部署文档v2

在所有节点

安装基础服务

curl、net-tools、vim

```
sudo apt install -y curl
sudo apt install -y net-tools
sudo apt install -y vim
```

关闭防火墙

```
systemctl stop firewalld
systemctl disable firewalld
```

关闭selinux

```
sed -i 's/enforcing/disabled/' /etc/selinux/config #
永久
sed -i 's/SELINUX=enforcing/SELINUX=disabled/g'
/etc/selinux/config
setforce 0 #临时
```

关闭swap

```
# 临时关闭
swapoff -a
# 永久关闭
sudo sed -i '/ swap /s/^/#/' /etc/fstab
#重启
```

设置主机名

hostnamectl set-hostname

修改/etc/hosts

将桥接的流量传递到iptables

/etc/sysctl.d/k8s.conf

```
net.bridge.bridge-nf-call-ip6tables=1
net.bridge.bridge-nf-call-iptables=1
```

sysctl --system

时间同步

```
apt install ntpdate
ntpdate time.windows.com
```

修改DNS服务器

手动修改 DNS 配置，使用公共 DNS 服务器（如 Google DNS 或 Cloudflare DNS）：

```
sudo vim /etc/resolv.conf
```

添加以下内容：

```
nameserver 8.8.8.8
nameserver 8.8.4.4
```

服务端

安装sealos

```
sudo curl -o /usr/bin/sealos https://deepflow-ce.oss-cn-beijing.aliyuncs.com/sealos/sealos

sudo chmod +x /usr/bin/sealos
```

部署集群

此处和官网部署文档不一样（需要换源下载）

```
sudo sealos run registry.cn-shanghai.aliyuncs.com/labring/kubernetes:v1.24.0 registry.cn-shanghai.aliyuncs.com/labring/calico:v3.22.1 --masters 192.168.239.145 -p 123456
```

删除主节点污点

```
kubectl taint node node-role.kubernetes.io/master-node-role.kubernetes.io/control-plane- --all
```

安装helm

```
sudo sealos run registry.cn-shanghai.aliyuncs.com/labring/helm:v3.8.2
```

安装LTS版本All-in-One DeepFlow

```
sudo helm repo add deepflow
https://deepflowio.github.io/deepflow

sudo helm repo update deepflow
cat << EOF > values-custom.yaml
global:
  allInOneLocalStorage: true
EOF
sudo helm install deepflow -n deepflow
deepflow/deepflow --version 6.5.012 --create-
namespace -f values-custom.yaml
```

访问 Grafana 页面

```
NODE_PORT=$(sudo kubectl get --namespace deepflow -o
jsonpath="{.spec.ports[0].nodePort}" services
deepflow-grafana)
NODE_IP=$(sudo kubectl get nodes -o jsonpath="
{.items[0].status.addresses[0].address}")
echo -e "Grafana URL: http://$NODE_IP:$NODE_PORT
\nGrafana auth: admin:deepflow"
```

```
root@deepflowserver-virtual-machine:/home/deepflow-server/Desktop# NODE_PORT=$(kubectl get --namespace deepflow
-o jsonpath="{.spec.ports[0].nodePort}" services deepflow-grafana)
NODE_IP=$(kubectl get nodes -o jsonpath="{.items[0].status.addresses[0].address}")
echo -e "Grafana URL: http://$NODE_IP:$NODE_PORT  \nGrafana auth: admin:deepflow"
Grafana URL: http://192.168.239.145:30810
Grafana auth: admin:deepflow
```

即可进入相应网址

下载 deepflow-ctl

```
sudo curl -o /usr/bin/deepflow-ctl https://deepflow-
ce.oss-cn-
beijing.aliyuncs.com/bin/ctl/stable/linux/$(arch |
sed 's|x86_64|amd64|' | sed
's|aarch64|arm64|')/deepflow-ctl
sudo chmod a+x /usr/bin/deepflow-ctl
```

客户端

curl、net-tools、vim

创建 Host Domain

server端上

创建一个专门用于同步服务器的 domain

```
unset DOMAIN_NAME
DOMAIN_NAME="legacy-host1"  # FIXME: domain name

cat << EOF | deepflow-ctl domain create -f -
name: $DOMAIN_NAME
type: agent_sync
EOF
```

创建采集器组

```
unset AGENT_GROUP
AGENT_GROUP="legacy-host1"  # FIXME: domain name

deepflow-ctl agent-group create $AGENT_GROUP
deepflow-ctl agent-group list $AGENT_GROUP # Get
agent-group ID
```

id为g-En3MAZtidP

创建采集器组配置文件 agent-group-config.yaml，指定 vtap_group_id 并开启 platform_enabled 让 deepflow-agent 将服务器的网络信息同步至 deepflow-server

```
vtap_group_id: g-hwplvPDanL # 根据上面的组id修改
platform_enabled: 1
```

创建采集器组配置：

```
deepflow-ctl agent-group-config create -f agent-group-config.yaml
```

部署 DeepFlow Agent

下载 deepflow-agent（deb）

```
curl -O https://deepflow-ce.oss-cn-beijing.aliyuncs.com/deb/agent/stable/linux/$(arch | sed 's|x86_64|amd64|' | sed 's|aarch64|arm64|')/deepflow-agent-deb.zip
unzip deepflow-agent-deb.zip
sudo dpkg -i x86_64/deepflow-agent-1.0*.systemd.deb
```

修改 deepflow-agent 的配置文件 `/etc/deepflow-agent.yaml` ：

```
controller-ips:
  - 192.168.239.145 # server端ip
vtap-group-id-request: 'g-En3MAZtidP' # 根据上面的组id修改
```

启动 deepflow-agent

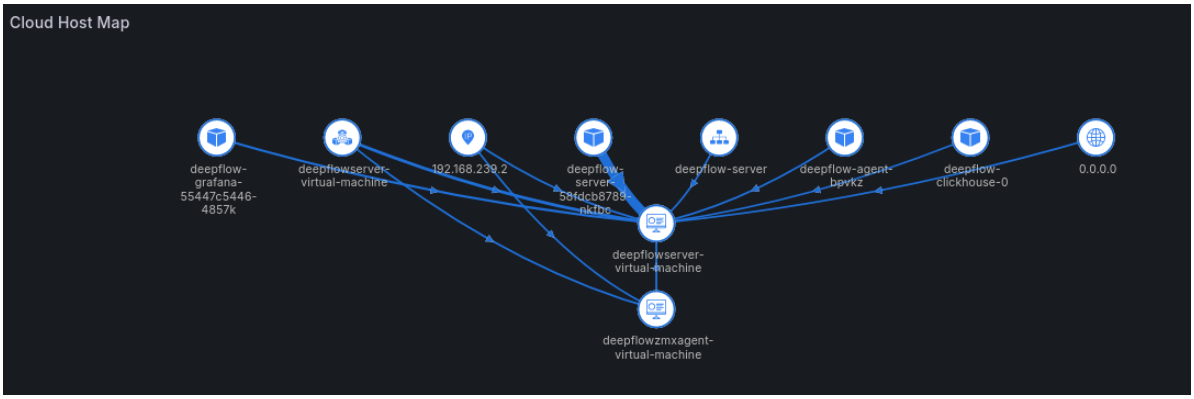
```
systemctl enable deepflow-agent
systemctl restart deepflow-agent
```

在master节点上查看已部署的agent

sudo deepflow-ctl agent list

```
deepflow-server@deepflowserver-virtual-machine:~/Desktop$ sudo deepflow-ctl agent list
ID   NAME                              TYPE       CTRL_IP           CTRL_MAC           STATE    GROUP          EXCEPTIONS   REVISION     UPGRADE_REVISION
1    deepflowserver-virtual-machine-V1 K8S_VM     192.168.239.145   00:0c:29:3b:dd:e3  NORMAL   default                     v6.5 10970
3    deepflowzmxagent-virtual-machine-W2 CHOST_VM 192.168.239.144   00:0c:29:ad:26:6f  NORMAL   legacy-host1
```

在grafana的network-cloud host map中也可以看到相应节点



将不同node加入集群

检查 Kubernetes 主节点状态

master上

kubectl get nodes

获取加入命令

sudo kubeadm token create --print-join-command

输出为

kubeadm join apiserver.cluster.local:6443 --token vxqr1d.t0daq2oatkhab187 --discovery-token-ca-cert-hash sha256:45b39767ba492a4d458674ab335cf375ccc8320f0e977e3d3faf0eab9d7f5c72

从节点上

启用root和允许ssh远程连接

sudo apt install openssh-server -y

sudo systemctl start ssh

修改ssh服务的配置运行root用户登录

sudo vim /etc/ssh/sshd_config

PermitRootLogin yes

刷新ssh服务

sudo service ssh restart

激活root用户，并且设置123456为root用户密码

sudo passwd root

增加node

在第1个master上进行操作，因为它安装了sealos软件。

切换到root用户进行。

sealos add --masters 192.168.239.144

验证集群状态

在主节点上运行以下命令，验证工作节点是否成功加入：

```
sudo kubectl get nodes
```

```
deepflow-server@deepflowserver-virtual-machine:~/Desktop$ sudo kubectl get nodes
[sudo] password for deepflow-server:
NAME                              STATUS   ROLES           AGE   VERSION
deepflowagent-virtual-machine     Ready    control-plane   61s   v1.24.0
deepflowserver-virtual-machine    Ready    control-plane   59m   v1.24.0
```

实际观测

nignx

通过部署demo来模拟实际场景

下载heml

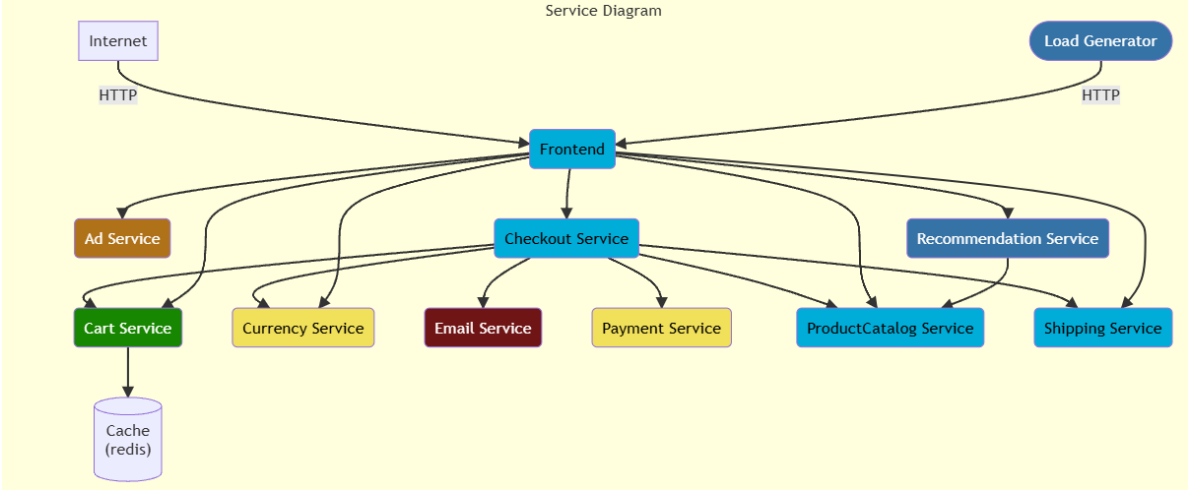
```
wget https://get.helm.sh/helm-v3.8.2-linux-
amd64.tar.gz
tar -zxvf helm-v3.8.2-linux-amd64.tar.gz
mv linux-amd64/helm /usr/local/bin/helm
helm version
```

下载nginx

```
helm repo add stable
https://mirror.azure.cn/kubernetes/charts/
sudo helm repo update
sudo helm install nginx stable/nginx-ingress
```

Open Telemetry -demo

部署一套 Open Telemetry -demo应用，让 DeepFlow Agent 有可采集到的应用数据， Demo 由 Go、C#、Node.js、Python、Java 等语言实现的十多个微服务组成，它的应用架构如下：



---

github仓库

[currency] Rename currencyservice to currency (#1858) · open-telemetry/opentelemetry-demo@e3548c6

官方文档

https://opentelemetry.io/zh/docs/what-is-opentelemetry/

下载部署open-telemetry-demo

```
sudo helm repo add open-telemetry https://open-telemetry.github.io/opentelemetry-helm-charts
sudo helm repo update open-telemetry
sudo helm install my-otel-demo open-telemetry/opentelemetry-demo
```

错误重试指令+重启（eof缺失需要关代理/connect-fused需要重连）

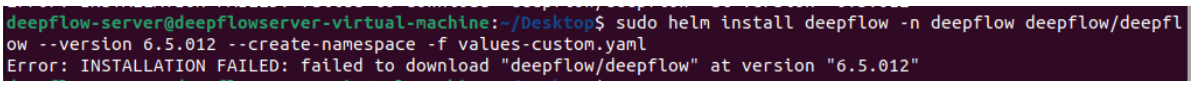
```
sudo helm repo remove open-telemetry
sudo helm repo add open-telemetry https://open-telemetry.github.io/opentelemetry-helm-charts
sudo helm repo update open-telemetry
sudo helm install my-otel-demo open-telemetry/opentelemetry-demo --debug
```

或者手动

```
curl -LO https://github.com/open-telemetry/opentelemetry-helm-charts/releases/download/opentelemetry-demo-0.33.8/opentelemetry-demo-0.33.8.tgz
helm install my-otel-demo ./opentelemetry-demo-0.33.8.tgz
```

问题汇总

问题1：无法下载



使用 Debug 模式

```
helm install deepflow -n deepflow deepflow/deepflow
--version 6.5.012 --create-namespace -f values-
custom.yaml --debug
```

错误的根本原因是 `repo deepflow not found`，这表明 Helm 找不到名为 `deepflow` 的仓库

```
helm pull deepflow/deepflow --version 6.5.012
helm install deepflow ./deepflow-6.5.012.tgz -n
deepflow --create-namespace -f values-custom.yaml
```

解决

1、手动下载并安装

```
helm pull deepflow/deepflow --version 6.5.012
helm install deepflow ./deepflow-6.5.012.tgz -n
deepflow --create-namespace -f values-custom.yaml
```

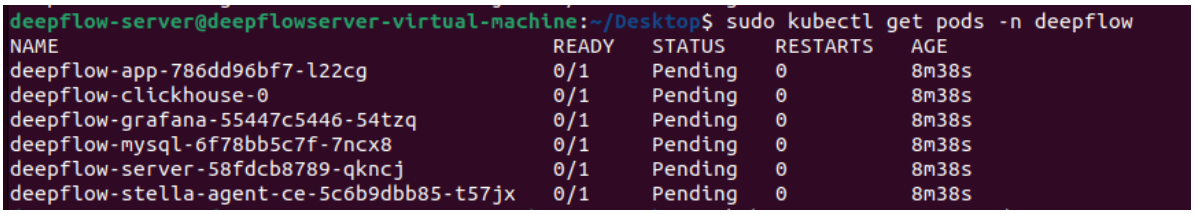
2、确保权限一致

安装LTS版本All-in-One DeepFlow时要保证权限一致，不同权限加入的库不同，需要均在root下运行

```
sudo helm repo add deepflow
https://deepflowio.github.io/deepflow

sudo helm repo update deepflow
cat << EOF > values-custom.yaml
global:
  allInOneLocalStorage: true
EOF
sudo helm install deepflow -n deepflow
deepflow/deepflow --version 6.5.012 --create-
namespace -f values-custom.yaml
```

问题2：各节点一直处于pending 状态



查看 Kubernetes 事件

kubectl get events -n deepflow --sort-by='.metadata.creationTimestamp'

Pod 无法调度的原因是节点存在 `untolerated taint {node.kubernetes.io/not-ready: }`，这表明节点的状态是 "NotReady"，并且没有适当的 `tolerations` 配置来容忍这个污点。

1、查看当前节点的污点

kubectl describe nodes | grep Taints

显示意味着节点因磁盘压力问题被标记为不可调度，导致无法在该节点上调度 Pod，增加磁盘容量

2、重启查看

sudo kubectl get pods -n deepflow

问题3：打不开相应网址、top中无对应进程（初始化失败）

1、查看相应node

sudo kubectl get pods -n deepflow

```
deepflow-server@deepflowserver-virtual-machine:~/Desktop$ sudo kubectl get pods -n deepflow
NAME                                      READY   STATUS              RESTARTS   AGE
deepflow-agent-7jdc4                      0/1     PodInitializing     0          16m
deepflow-app-786dd96bf7-h6944             0/1     ContainerCreating   0          16m
deepflow-clickhouse-0                     0/1     Init:0/1            0          16m
deepflow-grafana-55447c5446-zbl67         0/1     Init:0/2            0          16m
deepflow-mysql-6f78bb5c7f-mqgvt           0/1     PodInitializing     0          16m
deepflow-server-58fdcb8789-n44tp          0/1     ContainerCreating   0          16m
deepflow-stella-agent-ce-5c6b9dbb85-tc5pd 0/1     ContainerCreating   0          16m
```

2、查看 Kubernetes 事件

kubectl get events -n deepflow --sort-by='.metadata.creationTimestamp'

```
3s          Normal   Pulling            pod/deepflow-server-58fdcb8789-nkfbc            Pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflow-server:v6.5"
17s         Normal   Pulling            pod/deepflow-agent-bpvkz                        Pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflow-agent:v6.5"
64s         Normal   Pulling            pod/deepflow-clickhouse-0                       Pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/clickhouse-server:23.8.7.24"
64s         Normal   Pulling            pod/deepflow-stella-agent-ce-5c6b9dbb85-9dp5l   Pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflowio-stella-agent-ce:latest"
64s         Normal   Pulling            pod/deepflow-grafana-55447c5446-4857k           Pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflowio-init-grafana:v6.5"
63s         Normal   Pulling            pod/deepflow-app-786dd96bf7-vjq89               Pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflow-app:v6.5"
63s         Normal   Pulling            pod/deepflow-mysql-6f78bb5c7f-km4zl             Pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/mysql:8.0.31"
33s         Warning  Failed             pod/deepflow-agent-bpvkz                        Error: ErrImagePull
33s         Warning  Failed             pod/deepflow-agent-bpvkz                        Failed to pull image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflow-agent:v6.5": rpc error: code = DeadlineExceed
ed desc = failed to pull and unpack image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflow-agent:v6.5": failed to copy: httpReadSeeker: failed open: failed to do request: Get "http://alireglstry-cn-hongkong
.oss-cn-hongkong.aliyuncs.com/docker/registry/v2/blobs/sha256/1f/1f81e7059338430bc079b2848835e5d2d7bc3ba08857f742524ade1799f76b71/data?Expires=1736705309&OSSAccessKeyId=LTAI4FyN99sZuwGQsaDkPxbv&Signature=cUI3pqlIl
ah3YtUHZ9YgEMxlZ18%!D(MISSING)&x-oss-traffic-limit=41943040": dial tcp 47.79.65.240:80: i/o timeout
32s         Normal   BackOff            pod/deepflow-agent-bpvkz                        Back-off pulling image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflow-agent:v6.5"
32s         Warning  Failed             pod/deepflow-agent-bpvkz                        Error: ImagePullBackOff
10s         Normal   Started            pod/deepflow-server-58fdcb8789-nkfbc            Started container deepflow-server
10s         Normal   Pulled             pod/deepflow-server-58fdcb8789-nkfbc            Successfully pulled image "registry.cn-hongkong.aliyuncs.com/deepflow-ce/deepflow-server:v6.5" in 54.272865675s
10s         Normal   Created            pod/deepflow-server-58fdcb8789-nkfbc            Created container deepflow-server
```

解决：

1、**检查 ConfigMap 是否存在** 确保相关的 ConfigMap（如 `deepflow-grafana`）已经正确创建，并且内容没有问题。可以使用以下命令检查：

```
kubectl get configmap deepflow-grafana -n deepflow
```

如果 ConfigMap 不存在或损坏，可以重新创建。

2、**网络连接问题，关闭代理**

问题4：helm安装otel-demo时

对应issue：[OpenTelemetry Demo Failing to Install using Helm Chart · Issue #1494 · open-telemetry/opentelemetry-helm-charts]

```
Error: INSTALLATION FAILED: template: opentelemetry-demo/templates/component.yaml:12:12: executing "opentele
metry-demo/templates/component.yaml" at <include "otel-demo.deployment" $config>: error calling include: tem
plate: opentelemetry-demo/templates/_objects.tpl:106:16: executing "otel-demo.deployment" at <include "otel-
demo.pod.env" .>: error calling include: template: opentelemetry-demo/templates/_pod.tpl:43:4: executing "ot
el-demo.pod.env" at <tpl (toYaml $allEnvs) .>: error calling tpl: cannot retrieve Template.Basepath from val
ues inside tpl function: - name: OTEL_SERVICE_NAME
```

这个错误提示表明在执行 Helm 安装时，模板渲染过程中的某个步骤发生了错误。错误发生在 `opentelemetry-demo/templates/component.yaml` 文件中的某个 `include` 函数调用。

解决：更新helm到3.14.0

问题5：无法访问6443端口的问题

在虚拟机重启、甚至是回到快照之时，依然会存在无法访问6443端口的问题

The connection to the server apiserver.cluster.local:6443 was refused - did

目前无法查明原因

需要多node 同时回到快照节点