

kubernetes 部署Promethues + Grafana

k8s学习

1. kubernetes 集群环境

```
1. master 节点:
2. IP: 192.168.153.31 主机名: k8s-master 版本 v1.18.4 系统: cent
   os 7.5
3.
4. node 节点:
5. IP: 192.168.153.32 主机名: k8s-node1 版本 v1.18.4 系统: cent
   os 7.5
6. IP: 192.168.153.33 主机名: k8s-node2 版本 v1.18.4 系统: cent
   os 7.5
```

2. 部署Promethues 的数据持久化

2.1 安装nfs服务（所有节点执行）

```
1. 安装nfs-utils
2. # yum install nfs-utils
3. 启动nfs服务
4. # systemctl start nfs
5. 设置nfs服务开机自启
6. systemctl enable nfs
```

2.2 部署nfs pv

2.2.1上传yaml文件

```
[root@master1 nfs-client]# ls
class.yaml  deployment.yaml  rbac.yaml
```

2.2.2 开始部署

```
1. # kubectl apply -f rbac.yaml
2. # kubectl apply -f class.yaml
```

```
[root@master1 nfs-client]# kubectl get sc
NAME                                PROVISIONER      AGE
managed-nfs-storage                fuseim.pri/ifs    5m18s
[root@master1 nfs-client]#
```

2.2.3 设定一台主机为nfs server端（可以为任意一台主机,我这里是k8s-node1节点）

```
1. cat > /etc/exports << EOF
2. /mnt *(rw,no_root_squash)
3. EOF
4.
5. systemctl restart nfs
6.
7. # 创建nfs挂载的目录
8. mkdir /mnt/kubernetes
```

修改 deployment.yaml 里的地址和路径（与上面节点的ip保持一致）：

```
env:
  - name: PROVISIONER_NAME
    value: fuseim.pri/ifs
  - name: NFS_SERVER
    value: 192.168.153.32
  - name: NFS_PATH
    value: /mnt/kubernetes
volumes:
  - name: nfs-client-root
    nfs:
      server: 192.168.153.32
      path: /mnt/kubernetes
```

修改

29, 13

底端

部署deployment.yaml

```
1. kubectl apply -f deployment.yaml
```

```
[root@k8s-master nfs-client]# kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nfs-client-provisioner-7bffb47997-7dkth  1/1     Running   0          22s
[root@k8s-master nfs-client]#
```

3. 在 K8S 平台部署 Prometheus+Grafana

3.1 上传相关 yaml 文件

```
[root@master1 prome]# ll
总用量 68
-rw-r--r-- 1 root root 657 11月 24 17:18 alertmanager-configmap.yaml
-rw-r--r-- 1 root root 2183 11月 23 22:56 alertmanager-deployment.yaml
-rw-r--r-- 1 root root 331 11月 23 22:56 alertmanager-pvc.yaml
-rw-r--r-- 1 root root 392 11月 23 22:56 alertmanager-service.yaml
drwxr-xr-x 3 root root 180 2月 7 23:32 ansible_node_exporter
-rw-r--r-- 1 root root 1198 11月 23 22:56 grafana.yaml
drwxr-xr-x 2 root root 156 2月 7 23:32 grafana仪表盘
-rw-r--r-- 1 root root 2378 11月 23 22:56 kube-state-metrics-deployment.yaml
-rw-r--r-- 1 root root 2576 11月 23 22:56 kube-state-metrics-rbac.yaml
-rw-r--r-- 1 root root 506 11月 23 22:56 kube-state-metrics-service.yaml
-rw-r--r-- 1 root root 1495 11月 23 22:56 node-exporter-ds.yml
-rw-r--r-- 1 root root 425 11月 23 22:56 node-exporter-service.yaml
-rw-r--r-- 1 root root 5124 11月 24 22:53 prometheus-configmap.yaml
-rw-r--r-- 1 root root 1080 11月 23 22:56 prometheus-rbac.yaml
-rw-r--r-- 1 root root 4884 11月 24 22:44 prometheus-rules.yaml
-rw-r--r-- 1 root root 392 11月 23 22:56 prometheus-service.yaml
-rw-r--r-- 1 root root 3259 11月 23 22:56 prometheus-statefulset.yaml
[root@master1 prome]#
```

3.2 编辑文件并部署 promethues

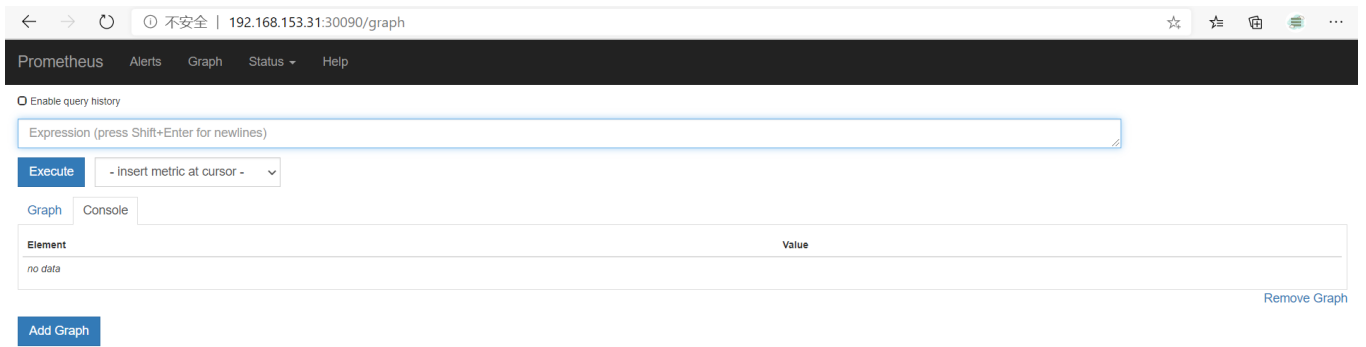
vim prometheus-configmap.yaml

修改监控节点 ip

```
[root@k8s-master prometheus]# cat prometheus-configmap.yaml | grep 192.168
- 192.168.153.31:9100
- 192.168.153.32:9100
- 192.168.153.33:9100
```

1. kubectl apply -f prometheus-configmap.yaml
2. kubectl apply -f prometheus-rbac.yaml
3. kubectl apply -f prometheus-rules.yaml
4. kubectl apply -f prometheus-service.yaml
5. kubectl apply -f prometheus-statefulset.yaml

访问 promethues : node_ip:30090



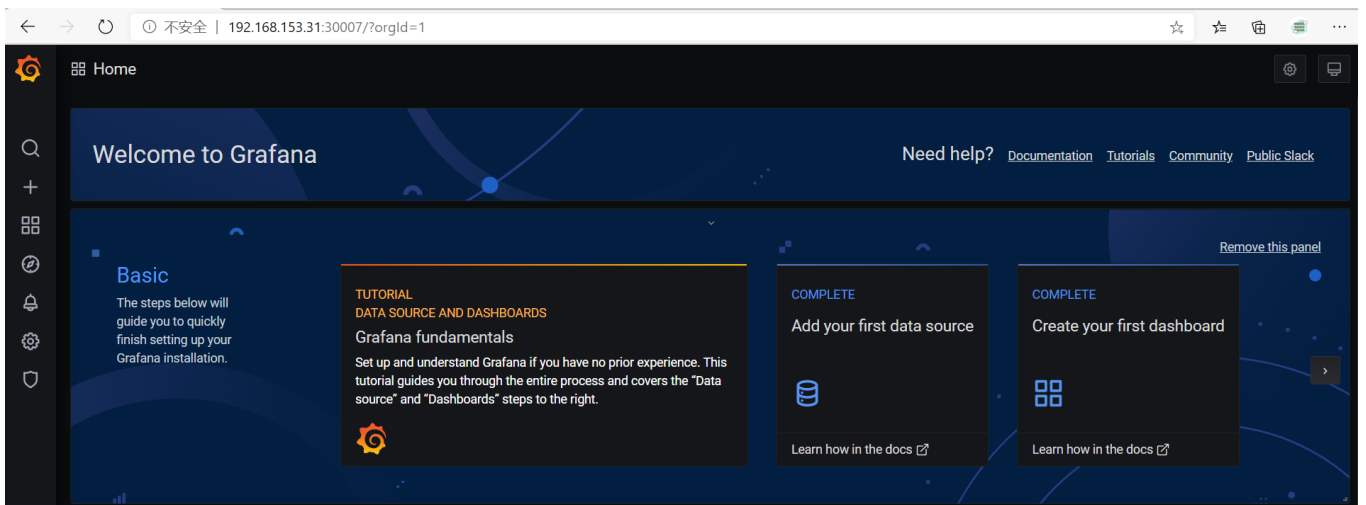
3.3 部署grafana

```
1. kubectl apply -f grafana.yaml
```

访问：node_ip:30007

初始账号密码都是：admin

本机重置：admin123admin



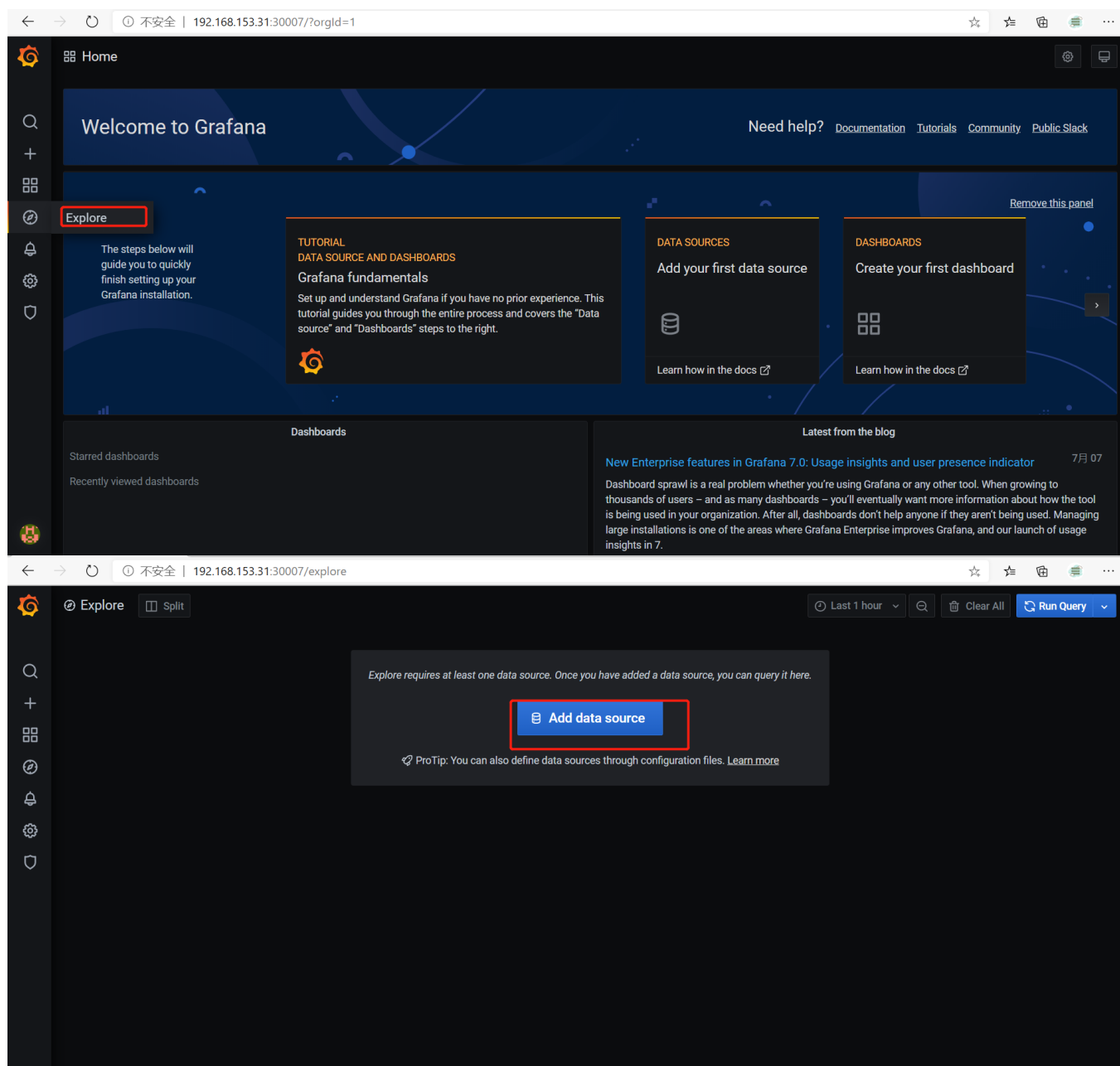
3.4 添加监控 node 节点

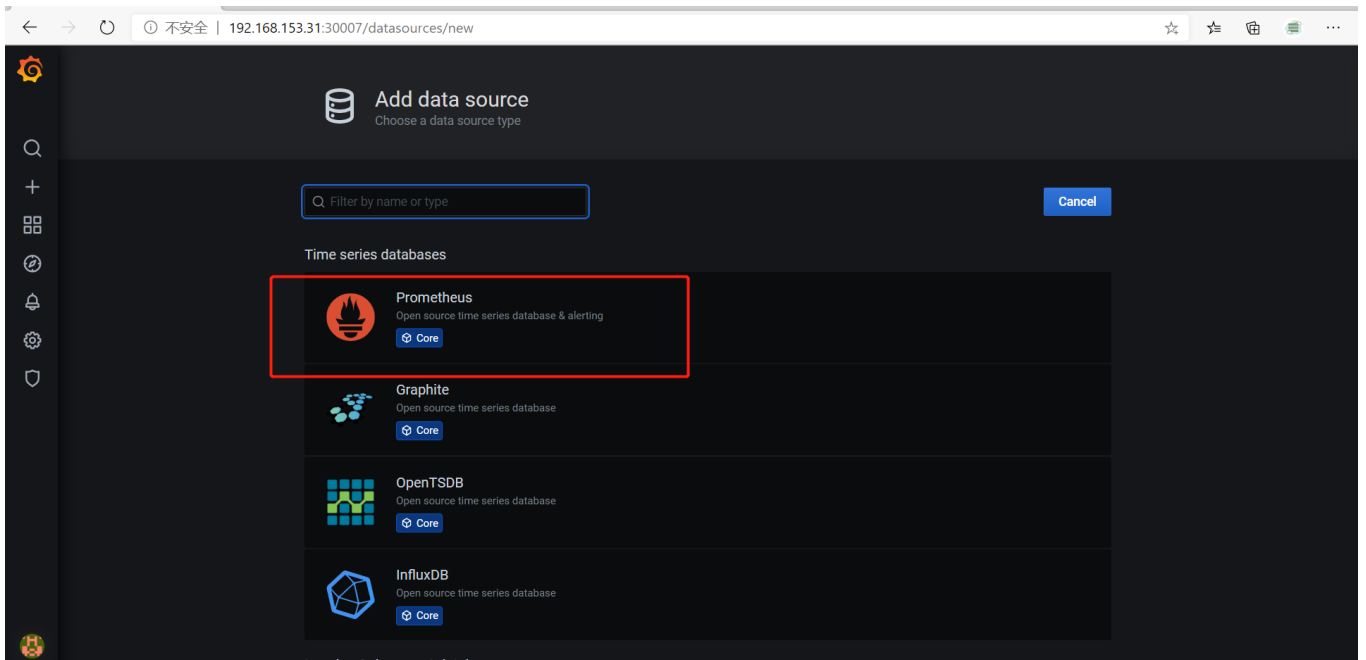
```
1. kubectl apply -f node-exporter-ds.yaml
```

3.5 添加master节点监控

```
1. # kubectl apply -f kube-state-metrics-rbac.yaml
2. # kubectl apply -f kube-state-metrics-service.yaml
3. # kubectl apply -f kube-state-metrics-deployment.yaml
```

3.6 导入模板

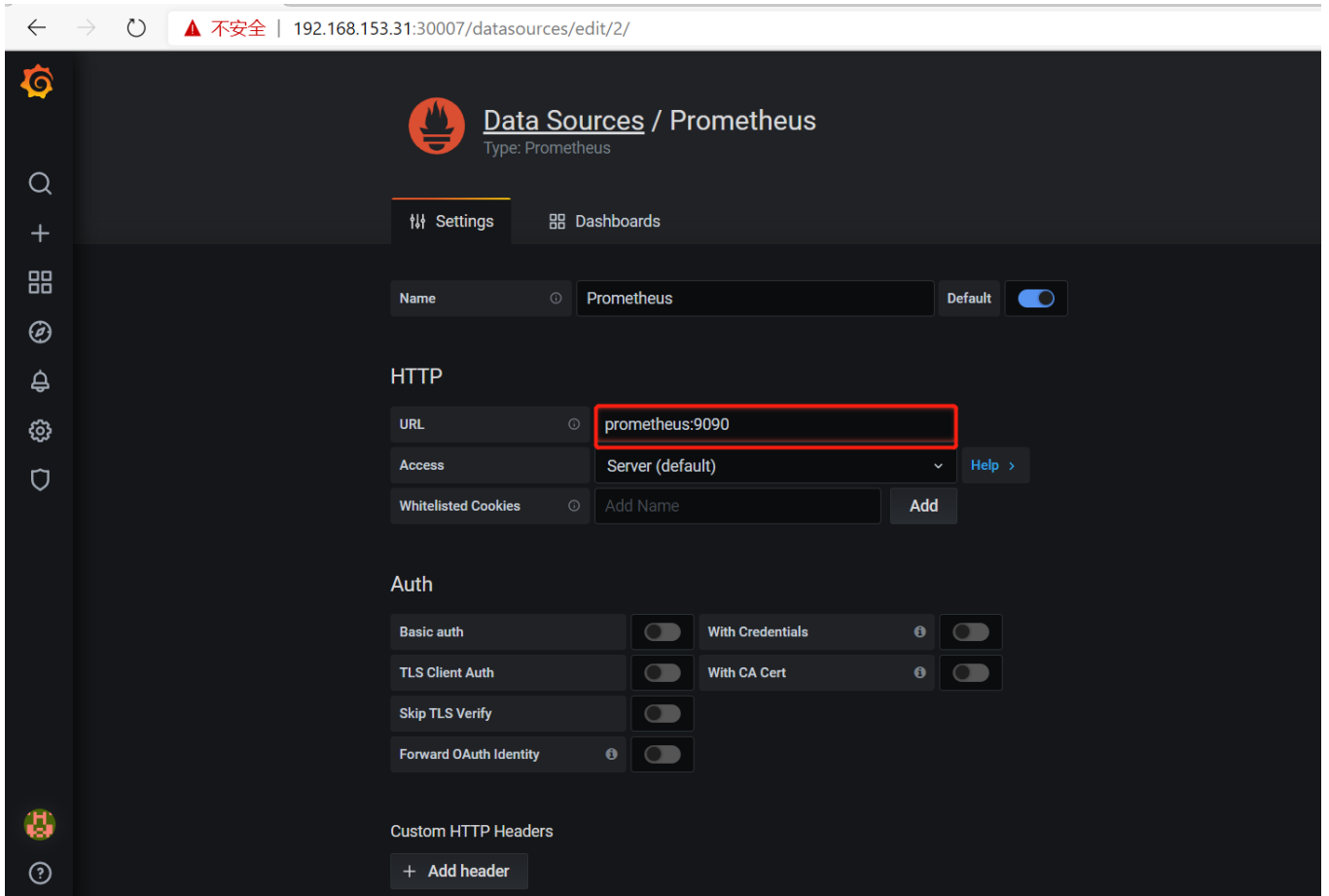




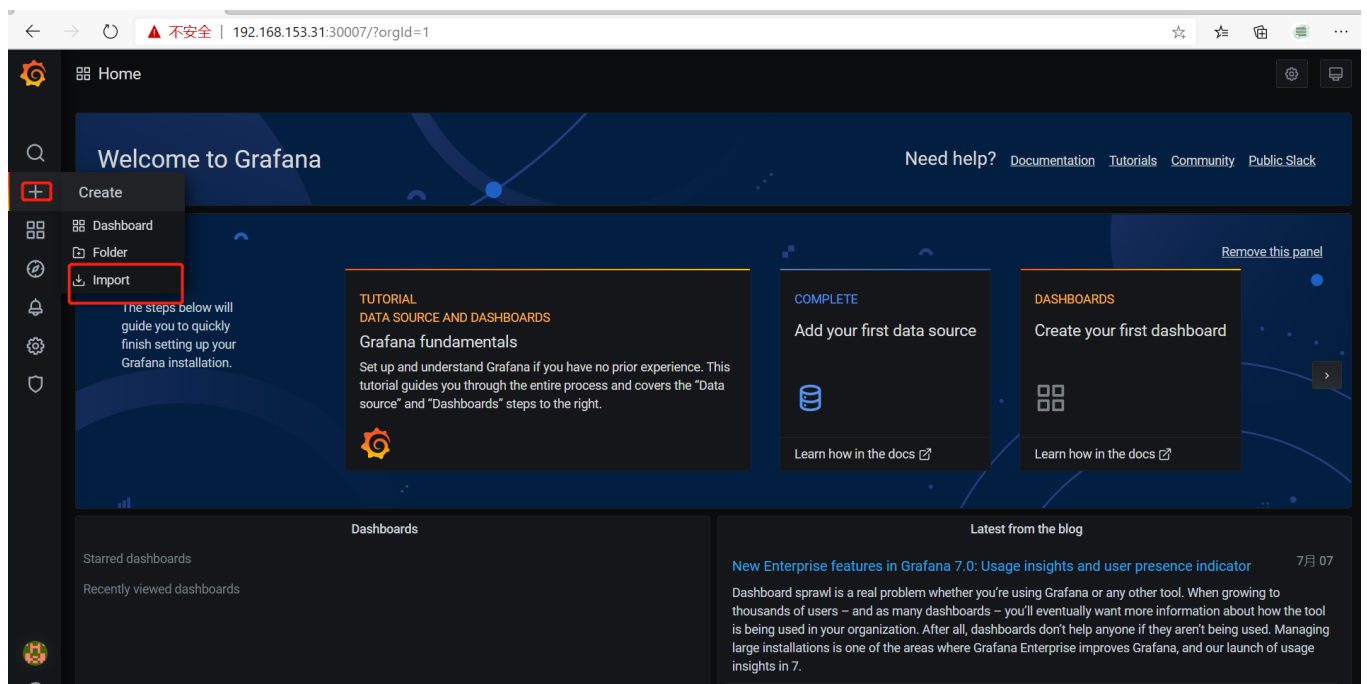
在master节点查看promethues svc

```
[root@k8s-master prometheus]# kubectl get svc -n kube-system | grep pro
prometheus      NodePort    10.0.0.248    <none>      9090:30090/TCP    25h
```

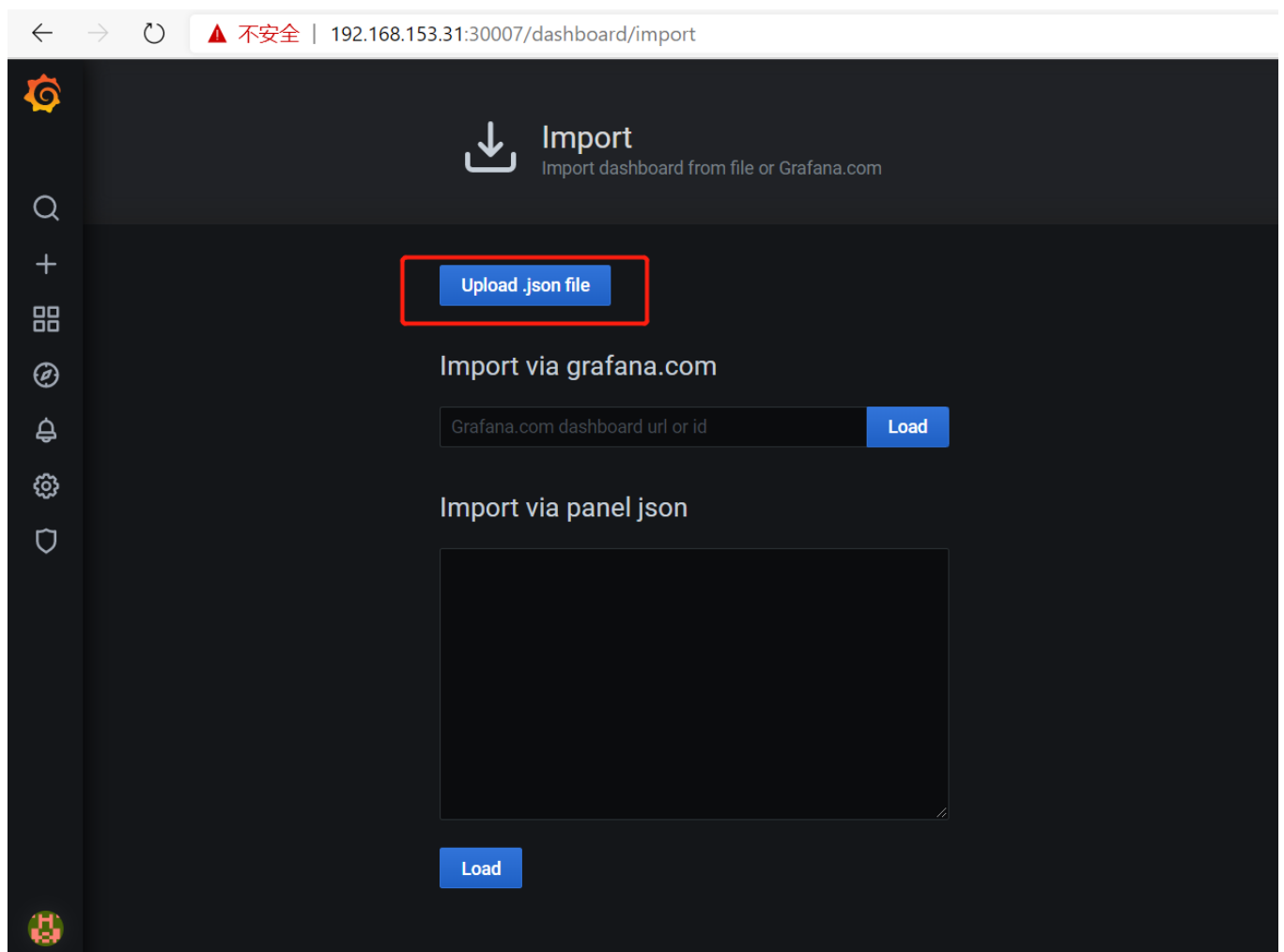
通过promethues svc名字和端口就可以连接






导入模板



依次导入



名称	修改日期	类型	大小
 K8S工作节点监控-1574605167595.json	2019/11/24 22:19	JSON 文件	59 KB
 K8S集群资源监控-1574605157483.json	2019/11/24 22:19	JSON 文件	58 KB
 K8S资源对象状态监控-157460517996...	2019/11/24 22:19	JSON 文件	59 KB

3.7 最后效果

