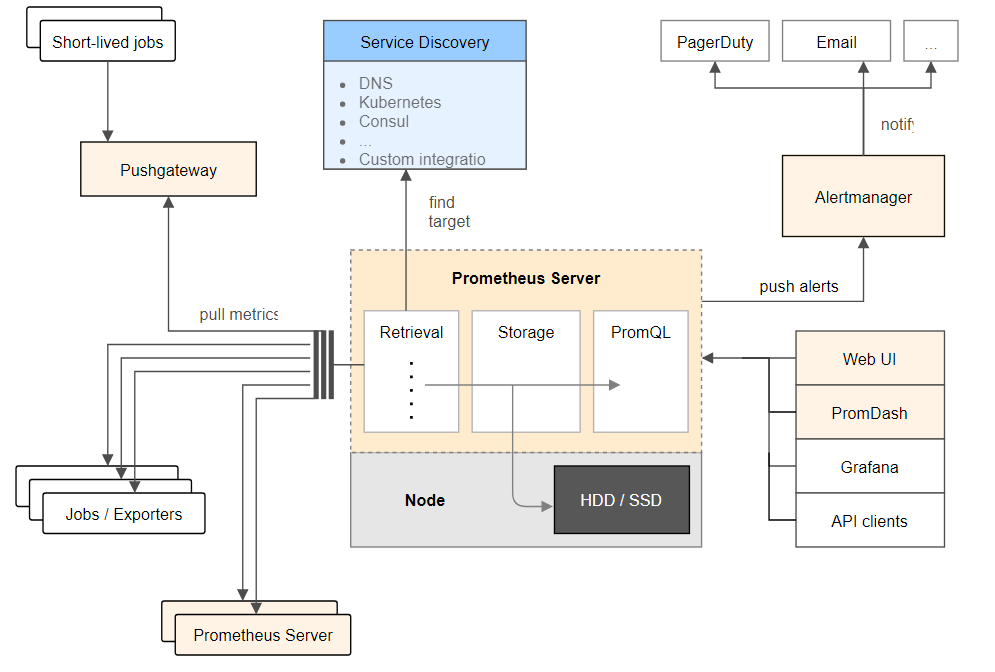
**prometheus数据采集agent编译部署**

**一、Prometheus部署架构图：**



**1、pull方式**

Prometheus采集数据是用的pull也就是拉模型,通过HTTP协议去采集指标，只要应用系统能够提供HTTP接口就可以接入监控系统，相比于私有协议或二进制协议来说开发、简单。

**2、push方式**

对于定时任务这种短周期的指标采集，如果采用pull模式，可能造成任务结束了，Prometheus还没有来得及采集，这个时候可以使用加一个中转层，客户端推数据到Push Gateway缓存一下，由Prometheus从push gateway pull指标过来。(需要额外搭建Push Gateway，同时需要新增job去从gateway采数据)

**3、安装prometheus**

cd /usr/local/src/

wget https://github.com/prometheus/prometheus/releases/download/v1.7.1/prometheus-1.7.1.linux-amd64.tar.gz

tar -zxf prometheus-1.7.1.linux-amd64.tar.gz

mv prometheus-1.7.1.linux-amd64 /data/app/

ln -s /data/app/prometheus-1.7.1.linux-amd64/ /data/app/prometheus

**4、启动prometheus**

./prometheus -config.file=prometheus.yml

1. **prometheus数据库指标采集agent**
2. **编译环境go环境准备**

下载go(我的当前目录是/data/work)

$wget <https://studygolang.com/dl/golang/go1.10.1.linux-amd64.tar.gz>

$tar -xvf go1.10.1.linux-amd64.tar.gz

设置环境变量

$vim /etc/profile

添加

export GOROOT=/data/work/go

export GOPATH=/data/work/gopath

export PATH=$PATH:$GOROOT/bin:$GOPATH/bin

export GO111MODULE=on

export GOPROXY=https://goproxy.io

1. **编译及运行mysql\_expoerter**

git clone [git@github.com:percona/mysqld\_exporter.git](mailto:git@github.com:percona/mysqld_exporter.git)

进入仓库目录

go build mysqld\_exporter.go

即可在当前路径生成mysqld\_exporter二进制文件了

./mysql\_exporter -h查看帮助

*$vim .my.cnf*   
[client]  
user=exporter  
password=Cds20180808!

启动agent ./mysqld\_exporter --config.my-cnf="/root/.my.cnf"

1. **编译及运行mongodb\_expoerter**

git clone [git@github.com:percona/mongodb\_exporter.git](mailto:git@github.com:percona/mongodb_exporter.git)

进入仓库目录

go build mongdb\_exporter.go

即可在当前路径生成mongdb\_exporter二进制文件了

./mongdb\_exporter -h查看帮助

启动agent ./mongodb\_exporter --mongodb.uri=mongodb://admin:admin@127.0.0.1:27017/admin