固定IP实现详细流程和测试重点

1. 特性说明
2. 支持多K8S集群。数据带有ClusterID，表明资源的集群归属。
3. 实现Deployment，Job/CronJob的地址分配和释放。Statefulset二期实现。
4. 支持地址租期管理。时间可配置，到期后地址才真正归还NSP，到期前地址可复用。
5. 服务多实例，并行，容损, 高可用。接口服务无状态，回收逻辑做分区。
6. 运维、状态、监控可视化，预留接口，后续可实现。
7. Deployment
8. 创建

AdmissionWebHook截获deployment创建请求，解析出网络参数（网络域ID，子网ID，子网网关地址，子网CIDR，集群ID）。向IPResMgrSrv发起地址资源创建请求。

IPResMgrSrv收到请求后，依据网络参数向NSP请求副本数量的地址资源（IP、Mac、PortID），将资源写入表tbl\_K8SResourceIPBind，每个资源是一条记录，状态为未绑定，落表完毕给AdmissionWebHook返回结果：地址池创建成功或者否则失败。

问题：

1. 地址资源怎么获取，失败回滚，怎么做？

副本数量很多时，会分批获取地址，例如每批20个，一旦后续获取发生错误，之前获取的的地址资源做defer回滚操作。

1. 前端要过滤用户IP数是否满足副本数量。
2. 删除

AdmissionWebHook截获消息后，通知IPResMgrSrv释放deployment对应的地址资源。

IPResMgrSrv在回收记录表tbl\_K8SResourceIPRecycle插入一条记录，记录实际要回收时间（在配置文件中配置了租期的时间，可动态reload），同时在服务的定时器队列中（内存中）插入一条记录。如果服务重启，会从数据表中恢复属于该服务的定时回收任务。

租期到期，如果到期之前没有相同的deployment创建，IPResMgrSrv将会按tbl\_K8SResourceIPRecycle的记录信息将IP地址释放给NSP。

异常：

1. 如果租期到期，还有未解绑的地址资源怎么办。错误日志、告警（邮件、短信）、输出未解绑pod的信息（node信息，pod信息）、运维介入，待实现。
2. 租期未到时创建

Deployment被删除后，原有的地址资源会按租期时长保留，以便让用户能重复使用。这时会在tbl\_K8SResourceIPRecycle表中插入一条对应记录。

这时如果用户重新创建这个deployment，IPResMgrSrv会判断tbl\_K8SResourceIPRecycle是否有待回收的数据，如果有就会分三种情况来处理。

1. 副本数相同。直接恢复，将回收记录表tbl\_K8SResourceIPRecycle中记录删除。
2. 当前请求的副本数小于原有的副本数。在表tbl\_K8SResourceIPBind表中选择未绑定的IP且数量为差额的记录，将其归还给NSP，同时删除该记录。

将回收记录表tbl\_K8SResourceIPRecycle中记录删除。

1. 当前请求的副本数大于原有的副本数。计算所需增量，向NSP请求地址数据，将多条记录插入tbl\_K8SResourceIPBind。

将tbl\_K8SResourceIPRecycle中回收记录删除。

问题：

1. 这是定时器队最小堆中会有一条回收任务，这个任务到期会怎么办？

细节就不过多描述，已经模拟测试。

1. 租期未到时反复的创建和删除。

流程如上。

问题：

1. 会在多个实例的定时器最小堆中有多条回收任务，这些任务到期怎么办？

细节就不过多描述，已经模拟测试。

1. 如果服务重启，这些定时器最小堆中任务怎么恢复？

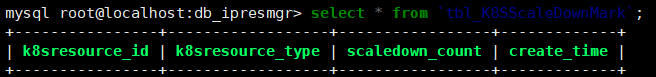
按服务实例id从回收记录表tbl\_K8SResourceIPRecycle中恢复。

1. 在线扩容，kubectl edit deployment xxx，修改replicas，变大。

响应WebHook的ScaleIPPool请求，向NSP获取增量的网络数据，插入新记录到地址资源表tbl\_K8SResourceIPBind，状态为未绑定。

1. 在线缩容，kubectl edit deployment xxx，修改replicas，变小。

响应WebHook的ScaleIPPool请求，在缩容标记表tbl\_K8SScaleDownMark插入记录，标记资源ID，类型，缩容数量，创建时间。IPAM释放IP时会递减缩容数量，这是个并发操作要保证数据表操作的事务化，如果大于0，立即将该记录从表tbl\_K8SResourceIPBind删除，将网络资源归还给NSP。



1. IPAM申请IP

这是个并发操作，多个IPAM会同时向多个IPResMgrSrv实例发起申请地址信息的请求。对mysql事务串行化是实现的重点。

这时tbl\_K8SResourceIPBind中会有副本数量的多条记录，状态都是未绑定。

表操作重点：

1. Transaction的隔离等级设置为LevelSerializable。
2. 使用SELECT … FOR UPDATE语句。
3. 选择后，设置该条记录为绑定状态。

测试重点：

1. IP是否有重复分配，错误情况：多个pod得到相同的IP。正确情况：IP和POD一一对应。
2. 是否有数据库死锁（不能有），这个通过log（deadlock）可以观察，也可以通过以下命令观察：

show status like '%lock%';

show OPEN TABLES where In\_use > 0;

show processlist;

1. 每条记录绑定标志是否为1，绑定PODName是否和kubectl中一致。

优化：

由于使用了LevelSerializable，这相当于是表锁，而不是gap lock，这样会影响申请的高并发。咨询了数据库专家，mysql8.0有skip locked，SELECT … FOR UPDATE SKIP LOCKED，同时保持原有的事务隔离等级repeatable\_read。

IPAM申请IP成功后，tbl\_K8SResourceIPBind表中对应记录绑定标志设置为1，同时设置绑定的podUniqueName、绑定时间。

异常：

1. 相同的deployment，一个pod被delete失败，k8s同时创建一个pod。IP被挂住没有正常释放，而新的POD因为无法获取IP而启动失败。怎么处理？

查询pod（删除中）、node状态，同时判断node机器是否存活，错误日志，告警（邮件，短信），输出未解绑pod的信息（node信息，pod信息），运维介入。待实现。

1. kubelet crash，节点没问题，网络没问题。这个时候pod还是可以服务的，但是5m之后k8s会重新创建pod。同样新pod无法获取IP，怎么处理？

查询pod、node状态（noready），同时判断node机器是否存活。通过agent或者什么方式将这个pod释放掉。待实现。

1. 节点失效。Pod是没有运行的，也没有解绑，5m之后k8s会重新创建pod，同样新pod无法获取IP，怎么处理。

查询pod、node状态（noready），同时判断node机器是否存活。通过运维接口强制解绑。待实现。

1. IPAM释放IP

POD被删除时，IPAM会调用IPResMgrSrv的释放接口，IPResMgrSrv收到请求后，根据podUniqueName去地址资源表tbl\_K8SResourceIPBind中找到对应的记录，将bind标志设置为0。

问题：

1. Bridge网络插件，释放IP时会重复调用4次，代码注释中有明确说明，释放不要返回error。和calico不同。我很难判断释放失败的情况，例如netns删除失败，k8s会反复调用释放接口。如何做出正确的运维告警判断？
2. Job、CronJob。不支持固定IP，地址直接从NSP中获取，解绑直接归还NSP
3. 创建：记录网络参数，用于向NSP申请地址信息所用。
4. 删除：删除网络参数。
5. IPAM申请IP，获取网络参数，向NSP获取地址信息，返回给IPAM。同时落表记录绑定关系。
6. IPAM释放IP，获取网络参数，向NSP归还地址，删除记录，释放绑定关系。
7. 系统管理，做到系统可运维可管理。

可视化：展示资源使用，IP分配使用的全景。

可视化告警：告警的分类、展示、统计、追踪。

可视化运维：针对告警情况，人工主动解绑、主动释放。

1. 相关资料

Kubelet状态更新机制。<https://www.qikqiak.com/post/kubelet-sync-node-status/>