解决方案实践

灵雀云全栈云原生开放平台解决方案实践

文档版本 1.0

发布日期 2024-03-18

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：
https://www.huawei.com/cn/psirt/vul-response-process
如企业客户须获取漏洞信息，请参见如下网址：
https://securitybulletin.huawei.com/enterprise/cn/security-advisory

目 录

1 方案概述

应用场景

传统企业在今天面临着新兴业务模式的剧烈冲击，同质化的竞争手段已无法让企业在愈演愈烈的竞争中脱颖而出，包括金融、能源、制造、汽车以及政府机构在内的传统企业，纷纷致力于数字化转型。通过数字化转型，企业能够快速感知用户的需求并做出调整，加速产品迭代更新，不断地提升用户体验和满意度，从而获得或提高市场差异化竞争优势。企业数字化转型中面临诸多痛点：

- 管理缺支撑，技术依赖外包，基础设施烟囱式管理成本高；多云等复杂基础设施场景管理难
- 业务开发要求资源快速供给满足生产级要求，自研体系不完善，业务要求弹性架构
- K8s运维变更难度大、易出错，业务故障处理周期长，资源利用率低

云原生解决方案通过使用容器、Kubernetes、微服务等这些先进技术，能够大幅加快软件的开发迭代速度，提升应用架构敏捷度，提高IT资源的弹性和可用性，帮助企业客户加速实现数字化转型。

方案架构

灵雀云企业级全栈云原生开放平台（以下简称“ACP全栈云平台”）打破传统以容器平台、DevOps 和微服务为技术框架理念的“云原生三驾马车”固定框架，基于以Kubernetes为核心的云原生技术，助力企业快速打造新一代ACP全栈云平台，ACP全栈云平台涵盖云原生基础设施、云原生应用管理、云原生DevOps、统一运维、统一治理等模块。ACP全栈云平台采用容器原生架构，以K8s为底座和控制平面，支持一键部署、自动运维、持续升级，并具备开放、灵活、可扩展等特性。

ACP全栈云平台屏蔽基础设施的差异性，天然适合做混合云的控制面板，支持编排各种不同的云环境和基础设施，推动以应用为中心的混合云到来。ACP全栈云平台以应用为中心，覆盖应用全生命周期，基础设施、应用管理、DevOps工具等模块作为平台组件，一体全栈、开箱即用，快速搭建。

图 1-1 业务架构

方案优势

- **混合云和多云管理**

  在任何环境中管理任何 K8s 发行版，大规模自动化 Kubernetes 集群生命周期管理，确保所有环境的配置、安全性和合规性保持一致，在所有环境中提供统一的操作和一致的体验。

- **开发者自服务**

  完全可定制的应用程序和基础架构抽象，拥抱基础设施即代码和 GitOps 最佳实践，抽离出基础设施的复杂性，减轻认知负担，内置和可定制的内部开发人员门户，提供良好的开发人员体验。

- **快速搭建PaaS容器平台**

  为各领域更多的企业客户快速搭建PaaS容器平台将现有的基础设施一键升级成新一代的容器云平台，管理容器的全生命周期。

- **持续扩展、集成Kubernetes生态工具的能力**

  平台深度对接Kubernetes，以Kubernetes原生架构作为开发框架，可兼容或集成越来越多的Kubernetes生态工具、系统、插件和解决方案，将源源不断的为使用平台的企业提供前沿且竞争力强的技术，从而保证企业的核心竞争力。

- **自动化的容器调度**

  自动创建Kubernetes资源对象。建立贴合业务需求的应用模型以及应用创建流程，将Kubernetes 资源的创建过程与业务流程很好地结合。

- **将安全融入DevOps**

  DevSecOps，在应用的整个生命周期内确保安全性。实现安全防护自动化，以保护整体环境和数据；同时实现持续集成/持续交付流程，并确保容器中应用的安全性。实现用户身份和访问控制功能的集中化。支持集成适用于容器的安全性扫描程序。

- **持续提升IT运营能力**

  以质量和安全为基础支撑保障，覆盖从需求到部署上线的软件全生命周期管理。将线下IT生产过程转变为线上高度自动化、可视化的IT生产流水线，提升产品研发效率，快速响应业务需求，持续提升IT运营能力。

2 资源和成本规划

表 2-1 资源和成本规划

云服务类型	用途	区域	规格	数量	计费模式	计费周期	计费周期单位	参考价格
弹性云服务器	Global集群，承载ACP平台所有功能 (平台组件，ui界面，kubernetes管理节点资源，日志，监控等)	亚太-曼谷	规格：X86计算 | 通用计算增强型 | c6s.2xlarge.2 | 8核 | 16GB 镜像：CentOS | CentOS 7.9 64bit 系统盘：通用型SSD | 150GB 数据盘1：通用型SSD | 500GB 数据盘2：通用型SSD | 100GB	3	包周期	1	年	¥54,606.21
弹性云服务器	客户的业务服务集群master (kubernetes管理节点资源，日志，监控等)	亚太-曼谷	规格：X86计算 | 通用计算增强型 | c6s.xlarge.2 | 4核 | 8GB 镜像：CentOS | CentOS 7.9 64bit 系统盘：通用型SSD | 100GB 数据盘1：通用型SSD | 100GB	3	包周期	1	年	¥23,077.95
弹性云服务器	客户的业务服务集群监控节点prometheus	亚太-曼谷	规格：X86计算 | 通用计算增强型 | c6s.xlarge.2 | 4核 | 8GB 镜像：CentOS | CentOS 7.9 64bit 系统盘：通用型SSD | 100GB 数据盘1：通用型SSD | 100GB	1	包周期	1	年	¥7,692.65

云服务类型	用途	区域	规格	数量	计费模式	计费周期	计费周期单位	参考价格
弹性云服务器	客户的业务服务集群slave（此处为单台价格，实际数量视情况而定）	亚太-曼谷	规格：X86计算 | 通用计算增强型 | c6s.2xlarge.2 | 8核 | 16GB 镜像：CentOS | CentOS 7.9 64bit 系统盘：通用型SSD | 100GB 数据盘1：通用型SSD | 100GB	1	包周期	1	年	¥13,785.57
虚拟IP	kubernetes集群高可用需要一个vip	亚太-曼谷	/	2	包周期	1	年	¥0.00
弹性公网IP	平台访问地址	亚太-曼谷	带宽费用：独享 | 全动态BGP | 按带宽计费 | 5Mbit/s	1	包周期	1	年	¥1,790.20

说明

1. 该价格仅为参考，实际需要以华为云控制台显示为准；
2. 以上为部署ACP平台最低建议配置，请按实际业务情况规划资源，ACP平台价格请咨询客户经理。

表 2-2 硬件配置（实际根据业务压力大小有所调整）

服务器角色	master&slave&global&log&monitor	业务集群master	业务集群prometheus	业务集群slave
服务器数量	3	3*2(两个业务集群)	1*2	10*2
服务器用途	承载平台所有功能	客户的业务服务集群master	客户的业务服务集群监控节点	客户的业务服务集群slave
是否可选	必须	必须	必须	必须
CPU数量	8	4	4	8

服务器角色	master& slave& global& log&monitor	业务集群 master	业务集群 prometheus	业务集群 slave
内存容量	16G	8G	8G	16G
/分区可用空间	150G	50G	50G	50G
/cpaas/data/	500G 日志，单独的块设备	/	/	/
/cpaas/monitoring/	100G 监控	/	50G	/
/var/lib可用空间	50G	50G	50G	50G
/opt可用空间	30G	30G	30G	30G
/var/lib/docker 或 /var/lib/containerd	100G	100G	100G	100G
/var/lib/docker 或 /var/lib/containerd	xfs	xfs	xfs	xfs
单独的块设备（可选）	/	100G topolvm 用	/	50G *2

表 2-3 硬件要求

硬件	具体要求	最低型号或配置
CPU	主频不小于 2.5GHz，在 iaas 层不得超售。如果不满足，cpu 数量需酌情增加。如果是 arm 的 cpu，数量增加1.5 倍，建议增加2倍。	Intel 8255c
内存	在 iaas 层不得超售。	六通道 DDR4
硬盘	单个块设备的iops > 2000 ； 吞吐量 > 200MB/s 。	ssd
GPU	仅对 418.87.00 CUDA Version: 10.1 驱动的 GPU进行了充分测试。	Nvidia

表 2-4 网络资源要求

资源	可选	数量	说明
证书	可选	1	如果不提供证书，部署脚本会自动生成一个证书，但是浏览器访问平台 UI 会提示安全警告，因为证书不是认证机构签发的。
平台访问地址(external IP)	必须	1	域名或 ip 地址，详细介绍请参考 *名词解释* 中"平台访问地址"。
global VIP	必须	1	详细介绍请参考 *名词解释* 中"global VIP"的介绍。
Kubernetes api server VIP	必须	多个	生产环境必须，给高可用的 Kubernetes 集群的 kube-api 使用，每一个高可用的 Kubernetes 集群都需要一个 vip。
ALB VIP	必须	多个	如果客户使用 alb 有高可用需求，这是必须的资源。每个客户业务服务集群的负载均衡器需要一个 VIP（注意，是每个负载均衡器需要一个 vip，不是每个 alb 实例需要一个 vip）。
ASM istio 网关 vip	可选	多个	每个部署了 asm istio 的业务服务集群，如果客户要求 istion 的网关是高可用的，那么就是必须提供一个 vip。
内网 LB	必须	1	生产环境必须，否则无法达到高可用要求。类似 F5 的负载均衡设备，Kubernetes api server vip 配置到这个负载均衡设备上，global vip 也配置到这个负载均衡上。
外网 LB	必须	1	生产环境必须，否则无法达到高可用要求。如果客户没有内外网区别，可以和内网 lb 复用。external address 配置到这个负载均衡设备上。
更多的访问地址	可选	多个	如果想通过external address 之外的更多的 ip 或域名访问 global 平台，请准备好域名和 ip，部署平台的时候在安装页面的高级设置中添加。

表 2-5 网络配置要求

类型	要求说明
网络速率	不低于千兆，建议万兆。如果global 平台和业务服务集群在不同的数据中心内或是混合云，global 与业务服务集群之间网络速率不低于百兆，建议用千兆。如果没有搜集业务服务集群上的服务日志、审计等数据的需求，速度还可酌情降低。
网络时延	不大于2ms。如果global 平台和业务服务集群在不同的数据中心内或是混合云， global 与业务服务集群之间网络的延迟请保证在30ms内，最大不要超过100ms。

类型	要求说明
安全及防火墙	Global平台的服务器之间无防火墙限制。 业务服务集群的服务器之间没有防火墙限制。 业务服务集群和平台之间建议无防火墙，如果有，请参考本章节转发规则，将端口在防火墙上放开。 calico使用 ipinip 协议，如果业务集群使用 calico 插件，不得限制 ipinip 协议。
ip 地址范围	部署平台的服务器，不得使用 172.16-32 网段的 ip，如果已经使用，无法更改，就需要修改每一台服务器上的 docker 的配置，加上 bip 参数，规避该网络段。
协议	如果客户打算使用双栈网络，那么就要求支持 IPv6。
路由	服务器有 default 或指向 0.0.0.0 这个地址的路由。
转发	转发端口全部放开

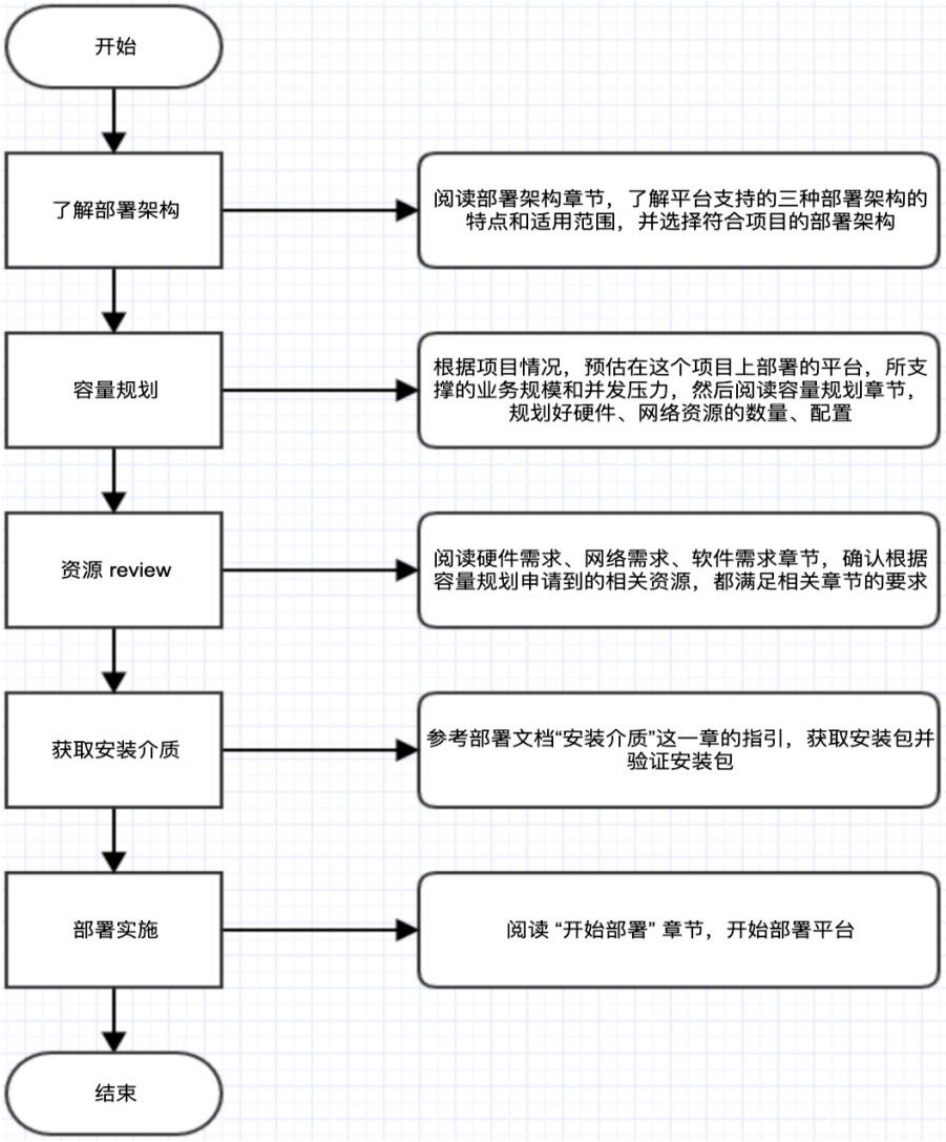
表 2-6 操作系统和内核版本

架构	CPU型号	支持的操作系统	版本信息	备注
ARM	鲲鹏920	银河麒麟	● Kylin Linux Advanced Server Release（麒麟 V10）内核版本： 4.19.90-11.ky10.aarch64 ● Kylin Linux Advanced Server Release（麒麟 V10 SP1）内核版本： 4.19.90-17.ky10.aarch64 ● Kylin Linux Advanced Server Release（麒麟 V10 SP2）内核版本： 4.19.90-24.4.v2101.ky10.aarch64	不支持虚拟化
		openEuler	openEluer 20.03 SP3 内核版本： 4.19.90-2112.8.0.0131.oe1.aarch64 openEluer 22.03 SP3 内核版本： 5.10.0-60.18.0.50.oe2203.aarch64	/
x86	/	银河麒麟	● Kylin Linux Advanced Server Release（麒麟 V10）内核版本： 4.19.90-11.ky10.x86_64 ● Kylin Linux Advanced Server Release（麒麟 V10 SP1）内核版本： 4.19.90-23.8.v2101.ky10.x86_64 ● Kylin Linux Advanced Server Release（麒麟 V10 SP2）内核版本： 4.19.90-24.4.v2101.ky10.x86_64	/

架构	CPU型号	支持的操作系统	版本信息	备注
		openEuler	• 20.03 SP3 内核版本：4.19.90-2112.8.0.0131.oe1.x86_64 • 22.03 SP3 内核版本：5.10.0-60.18.0.50.oe2203.x86_64	/
		Ubuntu	• Ubuntu 20.04 内核版本：5.4.0-124-generic • Ubuntu 22.04 内核版本：5.15.0-56-generic	/
		Red Hat	• Red Hat 7.8 内核版本：3.10.0-1127.el7.x86_64 • Red Hat 8.0 内核版本：4.18.0-80.el8.x86_64 • Red Hat 8.6 内核版本：4.18.0-372.9.1.el8.x86_64	/
		CentOS	CentOS 7.6、7.7、7.8、7.9 内核版本：3.10.0-1160、3.10.0-1127	/

3 实施步骤

图 3-1 ACP 平台部署操作流程



3.1 环境检查

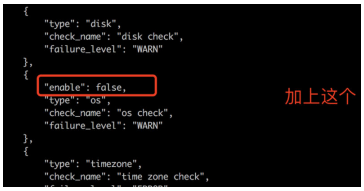
操作系统和内核版本

详见**3.2 安装介质**小节。

说明

1. 最小安装，只需要最基础的软件包。

   UOS 需要手动修改配置文件，开始部署后，修改/cpaas/conf/check_list.json 文件，找到 "type": "os" 这一行，在其上增加 "enable": false, 如下图：



2. xfs 碎片

3. kmem 问题链接：

https://access.redhat.com/solutions/532663

https://github.com/opencontainers/runc/issues/1725

https://github.com/kubernetes/kubernetes/issues/61937

https://github.com/kubernetes/kubernetes/issues/61937#issuecomment-567042968

https://github.com/alauda/kube-ovn/wiki/%E5%87%86%E5%A4%87%E5%B7%A5%E4%BD%9C

grub 启动参数

1. 解决 kmem

   编辑/etc/default/grub（centos\Red Hat\tlinux）或/boot/efi/EFI/kylin/grub.cfg（麒麟）文件，在GRUB_CMDLINE_LINUX= 这一行，在 crashkernel 后增加 cgroup.memory=nokmem 参数并执行grub2-mkconfig -o /boot/grub2/grub.cfg 命令并重启后，能在 /proc/cmdline 中找到增加的，即代表更改成功。

   说明

   https://github.com/opencontainers/runc/issues/1725

   https://github.com/kubernetes/kubernetes/issues/61937

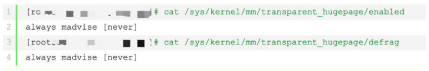
   https://github.com/kubernetes/kubernetes/issues/61937#issuecomment-567042968

2. 关闭大页

   编辑/etc/default/grub（centos\Red Hat\tlinux）或/boot/efi/EFI/kylin/grub.cfg（麒麟）文件，在GRUB_CMDLINE_LINUX加入选项 transparent_hugepage=never，并执行grub2-mkconfig -o /boot/grub2/grub.cfg 然后重启服务器。按说明这一列中的图片所述方式检查。

说明

arm 架构下，部署数据服务的 redis 服务，如果不关闭会严重影响性能



内核模块

网络内核模块要求：

- 如果是 Red Hat，且 version<4.18.0，或者不是 readhat，且version<4.19.0，则需要检查 nf_conntrack_ipv4，如果开启了 IPv6，还需要检查 nf_conntrack_ipv6；
- 如果使用kube-ovn，需要检查：“geneve”，“openvswitch”；
- 以下模块都需要检查：“ip_vs”，“ip_vs_rr”，“ip_vs_wrr”，“ip_vs_sh”。

说明

以 centos 7 为例，以 root 权限执行:

```
cat <<EOF > /etc/modules-load.d/cpaas.conf
iptable_nat
EOF
```

然后重启服务器，执行 lsmod | grep iptable_nat 命令发现有 iptables_nat 模块即待办配置成功。

用户权限

root。

说明

可以接受通过非 root 用户 ssh 登录，再 su - 成 root 用户

sshd 配置

- 各个服务器必须允许 global 集群的各个节点通过 ssh 远程登录；
- /etc/ssh/sshd_config 配置文件中 UseDNS 和 UsePAM 参数需为 no。

说明

如果不是 root 用户，需要配置 /etc/sudoers 文件，做到这个用户执行 sudo 命令，不需要输入密码；

如果 dns 没有设置反向解析，有概率因此造成超时失败

swap

关闭。

说明

如果不满足，系统会有一定几率出现 io 飙升，造成 docker 卡死

防火墙

关闭。

说明

Kubernetes 官方要求

selinux

关闭。

说明

Kubernetes 官方要求

时间同步

所有服务器要求时间必须同步，误差不得超过 10 秒。

说明

docker 和 Kubernetes 官方要求

时区

所有服务器时区必须统一。

说明

设置为 Asia/Shanghai

/etc/sysctl.conf 内核参数

- vm.max_map_count=262144
- net.ipv4.ip_forward=1
- vm.drop_caches=3
- net.ipv4.tcp_tw_recycle=0
- net.ipv4.tcp_mtu_probing=1
- ipv4.conf.all.rp_filter=0
- ipv4.conf.eth0.rp_filter=0
- net.ipv4.conf.default.rp_filter=0
- ipv6.disable=0

说明

vm.max_map_count是 es 运行的服务器的要求

net.ipv4.ip_forward是Kubernetes 要求

关闭

https://www.cnblogs.com/wx170119/p/11995533.html

https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=4396e46187ca5070219b81773c4e65088dac50cc

https://www.kernel.org/doc/Documentation/networking/ip-sysctl.txt

rp_filter 相关配置，是两个不同模式的calico子网内的组件互相通信需要

hostname 格式

获取节点的 hostname，当 hostname 做节点名称时不能重复，最多 36 个字符。并且满足以下要求：

- 只能包含小写字母、数字，以及 '-' 和 '.'
- 不能包含 ".-",".."和"-."
- 须以字母数字开头
- 须以字母数字结尾

说明

https://kubernetes.io/docs/concepts/overview/working-with-objects/names/

/etc/hosts

所有服务器可以通过 hostname 解析成 ip，可以将 localhost 解析成 127.0.0.1

注意

hosts 文件内，不能有重复的 hostname

core 文件

关闭core文件的生成，执行ulimit -c 0 关闭，并且在/etc/profile 文件内增加'ulimit -S -c 0' 这一行

说明

某些情况下，pod 内的进程重启，会在 pod 内创建 core 文件，大量占用磁盘空间，最终 pod 异常退出，甚至拖累宿主机

/etc/resolv.conf 的要求

如果有search 域 ，可能会造成解析 svc 错误，需要删掉这个文件中 search 字段。

/etc/resolv.conf 文件必须存在，并且有 nameserver 的配置项，并且不允许配置 127 开头的地址。

DefaultTasks

执行systemctl show --property=DefaultTasksMax 命令，如果返回的值不是infinity或18446744073709551615 这样的很大的数字，就需要更改

说明

修改/etc/systemd/system.conf 文件，将DefaultTasksMax 改成DefaultTasksMax=infinity。

影响范围：在单点或标准部署架构下，global 平台也当业务集群使用，这个配置会影响客户业务服务的数量，会在客户起了较多业务服务的时候，部分 pod 异常。

apparmor

systemctl stop apparmor.service && systemctl disable apparmor.service;

修改 /etc/default/grub，在GRUB_CMDLINE_LINUX 中增加 apparmor=0。

说明

如果使用 UOS 操作系统，且 runtime 是 containerd 1.6.4或之后的版本，需要禁用 apparmor ，否则部署平台的时候会报错。

软件包&系统工具

- 主机上必须存在的系统工具有：ip, ss, tar, swapoff, modprobe, sysctl, md5sum，scp 和 sftp 二者中必须包含其中一个；
- 欧拉系统必须安装bc；
- 用户部署 topolvm 和 rook 的时，必须安装 lvm2。

移除软件包

麒麟操作系统（ kylin ）默认安装了runc，和平台部署的runc 冲突，在部署前必须移除runc

访问 tmp 目录

账号必须有权限在 /tmp 目录下执行 ls 和 cat 命令。

GPU 设备

使用 GPU 设备时检查设备是否存在。

CPU 核

CPU 核数必须不小于 2。

内存大小

内存不小于 2G。

检查 kubelet 服务

不能存在 /etc/systemd/system/kubelet.service 文件。

默认路由

服务器有 default 路由或指向 0.0.0.0 这个地址的路由。

端口是否被占用

检查以下监听端口是否被占用：

- 所有须检查的端口：10249, 10250, 10256；
- master节点：2379, 2380, 6443, 10249, 10250, 10251, 10252, 10256；
- kube-ovn端口：6641, 6642；
- calico端口：179。

网卡

集群和节点配置的网卡都存在。

硬件架构

节点的硬件架构（X86、ARM）必须与集群的硬件架构相同。

IP 地址

IP 地址必须存在。如果开启了 IPv6 地址，IPv6 地址也必须存在。

节点ip不可为回环ip

- 127.0.0.1
- 0:0:0:0:0:0:0:0或::节点ip不可为组播地址
- 224.0.0.0到239.255.255.255
- FF开头的IPv6地址，节点ip不可为链路本地地址
- 169.254.0.0/16* 地址块
- fe80::/10*地址块：节点ip不可为全0地址；节点ip不可为广播地址
- 255.255.255.255

IP 段

Docker 所需网段 172.16.x.x - 172.32.x.x 的 IP 未被占用。

如果该网段的 IP 已被占用且无法更改，请修改所有节点上的 Docker 的配置文件，增加 bip 参数，避免 Docker 使用被占用 IP。

节点联通性

节点及其 SSH 端口能够正常访问。

节点是否能访问平台

节点能够通过平台地址访问平台。

节点是否能访问平台镜像仓库

节点能够访问平台的镜像仓库。

节点 IP 是否在已配置的任一网段中

节点 IP 不在已配置的任一网段（默认子网网段、容器网段、Service 网段、Join 网段）中。

检查 CIDR

- 默认子网为 underlay 场景下，对于节点 IP 是否在默认子网、Service 网段、Join 网段内不做校验。
- 对于默认子网为非 underlay 场景下需满足：节点 IP（包括 IPv6） 不在已配置的任一网段（默认子网网段、容器网段、Service 网段、Join 网段）中。

访问 Master 端口

检查机器到所有master的各个端口连通性：[6443, 2379, 2380]。

检查 pki 目录

查看目标机器上，/var/lib/kubelet/pki 目录能为空或者不存在。

检查 cri 目录空间

检查指定目录（/var/lib/containerd或/var/lib）当前可用大小。

检查超时

目前检查默认是 110s 左右。

检查不能存在的目录

/var/log/pods 目录不能存在。

/usr/bin 服务

"docker", "containerd", "runc" 这些服务如果存在，必须是在 /usr/bin 下。

3.2 安装介质

下载介质

请使用租户账号访问平台下载安装包和相关文档，或联系服务经理。访问地址：**https://cloud.alauda.cn**

验证介质

说明

可选步骤，可以依据本章内容校验安装包是否安全可信，未被篡改。

步骤1 背景介绍

要了解什么是GPG，就要先了解**PGP**。1991年，程序员**Phil Zimmermann**为了避开政府监视，开发了加密软件PGP。这个软件非常好用，迅速流传开来，成了许多程序员的必备工具。但是，它是商业软件，不能自由使用。所以，自由软件基金会决定，开发一个PGP的替代品，取名为GnuPG。这就是GPG的由来。作为PGP的替代，如今已经有一个开放源代码的类似产品可供使用。GPG（Gnu Privacy Guard），它不包含专利算法，能够无限制的用于商业应用。官方 HOWTO **https://www.gnupg.org/howtos/zh/index.html**。

步骤2 安装 GPG

GPG有两种安装方式。

可以**下载源码**，自己编译安装：

```
./configure
make
make install
```

也可以安装编译好的二进制包：

```
Ubuntu:
sudo apt-get install gnupg
Centos:
yum install gnupg -y
Mac:
brew install gpg
```

步骤3 导入公钥

注意

- 公钥是验证安装介质没有被篡改的可信方法，请从**www.alauda.cn** 的公网下载
- 公钥 MD5 是2eaddfab97d2951a8915f327acb53562 ，请下载后验证，确保不被篡改
- 导入公钥后，执行gpg --list-keys 查看公钥 ID 是不是BB097AE6 ，确保不被篡改

```
curl https://www.alauda.cn/download/verify-key.pub | gpg --import
# 执行上面的命令，会有如下输出
gpg: key BB097AE6: public key "cpaas (Special for packing) <wht@126.com>" imported
```

```
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

步骤4 查看公钥

```
gpg --list-keys
# 执行上面的命令，会输出公钥信息
/root/.gnupg/pubring.gpg
------------------------
pub   4096R/BB097AE6 2020-08-11
uid                  cpaas (Special for packing) <wht@126.com>
sub   4096R/3750351A 2020-08-11
```

步骤5 核对公钥签名

```
gpg --fingerprint BB097AE6
# 执行上面的命令，会输出公钥信息
pub 4096R/BB097AE6 2020-08-11
Key fingerprint = 09EE E7B9 A30C F4B3 5E31 A91B 2704 1C16 BB09 7AE6
uid cpaas (Special for packing) <wht@126.com>
sub 4096R/3750351A 2020-08-11
```

步骤6 签名验证

下载签名文件，参考安装介质，获取签名文件。

```
gpg --verify <签名文件> <安装包>
# 如下为校验正常的输出结果
gpg: Signature made Thu 03 Sep 2020 03:51:35 PM CST using RSA key ID BB097AE6
gpg: Good signature from "cpaas (Special for packing) <wht@126.com>"
gpg --verify finger/cpaas-devops-2.14.0-20200901.tgz.sig   42.50s user 3.08s system 68% cpu 1:06.76 total
# 如果出现如下 warning 提示，请核对公钥的，如果公钥确实与下载链接中给出的一致，即可忽略。
gpg: WARNING: This key is not certified with a trusted signature!
```

步骤7 快速配置命令

使用 root 用户，将解压安装包后 res 目录下的 init.sh 脚本在每个节点上执行一次，即可快速配置操作系统。

⚠ 注意

- 快速配置命令不包括修改 /etc/hosts、升级内核版本、配置 ntp 服务。
- 快速命令无法满足所有软件需求，执行了快速配置脚本并不能保证一定会部署成功，必须按照软件需求章节检查并保证所有要求项都按文档配置成功后，才不会出现因软件配置不符合要求造成的部署失败。

----结束

3.3 开始部署

说明

本节将使用平台默认的镜像仓库部署平台，需从 外部 镜像仓库拉取集群所需的平台组件镜像，请提前搭建好存储平台组件镜像的镜像仓库，具体实施方案请联系技术支持人员。

部署过程

步骤1 将主安装包解压缩到global 平台的第一台master节点的/root/cpaas-install（也可以解压缩到其他目录，要求存储解压缩文件的目录最少有100G 的空间，部署完毕后可以删除该目录）。

步骤2 在第一台master上执行如下命令，解压并进入安装目录：

```
tar -xvf <安装包文件地址，例如：installer-v3.6.0.tar> -C /root/cpaas-install     #解压安装包
cd /root/cpaas-install/installer
```

步骤3 选择global集群网络，执行相应安装命令。

- 使用Kube-OVN Overlay网络部署global集群。
  ```
  bash setup.sh
  ```
- 使用Calico网络部署global集群。
  ```
  bash setup.sh --network-mode calico
  ```

步骤4 根据命令行中的回显，使用浏览器输入访问地址，访问平台部署页面。

说明

等待 minialauda ready 的步骤大约需要 5 分钟。

图 3-2 基础设置

图 3-3 高级设置

表 3-1 参数配置说明

配置	参数	说明	备注
平台设置	Kubernetes版本	选择集群的Kubernetes组件版本。	服务网格依赖 Kubernetes 新版本，选择版本时需要充分考虑功能依赖，可参考产品基线文档获取功能依赖关系。
	运行时组件	选择容器运行时版本，以下为您列举了几个不同运行时版本间的优势点，帮助您快速选型： • Containerd 调用链更短、组件更少、更稳定、占用节点资源更少。 • Docker 1. 可使用 docker in docker； 2. 可在节点上使用 docker build/push/save/load 等命令； 3. 可调用 docker API； 4. 可使用 docker compose 或 docker swarm。	/

配置	参数	说明	备注
	集群地址	global所在Kubernetes集群暴露的apiserver地址。请提前参考 *容量规划* 章节确定部署架构，以下为您简单例举了两种部署模式配置： ● 高可用部署模式配置，以下模式二选一： 1. 使用 keepalived 实现集群地址高可用：开启自建 VIP（虚拟路由器 ID）开关，并设置 IP 地址/域名 为已申请的 VIP，集群使用自建 VIP 并创建成功后，集群的访问地址则为 <VIP>:6443。 2. 使用负载均衡器实现集群高可用：关闭自建 VIP开关，配置 IP 地址/域名 为提前准备的集群外负载均衡器（例如：F5 设备、IaaS 层的负载均衡器或 HAProxy 软件等）的 IP 或域名。 ● 单节点部署模式配置：使用一个节点作为控制节点，也作为计算节点，快速体验平台功能。您可关闭自建 VIP开关，配置 IP地址/域名 为准备的一个节点 IP。 提示：如有在集群创建成功后添加控制节点的计划，为方便后续扩容，建议您在 IP地址/域名框中输入提前准备的集群外的负载均衡的 IP 或域名。	注意： ● 开启自建 VIP 时，keepalived 需要主机网络支持 VRRP（Virtual Router Redundancy Protocol），并且所有节点与 VIP 在同一子网。 ● 如需使用自建 VIP作为集群地址，使集群地址高可用，请提前联系平台的网络管理员或运维人员申请 VIP（Virtual IP）及 VRID（虚拟路由器 ID），且同一子网下 VRID 不允许重复。
	GPU 类型	如果该节点需要使用GPU，请选择相应GPU类型。	请确保该节点中已安装GPU驱动。
	硬件架构	支持X86和ARM。	当前集群的硬件架构标识，选择对应的硬件架构后，在该集群下后续仅支持添加对应硬件架构的节点。

配置	参数	说明	备注
	平台访问地址	平台的访问地址，也是global集群与业务集群交互时使用的地址，默认与集群地址相同，如果平台访问地址有以下需求，可参考配置： ● 有内外网访问需求：则此地址需要填写内网地址。平台其他访问地址可以在本安装页面的高级设置中添加。 ● 有容灾需求：则此地址必须为域名。 ● 使用合法证书：需填写域名并在证书参数中，配置自签证书或上传认证机构颁发的证书。	默认使用HTTPS协议部署平台，如果确认使用不安全的HTTP协议部署平台，可在高级设置中打开开关。
镜像仓库	镜像仓库	存储平台组件镜像的仓库。 ● 平台默认：部署global时将创建内置镜像仓库，平台所有组件镜像将从内置镜像仓库中拉取。 ● 外部：填写外部镜像仓库信息，在容灾环境中必须保证仓库地址在所有容灾节点都可以访问，请提前搭建好存储平台组件镜像的镜像仓库，具体实施方案请联系技术支持人员。	/
容器网络	IPv4/IPv6双栈	IPv4/IPv6双栈可以有效弥补IPv4网络地址资源有限的问题。 提示：开启双栈请确保所有节点正确配置了 IPv6 网络地址，集群创建后将无法还原为 IPv4 单栈。 当您的业务应用涉及到以下场景时，建议您启用双栈： ● 您的应用需要为使用 IPv6 终端的用户提供访问服务。 ● 您需要对使用 IPv6 终端访问应用提供的服务的访问来源进行数据分析处理。 ● 如果您的应用系统与其他系统（例如：数据库系统）、应用系统之间需要使用 IPv6 进行内网访问。	平台global默认使用Kube-OVN Overlay网络，您需确保容器网络和宿主机网络属于不同网段，否则系统部署可能会出现异常。
	子网网段	表示默认子网网段。即Cluster CIDR。	
	Service网段	供类型为ClusterIP的Kubernetes Service使用的IP地址段，不可与默认子网的网段重叠。	

配置	参数	说明	备注
	Join 网段	Kube-OVN Overlay传输方式下，供节点与容器组间通信使用的IP地址段。不可与默认子网、Service网段重叠。	
节点设置	网卡名称	集群网络插件所使用的主机网卡，如果不填写系统将自动获取节点默认路由所对应的网卡。	/
	节点名称	可选节点IP作为节点名称或主机名称作为节点名称。	选择主机名称作为节点名称时，您需确保主机名称在当前集群中唯一。
	Global 集群平台节点隔离	当您既需要 global 集群运行平台组件，又需要该集群部署业务应用时，为避免平台组件与业务应用抢占资源，可开启本功能。 开启后，您需要在添加节点时，设置 平台独占 节点，表示该节点仅部署平台组件，并且平台组件（个别守护进程集除外）不会调度到非 平台独占 节点上。	/
	添加节点	• 控制节点仅支持 1 或 3 个，当控制节点为 3 个时 global 集群为高可用集群。 • 平台独占： 1. 控制节点: 平台独占 开启时，可部署应用强制关闭，不允许部署业务应用。 平台独占 关闭时，可部署应用默认关闭，可灵活调整是否部署业务应用。 2. 计算节点: 平台独占 开启时，可部署应用强制关闭，不允许部署业务应用。 平台独占 关闭时，可部署应用强制开启，允许部署业务应用。 • 添加控制节点或计算节点时，如果开启 GPU 节点，需要手动安装 GPU 驱动和容器运行时。 • 在使用 Kube-OVN Overlay 网络部署平台时，如果指定了节点网卡名称，则该节点将使用设置的节点网卡。	• 请先参考 *容量规划* 章节进行部署架构选择。 • 单击 添加 后，平台将对节点进行可用性检查，如果校验未通过，请根据提示信息调整相应配置并重新添加该节点。

配置	参数	说明	备注
日志监控设置	监控组件类型	推荐您选择Prometheus作为平台监控组件，选择节点时建议您选择非控制节点搭建监控服务。 选择VictoriaMetrics监控组件时，您须配置部署VictoriaMetrics代理实例数，即VMAgent的数量。推荐添加一个，最多支持添加三个。	为确保监控节点的配置，您需要符合本文档硬件需求章节内的要求。可靠性要求高的场景可以部署高可用的监控组件。 监控组件默认以小规模部署，后续需要调整时，可在部署成功后的平台管理中关闭监控组件，选择重新部署，再选择节点部署不同规模的监控组件。
	日志存储节点	在该节点上部署Elasticsearch等组件，建议选择非控制节点进行日志服务的搭建；默认配置的日志服务支持规模有限，如果日志规模较大，请联系相关人员。	日志节点仅支持选择 1 个或 3 个，如果日志组件有更多节点的需求，请参考 *修改 Es 节点数* 小节。
其他设置	平台其他访问地址	可输入多个 IP 或域名。	• 如果填写 IP 地址，您须确保该 IP 地址可以转发到集群地址； • 如果填写域名，请确保已将域名解析至集群地址。
	Pod 数上限	每个节点 Pod 数最大值，默认为 110。 提示：部署架构为单节点时，为确保 Pod 有足够的 IP，可手动调整上限至 255 个。	/
	产品	选择需要部署的产品名称。	/
	扩展参数	不建议您手动设置扩展参数，设置后可能会导致集群不可用，且集群创建后无法修改，如果有相关需求，请联系技术支持人员。	/

上述参数输入完毕后，单击右下角的【 开始部署 】按钮即开始部署进程。

----结束

验证平台

检查 global 平台是否部署成功，在运行 global 组件的 Kubernetes 集群的 master 节点上执行以下命令：

```
# 检查 sentry 部署的 chart 是否成功：执行如下命令查找部署失败的 chart
kubectl get apprelease --all-namespaces
#检查 所有 pod 是否正常：执行如下命令查找失败的 pod
```

```
kubectl get pod --all-namespaces | awk '{if ($4 != "Running" && $4 != "Completed")print}' | awk -F'[/ ]+' '{if ($3 != $4)print}'
```

访问平台

说明

因浏览器的兼容性不同，使用不同的浏览器访问平台时，可能存在平台界面显示错误或功能无法正常使用的情况。兼容浏览器的版本说明如下：

- Google Chrome浏览器支持93 及以上版本；
- Firefox浏览器支持92 及以上版本。

1. 部署安装完毕后，在浏览器界面单击访问按钮，跳转至平台 Portal 页面，同时可下载并查看部署清单。
2. 平台成熟度设置（ Alpha 功能开关设置）：<平台访问地址>/console-platform/feature-gate

删除安装器

在部署平台的主机上执行 docker rm -f minialauda-control-plane，删除安装器，删除后将无法下载部署清单。

说明

您也可不手动删除该容器，等待10分钟后系统将自动删除。

3.4 配置平台

说明

部署完毕之后，平台某些功能或配置可能会不满足项目要求，就需要依据本章内容，对平台进行配置。

修改组件软件配置

说明

在 *容量规划* 这一章节，为满足知道规模/压力下的业务，不仅需要保证硬件配置符合相应章节的要求，还需要依据本章，修改组件的软件配置。

修改日志采集范围及日志、审计、事件等数据保存时间：

- **日志采集范围**

  平台已经支持在 UI 管理页面上修改，请参考用户手册进行修改。
- **日志保存时间**

  平台已经支持在 UI 管理页面上修改，请参考用户手册进行修改。
- **监控数据保存时间**

  平台已经支持在 UI 管理页面上修改，请参考用户手册进行修改。
- **修改组件 limit**

  ```
  # 在第一台 master 节点执行如下命令搜索要修改的组件：
  kubectl get deploy,sts,ds -A | grep apoll # 这条命令会找到 apollo 的资源名，然后执行如下命令修改：
  kubectl edit -n cpaas-system deployment.apps/apollo
  ```

- **修改apollo –es-enablealias 参数**

```
#  修改这个参数为 false 会提高查询日志的速度。参数默认值 false ，设置为 true 就支持 es 中索引命名不符合规范(log-workload-20230208)，但是有 log-workload 别名的场景，一般是对接客户自己的 es ，然后客户自己写 es 的场景
#  在第一台 master 节点执行如下命令：
kubectl edit prdb base
# 在 .spec 下找valuesOverride ，如果没有就增加这个key ，然后增加如下内容：
 valuesOverride:
   ait/chart-alauda-base:
     logging:
       esEnableAliases: false  # 这是布尔型参数，只有 true 和 false 两个值
```

- **修改 Es分片数(ALAUDA_ES_SHARDING)**

  请参考用户手册中，平台中心 > 平台管理 > 集群管理 > 集群 > 插件管理 > 部署日志存储组件 的相关内容。

- **修改 Es 节点数**

  平台已经支持在 UI 管理页面上修改，请以管理员身份登录平台，然后进入 平台管理 > 产品管理 ，单击 操作 ，在下拉菜单中选择 "更新日志存储组件" 进行修改。

  说明

  目前仅支持1节点和3节点的Es。无法修改为更多节点。如果期望使用多于3个节点的Es，目前需要手动修改部分参数。

  1. 在命令行中执行 moduleinfo(kubectl get moduleinfo |grep logcenter | grep <cluster-name>) 并获取 spec.config.components.elasticsearch.nodes 部分。
  2. 将需要添加的节点名称加到下面即可。（ 获取节点名称可以使用命令 kubectl get nodes ）

4 修订记录

表 4-1 修订记录

发布日期	修订记录
2024-03-18	第一次正式发布。