Sesión 6 : ansible

ansible

become

```
- name: servicio
  service:
    name: httpd
    state: started
  become: yes
```

```
- name: servicio
  service:
    name: httpd
    state: started
  become: yes
  become_user: apache
```

adduser

add-user.yaml

```
- hosts: all
  tasks:
    - include: add-user-debian.yaml
      when: ansible_os_family == 'Debian'
    - include: add-user-redhat.yaml
      when: ansible_os_family == 'RedHat'
```

add-user-debian.yaml

```
    - name: add user
      user:
        name: operador
        comment: nuestro usuario operador
        shell: /bin/bash
        home: /home/operador
        groups: sudo
        append: yes
      become: true

    - name: Set authorized key taken from file
      authorized_key:
        user: operador
        state: present
        key: "{{ lookup('file', '/home/devops/.ssh/id_ed25519.pub') }}"
      become: true
```

add-user-redhat.yaml

```
    - name: check centos correct suoders line
      lineinfile:
        path: /etc/sudoers
        state: present
        regexp: '^%wheel\s'
        line: '%wheel ALL=(ALL) NOPASSWD: ALL'
      become: true
      when: ansible_os_family == 'RedHat'
    - name: add user
      user:
        name: operador
        comment: nuestro usuario operador
        shell: /bin/bash
        home: /home/operador
        groups: wheel
        append: yes
      become: true

    - name: Set authorized key taken from file
      authorized_key:
        user: operador
        state: present
        key: "{{ lookup('file', '/home/devops/.ssh/id_ed25519.pub') }}"
      become: true
```

adduser2

otra-opcion.yaml

```
- hosts: "all"
  sudo: true
  vars:
    users:
    - "operador1"
    - "operador2"
    - "operador3"
  tasks:
  - name: "crear grupo operador"
    become: yes
    group:
      name: operador
      state: present

  - name: "Create user accounts"
    user:
      name: "{{ item }}"
      groups: "operador"
    with_items: "{{ users }}"

  - name: "Add authorized keys"
    authorized_key:
```

```
      user: "{{ item }}"
      key: "{{ lookup('file', 'files/'+ item + '.pub') }}"    #
files/operador1.pub files/operador2.pub ...
    with_items: "{{ users }}"

  - name: "Allow admin users to sudo without a password"
    lineinfile:
      dest: "/etc/sudoers"
      state: "present"
      regexp: "^%operador"
      line: "%operador ALL=(ALL) NOPASSWD: ALL"
    become: yes

  - name: Install apache httpd but avoid starting it immediately
(state=present is optional)
     package:
      name: httpd
     when: ansible_os_family == 'RedHat'
     become: yes
```

mysql install

mysql-install.yaml

```
#        Ansible needs python-mysqldb
- name: Install MySQL
  apt: pkg={{item}} state=latest update_cache=false
  register: ispconfig_install_step1
  with_items:
    - pwgen
    - mysql-client
    - mysql-server
    - python-mysqldb

#        Requires a system with pwgen, included in our base system
- name: Generate MySQL Random Password
  command: /usr/bin/pwgen -s 16
  register: mysql_root_password

- name: update mysql root password for all root accounts
  mysql_user: name=root host={{ item }}
password={{mysql_root_password.stdout}}  update_password=always state=present
  with_items:
    - "{{ inventory_hostname }}"
    - 127.0.0.1
    - ::1
    - localhost
  notify:
    - Restart MySQL

- name: copy my.cnf file with root password credentials to /root/.my.cnf
  template: src=my.cnf dest=/root/.my.cnf owner=root mode=0600
```

```
- name: Configure MySQL to listen on *:3306
  replace: dest=/etc/mysql/my.cnf regexp='bind-address' replace='#bind-
address'
```

- https://docs.ansible.com/ansible/latest/modules/mysql_user_module.html?highlight=mysql

roles

- https://docs.ansible.com/ansible/latest/user_guide/playbooks_reuse_roles.html#role-directory-structure
- https://github.com/kpeiruza/ansible

```
- hosts: all
  roles:
    - rol1
    - rol2
    - rol3
```

add-user-with-rol.yaml

```
- hosts: all
  roles:
    - add-user-rol
```

add-user-rol/tasks/main.yaml

```
- include: add-user-debian.yaml
  when: ansible_os_family == 'Debian'
- include: add-user-redhat.yaml
  when: ansible_os_family == 'RedHat'
```

add-user-rol/tasks/add-user-debian.yaml

```
- name: add user
  user:
    name: operador
    comment: nuestro usuario operador
    shell: /bin/bash
    home: /home/operador
    groups: sudo
    append: yes
  become: true

- name: Set authorized key taken from file
  authorized_key:
    user: operador
    state: present
# la clave pública en el directorio files del rol
    key: "{{ lookup('file', '{{ role_path }}/files/id_ed25519.pub') }}"
  become: true
```

add-user-rol/tasks/add-user-redhat.yaml

```
- name: check centos correct suoders line
  lineinfile:
    path: /etc/sudoers
    state: present
    regexp: '^%wheel\s'
    line: '%wheel ALL=(ALL) NOPASSWD: ALL'
  become: true
  when: ansible_os_family == 'RedHat'
- name: add user
  user:
    name: operador
    comment: nuestro usuario operador
    shell: /bin/bash
    home: /home/operador
    groups: wheel
    append: yes
  become: true

- name: Set authorized key taken from file
  authorized_key:
    user: operador
    state: present
    key: "{{ lookup('file', lookup('env','HOME')+'/.ssh/id_ed25519.pub') }}"
  become: true
```

git

- HEAD : RAMA+COMMIT
- HEAD~2 : 2 commits atrás del HEAD actual
- comandos
    - `git clone`
    - `git add .`
    - `git log`
    - `git commit [-m «mensaje»]`
    - `git push`
    - `git rm <fichero>`
    - `git tag <version o tag> <commit>`
    - `git checkout [-b] <rama>` : crear o cambiar de rama
    - `git diff <tag/commit> <commit> [<fichero>]`
    - `git init`
    - `git remote set-url origin <direccion>`

docker-machine

instalar docker en máquinas de diferentes proveedores:

- máquina remota
- digital-ocean
- amazonec2

info:

- https://docs.docker.com/machine/
- https://docs.docker.com/machine/get-started/

comandos:

- `docker-machine create –driver digitalocean –digitalocean-access-token xxxxx docker-sandbox`
- `docker-machine create –driver virtualbox default`
- `docker-machine create –driver amazonec2 –amazonec2-access-key AKI******* –amazonec2-secret-key 8T93C******* aws-sandbox`
- ```
  docker-machine create \
    --driver generic \
    --generic-ip-address=203.0.113.81 \
    --generic-ssh-key ~/.ssh/id_rsa \
    vm
  ```

boot2docker.iso : imagen usada en el driver por defecto para lanzar un docker. Usada en windows/mac

https://docs.yugabyte.com/latest/deploy/docker-swarm/

Prometheus

- https://prometheus.io : motor de métricas
- monitoring stack
- federados + árbol
- alertar por umbrales → alertmanager
- recepción de métricas de monitorización → pushgateway
- se usa Grafana para acceder a los datos
- origen de las métricas:
  - node_exporter
  - cadvisor prometheus
  - docker metrics
- despliegue en docker swarm: https://github.com/stefanprodan/swarmprom

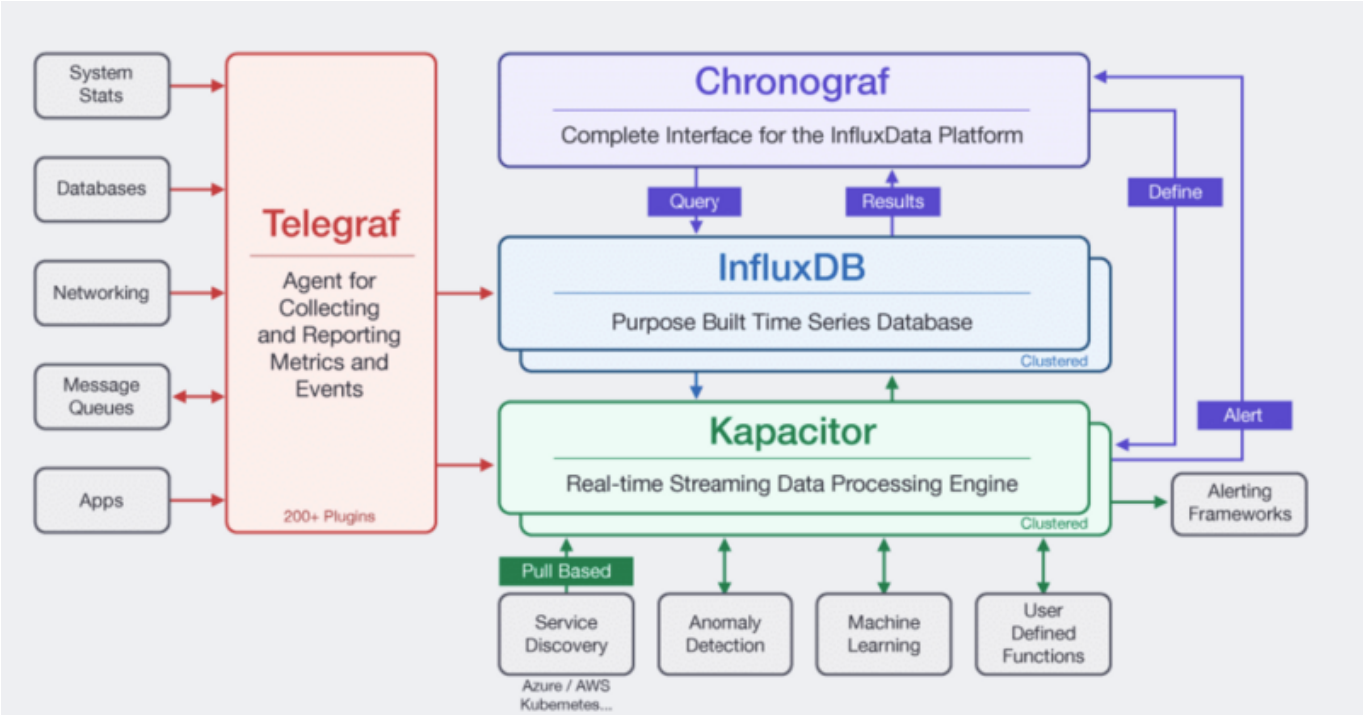
otros

- `cat /bin/bash > /dev/dsp`
- https://ed25519.cr.yp.to/
- vim-nox
- no confirmar claves de sistemas remotos:

  ~/.ssh/config

  ```
  host *
      StrictHostKeyChecking no
  ```

- TICK : Telegraf + InfluxDB + Chronograf + Kapacitor
  ```



- https://hackertyper.net/