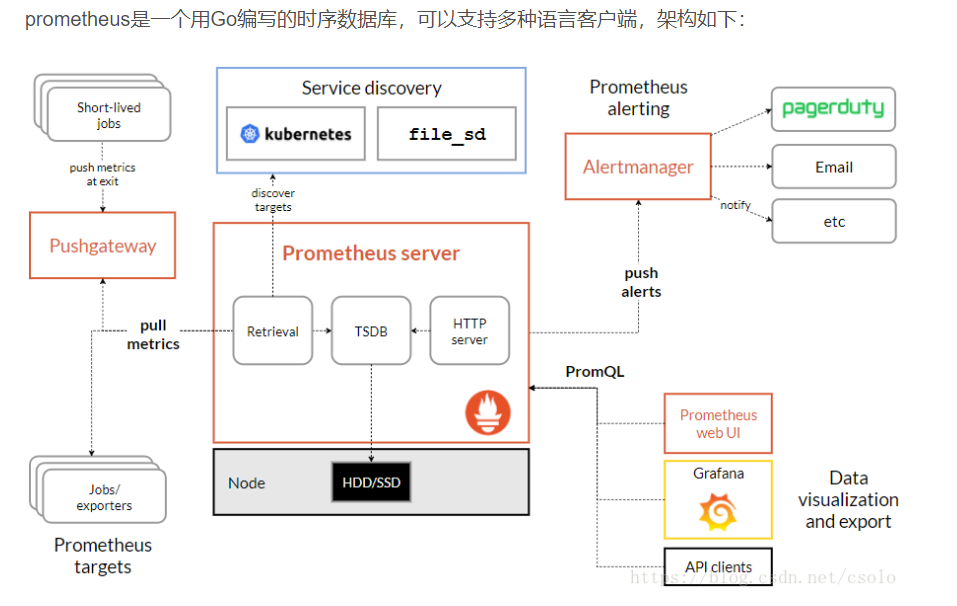
**prometheus安装**

1. 在业务层用作埋点系统:Prometheus支持多种语言（Go，java，python，ruby官方提供客户端，其他语言有第三方开源客户端）。我们可以通过客户端方面的对核心业务进行埋点。如下单流程、添加购物车流程
2. 在应用层用作应用监控系统:一些主流应用可以通过官方或第三方的导出器，来对这些应用做核心指标的收集。如redis,mysql
3. 在系统层用作系统监控:除了常用软件， prometheus也有相关系统层和网络层exporter,用以监控服务器或网络
4. 集成其他的监控:prometheus还可以通过各种exporte，集成其他的监控系统，收集监控数据，如AWS CloudWatch,JMX，Pingdom等等



安装go 语言环境

在监控服务器上安装prometheus

在被监控环境上安装export

安装grafana

1. 由于Prometheus 是用golang开发的，所以首先安装一个go环境，Go语言是跨平台，支持Windows、Linux、Mac OS X等系统，还提供有源码，可编译安装

在所有主机都安装go语言环境

#tar -C /usr/local -xzf go1.8.3.linux-amd64.tar.gz

#sed -i '$a\export PATH=$PATH:/usr/local/go/bin' /etc/profile

#source /etc/profile

#/usr/local/go/bin/go version && #go version

**在监控服务器上安装prometheus**

开始安装prometheus，首先先去官网下载对应系统的版本：<https://prometheus.io/download/>

#tar -vxf prometheus-2.3.2.linux-amd64.tar.gz -C /usr/local/

为了以后进入目录方便，做了一个软连接：

#ln -sv /usr/local/prometheus-2.3.2.linux-amd64/ /usr/local/prometheus

接着要配置一下监控的配置文件：prometheus.yml 示例

- job\_name: '10.0.3.105' # 给被监控主机取个名字，我这里直接填的IP

static\_configs:

- targets: ['10.0.3.105:9100'] # 这里填写被监控主机的IP和端口

- job\_name: '10.0.3.115'

static\_configs:

- targets: ['10.0.3.115:9100']

- job\_name: 'node\_exporter'

static\_configs:

- targets:

- '172.31.182.140:9100'

- '172.31.182.141:9100'

配置完成后，只需要运行起来这个软件：

#nohup ./prometheus --config.file＝/usr/local/prometheus/prometheus.yml [--web.enable-admin-api](https://www.cnblogs.com/dylanchu/p/11824078.html) --log.level=info --storage.tsdb.path=/usr/local/prometheus/data/ --storage.tsdb.retention.time=2h --storage.tsdb.wal-compression --storage.tsdb.max-block-duration=2h --storage.tsdb.min-block-duration=2h --web.enable-lifecycle &

--config.file：指定配置文件的路径

--storage.tsdb.path：指定Prometheus本地存储的路径。最好是指定较大分区下的目录，因为Prometheus收集的监控数据都会存储在这个目录下。

--log.level：指定日志级别，为[debug, info, warn, error]中的一种。

--web.enable-lifecycle：支持热加载新配置。从2.0开始，hot reload功能是默认关闭的，如需开启，需要在启动Prometheus的时候，添加--web.enable-lifecycle参数，热加载的方式为curl -X POST localhost:9090/-/reload

#netstat -ntulp | grep 9090

--web.enable-admin-api 这个组件很重要,配置文件prometheus.yml修改了job\_name导致grafana页面数据库收集信息报错

mysqld\_export 显示Only queries that return single...

#但如果你仅仅修改了某个job\_name（而没有修改ip），比如把web-mysql改为k8s-mysql，那么grafana界面中的singlestat panel将不能正确显示，显示“Only queries that return single...”，

这是因为singlestat只能显示一个结果，而查询语句查到了两个结果。解决方式是删除之前的数据系列

首先停止prometheus服务 #pkill prometheus

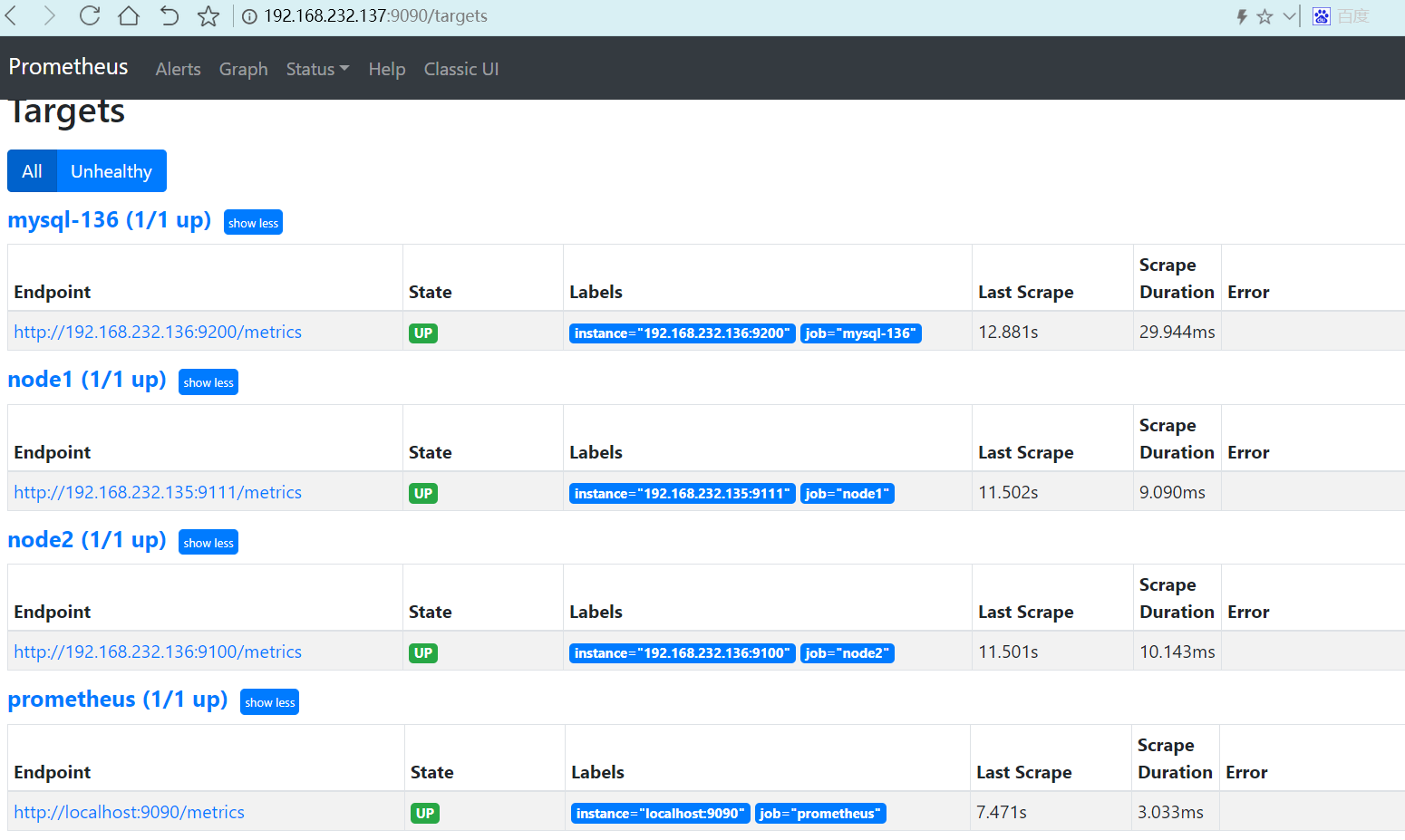
curl -X POST -g 'http://localhost:9090/api/v1/admin/tsdb/delete\_series?match[]={instance="localhost:9100"}'

如果启动prometheus没有启用--web.enable-admin-api那么这条命令显示error，坑啊

或者curl -X POST -g 'http://localhost:9090/api/v1/admin/tsdb/delete\_series?match[]={job="node\_exporter"}'

通过如下URL可以打开prometheus的自带监控界面： IP:9090，点击targets (IP:9090/targets)跳转到监控目标，这里展现了多个监控对象。红框的表示部署的prometheus

显示蓝色是因为已经安装了export，其他未安装的显示为红色



**在系统层用作系统监控exporter**

被监控主机监控OS信息

#tar -C /usr/local -xzf go1.8.3.linux-amd64.tar.gz

#sed -i '$a\export PATH=$PATH:/usr/local/go/bin' /etc/profile

#source /etc/profile

#/usr/local/go/bin/go version && #go version

#tar -xvf node\_exporter-0.16.0.linux-amd64.tar.gz -C /usr/local/

#ln -sv node\_exporter-1.0.1.linux-amd64/ node\_exporter

#nohup /usr/local/node\_exporter-0.16.0.linux-amd64/node\_exporter --web.listen-address=”:9100” &

--web.listen-address=":9111" 端口前面不用**加IP(192.168.232.135:9111)，默认9100，我改成9111**

--web.telemetry-path="/metrics"

--log.level="info" 设置日志级别 为[debug, info, warn, error]中的一种

--log.format="logger:stderr"

设置打印日志的格式，若有自动化日志提取工具可以使用这个参数规范日志打印的格式

启动后，可以看到136主机已经变成蓝色，正常up 状态。其他几台主机也通过先安装go环境，再安装export的方式，进行监控起来

**监控远程MySQL**

mysql安装

#rpm -qa | grep mariadb && #rpm -e mariadb-libs --nodeps

#tar -xvf mysql-5.7.32-linux-glibc2.12-x86\_64.tar.gz -C /usr/local

#groupadd -r mysql

#useradd -r -g mysql -s /bin/false -M mysql

#ln -sv /usr/local/mysql-5.7.32-linux-glibc2.12-x86\_64 /usr/local/mysql-5.7.32

#mkdir -p /usr/local/mysql-5.7.32/data

#sed -i '$a\export PATH=$PATH:/usr/local/mysql-5.7.32/bin' /etc/profile

#source /etc/profile

#chown -R mysql:mysql /usr/local/mysql/

#cat > /etc/my.cnf << EOF

[mysqld]

basedir=/usr/local/mysql-5.7.32

datadir=/usr/local/mysql-5.7.32/data/

pid-file=/usr/local/mysql-5.7.32/data/mysqld.pid

log-error=/usr/local/mysql-5.7.32/data/mysqld.err

socket=/tmp/mysql.sock

EOF

#/usr/local/mysql-5.7.32/bin/mysqld --initialize --user=mysql --basedir=/usr/local/mysql-5.7.32 --datadir=/usr/local/mysql-5.7.32/data MySQL初始化

#cp /usr/local/mysql-5.7.32/support-files/mysql.server /etc/init.d/mysqld

#chmod +x /etc/init.d/mysqld

#service mysqld start

#ss -ntulp | grep 3306

# a=`grep password /usr/local/mysql-5.7.32/data/mysqld.err | awk 'NR < 2 {print $NF}'`

或者#grep -r password /usr/local/mysql-5.7.32/data/mysqld.err | awk -F " " '{print $11}'

#echo ${a}

#mysql -uroot -p"${a}" -e "alter user root@localhost identified by '123456' " --connect-expired-password 修改数据库密码

grant all on \*.\* to 'root'@'%' identified by '123456';

登陆Navicat

**被管理MySQL服务器上安装mysqld\_exporter组件**

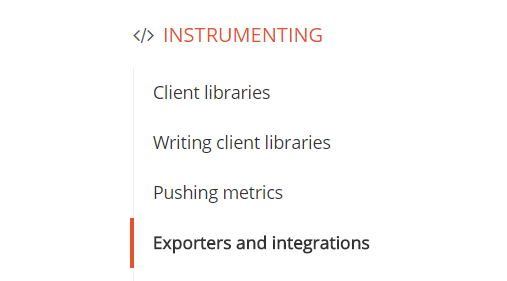
注意：OS监控安装node\_export,如果要监控OS和数据库，那么mysqld\_exporter两者都要安装

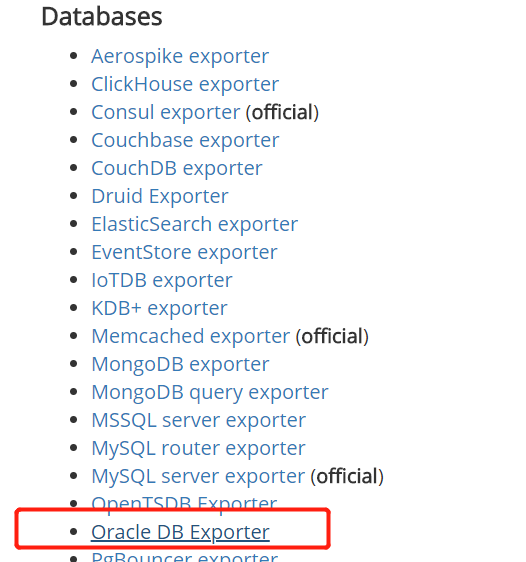
mysql监控安装https://prometheus.io/download/

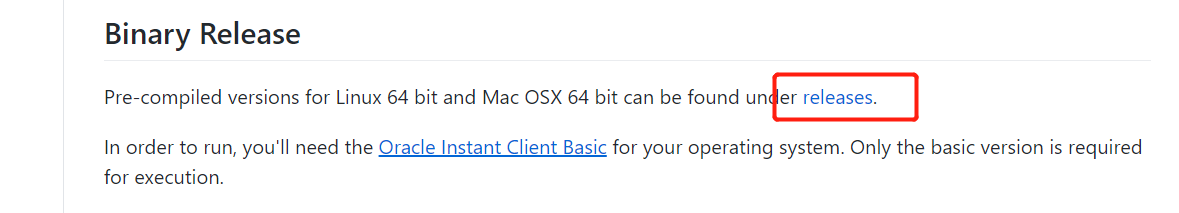
mysqld\_export,mysqld\_exporter-0.12.1.linux-**amd64**

oracle监控安装https://prometheus.io/docs/instrumenting/exporters/

oracledb\_exporter.linux-**amd64**









这里我们监控mysql数据库

#tar -xf mysqld\_exporter-0.12.1.linux-amd64.tar.gz -C /usr/local

#ln -sv /usr/local/mysqld\_exporter-0.12.1.linux-amd64 /usr/local/mysqld\_exporter

#mysql -uroot -p123456

mysql> grant replication client, process on \*.\* to “mysql\_monitor”@"%" identified by "123456"

或者mysql -uroot -p123456 -e "grant replication client,process on \*.\* to "mysql\_monitor"@"localhost" identified by '123456' "

#mysql -uroot -p123456 -e "grant replication client,process on \*.\* to "mysql\_monitor"@"localhost" identified by '123456' "

mysql> grant all on \*.\* to mysql\_monitor@"localhost" identified by ‘123456’;

mysql> flush privileges;

#vim /usr/local/mysqld\_exporter/.my.cnf

[client]

user=mysql\_monitor

password=123456

#nohup ./mysqld\_exporter --config.my-cnf=.my.cnf --web.listen-address＝"ip: 9112" &

# netstat -lnptu | grep 9200 默认端口9104,这里改成9112

**给prometheus换一个面板---grafana** https://grafana.com/grafana/download

<https://dl.grafana.com/oss/release/grafana-7.3.7-1.x86_64.rpm>

#yum localinstall grafana-5.2.3-1.x86\_64.rpm -y

接着把grafana加入到系统服务，将服务启动

#/sbin/chkconfig --add grafana-server

# service grafana-server start

启动后，打开如下URL，端口采用默认的3000

输入默认的admin/admin

进去后会要求修改密码，然后点击add datasource，选中 Prometheus 2.0 Stats后，就可以呈现如下的监控面板了

Grafana官方提供模板地址：<https://grafana.com/grafana/dashboards>

本次要导入的模板：https://grafana.com/grafana/dashboards/11074

MySQL监控做的最好当属percona公司的了

Grafana官网地址：<https://grafana.com/grafana/dashboards/7362>

