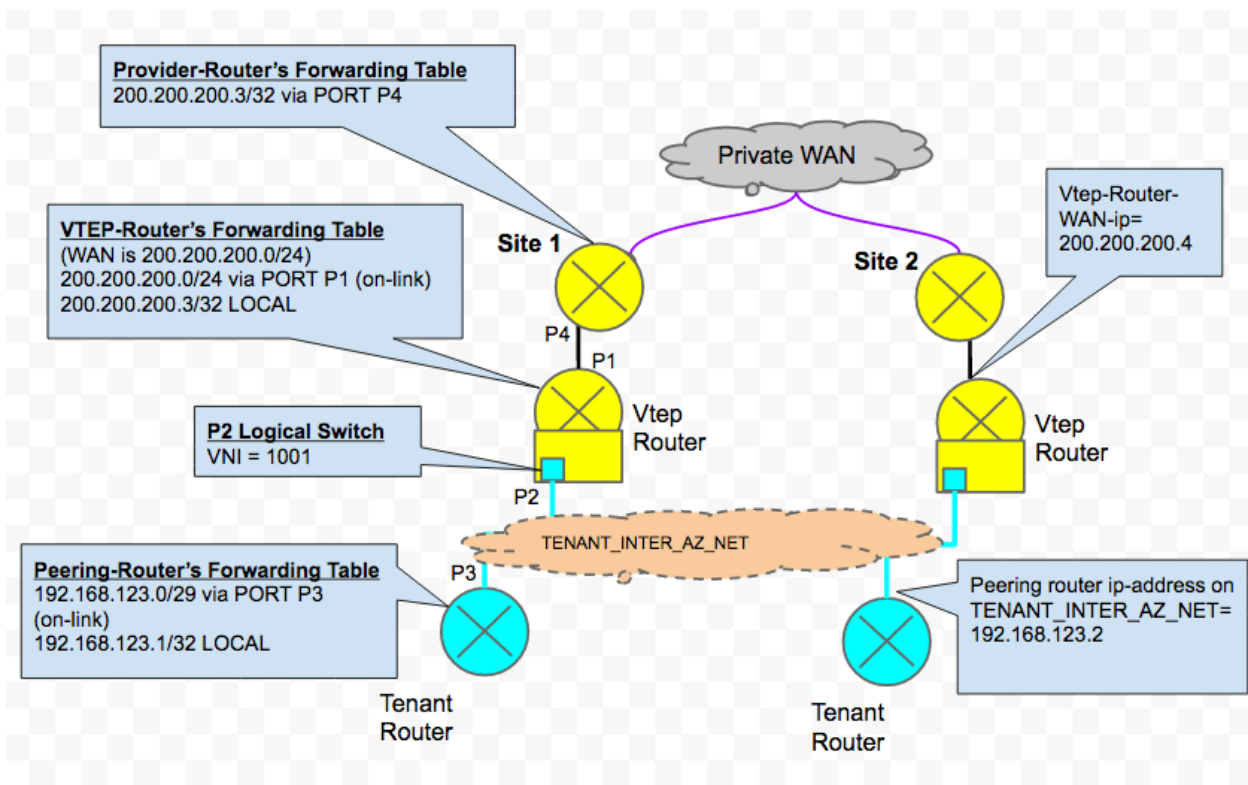
Option 1 - Provider router for Internet and WAN traffic



Option 2 - VTEP-Router is an edge-router to the WAN



Forwarding Tables



Operations to Neutron API mapping Table

#	Operation	API	Comments	Authority
1	Create VTEP-Router	neutron router-create VTEP_ROUTER_NAME		Admin
2	Define gateway device	neutron gateway-device-create --resource-id = VTEP_ROUTER_UUID	New API to manage VTEP-router device.	Admin
3	Create Logical GW	neutron 'l2'-gateway-create --device=GW_DEVICE_UUID or GW_DEVICE_NAME --segmentation_id = VNI GATEWAY-NAME	Create Logical GW to connect Tenant via VTEP-Router to other AZs	Admin
4	Create network/subnet for cross AZ Tenant Router connectivity	neutron network-create TENANT_INTER_AZ_NET neutron subnet-create CIDR TENANT_INTER_AZ_NET	CIDR to use cross AZ Provided by tenant (must be consistent across all AZs)	Tenant
5	Create tenant's peering-router	neutron router-create TENANT_ROUTER		Tenant
6	create port on TENANT_INTER_AZ_NET	neutron port-create --ip_address=IP --name=INTER_AZ_NET_PORT TENANT_INTER_AZ_NET	Create port on Inter-AZ Tenant Network with specific IP (orchestration layer managed)	Tenant
7	Connect Tenant's peering- Router to TENANT_INTER_AZ_NET	neutron router-interface-add TENANT_ROUTER INTER_AZ_NET_PORT	Connect Tenant peering-Router with specific IP. Tenant peering-Router MAC address is MAC of port from (6).	Tenant

8	Update MAC-to-VTEP_IP Table of VTEP-Router	neutron vtep-device-update-remote-mac-table DEV_UUID <list-of-entries>	Update local MAC-to-VTEP_IP Table' in the 'VTEP-Router' with 'Remote Tenant Router' MAC address to 'Remote VTEP Capable Router' IP	Admin
9	Update IP-to-MAC Table	neutron l2-gateway-conn-arp-table GW_UUID <list-of-entries>	Update IP-to-MAC (ARP) Table' in the Bridge connected to the VTEP. This is: remote-peering router ip-address To mac-adderss Also, update the MAC-to-Port table in that bridge.	Admin
10	Create Logical GW Connection	neutron 'l2'-gateway-connection-create GW_UUID TENANT_INTER_AZ_NET	Connect the inter-AZ tenant peering router via the gateway	Tenant
11	Assign tunnel-ip to VTEP-Router	neutron vtep-device-update-tunnel-ip DEV-UUID tunnel_IP	To be used as the VTEP ip-address	Admin

Workflow

** Operation: Cloud owner adds a new AZ (to orchestration layer)

Option 1 - Provider router for Internet and WAN traffic

1. Create VTEP-router
2. Allocate WAN_IP_ADDRESS (floating-ip) to be used as the VTEP ip-address
3. Configure the WAN_IP_ADDRESS on the VTEP-router using vtep-device-management API
4. Create network between Provider router and VTEP-router (PROVIDER_VTEP_NETWORK)

5. Create private subnet for the network between Provider router and VTEP-router (PROVIDER_VTEP_SUBNET)
6. Connect Provider router to PROVIDER_VTEP_NETWORK network by adding an interface on the network
7. Set PROVIDER_VTEP_SUBNET default-gw to be the provider router
8. Create a port to connect VTEP-router to PROVIDER_VTEP_NETWORK network
9. Configure routes to WAN_IP on Provider and VTEP routers

Option 2 - VTEP-Router is an edge-router to the WAN

1. Create VTEP-router
2. Create a network type 'Uplink'
3. Allocate WAN_IP_ADDRESS to be used as the VTEP ip-address
4. Connect VTEP-router to the 'Uplink' network with WAN_IP_ADDRESS
5. Configure the WAN_IP_ADDRESS on the VTEP-router using vtep-device-management API

→ At that point we have a VTEP-router defined as a gateway assigned with a WAN ip-address, connected to the provider-router

**** Operation: Tenant enables cross-site peering**

1. Allocate VNI for the tenant

→ At that point we have a VNI allocated for the tenant

**** Operation: Tenant connects AZ to cross-site peering**

1. Create tenant peering-router
2. Create logical gateway; associate the allocated VNI with the tenant
3. Seed the ARP entry in the bridge connected to the VTEP. This entry is how to get to remote peering-router's mac-address:
remote peering-router's ip-address → remote peering-router's mac-address
4. Create TENANT_INTER_AZ_NET (and subnet)
5. Connect peering-router to TENANT_INTER_AZ_NET with
TENANT_INTER_AZ_IP_ADDRESS
6. Connect TENANT_INTER_AZ_NET to the logical-gateway
7. Synchronize MAC-VTEP-ip table across sites

→ At that point we have:

- A tenant peering-router connected to the VTEP-router (logical-gateway). I.e., there is a stretched L2 segment that connects tenant peering-routers across-sites
- VTEP-router's MAC-VTEP-ip tables are synchronized across all sites

**** Operation: Tenant connects a subnet to the peering-router**

1. Add routes to added subnet in all remote

2. Add routes to all subnets connected to tenant peering-routers in other AZs

Note: Will not be required when routing protocols are supported on tenant routers

→ At that point router peering is complete and connected subnets can communicate across-sites

VTEP device management API

1. Administrator should be able to define GW Device via API. For Router Peering use case, administrator should be able to define GW Device on Virtual Router.
2. Administrator should be able to configure Tunnel endpoint IPs for VTEP Device.
3. Administrator/Orchestration service should be able to configure Remote MAC entries of the VTEP device (i.e. `remote_mac_table` in `HW_VTEP` ovsdb schema)

Above capabilities are proposed to be managed via VTEP device management API neutron extension.

Resource = Gateway Device

Attribute = name; human readable name

Attribute = id; auto-generated unique id of the gateway device

Attribute = device type; can be either Virtual Router or HW VTEP (i.e. supporting L3 or L2)

Attribute = resource_id; virtual router uuid (for HW VTEP, administrator will have to provide management IP and Port)

Attribute = Tunnel IPs (normally there is one), CRU, must contain at least one IP address.

Attribute = Remote MAC Table (list of entries), CRU, can be an empty list.