CODA-19 Deployment Guide

Authors:

Louis-Antoine Mullie
Bruno Lavoie

Version 0.2.0
November 2020

Table of content

Deployment steps

1. Obtain required accesses

- Request the following accesses (e-mail louis.mullie@gmail.com):
    - GitHub organization: https://github.com/CODA-19
    - DockerHub organization: https://hub.docker.com/repository/docker/coda19
    - AirTable terminology server: https://airtable.com/tblVaHH9uS83GcjqB/viw8XsmYG3hlDsB8G
- Obtain remote desktop connection token for VALERIA
    - To allow VALERIA to connect via SSH to your VMs for maintenance
    - Need to grant site access to VALERIA via RTSS

2. Provision virtual machines

Please note that all these specifications are an estimate and can change in future.

- Provision red, orange, and green virtual machines
    - **Red VM specs:**
        - 4 cores and 8GB RAM
        - Disks: 500 GB
          Or simply enough to hold temporary extractions during anonymization jobs, if needed.
    - **Orange VM specs:**
        - 4 cores and 8GB RAM
        - Disks:
            - Operating System - 100 GB
            - DATA - 1 TB
    - **Green VM specs:**
        - 8 cores and 16 GB RAM
        - GPU: RTX 2080 or better NVIDIA-compatible GPU
        - Disks:
            - Operating System - 100 GB
            - DATA - 1 TB

NOTE: It is important to dedicate a separate block device (virtual disk) for data. Simply because it is best practice and it ensures an easier and more homogenous deployment across sites.

3. Set up RED VM

- Identify location of information in source systems corresponding to the following templates:
  https://github.com/CODA-19/fhir-templates

- Obtain site-specific terminology dictionary via: https://github.com/CODA-19/terminology

- Develop scripts to convert source system data to FHIR JSON, using terminology dictionary to insert appropriate codes:
  Example: https://github.com/CODA-19/csv-to-fhir

- Develop scripts to extract image data in HDF5 format

4. Networking requirements

The hub is located at VALERIA (Université Laval) and must be accessible from your ORANGE and GREEN VMs by HTTPS protocol. To reach the hub, your site must permit outbound HTTPS (TCP/443) connections from your VMS to these CIDR blocks:

- 132.203.231.8/29
- 132.203.189.10/32

Aidbox is a licensed product that requires access to a license server located at license.aidbox.io (34.98.76.51). We validated with the provider and this IP is mostly guaranteed to be stable.

If you operate in stringent environments that need to use a proxy server to access the internet, you'll need to provide this to the VALERIA team (see next step). Communication from green VMs to hub uses websockets and we must ensure that it works. Some proxy and firewalls don't manage this type of communication correctly. If it does not work, see with your site networking team to permit this kind of communication.

Also, because deployment scripts are fetched from GitHub you need to whitelist this website access.

4.1. Proxy whitelisting

Some sites want to manage proxy accesses with strict whitelists.

Here are key sites that you must be accessed:

Site	Purpose
http://mirror.centos.org http://centos.mirror.iweb.ca/ http://fedora-epel.mirror.iweb.com/	Installing software packages (yum)
https://github.com https://github-releases.githubusercontent.com	Pulling deployment code. Pulling various artifacts (ie: node_exporter)
https://license.aidbox.io	Validating Aidbox Licence
https://dl.min.io	Downloading Min.IO artifacts.
https://hub.docker.com https://production.cloudflare.docker.com https://registry-1.docker.io https://auth.docker.io	Pulling Docker images for Aidbox software
https://heartbeat.hub.coda19.com	Sending heartbeat to control plane (hub).

5. Gather technical details and submit to VALERIA

Before proceeding to next steps, please gather required technical information and send it to VALERIA <info@valeria.science>.

1. NTP servers:	
2. PROXY server:	
3. SMTP server:	
4. Block devices:	

1. Necessary to keep all servers time in sync and provides accuracy while doing distributed analysis. Using more than one source is a best practice. If no one is provided, deployment scripts will configure default sources outside your organization, outbound NTP traffic must be permitted for this to work correctly.
2. Necessary only if no direct internet access is permitted.
3. Necessary to send emails, not used for first deployment.
4. Execute `lsblk` on each virtual machine and paste output. If `lsblk` utility is not found install it first by doing `yum install -y util-linux`.

Following this, the VALERIA team will let you know when to proceed to next steps and provide you your vault password that will enable you to bootstrap your environment.

6. Bootstrap your site virtual machines

On the orange and green virtual machines, deploy Ansible environment built by VALERIA, simply follow bootstrap instructions at:

https://github.com/CODA-19/deploy-scripts

After a few minutes you'll be asked to provide this information:

- **Site ID:** this uniquely identifies your site and instructs scripts which vault to use.
- **Vault password:** needed to decrypt your site's vault.
- **Proxy:** if no direct internet access, needed for bootstrapping process, not used afterwards.
- **VM role:** needed for scripts to which VM to deploy (orange/green).

After that, a cron job executes each 10 minutes by pulling latest scripts and executes it by using an ansible-pull.

7. Set up data import

- Package FHIR in ndjson.gz format
- Import FHIR JSON into AidBox
- Create Cron job for data refreshes

8. Test AidBox functionality

- Explore FHIR data from AidBox console

Appendix A - Sites codification

This codification is necessary to have seamless integration across participating sites, simplify management and automation.

Name	Acronym	Site ID
VALERIA	HUB	100
Centre Hospitalier de l'Université de Montréal	CHUM	110
McGill University Health Centre	MUHC	111
Jewish General Hospital	JGH	112
Hôpital Maisonneuve-Rosemont	HMR	113
Hôpital Sacré-Coeur de Montréal	HSCM	114
Centre Hospitalier Universitaire de Québec	CHUQ	115
CISSS Chaudière-Appalaches	CISSS-CA	116
Centre Hospitalier Universitaire de Sainte-Justine	CHUSJ	117
The Ottawa Hospital	OH	118

Appendix B - Virtual machines services and ports

Orange VM

MinIO (S3) - TCP/9000 (local)
Aidbox - TCP/8888 (local)

Green VM

COMING SOON

Appendix B - Getting site specific credentials

Once the bootstrap process is terminated, you'll be able to consult your site credentials. By opening the vault with your vault password.

The vault is site specific, you only need to consult it once on either the green or orange virtual machine.

Important files are:

- The vault password file: `/etc/ansible/vault.pass`
- Your site's vault file: `/opt/coda19/deploy-scripts-pull/ansible/vaults/vault.<site_id>`

To you vault content execute this command, by carefully replace the site_id placeholder:

```
/opt/coda19/venv-ansible/bin/ansible-vault
  view \
  --vault-password-file /etc/ansible/vault.pass \
  /opt/coda19/deploy-scripts-pull/ansible/vaults/vault.<site_id>
```

MinIO Credential

Locate access and secret keys, defined as variables `ul_minio_server_access_key` and `ul_minio_server_secret_key`.

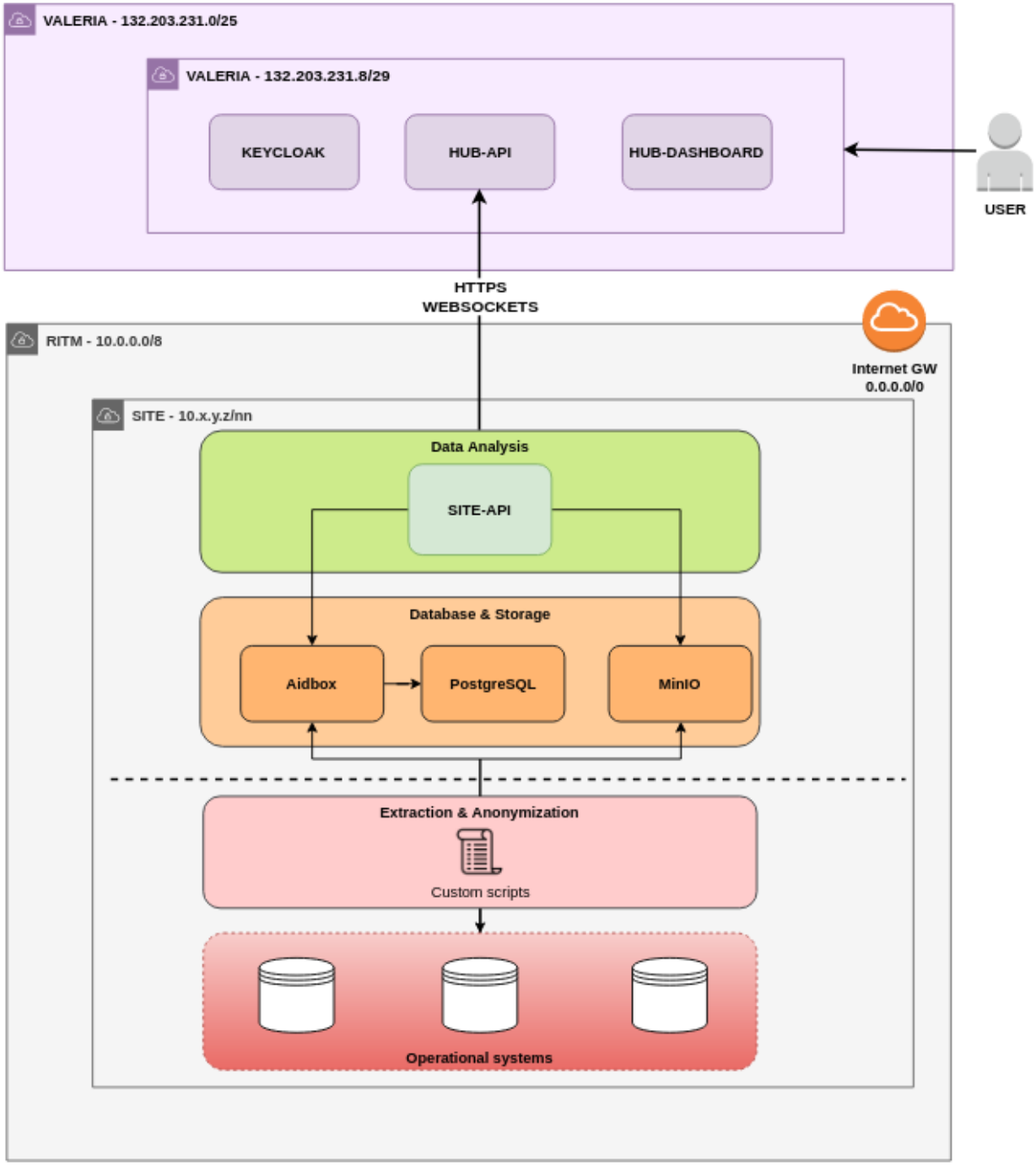
AidBox.one Credentials

Aidbox comes with an administrator and a client account with «admin» and «client» usernames respectively. Corresponding passwords are defined as `ul_aidbox_conf_admin_password` and `ul_aidbox_conf_client_secret` variables.

Appendix C - Architecture diagrams

Overall architecture

An arrow denotes «connects to», all other connection path are denied.



Websockets communication

0. Site initialize channel to HUB-API
1. Server emits request to SITE-API
2. SITE-API emits response to HUB-API socket server
3. HUB-API socket server stores request in WebSocket Bus
4. Initial request pools bus for awaiting response