Monito CPU, MEMORY(GRAFANA): https://blog.csdn.net/liudinglong1989/article/details/108571625  
ps -ef | grep 4739  
  
  
WINDOWS是需要安装PROMETHEUS和GRAFANA,安装如下。 NODE\_EXPORTER用来在LINUX运行，输出数据到PROMETHEUS  
ps -ef | grep node\_  
netstat -tunlp | grep 9100  
  
  
Grafana安装：https://grafana.com/grafana/download?platform=windows  
下载安装，再点EXE运行，检查， http://localhost:9090/targets  
  
Prometheus安装： https://prometheus.io/download/  
下载，直接点EXE运行，检查，http://localhost:3000  
先在CONFIGURATION里配置DATASOURCE，   
再配置DASHBOARD，IMPORT模板的ID，同时导入DATASOURCE,就能看到数据了  
  
Node\_exporter安装：https://github.com/prometheus/node\_exporter/releases  
下载相应版本，放到LIUNX里，./node\_exporter 运行，检查是不是正常运行，会有显示  
  
  
http://localhost:9090/targets 查看PERMETHEUS运行状态  
http://localhost:3000 查看GRAFANA界面  
  
  
https://grafana.com/grafana/dashboards  
下载DASHBOADR 模块，COPY ID 到GRAFANA里，记得同时加载DATA SOURCE

Grafana + Prometheus监控篇之Windows监控Linux服务器资源

　　今天这篇文章介绍一下windows本地环境监控linux服务器资源指标，写这个的动机是源于性能测试。大家做性能测试之前一定要搞清楚一些概念性的东西，比如说什么是指标，需要监控什么指标。在很久之前，我写过一篇性能测试中关键指标的监控与分析文章，

　　里面介绍了关于指标的概念。

　　闲话不多说，今天介绍的工具主要是监控linux服务器上的资源指标，例如：CPU，内存，硬盘，网络等资源。

　　工具介绍

　　Grafana是一款用Go语言开发的跨平台开源数据度量分析及可视化工具，可以做数据监控和数据统计，带有告警功能。

　　Grafana支持许多不同的时间序列数据（数据源）存储后端。每个数据源都有一个特定的查询编辑器。官方支持以下数据源：Graphite、infloxdb、opensdb、prometheus、elasticsearch、cloudwatch等。

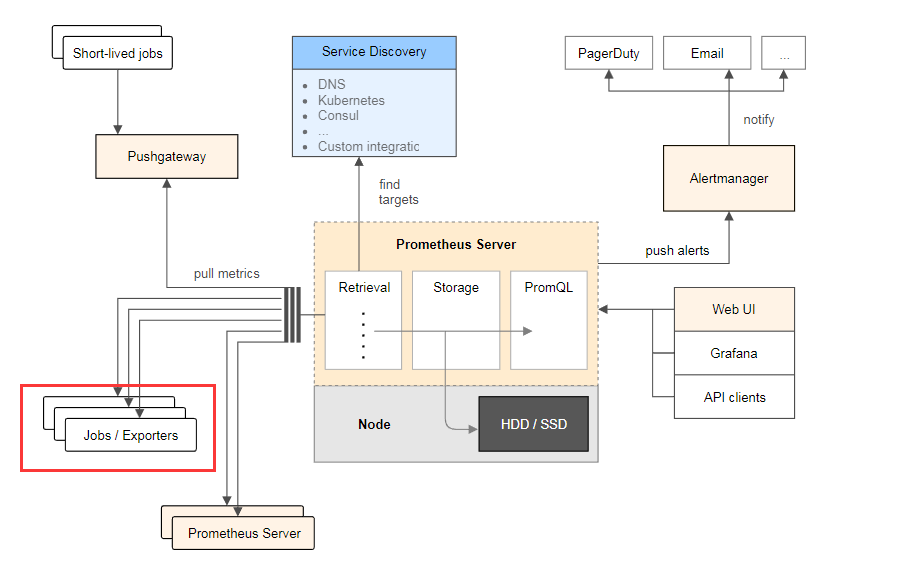
　　Prometheus（简称Prom）前身是SoundCloud的告警工具包，现已演化成一个独立的开源监控系统。属于Kurberntes所在的Cloud Native Computing Foundation。

　　Prometheus把一些常用数据库、系统、中间件等的指标数据的采集做成了一个个 exporter，在生产环境中，直接导入使用就可以。采用 http 协议，使用 pull 模式，拉取数据，简单易懂。 总而言之，它是用来拉取数据的。

　　Exporters：负责向prometheus server做数据汇报的程序统。而不同的数据汇报由不同的exporters实现，比如监控主机有node-exporters，mysql有MySQL server exporter等

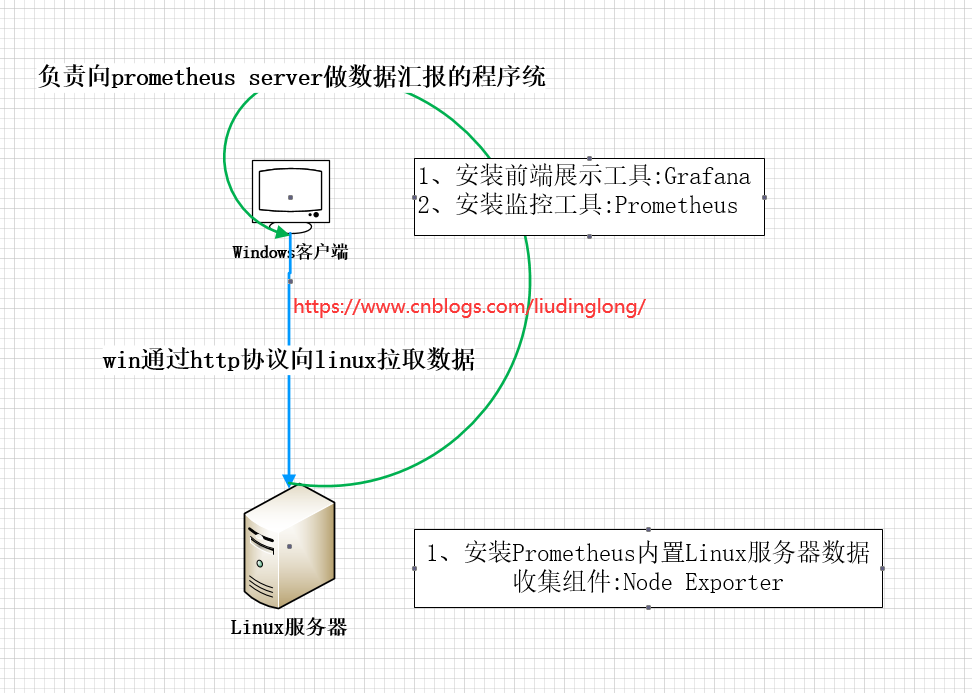
　　在Prometheus架构中，Exporters是负责收集数据并将信息汇报给Prometheus Server的组件。这篇文章我主要是监控Linux服务器资源，官方提供了node\_exporter内置了对主机系统的基础监控。

　　下面是Prometheus核心架构图，有兴趣的话可以了解一下： 如果看不明白，请移步官方中文文档：https://prometheus.fuckcloudnative.io/



　　监控方式

　　监控的流程，简单画一个草图，帮助理解 ，如图：



　　环境搭建

　　Grafana安装：https://grafana.com/grafana/download?platform=windows

　　Prometheus安装： https://prometheus.io/download/

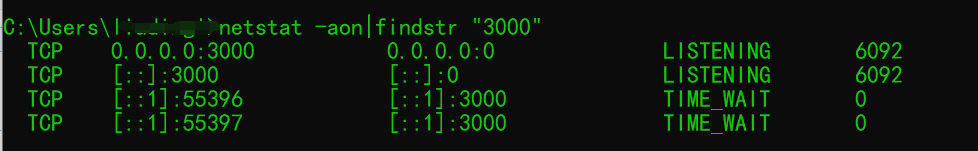
　　Node\_exporter安装：https://github.com/prometheus/node\_exporter/releases

　　总共安装上述三样程序，安装之后启动具体详情见下文：

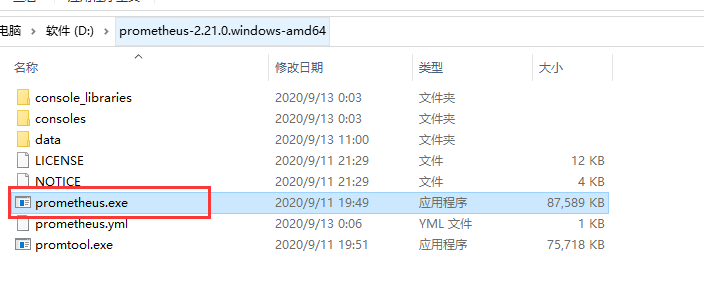
　　1、Grafana运行： 双击exe即可，默认端口3000，如图：



　　查看进程：



　　2、Prometheus运行：同上



　　启动方式

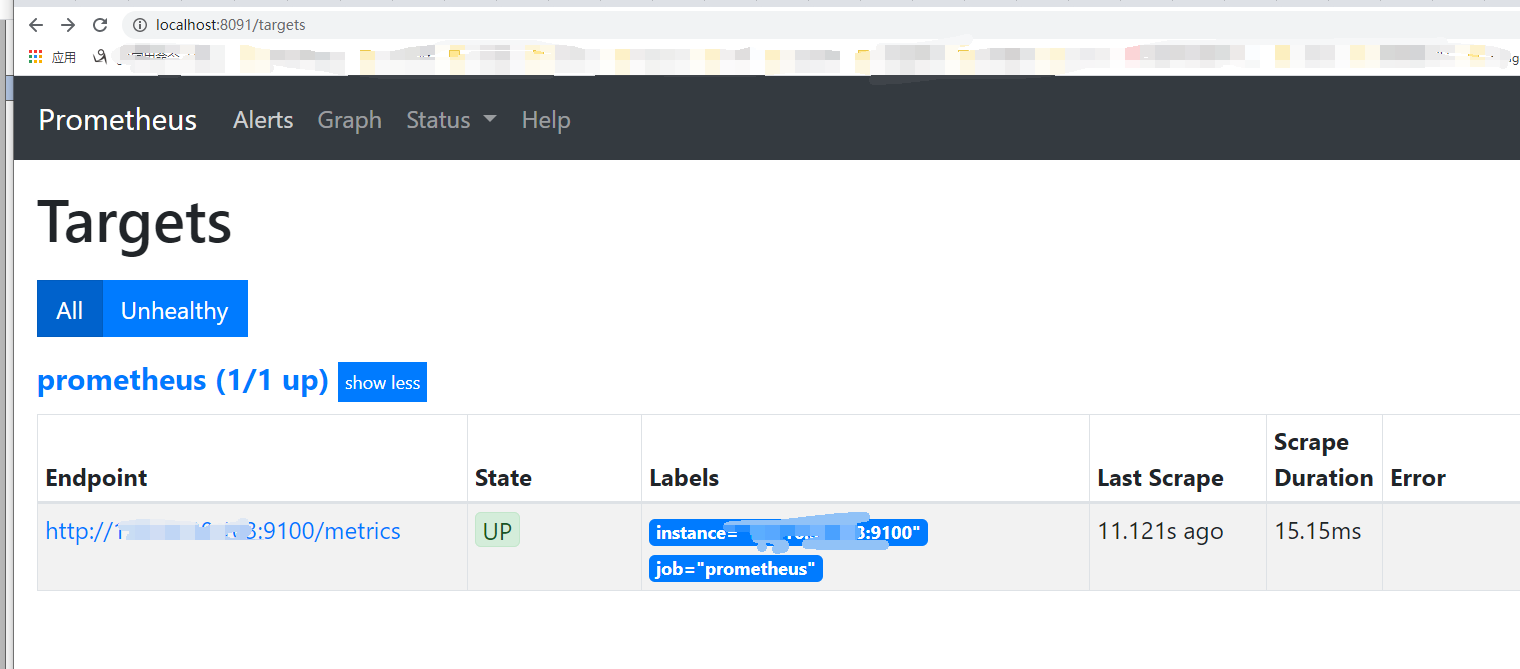
# 默认启动

直接双击exe文件，启动9090端口

# 自定义端口启动

prometheus.exe --config.file=prometheus.yml --web.listen-address=:8091 &

　　访问Ip+8091地址，status→targets出现UP正常启动成功，如图所示：



　　需要注意的地方：

　　在启动这个之前，需要修改Prometheus配置， prometheus.yml配置文件请求获取exporter采集器数据，比如我的node-exporter端口号是9100

　　prometheus.yml中一共定义了两个监控：一个是监控prometheus自身服务，另一个是监控Linux服务器。这里给个完整的示例：

scrape\_configs:

#prometheus自身监控

- job\_name: 'prometheus'

static\_configs:

- targets: ['localhost:9090']

#linux服务器监控

- job\_name: 'linux'

static\_configs:

- targets: ['NODE\_IP:9100']

labels:

instance: node1

　　3、Node-exporter启动方式：

#默认启动，起来后node\_exporter会监控在9100端口

./node\_exporter

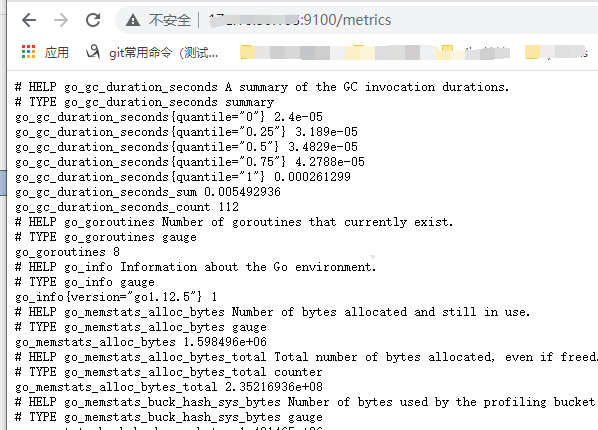
#自定义启动

nohup ./node\_exporter --web.listen-address=":9101" &

　　日志追加：

nohup ./node\_exporter > ./nohup.out 2>&1 &

　　如图：



　　进程查看命令：

ps -ef | grep node\_

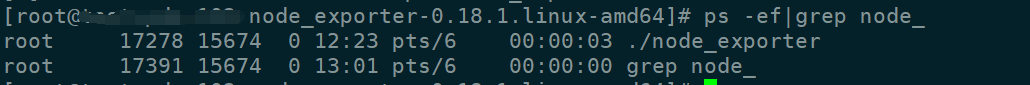
查看进程

netstat -tunlp | grep node

查看端口

netstat -tunlp | grep 9101

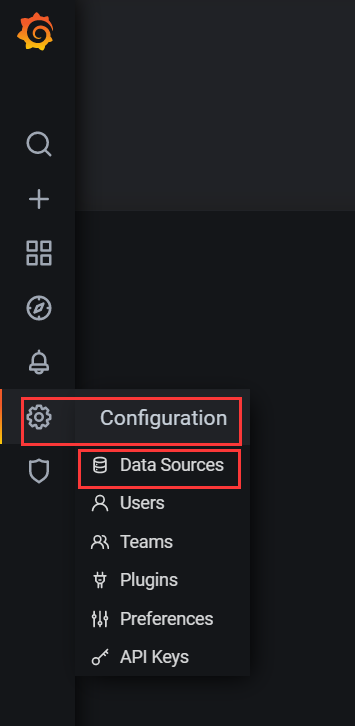
　　如图：



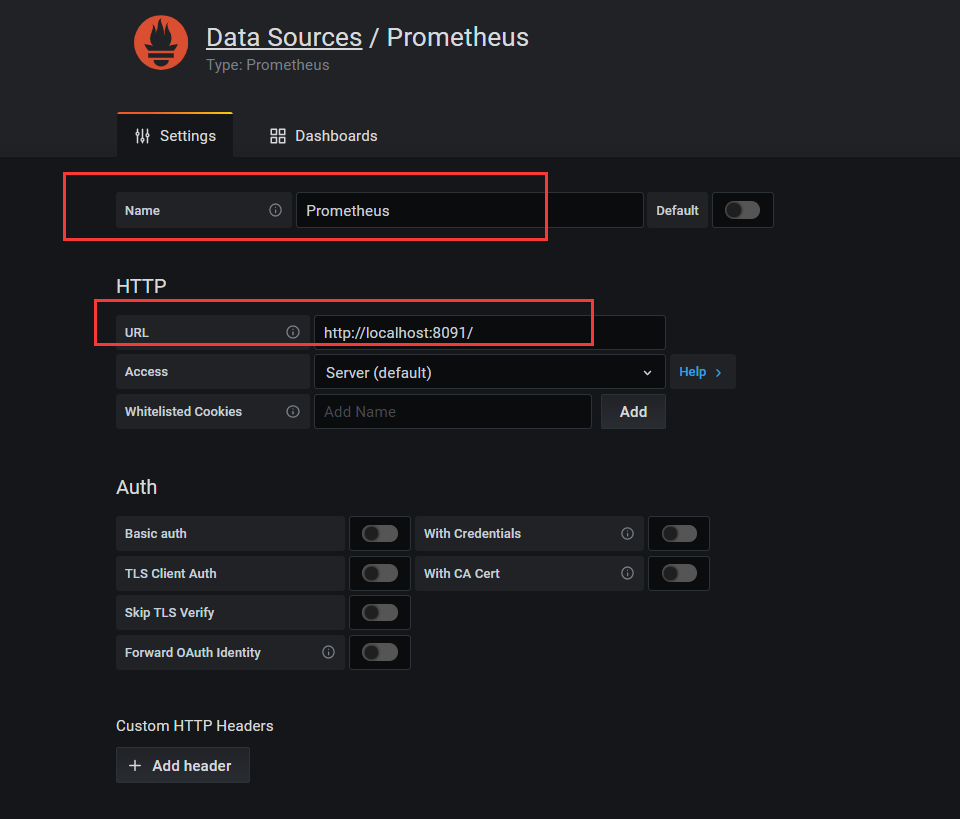
　　前端Grafana配置面板设置

　　启动grafana后，浏览器输入:ip+3000端口，默认账号:admin，密码:admin

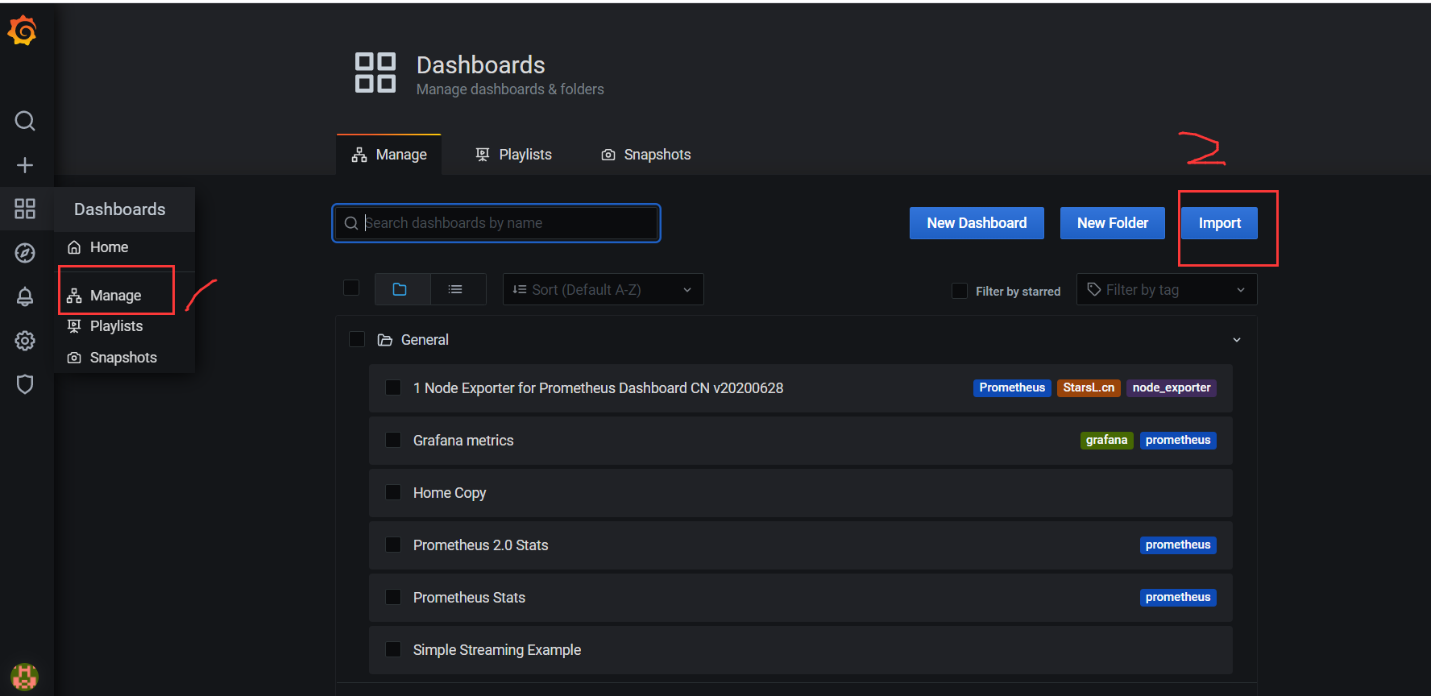
　　既然我们已经拿到了数据，那首先就配置数据源，如图：



　　主要是这个url配置对，这个8091就是prometheus服务的端口。

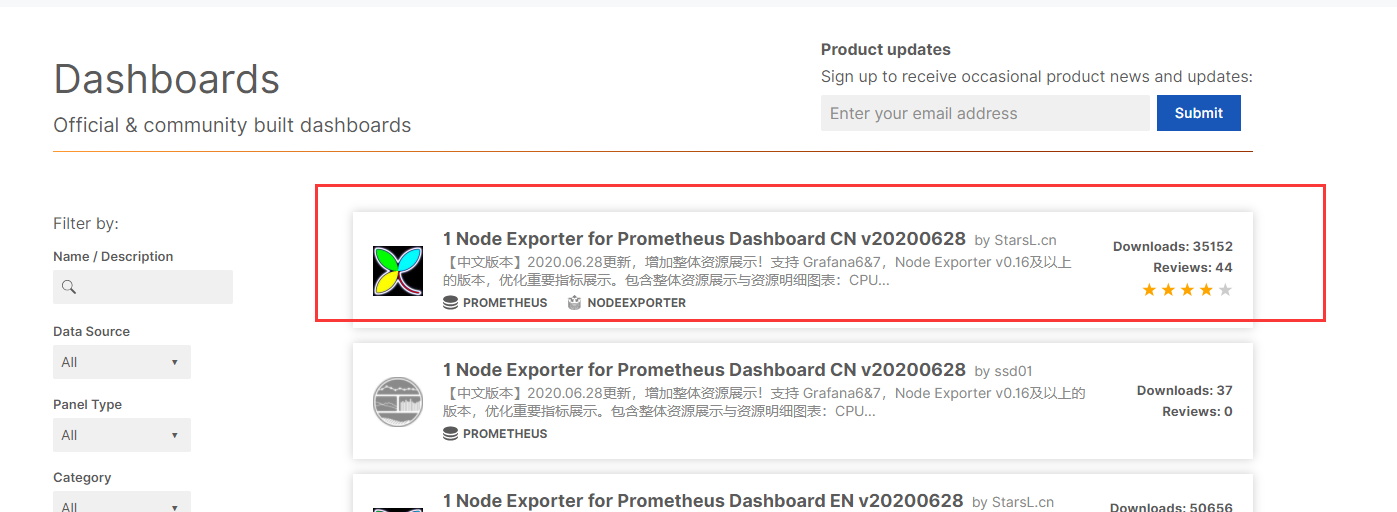


　　导入监控模板，如图：

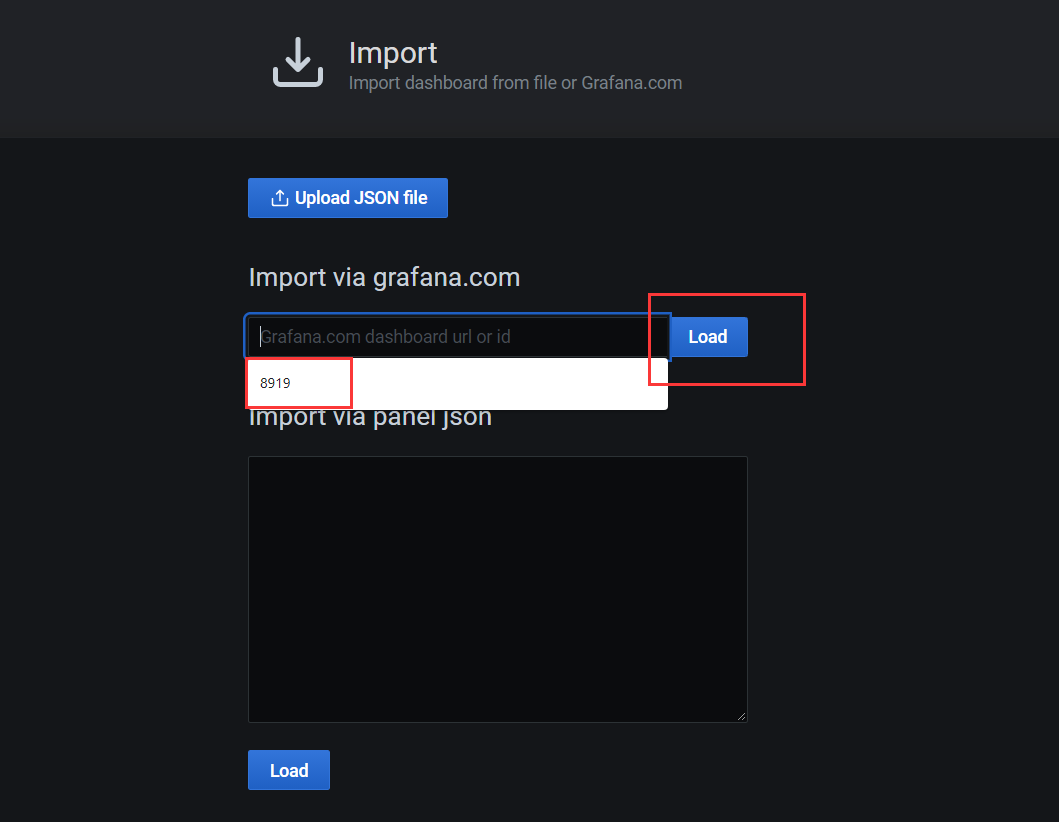


　　模板地址：https://grafana.com/grafana/dashboards

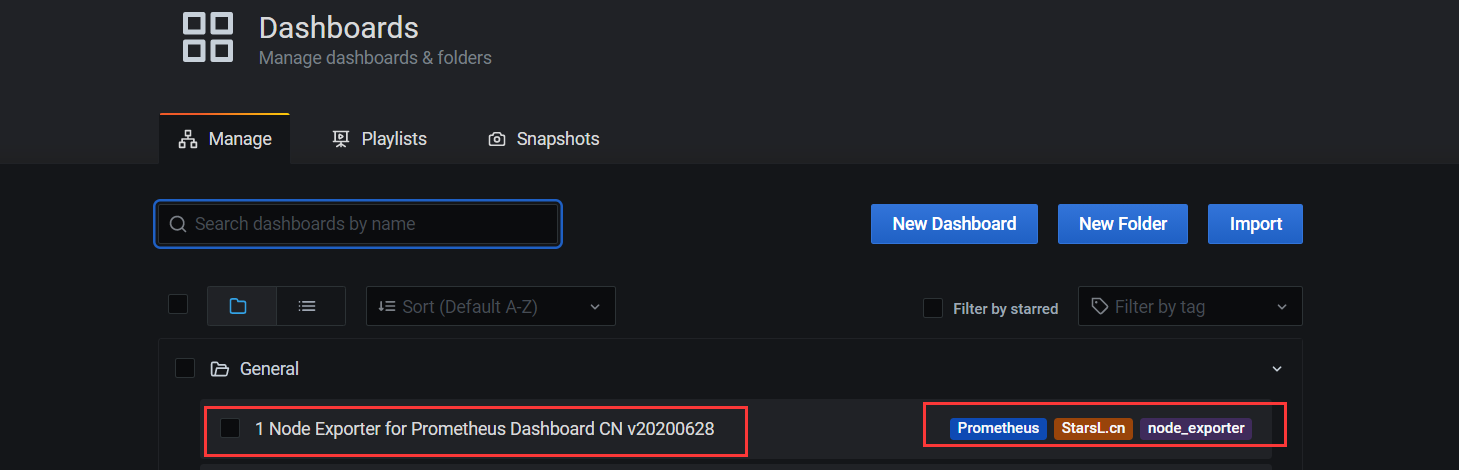
　　选择最新的模板：



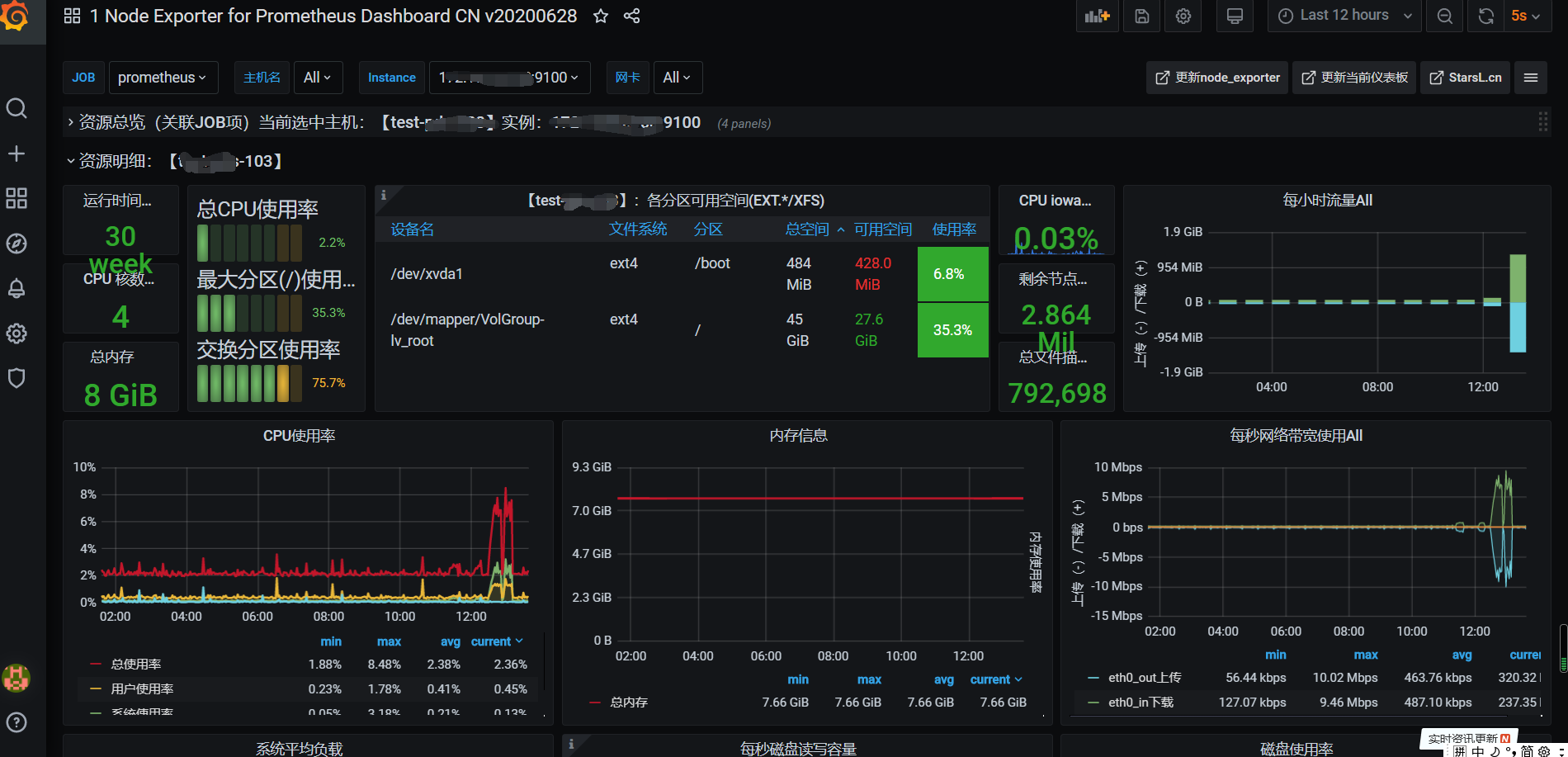
　　将id拷贝进来，如图：



　　成功之后会多一个面板：



　　点击进去就可以看到非常炫丽的图形界面：



　　总结

　　以上就是windows环境下对Linux服务器资源实时监控之Grafana实例，有兴趣的话，可以动动手实践一番。 后面会继续分享Linux下Grafana监控环境的搭建与使用。欢迎加入咋们测试开发交流群学习与进步~