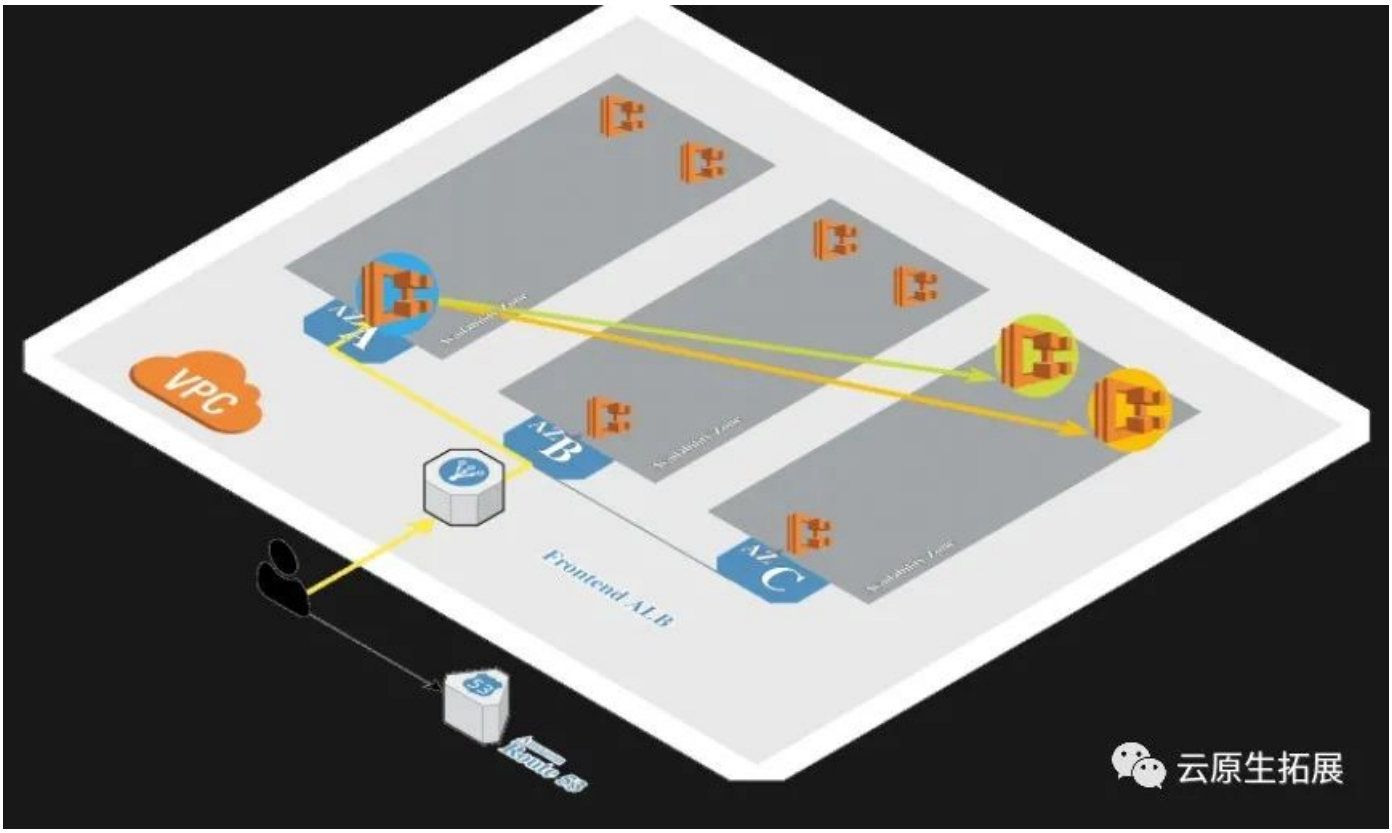
48Kubernetes 系列（四十五）Kubernetes 中的南北通信 — 客户端如何与集群内的服务通信？

Kubernetes 系列（四十五）Kubernetes 中的南北通信 — 客户端如何与集群内的服务通信?

从Pod ip 到云原生负载均衡器的逐步解决问题的方法

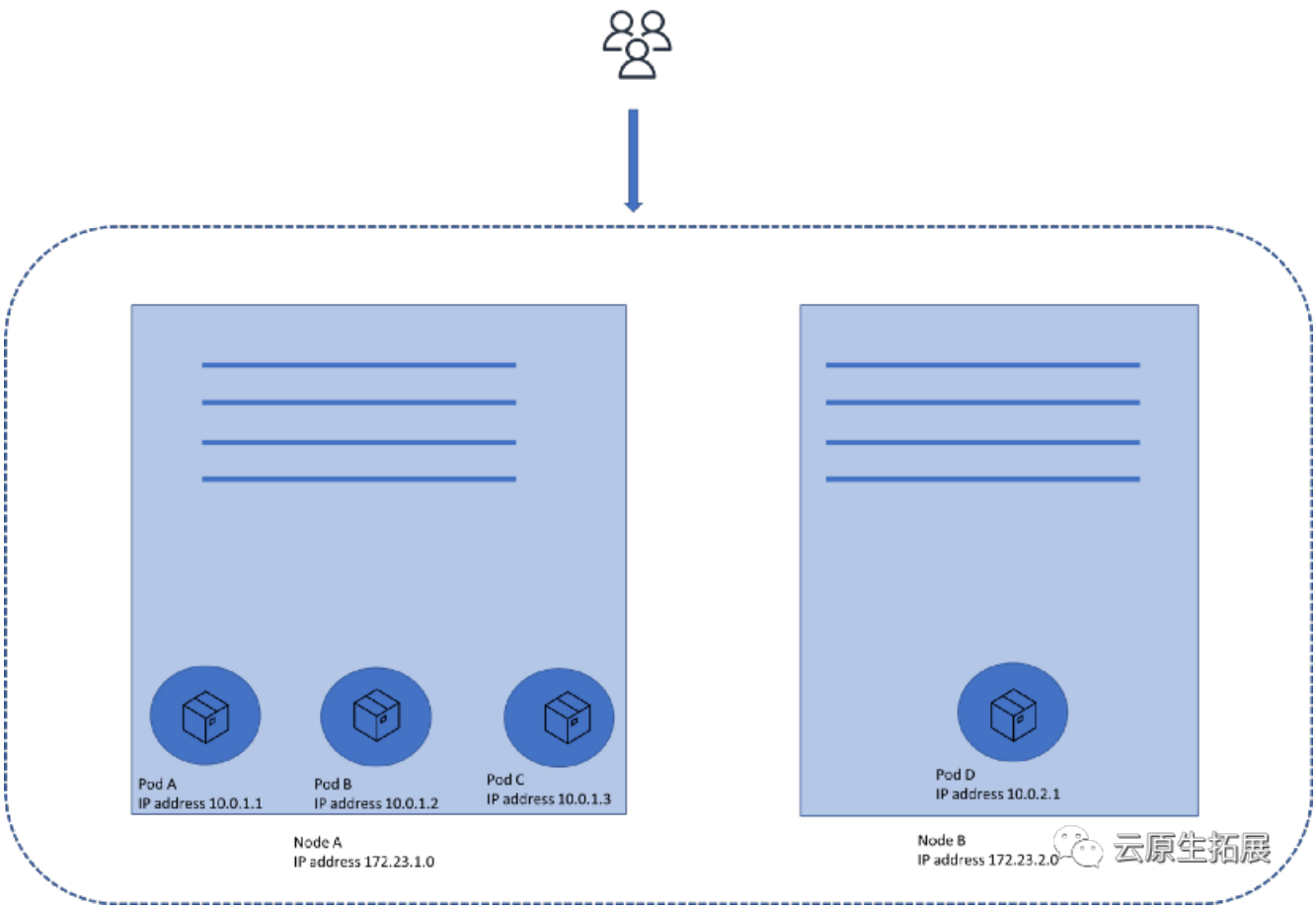


在开始交流之前，先快速澄清一下Kubernetes中使用的一些基本术语:

Pod - Kubernetes中最小的实体。每个 Pod 都有一个IP地址。为简单起见，可以将它们看作是运行应用程序的容器的包装器(就像在docker容器中运行的应用程序)。一个 Pod 可能有多个容器;附加的容器充当主应用程序容器的助手。

Nodes - 物理或虚拟的服务器/机器，在这些服务器/机器上部署pods并形成集群。

现在，假设这个pod正在运行一个web应用程序，两个节点和四个pod为它提供服务。



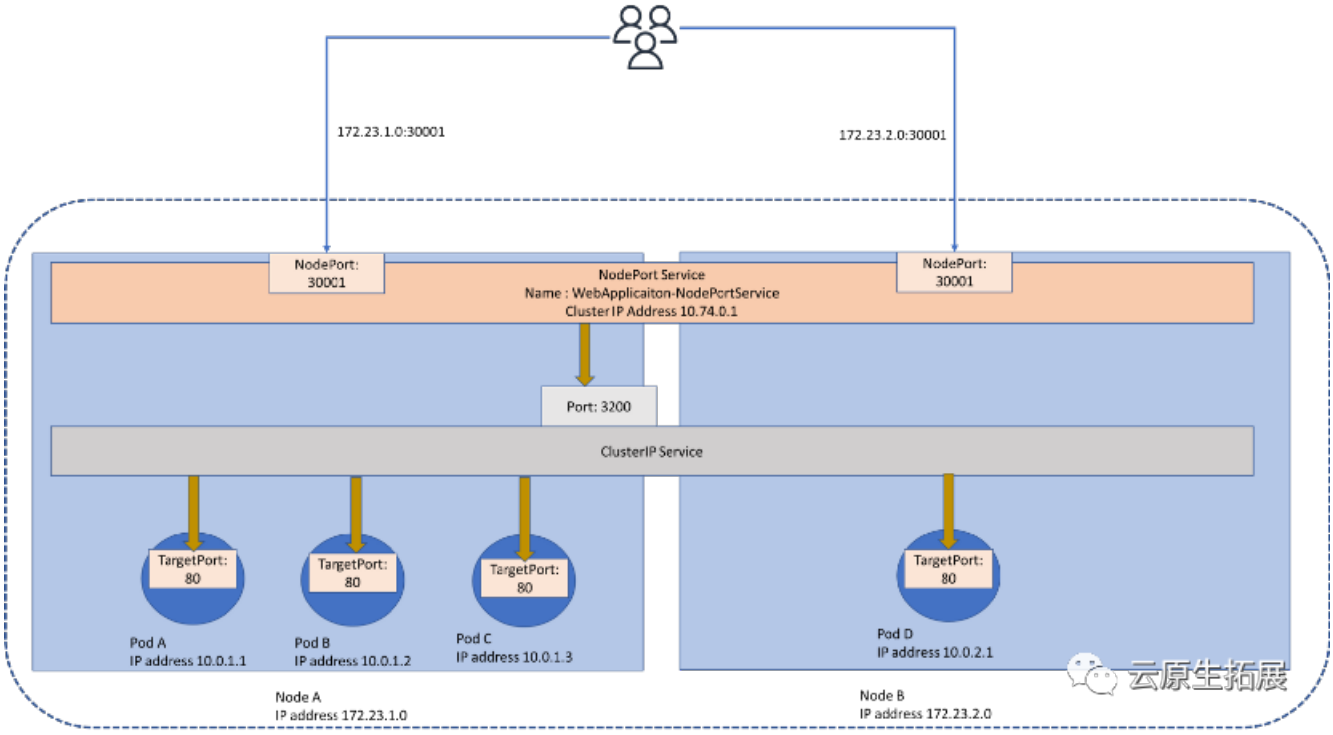
如果用户以某种方式获得了 pod 的公共IP地址，那么客户机是否可以从外部访问web应用程序?这些Pod 位于节点内部的单独网络上。其次，pod是不稳定的，每次创建一个pod，它都会获得一个新的IP地址。

我们正在讨论示例中的四个pod，那么我们要公开谁的IP地址呢?或者我们向外界公开四个IP地址，让客户端决定它想在哪个IP地址上碰碰运气?

1. NodePort 类型的 Service

我们需要在 Pod 的上面加一层。这一层称为NodePort Service，因为它在节点上创建一个端口来访问pods。

NodePort Service在集群中有自己的IP地址，它实际上是一组请求转发规则。它接受来自外部世界的请求，并将其传播到分布在一个或多个节点上的各个 Pod 中。



可以通过使用节点的IP地址和节点端口服务的配置端口来访问Pods。

它还负责使用随机算法和会话关联将流量分配到 Pod。在内部，NodePort服务使用另一个组件与pod通信，也就是ClusterIP服务。

所有问题都解决了吗?不完全是，还有三个问题:

从外部使用 Ip 地址访问 NodePort Service

使用节点的IP地址' <节点>的IP- address:< NodePort服务>的端口'。因此，如果节点是永久性的，我们可以给出两个节点的IP地址，客户机可以使用其中任何一个节点。第一个问题是，我们为它们提供了多个到达服务器的地址。

第二个问题是弹性和规模。因此，节点也是易变的，可能会停机(通常，可以通过使用对外公开的静态IP地址来克服这一限制。以及在伸缩事件中调用脚本，将此IP地址分配给集群中的一个节点)。

端口范围有限制

对于 NodePort Service，端口只能配置在30000 - 32,767的范围内。

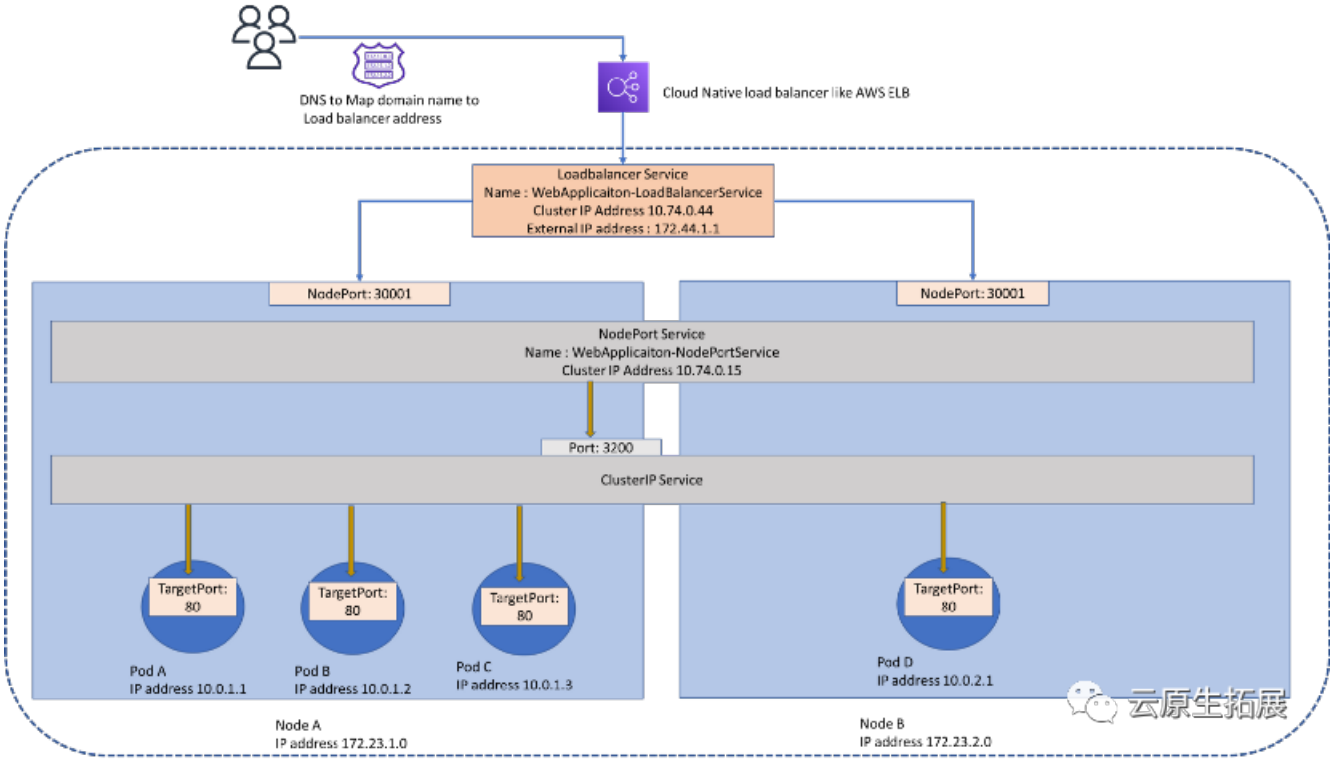
安全问题

由于工作节点的端口是直接对外开放的，因此产生了许多安全问题。

由于上述限制，在开发或测试环境中可能需要NodePort服务。当定制环境/集成中需要一些自由度时，它可能也很有用。让我们考虑一下其他选择:

2. Load Balancer 类型 Service

在NodePort Service之上的附加层可能会有所帮助。当集群部署在公共云(如AWS或谷歌)中时，我们可以使用Kubernetes提供的负载均衡器服务，它集成了AWS ELB等云本地负载均衡器。



用户连接到外部负载均衡器，将流量导向Pod。

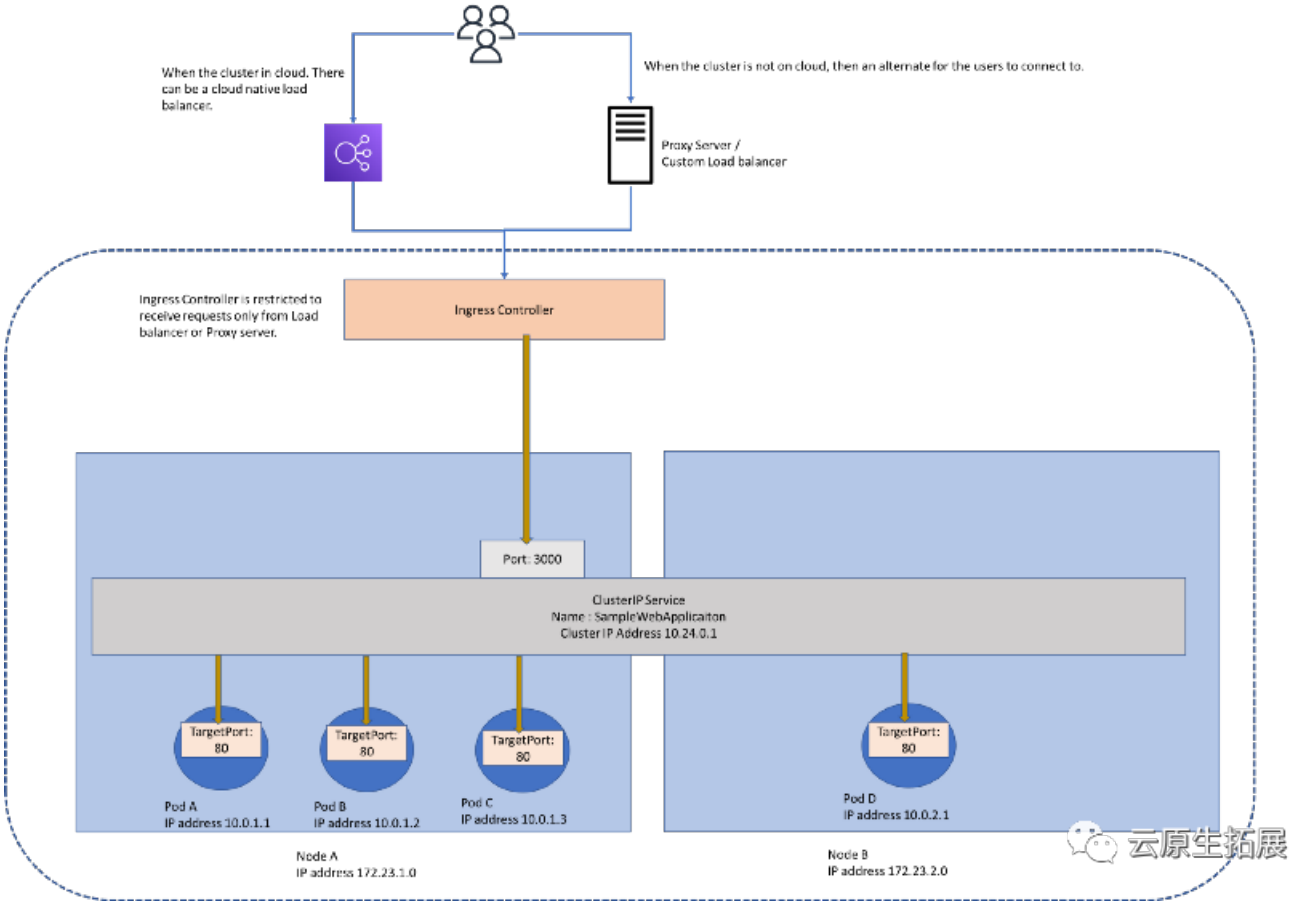
负载均衡器服务有一个集群IP地址和一个外部IP地址。负载均衡器服务是NodePort服务的抽象(NodePort服务是ClusterIP服务的抽象)。因此创建了NodePort Service，但是这些端口只能被云本地负载均衡访问——不能直接访问外部世界。

可能存在到负载均衡器的DNS路由，用户可以使用域名访问服务。还可以通过访问负载均衡器直接访问该服务。外部负载均衡器控制流量如何分配到 Pod。

作为旁注，根据负载均衡器的类型和云平台，可以指向节点/实例(并在节点级别有一个跳)，也可以直接获得pod本身的IP地址(避免跳，使其更高效)。例如，可以在AWS的网络负载均衡器中实现。

3. Ingress

如果集群不在云中，或者它在云中，但仅用于处理HTTP通信，则可以使用Kubernetes Ingress而不是负载均衡器。Ingress 是第7层(HTTP)抽象，它为传入请求指定路由规则。路由/重定向规则由另一个称为Ingress控制器的组件实现。



Ingress controller

Ingress 控制器充当集群内的反向代理/入口点。根据路由规则，来自客户机的任何请求通过负载均衡器到达Ingress控制器，在那里它被重定向到pods。不需要NodePort Service。

另外，值得强调的是，根据服务和控制器实现，还有其他可能的Ingress设置。

总结

Kubernetes使用服务来促进通信。我们研究了NodePort 类型服务、Load Balancer 类型服务，并触及了 ClusterIP 类型服务。我将在讨论集群通信的下一篇文章中重新讨论它。

Ingress是Kubernetes中的另一个对象，它可以在集群之外公开HTTP服务。提供路由、TLS终止和负载均衡功能。

此外，Kubernetes与云提供商集成得非常好，可以简化集群设置。

欢迎关注我的公众号“ **云原生拓展** ”，原创技术文章第一时间推送。