

Общество с ограниченной ответственностью «Зетра»

ОГРН 1237700278155 ИНН 7707488152 КПП 770701001

Юр. адрес: 127473, г. Москва, вн.тер.г. муниципальный округ Тверской, пер 1-й Волконский, д. 15, помещение 1/3

тел.: +7 (909) 909-17-73,

e-mail: Zetrasoft@mail.ru

Инструкция по установке программного обеспечения «Система оркестровки контейнеризированных приложений

ZETRAKUBER»

(«ZETRAKUBER»)

Москва

2023 г.

Процесс установки ZETRAKUBER

При выборе автоматической установки вникать в детали не понадобится, но требуется выделить достаточное количество системных ресурсов, чтобы платформа работала бесперебойно. Например, при небольшом количестве контейнеров и простой взаимосвязи достаточно 1-2 процессорных ядер, 2-4 Гб оперативки и двух виртуальных машин, выполняющих функции Master и Worker Node.

Настройка ОС.

Предварительно необходимо подготовить операционную систему к работе:

- 1) Добавить модули `overlay` `br_netfilter` в список модулей `/etc/modules-load.d/k8s.conf` и загрузить их.
- 2) настроить правила маршрутизации
- 3) отключить `swap`
- 4) установить пакеты `rsync` `wget` `yum-utils`, `socat` `conntrack` `etables` `ipset` `ipvsadm` `iproute-tc`
- 5) установить `containerd`
- 6) установить `cri-tools`
- 7) установить `container network interface (cni)`

Все вышеперечисленные шаги выполняются командой `sudo zetra/before-all`, с помощью скрипта, входящего в дистрибутив

После этого надо настроить `containerd` на загрузку образов с рабочего репозитория. Это может быть репозиторий `zetra` или собственный репозиторий. В случае использования репозитория компании `Zetra`, файл `/etc/containerd/config.toml` должен выглядеть так:

```
version = 2

[plugins]
[plugins."io.containerd.grpc.v1.cri"]
[plugins."io.containerd.grpc.v1.cri".cni]
  bin_dir = "/opt/cni/bin"
  conf_dir = "/etc/cni/net.d"
[plugins."io.containerd.grpc.v1.cri".registry]
[plugins."io.containerd.grpc.v1.cri".registry.mirrors]
  [plugins."io.containerd.grpc.v1.cri".registry.mirrors."registry.zetra.space"]
    endpoint = ["https://registry.zetra.space/"]
[plugins."io.containerd.internal.v1.opt"]
  path = "/var/lib/containerd/opt"
```

Для удобства работы следует настроить заход на Master и Worker Node по ssh и исполнение на этих машинах команд без пароля, на основе инфраструктуры открытых ключей.

Простая установка на Master Node.

Для установки ZETRAKUBER на мастер-ноду необходимо выполнить в каталоге с дистрибутивом ZETRAKUBER команду `zetra/deploy <имя хоста>`

Программа скопирует на целевой компьютер файлы kubeadm, kubelet и kubect1, установит их в папку `/usr/local/bin` и запустит процесс инсталляции ZETRAKUBER. В процессе инсталляции в терминале будут видны сообщения о процессе инсталляции. После успешной инсталляции на экран выведется команда для подключения Worker-nodes вида

Then you can join any number of worker nodes by running the following on each as root:

```
kubeadm join 192.168.250.13:6443 --token 2rjxiu.2uq201a57hsswovv \
--discovery-token-ca-cert-hash sha256:f0de1ba3c0d546072f2f11e67bd312797a385e9ec93a1514a6340384f3dadcd2
```

Здесь `192.168.250.13` - адрес Master Node, `2rjxiu.2uq201a57hsswovv` с и `sha256:f0de1ba3c0d546072f2f11e67bd312797a385e9ec93a1514a6340384f3dadcd2` - токены, сгенерированные для этого кластера. Их надо держать в секрете.

После установки система запросит подтверждение на запись файла конфигурации (`.kube/config`). Необходимо ответить `endthlbntkmyj ('y')`.

Инсталлятор установит сразу cni flannel

Простая установка на Worker Node.

Для установки *ZETRAKUBER* на мастер-ноду необходимо выполнить в каталоге с дистрибутивом *ZETRAKUBER* команду `zetra/deploy-worker <имя хоста с worke noder> <имя хоста с master node><token> <discovery-token-ca-cert-hash>`

Токены необходимо взять из результатов установки Master Node в предыдущем пункте. Если токены были утрачены, можно получить их заново на этой ноде, выполнив `kubeadm token list`

По умолчанию токены действует 24 часа. Если поставлена задача добавить новый узел по завершении периода, новый создается командой:

```
kubeadm token create
```

Проверка работоспособности кластера

Необходимо зайти на Master Node и выполнить команду `kubectl get nodes`

Если все сделано правильно и ОС совместима с *ZETRAKUBER*, то команда выведет в терминал список узлов примерно в таком виде

NAME	STATUS	ROLES	AGE	VERSION
ubuntu1	Ready	control-plane	2d4h	v1.26.0-rc.1.163+e7f8980d775f0d-dirty
ubuntu2	Ready		2d3h	v1.26.0-rc.1.163+e7f8980d775f0d-dirty

Создание кластера с поддержкой шифрования по ГОСТ

В `kubeadm` заложена возможность формировать ключи в соответствии с ГОСТ, выпускать сертификаты и подписывать их этими ключами. В дальнейшем, при работе подсистем *ZETRAKUBER* проверяется вид подписи и в зависимости от этого автоматически выбирается метод проверки сертификата и метод шифрования трафика. Для того, чтобы воспользоваться этой возможностью, необходимо очистить созданный ранее кластер (`kubeadm reset --force`) и запустить `kubeadm` с ключом `--gost` при генерации сертификатов для компонент *ZETRAKUBER*. При этом следует

иметь ввиду, что для генерации и подписания сертификатов для связи с etcd в этой версии необходимы стандартные для tls алгоритмы, поэтому для генерации и подписания сертификатов для etcd необходимо вызывать kubectl без ключа --gost. Для облегчения этого процесса в дистрибутив входит команда recreate-cert. Для создания кластера с поддержкой шифрования по ГОСТ необходимо выполнить на хосте с предварительно установленной Master Node следующие команды

```
kubectl/recreate-cert
```

```
kubectl/install
```

Заключение

После выполнения предыдущих шагов система готова к эксплуатации. Дальнейшие действия пользователя зависят от стоящих задач. Использование ZETRAKUBER аналогично использованию open-source-варианту Kubernetes