

PDP-12: Cloud-native PNDA

Take the work contributed to Red PNDA for containerisation and expand it to deliver a fully cloud-native PNDA.

PNDA Containerization Current State (red-pnda)

Red-pnda provisions a miniman set of PNDA components to enable developers writing apps targeted at the full PNDA stack. One of its last features is to deploy red-pnda as a set of docker containers to optimize resources compare to the monolithic Virtual Machine deployment.

A set of Dockerfiles are included in the red-pnda github repo to build the corresponding containers for each PNDA deployment unit. Dockerfiles currently admit a "version" build-arg to download the corresponding PNDA component release from its github repo.

From the identified PNDA Deployment units we Highlight the dockerized ones.

PNDA-specific stuff

- deployment manager (1 process). PARTIALLY_DONE. A docker image exists that only support jupyter-notebooks deployment. Deployment Manager should be re-thinked since it leverage on scp and systemd/sysv service definitions (not really cloud-friendly stuff).
- package repository (1 process). DONE.
- platform test agent & jmx proxy (2 processes). DONE. A container image for each process.
- console front end & nginx (2 processes) DONE. There is only 1 process (nginx) serving the files of console-frontend.
- console back end & graphite & redis (3 processes) PARTIALLY DONE. Console backend Image and redis Image. Graphite is not done.
- data service (1 process). DONE
- data curator agent (aka hdfs cleaner) (1 process) DONE

3rd party stuff

- kafka manager (1 process). DONE
- jupyter & jupyter proxy & livy (3 or more processes). a jupyter image is DONE. (We have also experimented with a jupyterhub docker image for multiple agent, but it is incomplete)
- gobblin & gobblin modules (1 process). PARTIALLY_DONE. A gobblin docker image was created but not tested.
- kafka & kafkat (1 process). DONE - We use official Kafka Images provided by Confluent.
- zookeeper (1 process). DONE - We use Zookeeper Images provided by Confluent.
- hadoop (many processes). DONE for hdfs-namenode, hdfs-datanode, hbase-master, hbase-region.
- opentsdb (1 process). DONE
- grafana (1 process). DONE - We use official Grafana Docker Image.
- ELK (2 processes + see logstash below). NOT_DONE

Plan

- Define a set of common rules for the creation of Dockerfile in each pnda component.
 - Set uniform environment variable names for container configuration (e.g., hdfs namenode URI must be configured with the same environment variable name in all components).
- Integrate the corresponding Dockerfile for each deployment unit into its repo. Dockerfile must be modified to get source code directly from its repo itself (COPY instruction) and extract the version maybe with a git describe.
- Integrate Apache Spark with kubernetes (supported by Spark 2.3 release).
- Dataset Persistence in Cloud. Study cloud storage alternatives to HDFS (S3, minio for on-prem). Improve Data-mgmt to work with Cold/Warm storage paradigms (S3 vs Glacier ...).
- Separate Infrastructure Orchestration from Container Orchestration.
 - Infrastructure Orchestration can be further divided into infrastructure provision and Kubernetes deployment. Candidates are Terraform, Ansible, SaltStack and for kubernetes deployment Kubernetes-kops, kube-adm.
 - Candidate tools to automate the orchestration of the multiple components/microservices of PNDA over Kubernetes are Ansible or Kuberenetes specific tools such as Kubernetes Operators or Helm Charts.
- Platform Monitoring. Study the Integration of [Prometheus](#).