

Лабораторная работа №5

Задание 1.

Скрины:

```
UW PICO 5.09
apiVersion: apps/v1
kind: Deployment
metadata:
  name: web-app
  labels:
    app: web-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: web-app
  template:
    metadata:
      labels:
        app: web-app
        environment: test
    spec:
      containers:
        - name: nginx-container
          image: nginx:1.25
          ports:
            - containerPort: 80
          env:
            - name: NGINX_PORT
              value: "80"
```

```
UW PICO 5.09
apiVersion: v1
kind: Service
metadata:
  name: web-service
  labels:
    service-type: internal
spec:
  type: ClusterIP
  selector:
    app: web-app
  ports:
    - name: http
      protocol: TCP
      port: 80
      targetPort: 80
```

```
ptashko@MacBook-Air--Artem K8S % kubectl get service web-service
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
web-service   ClusterIP   10.109.192.49   <none>       80/TCP      2m55s
ptashko@MacBook-Air--Artem K8S % kubectl get endpoints web-service
Warning: v1 Endpoints is deprecated in v1.33+; use discovery.k8s.io/v1 EndpointSlice
NAME      ENDPOINTS      AGE
web-service   10.244.0.49:80,10.244.0.50:80,10.244.0.51:80   2m58s
ptashko@MacBook-Air--Artem K8S % kubectl describe service web-service
Name:           web-service
Namespace:      test
Labels:         service-type=internal
Annotations:    <none>
Selector:       app=web-app
Type:          ClusterIP
IP Family Policy: SingleStack
IP Families:   IPv4
IP:             10.109.192.49
IPs:            10.109.192.49
Port:           http  80/TCP
TargetPort:     80/TCP
Endpoints:     10.244.0.51:80,10.244.0.49:80,10.244.0.50:80
Session Affinity: None
Internal Traffic Policy: Cluster
Events:         <none>
ptashko@MacBook-Air--Artem K8S %
```

Ответы на вопросы:

Какой ClusterIP был назначен сервису?

Ответ: 10.109.192.49

Сколько endpoints связано с сервисом и какие у них адреса?

Ответ: 3

10.244.0.49:80

10.244.0.50:80

10.244.0.51:80

Какой селектор использует Service для поиска Pod'ов?

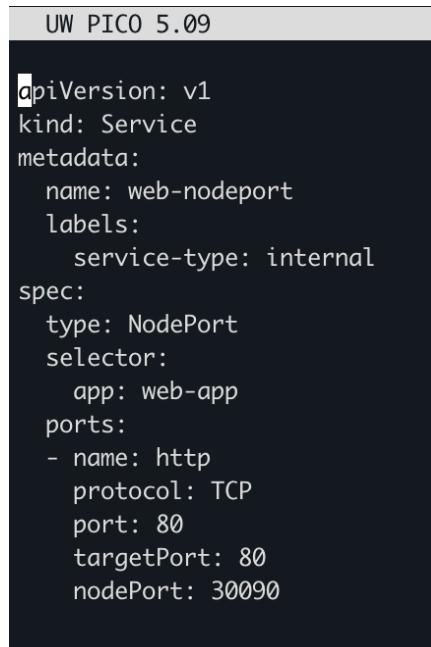
Ответ: app=web-app

Какой тип сервиса используется по умолчанию?

Ответ: ClusterIP

Задание 2.

Скрины:



```
UW PICO 5.09

apiVersion: v1
kind: Service
metadata:
  name: web-nodeport
  labels:
    service-type: internal
spec:
  type: NodePort
  selector:
    app: web-app
  ports:
    - name: http
      protocol: TCP
      port: 80
      targetPort: 80
      nodePort: 30090
```

Ответы на вопросы:

Какой NodePort был назначен сервису?

Ответ: http 30090/TCP

Доступен ли сервис извне кластера?

Ответ: да, так как это NodePort сервис

Какая разница между доступом через NodePort и port-forward?

Ответ: NodePort позволяет пользователям вне кластера получать доступ, а port-forward только локальной машине

Как проверить, что трафик балансируется между разными Pod'ами?

Ответ: через curl или используя LoadBalancer

Задание 3.

Скрины:

```
UW PICO 5.09                                         File: test2.y

apiVersion: apps/v1
kind: Deployment
metadata:
  name: multi-service-app
  labels:
    app: multi-service-app
spec:
  replicas: 2
  selector:
    matchLabels:
      app: multi-service-app
  template:
    metadata:
      labels:
        app: multi-service-app
        environment: test
    spec:
      containers:
        - name: frontend
          image: nginx:1.25-alpine
          ports:
            - containerPort: 80
        - name: backend
          image: httpd:2.4-alpine
          ports:
            - containerPort: 8080
          command: ["/bin/sh"]
          args:
            - "-c"
            - |
              cp /usr/local/apache2/conf/httpd.conf /usr/local/apache2/conf/httpd-custom.conf
              sed -i 's/^Listen 80/Listen 8080/g' /usr/local/apache2/conf/httpd-custom.conf
              sed -i 's/^#ServerName www.example.com:80/ServerName localhost:8080/g' /usr/local/apache2/conf/httpd-custom.conf
              echo "h1>Apache Server working on port 8080</h1>" > /usr/local/apache2/htdocs/index.html
              echo "<p>Pod: $(hostname)</p>" >> /usr/local/apache2/htdocs/index.html
              httpd -f /usr/local/apache2/conf/httpd-custom.conf -DFOREGROUND
```

```
UW PICO 5.09

apiVersion: v1
kind: Service
metadata:
  name: frontend-service
spec:
  type: ClusterIP
  selector:
    app: multi-service-app
  ports:
    - name: frontend
      protocol: TCP
      port: 80
      targetPort: 80

---
apiVersion: v1
kind: Service
metadata:
  name: backend-service
spec:
  type: ClusterIP
  selector:
    app: multi-service-app
  ports:
    - name: backend
      protocol: TCP
      port: 8080
      targetPort: 8080
```

```
UW PICO 5.09

apiVersion: v1
kind: Service
metadata:
  name: nodeport-external
spec:
  type: NodePort
  selector:
    app: multi-service-app
  ports:
    - name: frontend-external
      protocol: TCP
      port: 80
      targetPort: 80
      nodePort: 30081
    - name: backend-external
      protocol: TCP
      port: 8080
      targetPort: 8080
      nodePort: 30082
```

```
ptashko@MacBook-Air--Artem K8S % kubectl apply -f multi-nodeport.yaml
service/nodeport-external created
ptashko@MacBook-Air--Artem K8S % kubectl get service
NAME          TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)          AGE
backend-service ClusterIP 10.102.162.153 <none>        8080/TCP        9m8s
frontend-service ClusterIP 10.102.35.167 <none>        80/TCP          9m8s
kubernetes     ClusterIP 10.96.0.1     <none>        443/TCP         14d
nodeport-external NodePort   10.108.3.245 <none>        80:30081/TCP,8080:30082/TCP  21s
ptashko@MacBook-Air--Artem K8S % kubectl get endpoints backend*
zsh: no matches found: backend*
ptashko@MacBook-Air--Artem K8S % kubectl get endpoints backend-service
Warning: v1 Endpoints is deprecated in v1.33+; use discovery.k8s.io/v1 EndpointSlice
NAME          ENDPOINTS          AGE
backend-service 10.244.0.76:8080,10.244.0.77:8080  11m
ptashko@MacBook-Air--Artem K8S % kubectl get endpoints frontend-service
Warning: v1 Endpoints is deprecated in v1.33+; use discovery.k8s.io/v1 EndpointSlice
NAME          ENDPOINTS          AGE
frontend-service 10.244.0.76:80,10.244.0.77:80  11m
ptashko@MacBook-Air--Artem K8S % kubectl get endpoints nodeport-external
Warning: v1 Endpoints is deprecated in v1.33+; use discovery.k8s.io/v1 EndpointSlice
NAME          ENDPOINTS          AGE
nodeport-external 10.244.0.76:80,10.244.0.77:80,10.244.0.76:8080 + 1 more...  3m1s
ptashko@MacBook-Air--Artem K8S %
```

Ответы на вопросы:

Могут ли два сервиса указывать на одни и те же Pod'ы?

Ответ: нет, не могут

Как проверить, что каждый сервис перенаправляет трафик на правильный контейнер?

Ответ: необходимо подключиться к Pod'у с помощью kubectl exec и отправить запрос к сервису

Сколько всего endpoints создано для этого Deployment?

Ответ: 4

10.244.0.76:8080
10.244.0.77:8080
10.244.0.76:80
10.244.0.77:80

Как организовать доступ к обоим сервисам извне кластера?

Ответ: создать NodePort сервис

Задание 4.

Ответы на вопросы:

Какие признаки у сервиса, который не нашел Pod'ы?

Ответ: отсутствие endpoint'ов

Как быстро определить проблему с неправильными портами?

Ответ: через endpoint'ы, где будет указан порт сервиса

Какие команды наиболее эффективны для диагностики сетевых проблем?

Ответ: kubectl get endpoints

kubectl describe service pod deployment

kubectl logs <pod>

Как проверить, что DNS работает корректно внутри кластера?

Ответ: kubectl run test-client --image=busybox --rm -it --restart=Never -- sh

Задание 5.

Скрины:

UW PICO 5.09	UW PICO 5.09	UW PICO 5.09
<pre>apiVersion: apps/v1 kind: Deployment metadata: name: frontend labels: app: prod-app spec: replicas: 3 selector: matchLabels: app: frontend template: metadata: labels: app: frontend environment: production spec: containers: - name: frontend-cont image: nginx:1.25 ports: - containerPort: 80 env: - name: NGINX-PORT value: "80"</pre>	<pre>apiVersion: apps/v1 kind: Deployment metadata: name: backend labels: app: prod-app spec: replicas: 2 selector: matchLabels: app: backend template: metadata: labels: app: backend environment: production spec: containers: - name: backend-port image: httpd:2.4 ports: - containerPort: 8080 env: - name: APACHE_PORT value: "8080"</pre>	<pre>apiVersion: v1 kind: Service metadata: name: frontend-service labels: service-type: internal spec: type: ClusterIP selector: app: frontend ports: - name: front-port protocol: TCP port: 80 targetPort: 80</pre>
UW PICO 5.09	UW PICO 5.09	UW PICO 5.09
<pre>apiVersion: v1 kind: Service metadata: name: frontend-external labels: service-type: external spec: type: NodePort selector: app: frontend ports: - name: front-prod-node protocol: TCP port: 80 targetPort: 80 nodePort: 30080</pre>	<pre>apiVersion: v1 kind: Service metadata: name: backend-service labels: service-type: internal spec: type: ClusterIP selector: app: backend ports: - name: back-port protocol: TCP port: 8080 targetPort: 8080</pre>	

```

ptashko@MacBook-Air--Artem K8S % nano prod-front.yaml
ptashko@MacBook-Air--Artem K8S % kubectl apply -f prod-front.yaml
deployment.apps/foreground created
ptashko@MacBook-Air--Artem K8S % nano prod-back.yaml
ptashko@MacBook-Air--Artem K8S % kubectl apply -f prod-back.yaml
deployment.apps/backend created
ptashko@MacBook-Air--Artem K8S % ls -la
total 80
drwxr-xr-x@ 12 ptashko staff 384 23 окт. 15:49 .
drwxr-x---+ 67 ptashko staff 2144 23 окт. 14:31 ..
-rw-r--r--@ 1 ptashko staff 236 23 окт. 15:49 backend-service.yaml
-rw-r--r--@ 1 ptashko staff 417 17 окт. 14:31 frontend-clusterip.yaml
-rw-r--r--@ 1 ptashko staff 260 23 окт. 15:43 frontend-external.yaml
-rw-r--r--@ 1 ptashko staff 235 23 окт. 15:34 frontend-service.yaml
-rw-r--r--@ 1 ptashko staff 337 17 окт. 14:39 multi-nodeport.yaml
-rw-r--r--@ 1 ptashko staff 442 23 окт. 15:58 prod-back.yaml
-rw-r--r--@ 1 ptashko staff 442 23 окт. 15:57 prod-front.yaml
-rw-r--r--@ 1 ptashko staff 593 2 окт. 12:27 test1.yaml
-rw-r--r--@ 1 ptashko staff 1154 17 окт. 13:53 test2.yaml
-rw-r--r--@ 1 ptashko staff 243 17 окт. 13:27 testClusterIP.yaml
ptashko@MacBook-Air--Artem K8S % kubectl apply -f frontend-serv*.yaml
service/frontend-service configured
ptashko@MacBook-Air--Artem K8S % kubectl apply -f frontend-ex*.yaml
service/frontend-external created
ptashko@MacBook-Air--Artem K8S % kubectl apply -f backend-serv*.yaml
service/backend-service configured
ptashko@MacBook-Air--Artem K8S %

```

```

ptashko@MacBook-Air--Artem K8S % kubectl get all -l environment=production
NAME                               READY   STATUS        RESTARTS   AGE
pod/backend-6b676c6b87-pvlkk     0/1    ContainerCreating   0          2m10s
pod/backend-6b676c6b87-s44kw     0/1    ContainerCreating   0          2m10s
pod/foreground-86f5c94778-9b9d5  1/1    Running       0          3m16s
pod/foreground-86f5c94778-cdxk4  1/1    Running       0          3m16s
pod/foreground-86f5c94778-pxbrd  1/1    Running       0          3m16s

NAME                         DESIRED   CURRENT   READY   AGE
replicaset.apps/backend-6b676c6b87  2         2         0      2m10s
replicaset.apps/foreground-86f5c94778 3         3         3      3m16s
ptashko@MacBook-Air--Artem K8S % kubectl describe service frontend-external
Name:                     frontend-external
Namespace:                default
Labels:                   service-type=external
Annotations:              <none>
Selector:                 app=foreground
Type:                     NodePort
IP Family Policy:        SingleStack
IP Families:             IPv4
IP:                      10.102.106.250
IPs:                     10.102.106.250
Port:                    front-prod-node 80/TCP
TargetPort:               80/TCP
NodePort:                 front-prod-node 30080/TCP
Endpoints:                10.244.0.103:80,10.244.0.105:80,10.244.0.104:80
Session Affinity:         None
External Traffic Policy: Cluster
Internal Traffic Policy: Cluster
Events:                  <none>

```

Ответы на вопросы:

Как организовать сетевую изоляцию между frontend и backend?

Ответ: использовать разные селекторы

Как проверить, что backend доступен только изнутри кластера?

Ответ: через describe, например, так как это ClusterIP сервис

Задание 6.

Скрины:

```
ptashko@MacBook-Air--Artem K8S % kubectl get services --show-labels
NAME        TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)          AGE      LABELS
backend-service ClusterIP  10.102.162.153 <none>        8080/TCP        6d1h    service-type=internal
frontend-external NodePort   10.102.106.250 <none>        80:30080/TCP     3m37s   service-type=external
frontend-service ClusterIP  10.102.35.167 <none>        80/TCP          6d1h    service-type=internal
kubernetes     ClusterIP  10.96.0.1     <none>        443/TCP          20d     component=apiserver,provider=kubernetes
nodeport-external NodePort   10.108.3.245  <none>        80:30081/TCP,8080:30082/TCP 6d1h    <none>
ptashko@MacBook-Air--Artem K8S % kubectl get endpoints
Warning: v1 Endpoints is deprecated in v1.33+; use discovery.k8s.io/v1 EndpointSlice
NAME           ENDPOINTS                                     AGE
backend-service 10.244.0.106:8080,10.244.0.107:8080       6d1h
frontend-external 10.244.0.103:80,10.244.0.104:80,10.244.0.105:80 3m48s
frontend-service 10.244.0.103:80,10.244.0.104:80,10.244.0.105:80 6d1h
kubernetes      192.168.49.2:8443                         20d
nodeport-external 10.244.0.91:80,10.244.0.92:80,10.244.0.91:8080 + 1 more... 6d1h
ptashko@MacBook-Air--Artem K8S % kubectl describe service frontend-service
Name:           frontend-service
Namespace:      default
Labels:         service-type=internal
Annotations:   <none>
Selector:       app=frontend
Type:          ClusterIP
IP Family Policy: SingleStack
IP Families:   IPv4
IP:             10.102.35.167
IPs:            10.102.35.167
Port:           front-port 80/TCP
TargetPort:     80/TCP
Endpoints:     10.244.0.103:80,10.244.0.104:80,10.244.0.105:80
Session Affinity: None
Internal Traffic Policy: Cluster
Events:        <none>
ptashko@MacBook-Air--Artem K8S % kubectl run dns-check --image=busybox --rm -it --restart=Never -- nslookup frontend-service
Error from server (AlreadyExists): pods "dns-check" already exists
ptashko@MacBook-Air--Artem K8S % kubectl delete pod dns-check
pod "dns-check" deleted from default namespace
ptashko@MacBook-Air--Artem K8S % kubectl run dns-check --image=busybox --rm -it --restart=Never -- nslookup frontend-service
Server:        10.96.0.10
Address:       10.96.0.10:53

** server can't find frontend-service.cluster.local: NXDOMAIN
** server can't find frontend-service.svc.cluster.local: NXDOMAIN
** server can't find frontend-service.cluster.local: NXDOMAIN

Name:   frontend-service.default.svc.cluster.local
Address: 10.102.35.167

** server can't find frontend-service.svc.cluster.local: NXDOMAIN
pod "dns-check" deleted from default namespace
pod default/dns-check terminated (Error)
```

```
ptashko@MacBook-Air--Artem K8S % kubectl logs deployment/foreground --tail=10
Found 3 pods, using pod/foreground-86f5c94778-9b9d5
2025/10/23 12:57:44 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/10/23 12:57:44 [notice] 1#1: start worker processes
2025/10/23 12:57:44 [notice] 1#1: start worker process 29
2025/10/23 12:57:44 [notice] 1#1: start worker process 30
2025/10/23 12:57:44 [notice] 1#1: start worker process 31
2025/10/23 12:57:44 [notice] 1#1: start worker process 32
2025/10/23 12:57:44 [notice] 1#1: start worker process 33
2025/10/23 12:57:44 [notice] 1#1: start worker process 34
2025/10/23 12:57:44 [notice] 1#1: start worker process 35
2025/10/23 12:57:44 [notice] 1#1: start worker process 36
ptashko@MacBook-Air--Artem K8S % kubectl get events --field-selector involvedObject.kind=Service
No resources found in default namespace.
```

Ответы на вопросы:

Какие метрики важны для мониторинга сервисов?

Ответ: статусы node, deployment'ов и потребление ресурсов кластером

Как определить, что сервис работает корректно?

Ответ: сервис доступен и безошибочно работает

Какие инструменты наиболее эффективны для диагностики сетевых проблем?

Ответ: kubectl describe pod/<pod-name>

kubectl logs <pod-name>

kubectl run dns-check --image=busybox --rm -it --restart=Never -- nslookup frontend-service

kubectl get

kubectl exec

Контрольные вопросы

В чем основное отличие ClusterIP от NodePort?

Ответ: при использовании NodePort сервис доступен извне кластера, при использовании ClusterIP - только внутри

Как Service находит Pod'ы для балансировки нагрузки?

Ответ: Service использует selector, в котором ищет необходимы labels

Когда следует использовать несколько сервисов для одного Deployment?

Ответ: когда Pod'ы должны использовать разные порты или протоколы

Как организовать доступ к разным контейнерам в одном Pod?

Ответ: указать разные порты

Какие ограничения есть у NodePort сервисов?

Ответ: в значениях порта: 30000-32767

Какие три команды вы бы использовали для быстрой диагностики проблемного сервиса?

Ответ: kubectl get endpoints

kubectl get services

kubectl describe service <service-name>

Как определить, что проблема с сетью, а не с приложением?

Ответ: провести анализ сервисов и endpoint'ов

Какие логи наиболее полезны при проблемах с сервисами?

Ответ: логи, связанные с DNS или с балансировкой нагрузки, также логи Pod'ов