

Automating Zabbix with Netbox and Ansible

About me



- Working @ AP Hogeschool
- Network and security expert
- Network automation enthusiast

 @jefvantongerloo

 [Linkedin.com/in/jefvantongerloo](https://www.linkedin.com/in/jefvantongerloo)

 github.com/jefvantongerloo/zabbix-automation-demo

Network automation

What, Why & How?

- Use of programmable logic and code to describe the network
 - Intent based networking
 - Predictable state

Network automation

What, Why & How?

- Use of programmable logic and code to describe the network
 - Intent based networking
 - Predictable state
- Automation of network processes and operations
 - Configuration management
 - Device discovery and provisioning
 - Operational tasks

Network automation

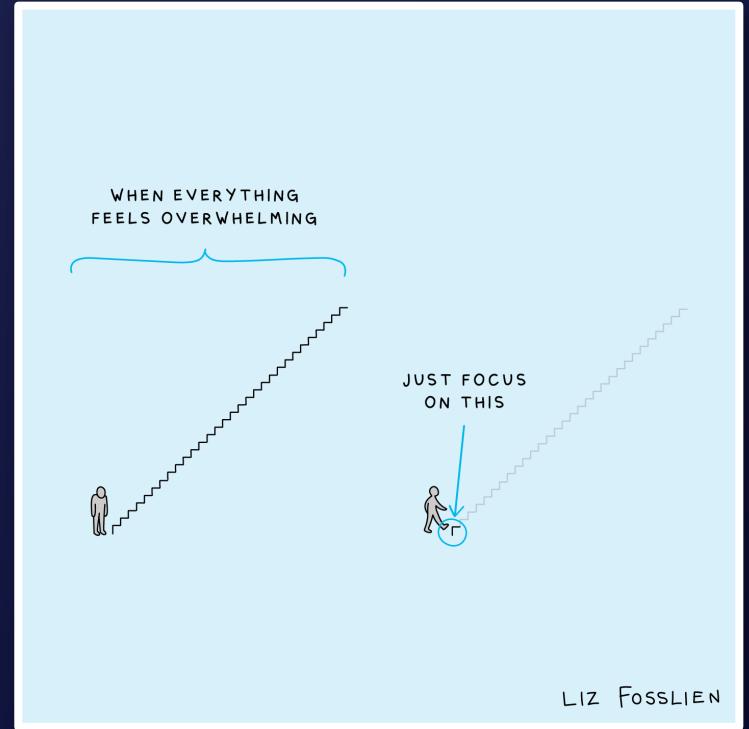
What, Why & How?

- Eliminates manual error prone tasks
- Improves efficiency
- Scales from cow to cattle
- Vendor independent

Network automation

What, Why & How?

- Think big, start small
- Start with read-only operations
- KISS – Keep it simple, stupid
 - Complexity lies into the problem, not the solution

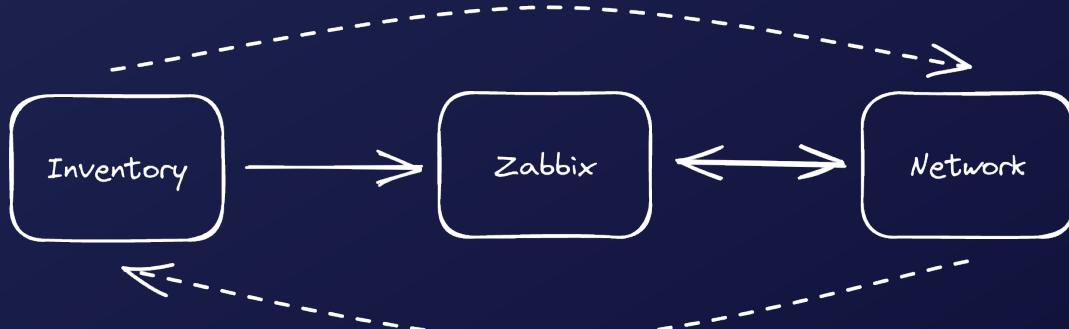


Liz Fosslien, <https://fosslien.com>

Case study

What do we want to achieve?

- An Automated flow for on-boarding, synchronizing and configuring network devices from our source of truth to Zabbix for enabling SNMP monitoring.
- A closed loop and hands-off approach, set and forget.



Case study

Automate all the things!

- Automated Zabbix host and interface creation
- Automated Zabbix templates and groups configuration
- Automated Network device SNMP configuration and validation
- Automated source of truth as inventory
- Automated callback to our inventory journal for logging

Case study

Open-source rules the world

- Infrastructure manager as source of truth
 - Network & device modeling
 - API integration
 - Device journal



Case study

Open-source rules the world

- Infrastructure manager as source of truth
 - Network & device modeling
 - API integration
 - Device journal
- Automating tasks and processes
 - Modular with Ansible collections
 - API interactions through libraries
 - Network CLI interaction capabilities
 - Configuration templating



Case study

Open-source rules the world

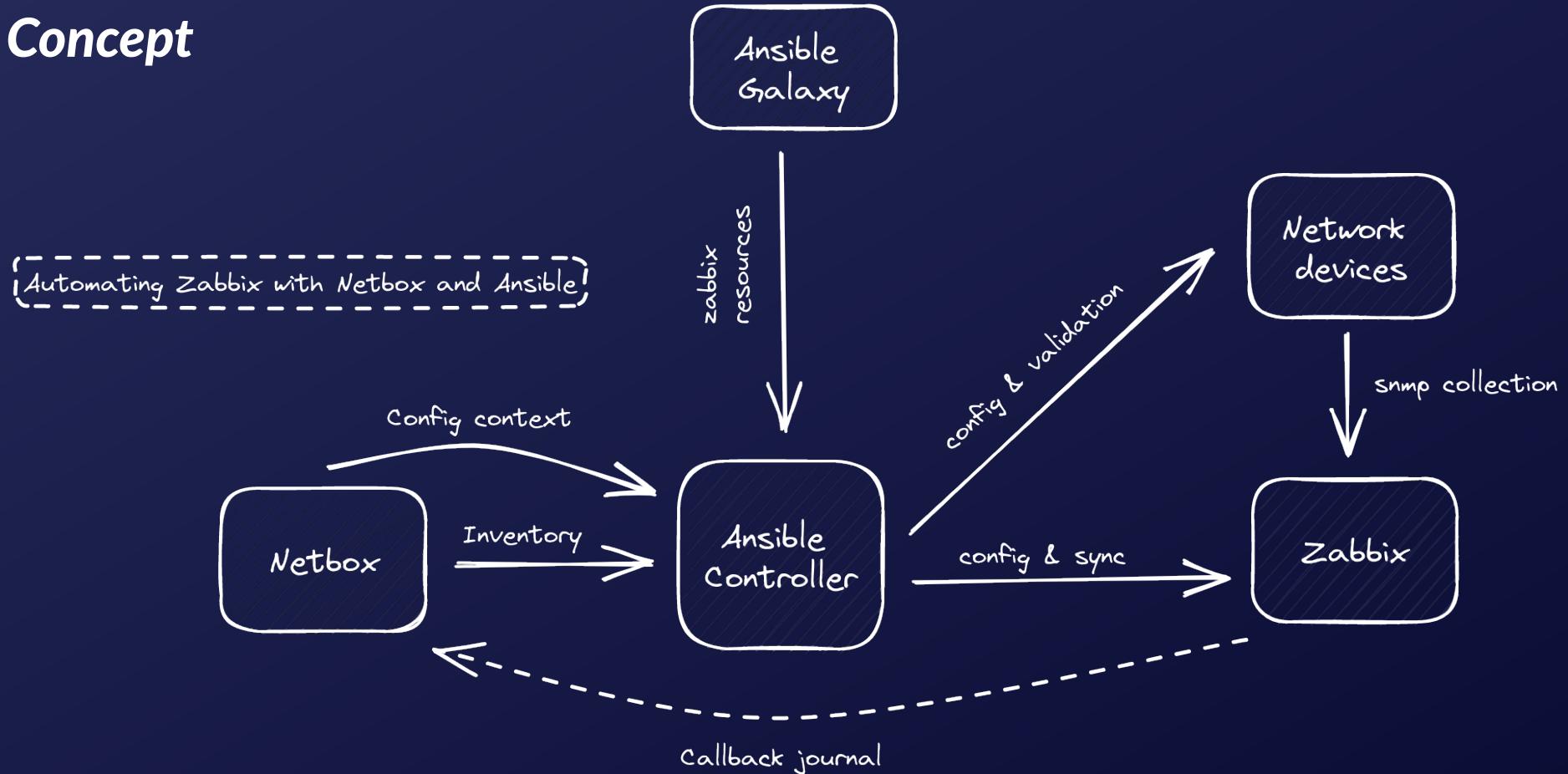
- Infrastructure manager as source of truth
 - Network & device modeling
 - API integration
 - Device journal
- Automating tasks and processes
 - Modular with Ansible collections
 - API interactions through libraries
 - Network CLI interaction capabilities
 - Configuration templating
- Monitoring solution
 - Community Ansible collection
 - API integration



ZABBIX

Case study

Concept



Case study

Ansible Galaxy

The screenshot shows the Ansible Galaxy interface. On the left is a sidebar with navigation links: Home, Search, Community, My Content, and My Imports. The main content area displays the details for the 'zabbix' collection under the 'community' category. The collection icon is a red circle with a white 'A'. The title is 'zabbix' and the description is 'Collection allowing to configure resources in Zabbix monitoring system'. The page includes tabs for Details, Read Me, and Content. Under the 'Info' section, there's an 'Installation' section with a command line snippet '\$ ansible-galaxy collection install community.zabbix' and a note about supported Ansible versions. It also features a 'Download tarball' button. The 'Install Version' dropdown shows '1.9.3 released 7 days ago (latest)'. The 'Tags' section lists 'monitoring' and 'zabbix'. To the right, there's a 'Content Score' card with a green bar indicating a score of 4.7 / 5 based on 3760508 downloads. Buttons for Unfollow Collection, Issue Tracker, Website, and Docs Site are also present.

Ansible Galaxy

https://galaxy.ansible.com/community/zabbix

GALAXY

Community Authors > community > zabbix

zabbix

Collection allowing to configure resources in Zabbix monitoring system

Details Read Me Content

Info

Installation

```
$ ansible-galaxy collection install community.zabbix
```

NOTE: Installing collections with ansible-galaxy is only supported in ansible 2.9+
[Download tarball](#)

Install Version

1.9.3 released 7 days ago (latest)

Tags

monitoring zabbix

4.7 / 5 Score 3760508 Downloads

Unfollow Collection Issue Tracker Repo

Website Docs Site

Content Score

Community Score 4.7 / 5

Based on 5 surveys. Show Details

Tell us about this collection

Quality of docs? Ease of use? Does what it promises?

Y N

Demo

Add device to Netbox

The screenshot shows the Netbox interface for adding a new device. The left sidebar is titled 'netbox' and includes sections for Organization, Devices, DEVICES (Devices, Modules, Device Roles, Platforms, Virtual Chassis), DEVICE TYPES (Device Types, Module Types), and DEVICE COMPONENTS (Interfaces). The 'Devices' section is currently selected.

The main page title is 'Add a new device'. A handwritten note 'Add "zabbix" tag' with an arrow points to the 'Tags' input field, which contains the tag 'zabbix' and has a blue oval around it.

The 'Device' section contains the following fields:

- Name: net-lab-swi-dis-1-1
- Device role*: net distribution switch
- Description: demo
- Tags: zabbix

The 'Hardware' section is partially visible at the bottom.

Demo

Add device to Netbox

The screenshot shows the Netbox web interface for managing network devices. On the left, there's a sidebar with navigation links for Organization, Devices, Device Types, and Device Components. The main area displays a device named "net-lab-swi-dist-1-1". The top navigation bar includes a search bar, a user profile for "Jef", and buttons for adding components, cloning, editing, and deleting the device. A handwritten note with an arrow points to the "Config Context" tab, which is highlighted with a blue oval.

Select 'config context'

Device		Management	Services
Region	Antwerpen	Status	Active
Site	lab	Role	net distribution switch
Location	—	Platform	alcatel_aos
Rack	—	Primary IPv4	10.10.20.11
Position	—	Primary IPv6	—
Tenant	AP Hogeschool / netwerk		
Device Type	Alcatel-Lucent Enterprise OS6560-P24X4 (1U)		
Description	—		
Airflow	—		

Demo

Add device to Netbox

Used for switch config

net-lab-swi-dist-1-1

Devices > lab

+ Add Components ▾ Clone Edit Delete configuration in gitlab

Device Interfaces 31 Config Context Status LLDP Neighbors Config Journal Changelog

Rendered Context

```
{ "cfg_snmp_stations": [ { "ip_address": "10.200.10.101", "protocol": "v3", "status": "enable", "udp_port": "162", "user": "snmpv3" } ] }
```

Local Context

```
{ "cfg_snmp_stations": [ { "ip_address": "10.200.10.101", "protocol": "v3", "status": "enable", "udp_port": "162", "user": "snmpv3" } ] }
```

netbox

Organization Devices

Devices Modules Device Roles Platforms Virtual Chassis

Device Types

Device Types Module Types Manufacturers

DEVICE COMPONENTS

Interfaces

Demo

Ansible inventory

```
Netbox


```
plugin: netbox.netbox.nb_inventory
validate_certs: false
config_context: true
flatten_config_context: true
compose:
 ansible_network_os: custom_fields.ansible_network_os
 device_id: id
 netbox_api: "{{ lookup('env', 'netbox_api') }}"
 netbox_user_id: '2'

group_names_raw: true
groups:
 netbox: ansible_host != ""
group_by:
 - sites

query_filters:
 - tag: zabbix
```


```

```
Ansible-inventory -i inventory/netbox.yml --list


```
{
 "_meta": {
 "hostvars": {
 "net-lab-swi-dist-1-1": {
 "ansible_host": "10.10.20.11",
 "cfg_snmp_stations": [
 {
 "ip_address": "10.200.10.101",
 "protocol": "v3",
 "status": "enable",
 "udp_port": "162",
 "user": "snmpv3"
 }
]
 }
 },
 "lab": {
 "hosts": [
 "net-lab-swi-dist-1-1"
]
 }
 }
}
```


```

Demo

Switch configuration

```
switch-config.yml
```

The diagram illustrates the 'parse' step between two files. On the left, a box highlights the first task in the Ansible playbook, labeled '1'. An arrow points from this task to the right, labeled 'parse', indicating the transition to the TextFSM template on the right.

```
name: Get >>> snmp stations
  ansible.utils.cli_parse:
    command: show snmp station
    parser:
      name: ansible.utils.textfsm
      set_fact: pre_snmp_stations
```

```
- name: Configure >>> snmp stations
  when: pre_snmp_stations != cfg_snmp_stations
  notify:
    - "Zabbix-sync : save configuration"
    - "Zabbix-sync : callback netbox journal - config diff"
  block:
    - name: Push >> no snmp stations
      ansible.netcommon.cli_command:
        command: "{{ lookup('template', 'no_snmp_stations.j2') }}"
    - name: Push >> snmp stations
      ansible.netcommon.cli_command:
        command: "{{ lookup('template', 'snmp_stations.j2') }}"
    - name: Info >> snmp configuration changes
      ansible.utils.fact_diff:
        before: "{{ pre_snmp_stations }}"
        after: "{{ cfg_snmp_stations }}"
      register: cfg_snmp_stations_diff
```

```
aos_show_snmp_station.textfsm
```

The diagram shows the TextFSM template and its resulting parsed output. The template defines regular expressions for various parameters like ip_address, udp_port, status, protocol, and user. It includes sections for 'Start', 'Station', and 'Record'. The 'parsed' section on the right shows the structured output generated by the template.

```
Value ip_address ((?:[0-9]{1,3}\.){3}[0-9]{1,3})/(?:[a-f0-9]{1,4}:){7}[a-f0-9]{1,4})
Value udp_port (\d{1,5})
Value status (enable/disable)
Value protocol (v1/v2/v3)
Value user (.*)
```

```
Start
  ^ipAddress/port
  ^[-+\.]+\s*$$ → Station
          status     protocol user$$
```

```
Station
  ^${ip_address}\/${udp_port}\s+${status}\s+${protocol}\s+${user}$$ → Record
  ^.\s* → Error
```

```
parsed:
- ip_address: 10.200.10.11
  protocol: v3
  status: enable
  udp_port: '162'
  user: snmpv3
```

Demo

Switch configuration

```
switch-config.yml
```

```
---
```

```
- name: Get >>> snmp stations
  ansible.utils.cli_parse:
    command: show snmp station
    parser:
      name: ansible.utils.textfsm
    set_fact: pre_snmp_stations
```

```
- name: Configure >>> snmp stations
  when: pre_snmp_stations != cfg_snmp_stations
  notify:
    - "Zabbix-sync : save configuration"
    - "Zabbix-sync : callback netbox journal - config diff"
  block:
    - name: Push >> no snmp stations
      ansible.netcommon.cli_command:
        command: "{{ lookup('template', 'no_snmp_stations.j2') }}"
```

```
- name: Push >> snmp stations
  ansible.netcommon.cli_command:
    command: "{{ lookup('template', 'snmp_stations.j2') }}"
```

```
- name: Info >> snmp configuration changes
  ansible.utils.fact_diff:
    before: "{{ pre_snmp_stations }}"
    after: "{{ cfg_snmp_stations }}"
  register: cfg_snmp_stations_diff
```

Source of truth

```
parsed:
- ip_address: 10.200.10.101
  protocol: v3
  status: enable
  udp_port: '162'
  user: snmpv3
```



Device state

```
parsed:
- ip_address: 10.200.10.11
  protocol: v3
  status: enable
  udp_port: '162'
  user: snmpv3
```

Demo

Switch configuration

```
switch-config.yml
```

```
---
```

```
- name: Get >>> snmp stations
  ansible.utils.cli_parse:
    command: show snmp station
    parser:
      name: ansible.utils.textfsm
    set_fact: pre_snmp_stations
```

```
- name: Configure >>> snmp stations
  when: pre_snmp_stations != cfg_snmp_stations
  notify:
    - "Zabbix-sync : save configuration"
    - "Zabbix-sync : callback netbox journal - config diff"
  block:
    - name: Push >> no snmp stations
      ansible.netcommon.cli_command:
        command: "{{ lookup('template', 'no_snmp_stations.j2') }}"
    - name: Push >> snmp stations
      ansible.netcommon.cli_command:
        command: "{{ lookup('template', 'snmp_stations.j2') }}"
    - name: Info >> snmp configuration changes
      ansible.utils.fact_diff:
        before: "{{ pre_snmp_stations }}"
        after: "{{ cfg_snmp_stations }}"
      register: cfg_snmp_stations_diff
```

3



```
template
```

```
snmp_stations.j2
```

```
{% for snmp_station in cfg_snmp_stations %}
  snmp station {{ snmp_station.ip_address }}
  {{ snmp_station.udp_port }}
  {{ snmp_station.user }}
  {{ snmp_station.protocol }}
  {{ snmp_station.status }}
{% endfor %}
```

```
! SNMP:
  snmp station 10.200.10.101 162 "snmpv3" v3 enable
```

Demo

Zabbix configuration



netbox.yml

```
name: Get >>> sites from netbox
run_once: true
block:
- name: Get >>> netbox sites
  ansible.builtin.set_fact:
    - netbox_sites: |
        "{{ query('netbox.netbox.nb_lookup',
        'sites',
        api_endpoint=netbox_api,
        token=netbox_token) }}"
- name: Add/update/remove >>> groups from netbox
  delegate_to: "{{ zabbix_api }}"
  community.zabbix.zabbix_group:
    - state: "{{ netbox_state_map[item.value.status.value] }}"
      host_groups:
        - "net/{{ item.value.slug }}"
  loop: "{{ netbox_sites }}
```

1

GET - HTTP 200 OK

2

Netbox:

Response body:

```
{
  "count": 1,
  "next": null,
  "previous": null,
  "results": [
    {
      "id": 1,
      "url": "api/dcim/sites/1/",
      "display": "lab",
      "name": "lab",
      "slug": "lab",
      "status": {
        "value": "active",
        "label": "Active"
      },
      ...
    }
  ]
}
```

Demo

Zabbix configuration

The screenshot shows the Zabbix web interface with the title "AP Zabbix Server: Config" in the browser tab. The left sidebar is titled "ZABBIX" and contains the following navigation items: Monitoring, Services, Inventory, Reports, Configuration (selected), Host groups, Templates, Hosts, Maintenance, Actions, Event correlation, Discovery, and Administration.

The main content area is titled "Host groups". A search bar at the top right says "Search with Google or enter address". A "Create host group" button is located in the top right corner of the main content area. A "Filter" button with a dropdown arrow is also present.

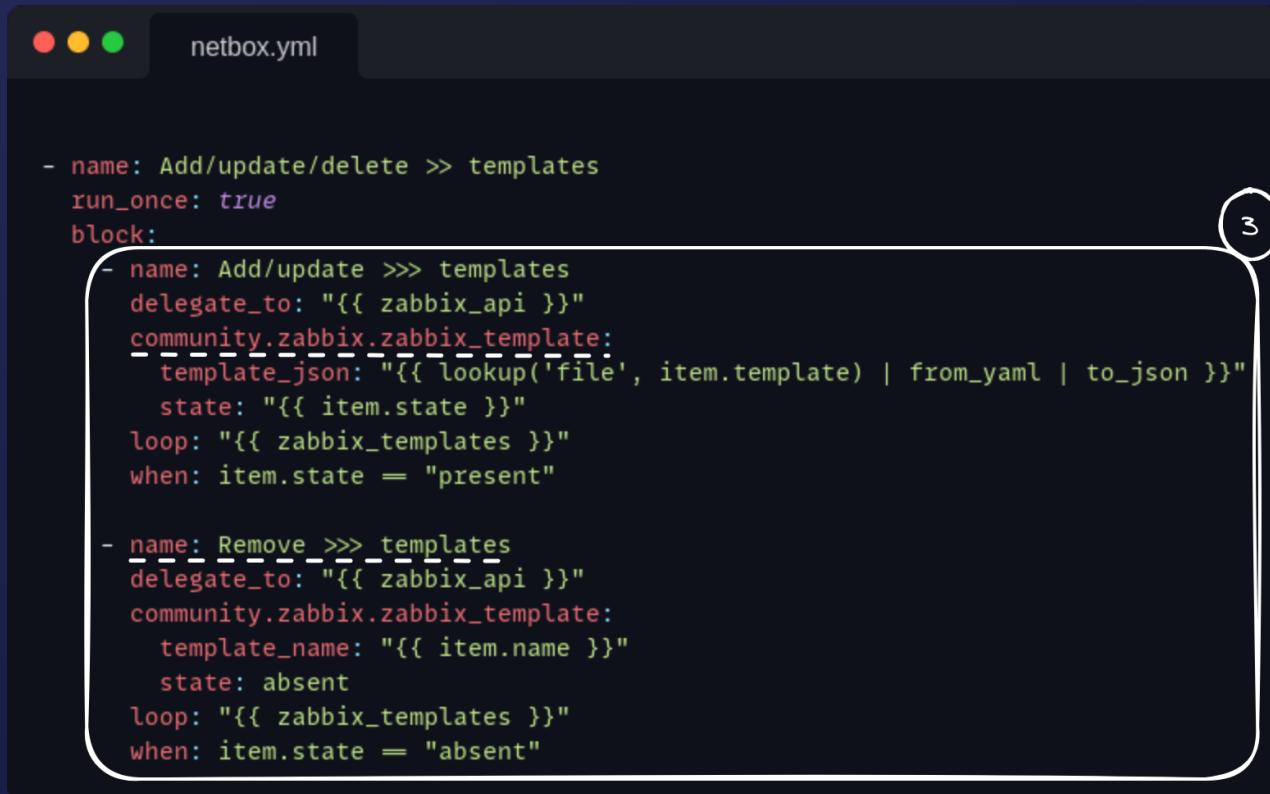
In the "Host groups" section, there is a search input field with the value "lab" and "Apply" and "Reset" buttons below it. The table displays one host group entry:

Name	Hosts	Templates	Members	Info
net/lab	Hosts	Templates		

Below the table, the status "Displaying 1 of 1 found" is shown. At the bottom of the page, there are buttons for "0 selected", "Enable hosts", "Disable hosts", and "Delete".

Demo

Zabbix configuration



netbox.yml

```
- name: Add/update/delete >> templates
  run_once: true
  block:
    - name: Add/update >>> templates
      delegate_to: "{{ zabbix_api }}"
      community.zabbix.zabbix_template:
        template_json: "{{ lookup('file', item.template) | from_yaml | to_json }}"
        state: "{{ item.state }}"
      loop: "{{ zabbix_templates }}"
      when: item.state == "present"

    - name: Remove >>> templates
      delegate_to: "{{ zabbix_api }}"
      community.zabbix.zabbix_template:
        template_name: "{{ item.name }}"
        state: absent
      loop: "{{ zabbix_templates }}"
      when: item.state == "absent"
```

Ansible variable:

```
zabbix_templates:
  - template: template_alcatel-lucent...
    name: Alcatel-Lucent Enterprise...
    platform: aos8
    state: present
```

Demo

Zabbix configuration

The screenshot shows the Zabbix Server Configuration interface with the following details:

- Header:** AP Zabbix Server: Config
- Search Bar:** Search with Google or enter address
- Left Sidebar (Configuration):**
 - Monitoring
 - Services
 - Inventory
 - Reports
 - Templates** (selected)
 - Host groups
 - Hosts
 - Maintenance
 - Actions
 - Event correlation
 - Discovery
 - Administration
- Main Content Area:**

Templates

Host groups: type here to search Tags: And/Or Or

Linked templates: type here to search

Name: aos

Buttons: Apply, Reset, Create template, Import, Filter

Name	Hosts	Items	Triggers	Graphs	Dashboards	Discovery	Web	Linked templates	Linked to templates	Tags
Alcatel-Lucent Enterprise OmniSwitch AOS 6.x	Hosts	Items 5	Triggers 1	Graphs	Dashboards	Discovery 4	Web			
Alcatel-Lucent Enterprise OmniSwitch AOS 8.x	Hosts	Items 5	Triggers 1	Graphs	Dashboards	Discovery 3	Web			

Displaying 2 of 2 found

Buttons at the bottom: 0 selected, Export, Mass update, Delete, Delete and clear.

Demo

Zabbix hosts

```
zabbix-hosts.yml
```

```
- name: Add/update >>> hosts to zabbix
  when: status.value in ["active", "offline", "planned", "staged"]
  notify: 'Zabbix-sync : callback netbox journal - added'
  block:
    - name: Add/update >>> hosts to zabbix
      delegate_to: "{{ zabbix_api }}"
      community.zabbix.zabbix_host:
        state: present
        host_name: "{{ inventory_hostname }}"
        visible_name: "{{ inventory_hostname }}"
        interfaces:
          - type: 2
            main: 1
            useip: 1
            ip: "{{ primary_ip4 }}"
            details:
              authpassphrase: "{{ cfg_snmp_passphrase }}"
              authprotocol: 1
              securityname: snmpv3
              privpassphrase: "{{ cfg_snmp_passphrase }}"
              privprotocol: 1
              securitylevel: 2
              version: 3
        host_groups:
          - "net/{{ sites[0] }}"
...
```

```
zabbix-hosts.yml
```

```
...
  - name: Add/update >>> linked templates
    delegate_to: "{{ zabbix_api }}"
    community.zabbix.zabbix_host:
      host_name: "{{ inventory_hostname }}"
      link_templates:
        - Alcatel-Lucent Enterprise OmniSwitch AOS 8.x
    when: platforms[0] = "alcatel_aos"

  - name: Add/update >>> host status
    notify: 'Zabbix-sync : callback netbox journal - status changed'
    delegate_to: "{{ zabbix_api }}"
    community.zabbix.zabbix_host:
      host_name: "{{ inventory_hostname }}"
      status: "{{ zabbix_status_map[status.value] }}"
    when: status.value in ["offline", "planned", "staged"]

  name: Delete >>> hosts
  notify: 'Zabbix-sync : callback netbox journal - deleted'
  delegate_to: "{{ zabbix_api }}"
  community.zabbix.zabbix_host:
    state: absent
    host_name: "{{ inventory_hostname }}"
  when: status.value in ["inventory", "decommissioning"]
```

Demo

Zabbix configuration

AP Zabbix Server: Config

Search with Google or enter address

ZABBIX AP Zabbix Server

Monitoring Services Inventory Reports Configuration Host groups Templates Hosts Maintenance Actions Event correlation Discovery Administration

Hosts

Create host Import Filter

Host groups: type here to search Select Monitored by: Any Server Proxy

Templates: type here to search Select

Name: lab

DNS:

IP:

Port:

Tags: And/Or Or Add tag Contains value Remove

Apply Reset

Name	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy	Templates	Status	Availability	Agent encryption	Info	Tags
net-lab-swi-dist-1-1	Items 5	Triggers 1	Graphs	Discovery 3	Web	10.10.20.11:161		Alcatel-Lucent Enterprise OmniSwitch AOS 8.x	Enabled	SNMP	None		

Displaying 1 of 1 found

0 selected Enable Disable Export Mass update Delete

Demo

Zabbix configuration

The screenshot shows the Zabbix Server Configuration interface for creating a new host. The left sidebar includes links for Monitoring, Services, Inventory, Reports, Configuration (selected), Host groups, Templates, Hosts, Maintenance, Actions, Event correlation, Discovery, and Administration.

The main area displays the 'Host' configuration form:

- Host tab:** Contains fields for Host name (net-lab-swi-dist-1-1) and Visible name (net-lab-swi-dist-1-1).
- Templates:** Associated with Alcatel-Lucent Enterprise OmniSwitch AOS 8.x.
- Groups:** Associated with net.lab.
- Interfaces:** An SNMP interface is configured with IP address 10.10.20.11, Port 161, and Security name snmpv3.

The right side of the interface shows a summary table for the host:

Host	IP	Port	Default
net-lab-swi-dist-1-1	10.10.20.11	161	None

At the bottom, a message indicates "Displaying 1 of 1 found".

Demo

Ansible handlers

```
handlers.yml
```

```
name: 'Zabbix-sync : save configuration'
vars:
  ansible_connection: ansible.netcommon.network_cli
  ansible_network_os: jefvantongerloo.alcatel-aos
  ansible.netcommon.cli_command:
    - write memory flash-synchro

name: 'Zabbix-sync : callback netbox journal - added'
ansible.builtin.uri:
  url: "{{ netbox_api }}/api/extras/journal-entries/"
  method: POST
  headers:
    Accept: application/json
    Authorization: "Token {{ netbox_token }}"
  status_code: 201
  body_format: json
  body:
    assigned_object_type: dcim.device
    assigned_object_id: '{{ device_id }}'
    created_by: '{{ netbox_user_id }}'
    kind: info
    tags:
      - name: zabbix
    comments: /
  zabbix - device added
```

On task 'changed':

```
notify:
  - "Zabbix-sync : save configuration"
```

On task 'changed':

```
notify:
  - 'Zabbix-sync : callback netbox journal - added'
```

Demo

Netbox callback

The screenshot shows the Netbox interface for a device named "net-lab-swi-dist-1-1". The left sidebar is collapsed, showing sections for Organization, Devices, DEVICES (Devices, Modules, Device Roles, Platforms, Virtual Chassis), DEVICE TYPES (Device Types, Module Types), and DEVICE COMPONENTS (Interfaces). The main content area shows the device details for "net-lab-swi-dist-1-1". The top navigation bar includes a search bar, a user dropdown for "Jef", and buttons for Add Components, Clone, Edit, and Delete. A link to "configuration in gitlab" is also present. Below the header, there are tabs for Device, Interfaces (31), Config Context, Status, LLDP Neighbors, Config, Journal (2), and Changelog. The Changelog tab is active, displaying two entries:

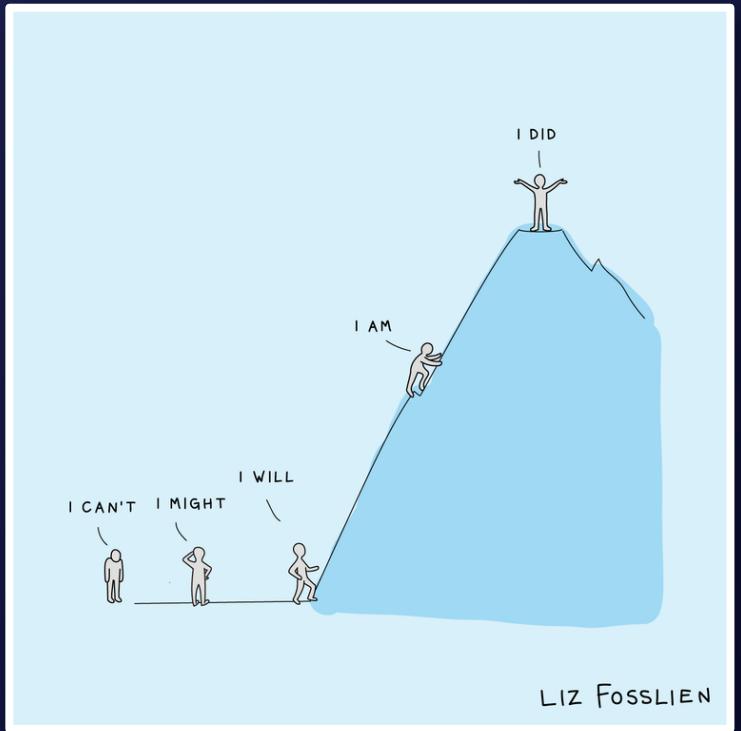
Created	Created by	Kind	Tags	Comments (Short)
2023-03-26 14:26	p064033	Info	config	zabbix - snmp config changed --- before +++ after @@ -1 +1 @@ -{user': 'snmpv3', 'status': 'enable', 'protocol': 'v3', 'udp_port': '162', 'ip_address': '10.200.10.11'} +{user': 'snmpv3', 'status': 'enable', 'protocol': 'v3', 'udp_port': '162', 'ip_address': '10.200.10.101'}
2023-03-26 14:26	p064033	Info	zabbix	zabbix - device added

At the bottom right, there are buttons for Per Page (dropdown) and a link showing "Showing 1-2 of 2".

What's next?

Integration

- Integrate in CI/CD pipeline services
 - Github, GitLab, Drone CI, ...
- Integrate into task manager
 - AWX, Jenkins, ...
- Sit back and relax



LIZ FOSSLIEN

Liz Fosslien, <https://fosslien.com>

Questions?

 @jefvantongerloo

 [Linkedin.com/in/jefvantongerloo](https://www.linkedin.com/in/jefvantongerloo)

 github.com/jefvantongerloo/zabbix-automation-demo