

```

# На всех br-srv1,2,3
apt install debian-archive-keyring

# В /etc/apt/sources.list добавить (без решётки):
# deb https://deb.debian.org/debian/ bookworm main contrib non-free non-free-firmware
apt update
apt install snapd
# sudo systemctl enable --now snapd.apparmor
snap install microk8s --classic --channel=1.35
sudo usermod -a -G microk8s $USER # От administrator
mkdir -p ~/.kube
chmod 0700 ~/.kube
su - $USER
sudo service snapd.apparmor start
sudo -i
microk8s status --wait-ready # Должно не вывести ошибку
microk8s kubectl get nodes # Должно вывести ноду

# Добавляем ноды в кластер
# На br-srv1 от root
microk8s add-node
# Ищем строку под Use the '--worker' flag to join a node as a worker not running the
control plane, eg и копируем её, вставляем на br-srv2 и br-srv3

microk8s join <IP BR-SRV1>:25000/<Токен, который был в выводе> --worker
# Ждём successful

# На br-srv1 от root:
microk8s enable metallb
# Запросит диапазон, вводим 192.168.2.1-192.168.2.10
nano address-pool.yaml
# Вставляем и меняем адр
apiVersion: metallb.io/v1beta1
kind: IPAddressPool
metadata:
  name: "metallb-address-pool"
  namespace: metallb-system
spec:
  addresses:
    - <ip br srv-1>/32
    - <ip br srv-2>/32
    - <ip br srv-3>/32

# Сохраняем и применяем
microk8s kubectl apply -f address-pool.yaml
microk8s enable ingress
nano ingress.yaml

# Вставляем:
apiVersion: networking.k8s.io/v1

```

```

kind: Ingress
metadata:
  name: python-app-ingress
  namespace: default
  annotations:
    nginx.ingress.kubernetes.io/force-ssl-redirect: "true"
spec:
  ingressClassName: public
  tls:
    - hosts:
        - portal.rea26.ru
      secretName: app-tls
  rules:
    - host: portal.rea26.ru
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: python-app
                port:
                  number: 80

```

Сохраняем и применяем:

```
microk8s kubectl apply -f ingress.yaml
```

Для проверки:

```
microk8s kubectl get ingress
```

Пример вывода, если всё ок:

#NAME	CLASS	HOSTS	ADDRESS	PORTS	AGE
#python-app-ingress	public	portal.rea26.ru	192.168.2.0	80, 443	2m43s

```
nano registry-ingress.yaml
```

вставить:

```

apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: registry-ingress
  namespace: kube-system
  annotations:
    nginx.ingress.kubernetes.io/force-ssl-redirect: "true"
spec:
  ingressClassName: public
  tls:
    - hosts:
        - registry.rea26.skills
      secretName: secret
  rules:
    - host: registry.rea26.skills
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:

```

```
service:
  name: registry
  port:
    number: 5000

# Применить
microk8s kubectl apply -f registry-ingress.yaml

microk8s enable registry
```

Почта

```
sudo apt install exim4-daemon-heavy dovecot-imapd dovecot-gssapi\
nano /etc/dovecot/conf.d/10-master.conf
# В секцию ServerAuth добавить
unix_listener auth-client {
  mode = 0600
  user = Debian-exim
}
sudo dpkg-reconfigure exim4-config
# Доставка только локальной почты без доступа к сети
# почтовое имя - rea26.skills
# ip адреса для входящих - оставить пустым
# другие места - rea26.skills
# релейная передача почты - 192.168.1.0/24
# сокращать dns - нет
# maildir
# Разделить - да
sudo chown -R Debian-exim:Debian-exim /var/spool/exim4/
```

Осмысленно доделать по ссылке

<https://wiki.astralinux.ru/pages/viewpage.action?pageId=94175339>