This is a Q&A document built from real questions asked on Slack. A “Q” followed by an “A” is for short questions followed by the answer, and a number suffix was added for questions with flow, e.g Q1 -> A1 -> Q2 -> A2 etc.

**Q:**

Greeting Team,

I have a question about If metallb+layer2 is used to expose a service, will machines in other network segments be unable to access the service through LB? Is this assumption correct?

**A:**

yes and no.

if you have the right routing to bring the traffic to the node and back to the client, it will work.

if you expect it to work out of the box, it won't. In general, I'd say the natural way to use l2 mode is when the client runs in the same subnet.

**Q:**

Greeting Team,

I have a question about a service has EXTERNAL-IP but no speaker announcing this address.

oc get svc -n jwang-udn-poc4

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

rhel-8-jwang-01-loadbalancer-ssh-service-kbdk LoadBalancer 172.30.63.64 10.120.88.52 22000:32739/TCP 15m



**A:**

try to follow [https://metallb.io/troubleshooting](https://metallb.io/troubleshooting/). one reason might be you don't have endpoints.

**Q:**

MetalLB is currently in *beta*. <https://metallb.universe.tf/concepts/maturity/>. We ship it as non tech preview but that beta note has customer concerned. Does this concern us?

**A:**

it's GA in OCP. The customer should not check upstream for supportability but the Openshift platform docs, where we fully support it.

**Q:**

Hello team, i am using metallb on OCP 4.16, after configuring metalLB, i cannot reach MetalLB LoadBalancer, the router service got their external-ip, but i cannot see any nodes interfaces advertised with Ip pool. Could you know how debug them.

**A:**

<https://metallb.io/troubleshooting/> is a good starter.

**Q:**

Are there any "hardcoded" requirements for the namespace name that metallb operator is installed into? <https://docs.redhat.com/en/documentation/openshift_container_platform/4.18/html/networking/networking-operators#metallb-operator-install> - my presumption is no, but if someone with more direct knowledge of the codebase can chime in and confirm, that would be nice.

Specifically [https://docs.redhat.com/en/documentation/openshift\_container\_platform/4.18/html/networking/networking-operators#nw-metallb-insta[…]-operator-install](https://docs.redhat.com/en/documentation/openshift_container_platform/4.18/html/networking/networking-operators#nw-metallb-installing-operator-cli_metallb-operator-install) - "recommended" is the word I'm asking to get clarified. Customer has a question if they can use a different name for the namespace.

**A:**

Not really. The only constraint is, the CRs should be created in the same namespace as the operator. I’d give it a try, if it doesn’t work, it should be treated as a bug.

**Q:**

Hi everyone, I’m learning RHCL and MetalLB to support a customer. This customer, who is configuring with our help a MetalLB with F5 Loadbalancer and RHCL, and they are asking when creating a new Gateway, will it get its own IP from the MetalLB pool? I believe yes, if they expose the Gateway via a service of type LoadBalancer, it will receive an IP from MetalLB. My understanding is that this allows the Gateway to be externally reachable and participate in the same traffic flow as the primary Envoy ingress they are setting up as well. Could someone please confirm this?

**A:**

I am not familiar with the terminology, but my understanding is that what you call a gateway sits on top of a LoadBalancer service. MetalLB works at cluster level and single instance - you can assign different pools to different namespaces, advertise some pools via BGP and others via L2. See <https://www.redhat.com/en/blog/advanced-metallb-configuration>.

**Q1:**

Hi, I got a question regarding metalLB BGP mode and response IP address.

Does externalTrafficPolicy:local <https://metallb.universe.tf/usage/#local-traffic-policy-1> lead to paket responses being done by the IP of the metalLB or are the packets then still sent via the nodeIPs?

client IP: 10.17.190.41

metalLB IP: 10.18.19.144

OCP node: 10.18.19.22

Customers application is having issues, when the response is sent from the nodes IP instead of the initial targetIP 10.18.19.144 . To my understanding the etp:local should help or not?

**A1:**

externalTrafficPolicy only affects the source IP the node sees, the reply packets should always have the LoadBalancer IP, and this is independent of using Layer2 or BGP. It is actually being implemented by OVN-Kubernetes (the CNI) and is not related to MetalLB.

**Q2**:

AFAIK the metalLB IP is not visible on the nodes. Thus I am having some problems understanding that this response can come from another IP than the nodes.

As this is SIP + RTP streams the SIP is doing the signaling and the RTP is the data.

The reply in paket 10 is coming from the metalLb IP, which is fine as it responds to the signaling.

The data stream is done via the nodes directly.

**A2:**

I understand, for this purpose we have EgressServices, a techpreview feature exactly for this usecase, where you want the pods to use the LoadBalancer Service’s IP for the egress traffic they initiate instead of the node’s IP. See [https://docs.redhat.com/en/documentation/openshift\_container\_platform/4.18/html/networking/load-balancing-with-metallb#metallb-confi[…]eturn-traffic](https://docs.redhat.com/en/documentation/openshift_container_platform/4.18/html/networking/load-balancing-with-metallb#metallb-configure-return-traffic)

**Q:**

Hello Channel, we are deploying RHOSO-18 with Multi DC (3-AZ) and customer looking for deploy the speaker pod only on Worker node, so just wanted to know below CR is good or do we need any additional configuration as well ?

apiVersion: metallb.io/v1beta1

kind: MetalLB

metadata:

name: metallb

namespace: metallb-system

spec:

nodeSelector: 1

node-role.kubernetes.io/worker: ""

speakerTolerations: 2

- key: "Example"

operator: "Exists"

effect: "NoExecute"



**A:**

I think you copied this directly from the documentation. Only the first bit is relevant (nodeSelector) without the number “1”, so that what you need is something similar to:

apiVersion: metallb.io/v1beta1

kind: MetalLB

metadata:

name: metallb

namespace: metallb-system

spec:

nodeSelector:

node-role.kubernetes.io/worker: ""



**Q1:**

Hello team,

I would like to know if we have any supported method to apply the following configuration:

* Is it possible to announce L2 LB service through an OVS bridge assigned to one secondary network? It is not part of OVN-K network, but one additional OVS bridge
* In a secondary network, is it required to assign one IPv4 address to announce one LB svc in this interface (L2 method)? I think yes, but I dont know if we have anything different in this perspective.

**A1:**

There should not be any special configuration, when you have second host network interfaces, metallb speaker as host network pod should be able to do its control-plane stuff (attract traffic in L2 or BGP on a VIP). Once traffic arrives, you might need OCP to be configured with RoutingViaHost (and enabled ip forwarding) to actually allow data traffic in out of the cluster.

2. Not sure, but the secondary interface is not required to have any IP address if you need only return traffic from VIP (ip packet leaving the host have src IP the VIP as a result from a NAT rule). At least for BGP, but I see no reason for it not to work for L2.

**Q2:**

I was able to explore this even more, so my observations:

* ovs-bridge and linux-bridge cant handle the traffic from announced LB services without any IP assigned to the interface. I can see that the arp is replied properly at least in linux-bridge method, but the communcation does not work internally in the node. routingviahost=true and ipForwarding=Global configured
* ovs-bridge even with IP assigned is not able to handle the traffic properly when it is configured as announcing interface

Is it expected? Can we have anything described in the documentation? Would we be able to handle it with a bug or RFE? I would like have the opinion of your guys. I have a cluster with this configuration applied, I can run some tests if needed.

**A2:**

I cannot tell what is expected for L2 in that setup tbh. An RFE sounds like a good idea.

**Q:**

Hi team, quick question: Do we have a short presentation on MetalLB network flow details for customer walkthroughs? Specifically need to cover AddressPools and the requirement for external network routing to LoadBalancer IPs via the BGP peer router. Thanks!

**A:**

Try <https://www.redhat.com/en/blog/advanced-metallb-configuration>.

**Q:**

Hello Team,

Does Red Hat support through updates and security patches if any vulnerabilities or issues or any fault arises with MetalLB ?

**A:**

Yes, <https://access.redhat.com/support/policy/updates/openshift_operators#platform-aligned>.

**Q1:**

Hi Team , I have a query from a customer wanting to use multiple bgp peering, need to know is this supported and if so how is it achievable?

~~~

~~~

There is a requirement that we need to configure MetalLB and stablish the BGP from specific worker-nodes.

**Our current deployment of MetalLB is as following**

In our cluster MetalLB operators and other configurations are already done.

In existing deployment, we have 13 worker node and MetalLB bgp peering working fine with all this worker nodes.

This BGP peering is stablish between MetalLB (running on all 13-worker nodes) and one of the vrf (oam vrf ) router side.

**New requirement for which we need your expert advice.**

We have added 3 more worker-node in the cluster.

From this three-worker node we have to configure the BGP neighbour words the switch in three different vrfs.

Having the bgp neighbour in different vrf is quite possible and tested in our LAB environment with Michael (Michael is the one who implemented this)

So till this stage everything is ok, the main concern is that we have to stablish this three bgp neighbours only between the newly added worker-nodes and routers’s different vrfs.

So please need your expert advice whether it is possible or not, or there is any limitation.

~~~

**A1:**

I assume different vrfs means different vlans on the new nodes. You can use the BGPPeer resource’s node selector field. You define the BGP sessions as usual, and they will be established only from those nodes, see <https://www.redhat.com/en/blog/advanced-metallb-configuration> and the docs.

**Q2:**

so the question is - can I create two BGP peer objects, with same ASN, but different peer addresses & I will use nodeSelectors so that certain workers can peer to one peer address and certain to the other peeraddress.

apiVersion: metallb.io/v1beta1

kind: BGPPeer

metadata:

name: BGP-A

namespace: metallb-system

spec:

peerAddress: 192.168.133.1

peerASN: 64521

myASN: 64520

nodeSelector:

matchLabels:

metallb.io/speaker: "vrf-a"

And

apiVersion: metallb.io/v1beta1

kind: BGPPeer

metadata:

name: BGP-A

namespace: metallb-system

spec:

peerAddress: 10.10.133.1

peerASN: 64522

myASN: 64520

nodeSelector:

matchLabels:

metallb.io/speaker: "vrf-b"



and we will label certain worker nodes `[metallb.io/speaker](http://metallb.io/speaker): “vrf-a”` and `[metallb.io/speaker](http://metallb.io/speaker): “vrf-b”`?

**A2:**

Yes, this is what they are meant for.

**Q:**

Hello folks, this is a tricky case causing an escalation with a disgruntled customer. there is a scenario we have where we are running 3 replicas of the KAS (kube-api server) of a hosted control plane on 3 worker nodes of the management cluster. The kube-apiserver service is exposed via metal LB load balancer operating in Layer-2 mode.

Now, the customer noticed in failure scenarios, the speaker would route the request to a different kube-apiserver (with preference to the colocated KAS. The externalTrafficPolicy is auto provioned so it is set to Cluster, so I am expecting the speaker to be able to route the request to a different working KAS as the externalTrafficPolicy is set to cluster and not Local.

The second thing we noticed is a problem with the ACI fabric causing the MAC addresses not to be updated correctly. For example if a node reboot then the old active MAC associated to an IP is still announced instead of using the new one.

So there are two questions here:

* I feel like the MAC flapping with ACI fabric is not a good sign as we are operating in Layer-2 here.
* Does the L2 speaker has some sort of built in preference to route traffic to its adjacent/ colocated EndPoint of a given service i.e. the kube-apiserver in this case running on the same node?

**A:**

First, it's not the speaker that redirects the traffic with ETP=cluster, but it's OVN-kubernetes (the CNI). For your questions:

1 - MetalLB is stateless. Receives an arp request, sends out an arp reply with the mac of the interface it got it. When a new election happens, a gratuitous arp is sent out to realign the new leader (ie because a node dies). It could be that they have some anti spoofing mechanism. In general the speaker log should tell if it receives an arp request / sends a GARP.

2 - not at all, as I wrote above the speaker only attracts traffic to the node (any in case of etp=cluster, or one with at least an active endpoint in case of etp=local). Once the traffic is on the node, OVN-kubernetes is the datapath.

**Q:**

Hi team. Is it possible to migrate the IP assigned by MetalLB for apiserver service from L2 mode to BGP for the HCP cluster?

**A:**

This in general and leaving any hcp considerations aside: that should be transparent. Given a service, you can define how you want to advertise it by adding a BGP advertisement / l2 advertisement. The BGP one will make the svc being advertised via BGP, and you can do both at the same time. If the ask is, given a service can I switch to advertise to bgp from l2 - then the answer is yes.

**Q1:**

Hi team. When using MetalLB with Pool and L2 advertisement on an OVS interface defined via NNCP. Does the OVS interface require an IP address for the host or can MetalLB assign an IP from its pool regardless?

**A1:**

MetalLB does not assign ips, it replies to arp requests received on interfaces. Not entirely sure if / how it would work in case of ovs.

**Q2:**

We document MetalLB as an on-premise solution when needing Services with type: LoadBalancer and an ExternalIP. How is that not assigning an IP?

**A2:**

it's not assigning the ip to the interfaces, it assigns the ip to the service - sorry I wasn’t accurate. The ExtneralIP part above is not correct. MetalLB serves only for services of type LoadBalancer, and can work with interfaces on the host without an ip - it receives the arp, replies with the mac of the interface it received the arp from.

**Q:**

Hi All! Just for double checking. If using MetalLB with BGP used subnet for MetalLB does not need to be same than where OpenShift hosts are located. Right?

**A:**

Right, the only thing you need to take care is routes for the reply, in case you are getting to the nodes via vlans or via an interface which is not the default gateway one, otherwise you’ll hit rpf.

**Q1:**

Hello MetalLB team,

we have a BM cluster with OCP 4.18.1 and metallb-operator.v4.18.0-202502041632 installed.

recently, requests toward LoadBalancer services provided by MetalLB stopped working when the request is made from outside of the cluster.

when the request is made from any of the cluster nodes (to the same IP:Port), it works.

the controller , speaker and frr pods do not show any particular error.

could you please show us how can we debug this issue?

**A1:**

When the request is made from any of the cluster nodes MetalLB is not involved,only OVN-kubernetes. <https://metallb.io/troubleshooting/> is a good starting point.

**Q2:**

if the metric metallb\_k8s\_client\_config\_stale\_bool does not show any datapoints, it means the config is valid?

**A2:**

yeah it should, if an invalid config was the cause you would have seen errors. Since you’re saying “it stopped working” I’m assuming the config is legit.

**Q3:**

ok i found the problem.

configuration issue, there was an outdated nodeSelector on the L2Advertisement

issue resolved. Thanks

**A3:** you’re welcome