Ansible at Empa

An overview

Patrik Burkhalter

Empa Dübendorf

May 19, 2022

Patrik Burkhalter

I'm at Empa since 10 years. I had various different jobs:

- I built up the old HPC cluster of Empa using PBS torque and Lustre
- Then I was focusing mainly on IT systems based on Linux
- After that I built up an Empa Linux client (based on Ubuntu)
- Now I'm focusing mostly on architecture and design of new systems based on 'new' technologies

Topic and reason of my presentation

Daniele Passerone forced me to have a speech :-)

But to be more serious:

- We implemented infrastructure as code in some parts at Empa IT
- We are using Ansible for that
- I'll give a short introduction about Ansible

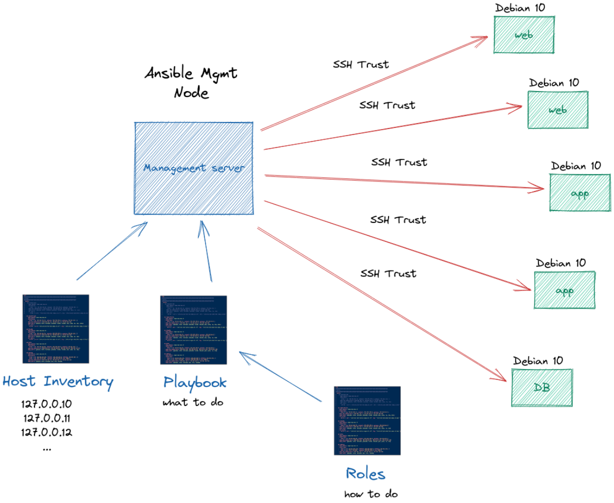
Overview

The situation at Empa is currently that:

- Empa has plenty of Linux servers
- We also deliver a Linux Client (Ubuntu)
- To configure those systems we are using Ansible
- Ansible is a provisioning, configuration and deployment software
- It enables us to implement Infrastructure as Code (IaC)
- We use Gitlab pipelines and docker to automate our work

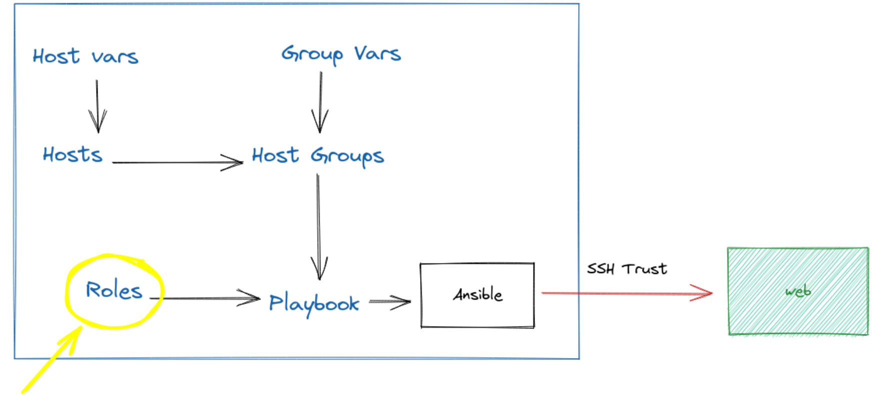
Ansible configuration

Ansible is configured using Yaml files



Ansible file structure

This is our Ansible structure:

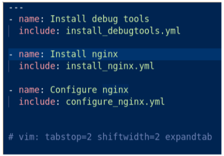


The role is where the engineering mainly happens!

Ansible role

An Ansible Role defines what to do exactly

main.yml



```
---
- name: Install debug tools
  include: install_debugtools.yml

- name: Install nginx
  include: install_nginx.yml

- name: Configure nginx
  include: configure_nginx.yml


# vim: tabstop=2 shiftwidth=2 expandtab
```

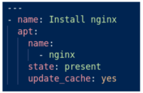
Note:

We do not have to define how a package is installed.

We can instead use the Ansible module `apt' to install packages via apt.

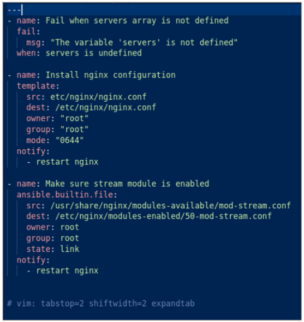
The most interesting module is probably `template'. This module takes a (local) file, fills in values and places it on the remote server

install_nginx.yml



```
---
- name: Install nginx
  apt:
    name:
      - nginx
    state: present
    update_cache: yes
```

configure_nginx.yml

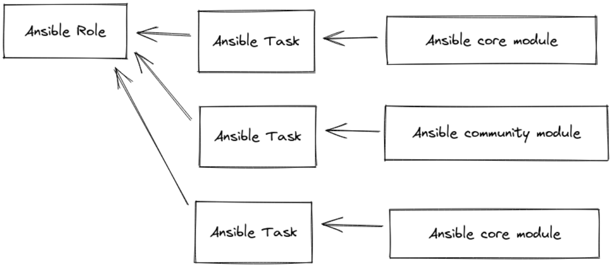


```
---
- name: Fail when servers array is not defined
  fail:
    msg: "The variable 'servers' is not defined"
  when: servers is undefined

- name: Install nginx configuration
  template:
    src: etc/nginx/nginx.conf
    dest: /etc/nginx/nginx.conf
    owner: "root"
    group: "root"
    mode: "0644"
  notify:
    - restart nginx

- name: Make sure stream module is enabled
  ansible.builtin.file:
    src: /usr/share/nginx/modules-available/mod-stream.conf
    dest: /etc/nginx/modules-enabled/50-mod-stream.conf
    owner: root
    group: root
    state: link
  notify:
    - restart nginx


# vim: tabstop=2 shiftwidth=2 expandtab
```

Ansible tasks and modules

Ansible uses modules which are implementing the functionality. Those modules are executed in an Ansible task.

An Ansible role can have a lot of tasks
and each task is using a module.



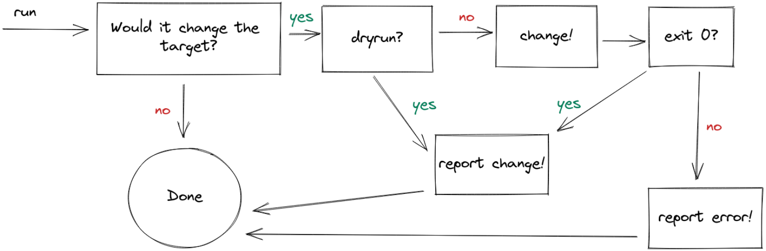
Plenty of Ansible modules available.
As a last resort, there is the module `command or cmd'

Idempotence

A change is only done when needed

Nice, but why are we using especially Ansible?

* A main point of using Ansible is, that Ansible checks the state on a target before it is doing anything!
    * We can run Ansible over and over on the same target without changing anything when everything is already ok.
        * This gets relevant when you just want to know if anything has changed on the system
* Infrastructure as code (goes to gitlab)
* Multiuse of roles



run

Would it change the target?

yes

no

dryrun?

no

yes

change!

exit 0?

yes

no

report change!

report error!

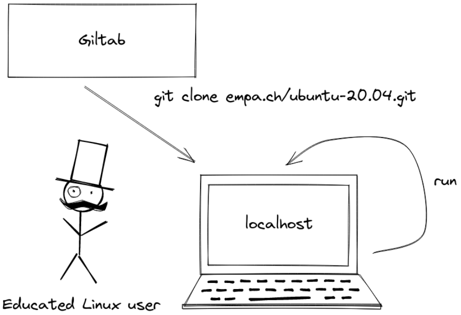
Done

Using Ansible on the client

We are using what we already know...

Why do I tell you something about Ansible?

- We also are providing Linux client installations at Empa
- Allthough Ansible is used mainly for servers, we also use Ansible for clients
- We do a "pull/local-push"



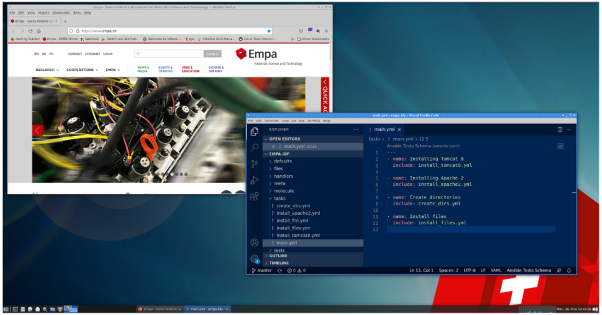
```
git clone empa.ch/ubuntu-20.04.git
```

Empa Linux client

An Ubuntu 20.04 LTS configured by Ansible

Client roles for Ubuntu

- Active Directory integration
- Kerberos
- Look&Feel
- Security settings (SSH, ...)
- Preinstalled software (docker, ...)

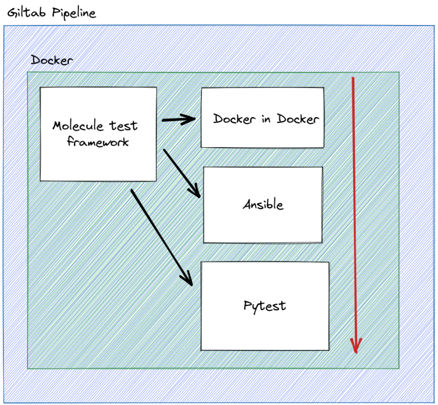


```yaml
---
- name: Installing Tomcat 8
  include: install_tomcat8.yml

- name: Installing Apache 2
  include: install_apache2.yml

- name: Create directories
  include: create_dirs.yml

- name: Install files
  include: install_files.yml

```

Using Ansible properly

Having things automated with Gitlab is a wonderful experience

Beyond that, we are using Gitlab and Docker to test our roles



Questions?

Any questions?