Kriten

API driven automation

working alongside: evolvere TECHNOLOGIES

10 years of Network Automation

Many platforms, modules and products now available

No single solution, even within an organisation.

Custom code necessary to automate networks.

evolvere
TECHNOLOGIES

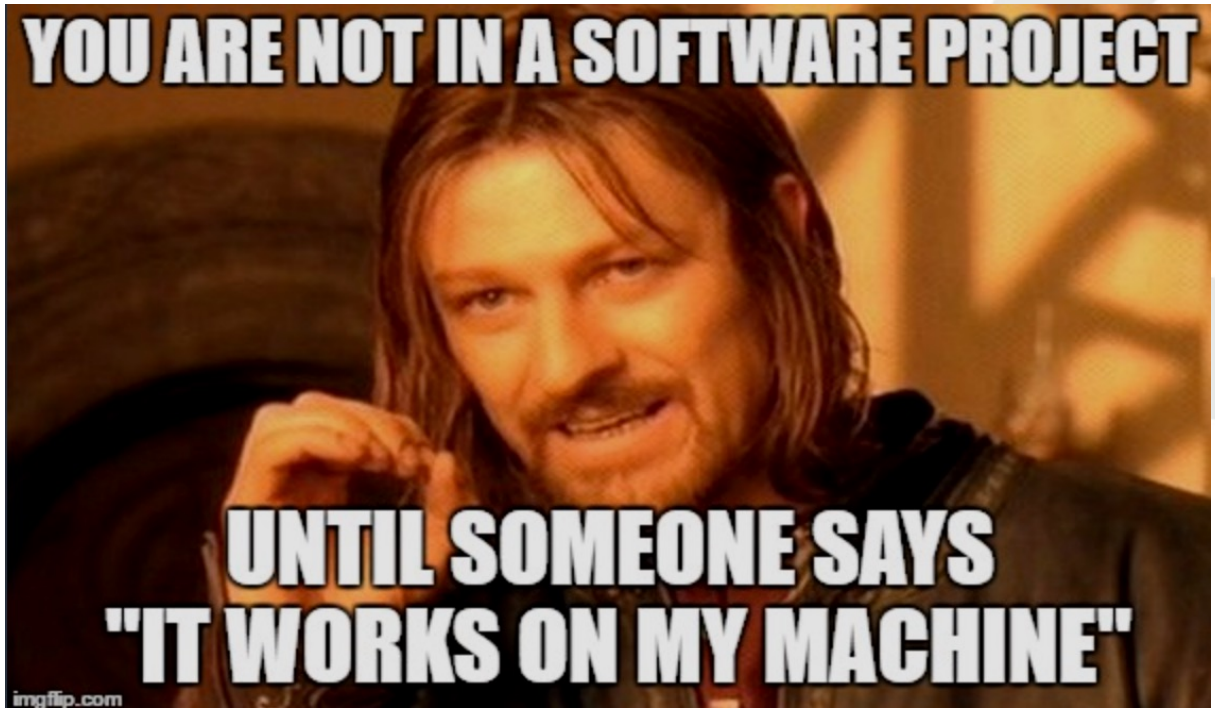
Common Problems

Where to run automation code?

How to make it available?

How to implement authentication/authorisation.

How to protect system/priviledged accounts?



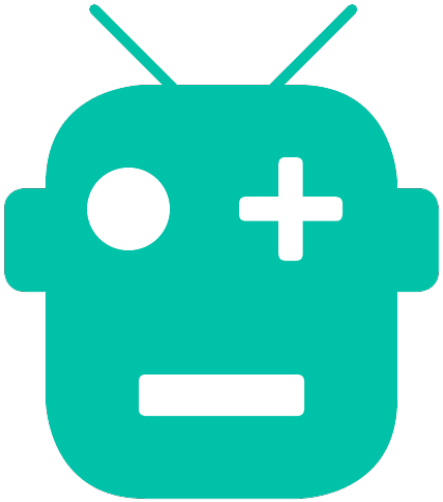
evolvere
TECHNOLOGIES

Kriten – The API-driven runtime environment

Kriten is a cloud-native code execution platform.

- Build and test your automation code.
- Build a container with your packages installed.
- Configure runner and task in Kriten.

You can now send consumers the Kriten task URL, which is all they need to run your automation code.



https://kriten.io/

evolvere TECHNOLOGIES

The show version example

You have proudly written an Ansible playbook to issue the “show version” command to a set of devices.

```yaml
---
- name: Read extra_vars
  hosts: localhost
  gather_facts: no
  vars:
    input_vars: "{{ lookup('ansible.builtin.env', 'EXTRA_VARS')}}"

  tasks:
    - name: Reading target hosts from input vars and storing as localhost fact
      set_fact:
         target_hosts: "{{ input_vars['target_hosts'] }}"

- name: Show Version
  hosts: "{{ hostvars['localhost']['target_hosts'] }}"
  gather_facts: false
  connection: network_cli

  tasks:
    - name: Run 'show version' command on remote devices
      eos_command:
        commands: "show version"
      register: show_version

    - name: debug
      debug:
        #msg: "{{ show_version.stdout.0 }}"
        var: show_version.stdout.0
```

https://github.com/kriten-io/kriten-examples/tree/main/netbox-ansible

evolvere TECHNOLOGIES

Run the playbook locally

```
python3 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
source venv/bin/activate
export NETBOX_TOKEN=a08684c4649c091e201f1eeb4eb5c18123123123
export NETBOX_API=http://192.168.10.55:8000/
export EXTRA_VARS='{"target_hosts": "all"}'
ansible-playbook -i netbox_inv.yml show_version.yml -e EXTRA_VARS
```

```
PLAY [Read extra_vars] *********************************************************************

TASK [Reading target hosts from input vars and storing as localhost fact] ******************
ok: [localhost]

PLAY [Show Version] ************************************************************************

TASK [Run 'show version' command on remote devices] ****************************************
ok: [evo-eos01]

TASK [debug] *******************************************************************************
ok: [evo-eos01] =>
  show_version.stdout.0: |-
    Arista vEOS-lab
    Hardware version:
    Serial number: CA0DFC7A91FA31E6C14199440F9B86C6
    Hardware MAC address: 0050.56f6.9bb9
    System MAC address: 0050.56f6.9bb9

    Software image version: 4.27.0F
    Architecture: x86_64
    Internal build version: 4.27.0F-24308433.4270F
    Internal build ID: 9088210e-613b-47db-b273-7c7b8d45a086
    Image format version: 1.0

    Uptime: 5 days, 23 hours and 13 minutes
    Total memory: 4002360 kB
    Free memory: 2836760 kB

PLAY RECAP *********************************************************************************
evo-eos01                  : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
localhost                  : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

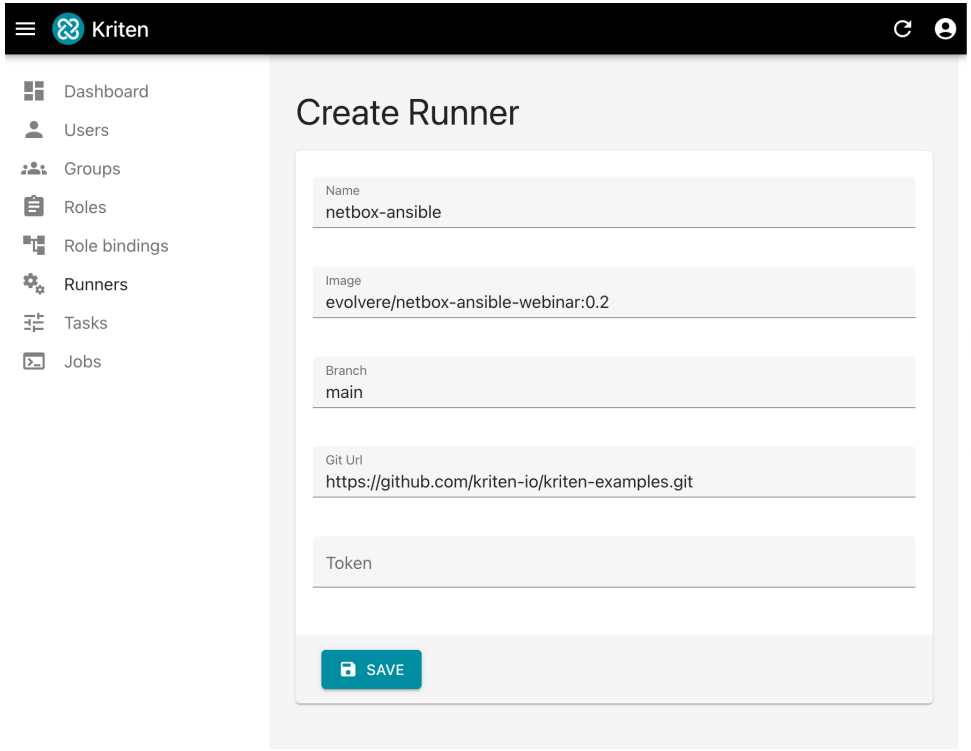
evolvere TECHNOLOGIES

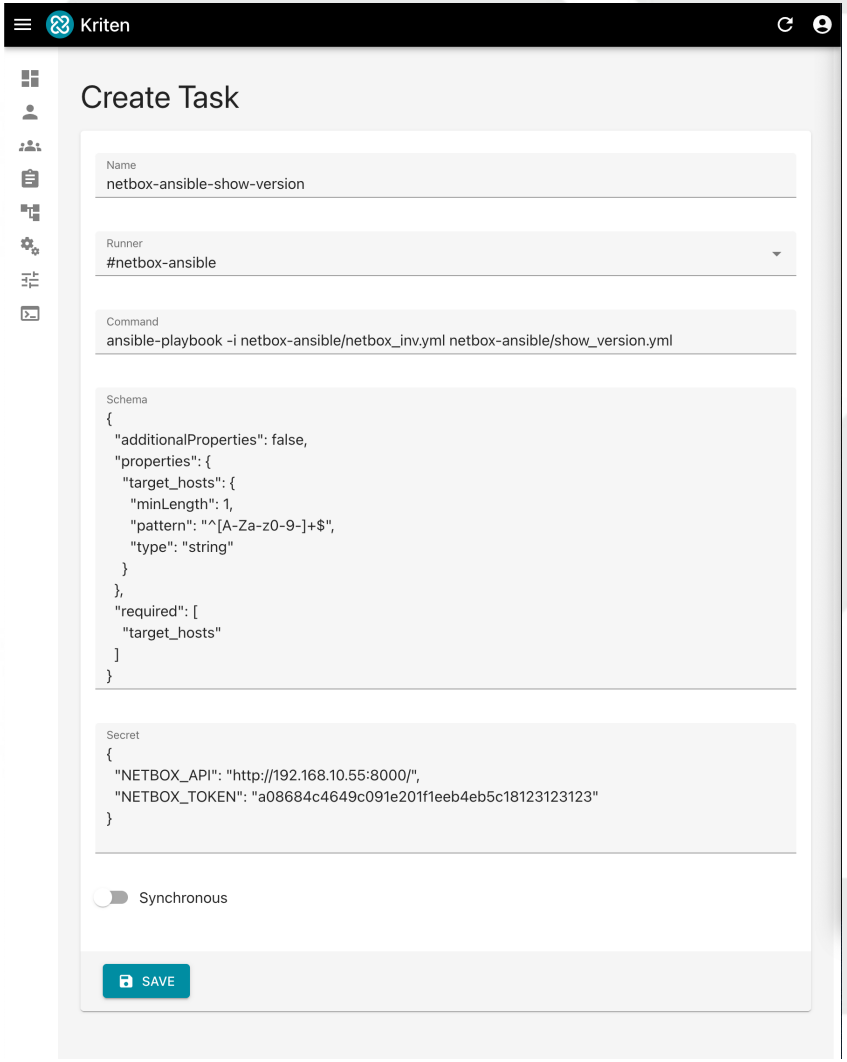
Build a container image

```
FROM python:3.9-slim
RUN mkdir /app
COPY requirements.txt /app
WORKDIR /app
RUN pip3 install -r requirements.txt
ENV ANSIBLE_HOST_KEY_CHECKING=False
CMD ["python", "ansible-playbook -i netbox_inv.yml show_version.yml"]
```

```
docker build -t evolvere/netbox-ansible-webinar:0.2 .
docker push evolvere/netbox-ansible-webinar:0.2
```

Create runner and task





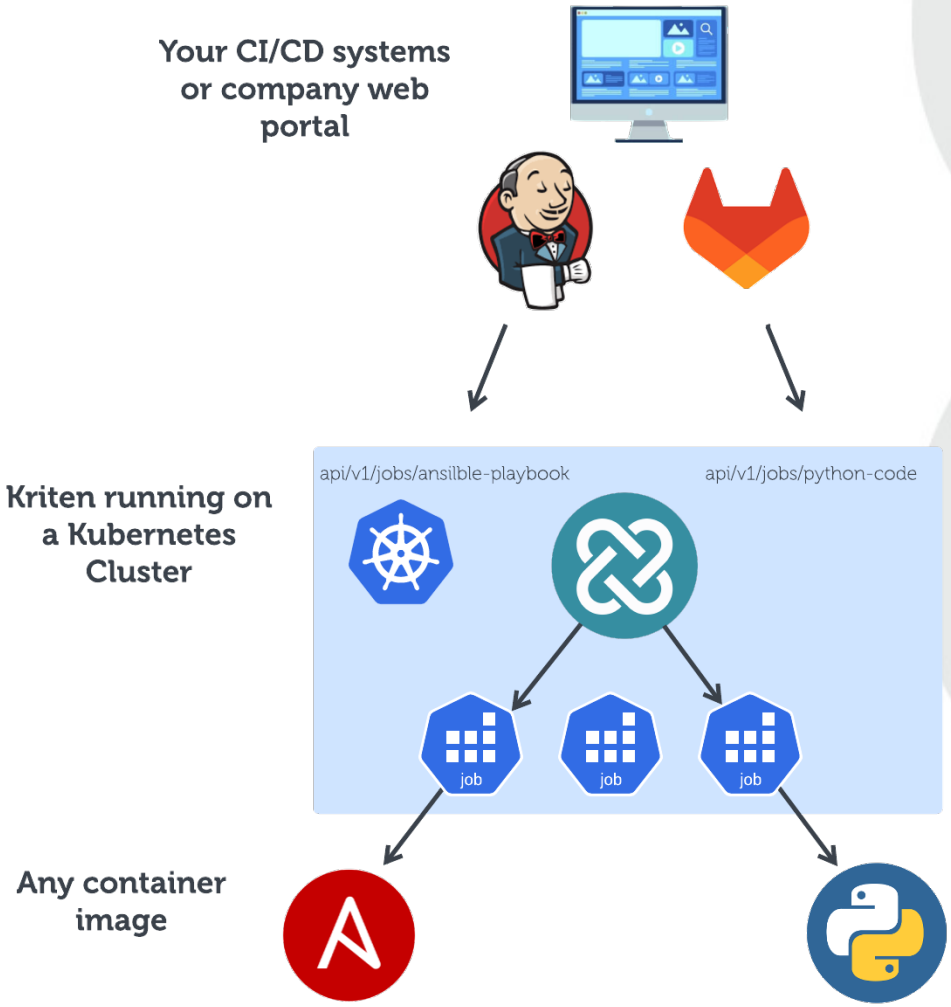
evolvere
TECHNOLOGIES

Run the playbook via Kriten API call

```
export KRITEN_URL=http://kriten.192.168.10.102.nip.io

curl -c ./token.txt $KRITEN_URL'/api/v1/login' \
--header 'Content-Type: application/json' \
--data '{
  "username": "root",
  "password": "root",
  "provider": "local"
}'
{"token":"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6InJvb3QiLCJ1c2VyX2lkIjoiZTc4ZDk3NDYtZTNiZi00MDdiLTk1NmYtNWQwNzg3ZGEyZTczIiwicHJvdmlkZXIiOiJsb2NhbCIsImV4cCI6MTcxMjc0ODgxNn0.oah0MVWXRyfYuF6dsJneV7bOq8HyEfnSo1hf-vQvjYY"}

curl -b ./token.txt $KRITEN_URL'/api/v1/jobs/netbox-ansible-show-version' \
--header 'Content-Type: application/json' \
--data '{
  "target_hosts": "evo-eos01"
}'
{"id":"netbox-ansible-show-version-7vdlf","msg":"job created successfully"}[evolvere@evo-management-05 ~]$

curl -b ./token.txt $KRITEN_URL'/api/v1/jobs/netbox-ansible-show-version-7vdlf' \
--header 'Content-Type: application/json'
{"id":"netbox-ansible-show-version-7vdlf","owner":"root","startTime":"Wed Apr 10 10:34:39 UTC
2024","completionTime":"Wed Apr 10 10:34:56 UTC 2024","failed":0,"completed":1,"stdout":"\nPLAY [Read
extra_vars] ***********************************************************\n\nTASK [Reading target hosts from input
vars and storing as localhost fact] ******\nok: [localhost]\n\nPLAY [Show Version]
***********************************************************\n\nTASK [Run 'show version' command on remote
devices] ***************************\nok: [evo-eos01]\n\nTASK [debug]
*********************************************************************\nok: [evo-eos01] =\u003e {\n
\"show_version.stdout.0\": \"Arista vEOS-lab\\nHardware version: \\nSerial number:
CA0DFC7A91FA31E6C14199440F9B86C6\\nHardware MAC address: 0050.56f6.9bb9\\nSystem MAC address:
0050.56f6.9bb9\\n\\nSoftware image version: 4.27.0F\\nArchitecture: x86_64\\nInternal build version:
4.27.0F-24308433.4270F\\nInternal build ID: 9088210e-613b-47db-b273-7c7b8d45a086\\nImage format version:
1.0\\n\\nUptime: 5 days, 23 hours and 52 minutes\\nTotal memory: 4002360 kB\\nFree memory: 2837448
kB\"\n}\n\nPLAY RECAP
*********************************************************************\nevo-eos01                  : ok=2
changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0   \nlocalhost                  :
ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0   \n\n","json_data":null}
```

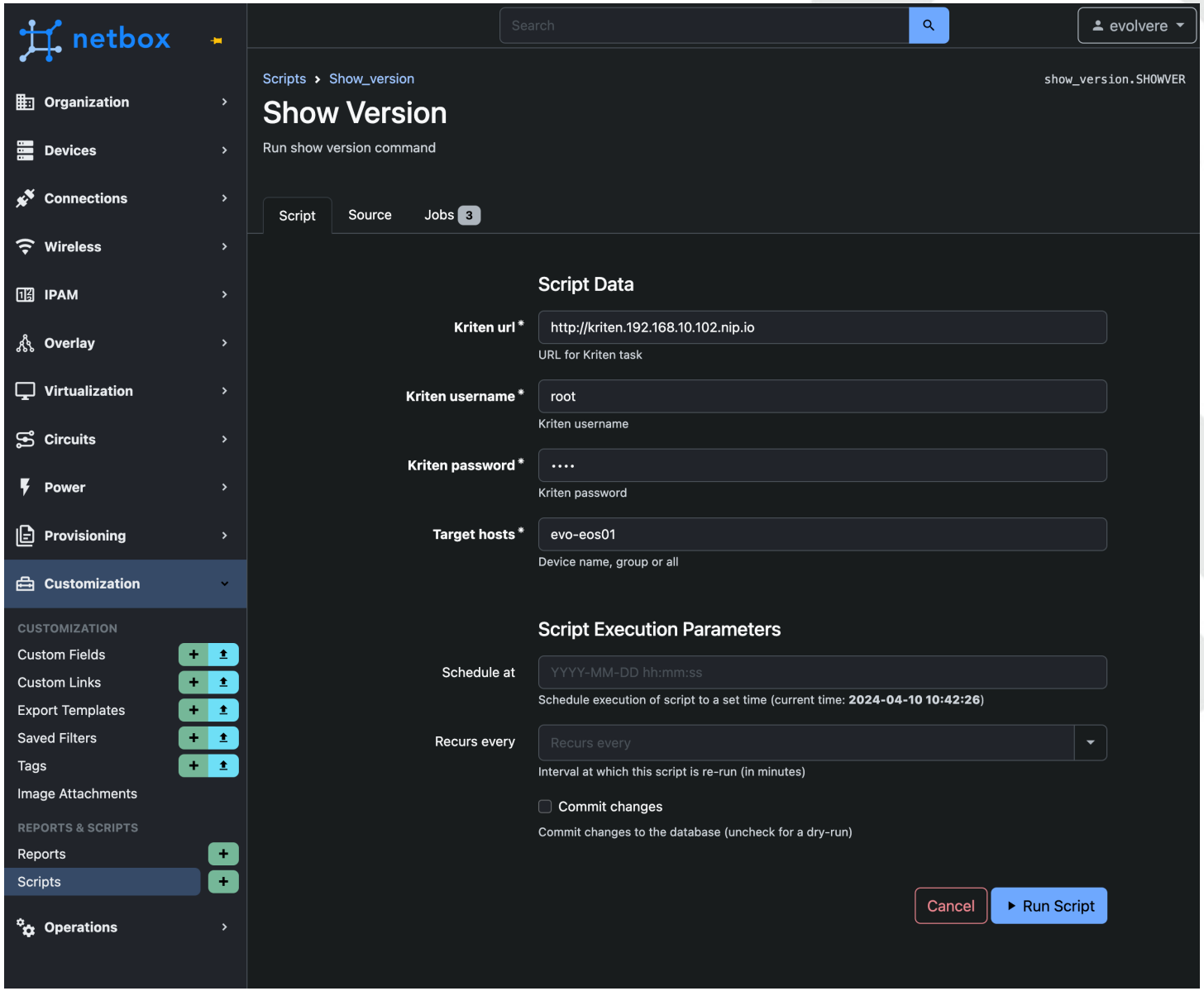
evolvere TECHNOLOGIES

Typical deployment scheme



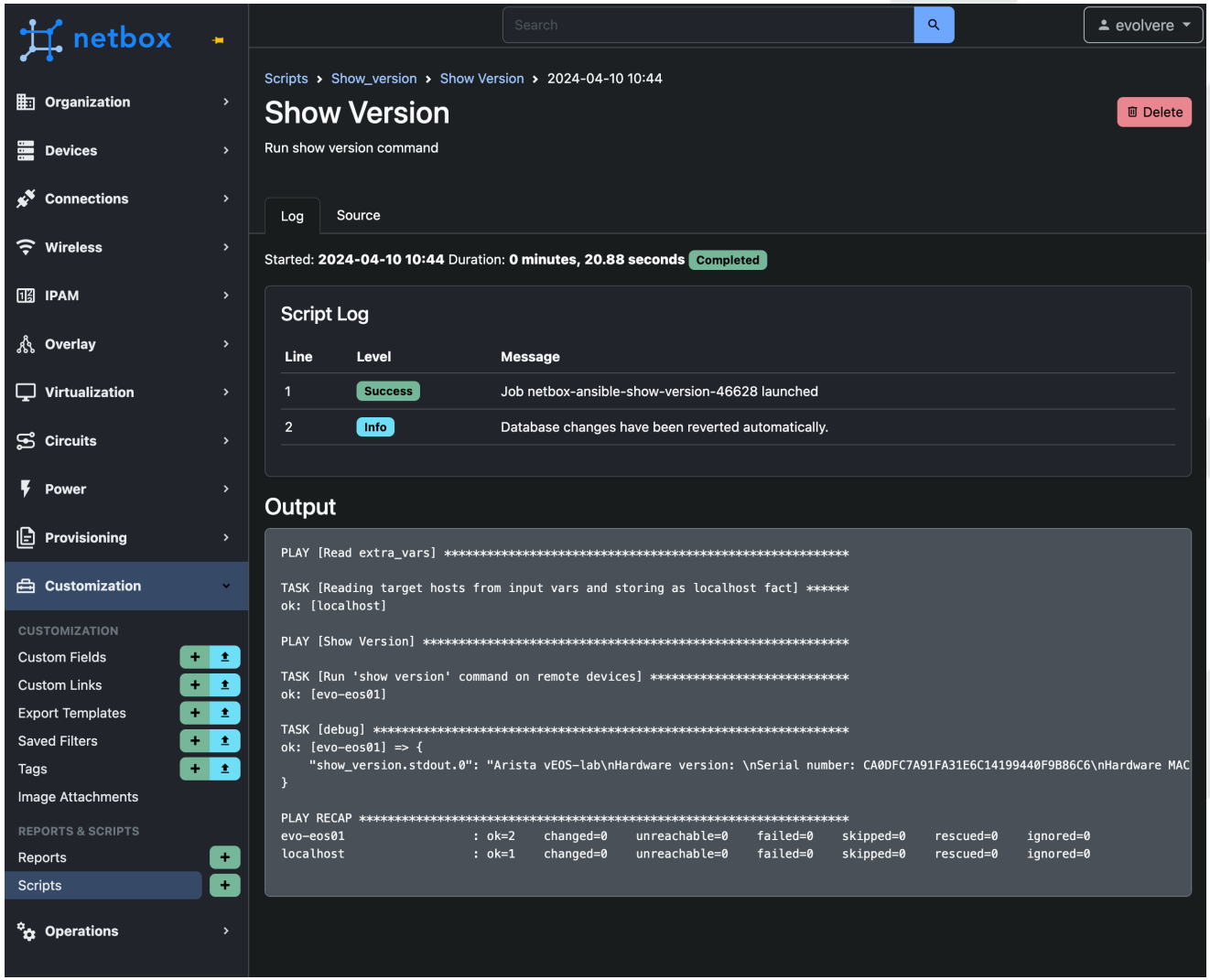
evolvere TECHNOLOGIES

Run from NetBox



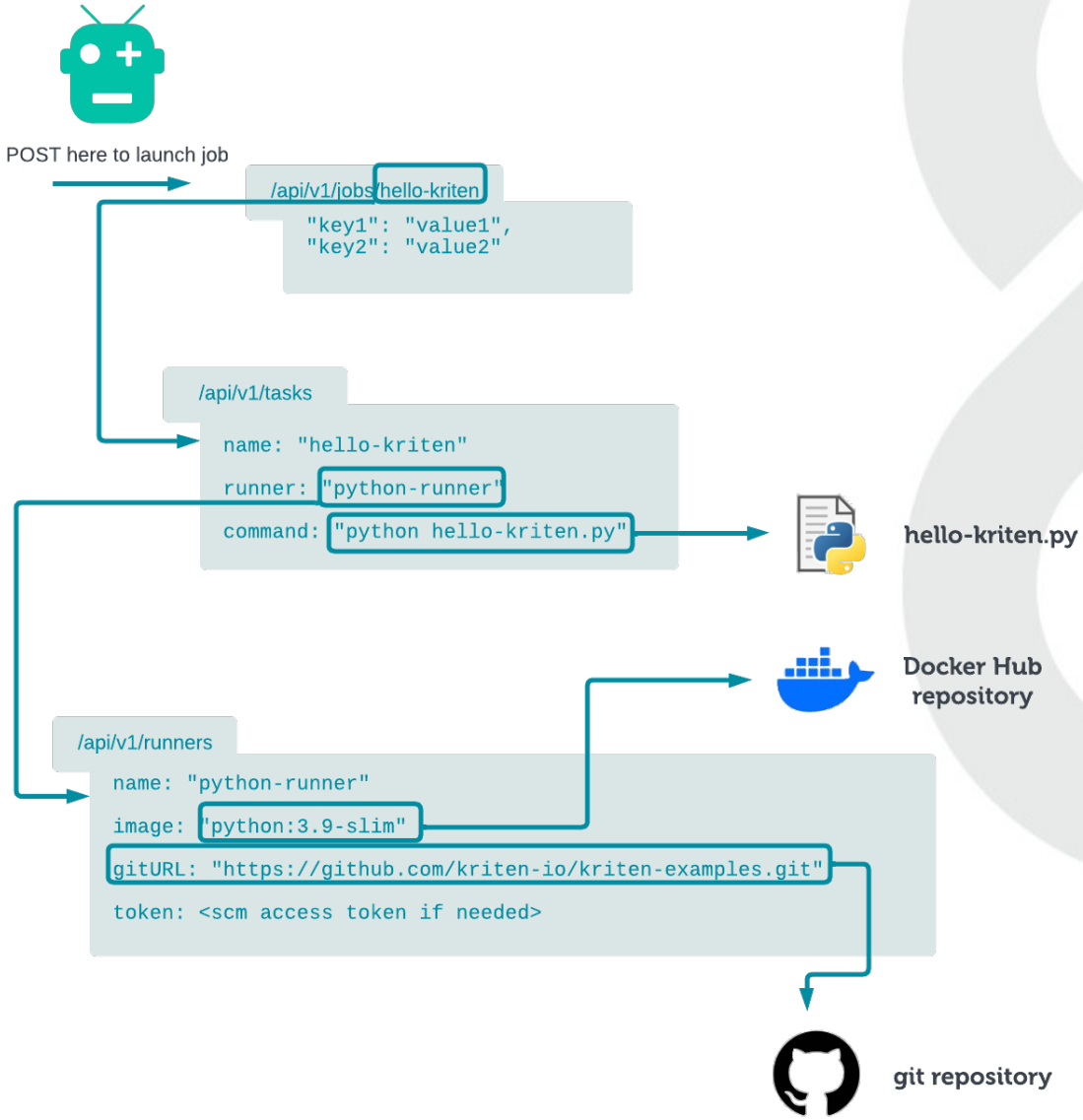
evolvere TECHNOLOGIES

Results from NetBox



evolvere
TECHNOLOGIES

Kriten runners and tasks



evolvere TECHNOLOGIES

Kriten Backlog

We have several features in the backlog for future release.

Scheduling
Deployments
Administrstion UI
Kritctl

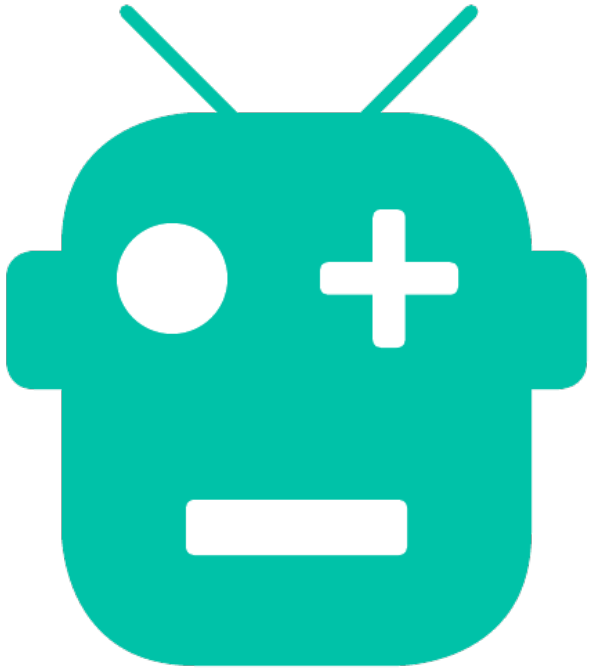
New feature and enhancement suggestions welcome.

Contact us at support@kriten.io

evolvere
TECHNOLOGIES

Need some help?

We have Network Automation and DevOps capabilities and a flexible approach to help wherever you are in the automation maturity model.



Contact us at info@evolvere-tech.co.uk.

evolvere
TECHNOLOGIES