

# Rapport : Déploiement sécurisé avec Ansible

## 1. Objectif du projet

Ce mini-projet vise à automatiser le **déploiement sécurisé** d'un serveur web local à l'aide d'Ansible.

L'objectif est d'installer un service (Apache), de durcir la configuration système (utilisateur non-root, pare-feu, désactivation de services inutiles), et de vérifier l'application effective des mesures de sécurité.

## 2. Environnement de test

- **Système** : Ubuntu 22.04 (VM locale ou WSL)
- **Outil principal** : Ansible
- **Langage de configuration** : YAML
- **Mode d'exécution** : Localhost ([inventory.ini](#))
- **Rôle personnalisé** : [harden\\_web](#) + common

## 3. Structure du projet

```
secure-web-deploy/  
├── inventory.ini  
├── site.yml  
├── roles/  
│   ├── common/  
│   │   └── tasks/users.yml  
│   └── harden_web/  
│       └── tasks/main.yml
```

## 4. Fichiers clés et explications

[inventory.ini](#)

```
[local]  
localhost ansible_connection=local
```

Permet d'exécuter les commandes localement (sans SSH).

site.yml

```
- hosts: local
  become: yes
  roles:
    - common
```

Ce fichier inclut les rôles nécessaires au déploiement et au durcissement.

roles/common/tasks/users.yml

```
# roles/common/tasks/users.yml

- name: Créer un utilisateur non-root
  ansible.builtin.user:
    name: webuser
    shell: /bin/bash
    create_home: yes
```

Ajoute un utilisateur dédié sans droits root.

roles/harden\_web/tasks/main.yml

```
- name: Installer Apache
  apt:
    name: apache2
    state: present
    update_cache: yes

- name: Ajouter un utilisateur non-root
  user:
    name: webuser
    shell: /bin/bash
    create_home: yes

- name: Désactiver services inutiles (exemple)
  service:
    name: bluetooth
    state: stopped
    enabled: no
  ignore_errors: yes

- name: Activer le pare-feu UFW
  ufw:
    state: enabled

- name: Autoriser HTTP (port 80) via UFW
  ufw:
    rule: allow
    port: 80
    proto: tcp

- name: Interdire tout le reste (par défaut)
  ufw:
    direction: incoming
    policy: deny

- name: Redémarrer Apache si nécessaire
  service:
    name: apache2
    state: restarted
```

## 5. Exécution de la commande

Commande :

```
ansible-playbook -i inventory.ini site.yml --ask-become-pass
```

```
narm@pcn:~/projets/mini-projets/secure-web-deploy$ ansible-playbook -i inventory.ini site.yml --ask-become-pass
```

```
narm@pcn:~/projets/mini-projets/secure-web-deploy$ ansible-playbook -i inventory.ini site.yml --ask-become-pass
BECOME password:

PLAY [Déploiement sécurisé du serveur web] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [harden_web : Installer Apache] *****
ok: [localhost]

TASK [harden_web : Ajouter un utilisateur non-root] *****
ok: [localhost]

TASK [harden_web : Désactiver services inutiles (exemple)] *****
fatal: [localhost]: FAILED! => {"changed": false, "msg": "Could not find the requested service bluetooth: host"}
...ignoring

TASK [harden_web : Activer le pare-feu UFW] *****
ok: [localhost]

TASK [harden_web : Autoriser HTTP (port 80) via UFW] *****
ok: [localhost]

TASK [harden_web : Interdire tout le reste (par défaut)] *****
ok: [localhost]

TASK [harden_web : Redémarrer Apache si nécessaire] *****
changed: [localhost]

TASK [common : Inclusion des tâches de création utilisateur] *****
included: /home/narm/projets/mini-projets/secure-web-deploy/roles/common/tasks/users.yml for localhost

TASK [common : Créer un utilisateur non-root] *****
ok: [localhost]

PLAY RECAP *****
localhost : ok=10  changed=1  unreachable=0  failed=0  skipped=0  rescued=0  ignored=1
```

## 6. Vérifications après déploiement

## Apache actif

curl http://localhost

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w
3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2022-03-22
    See: https://launchpad.net/bugs/1966004
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
      * {
        margin: 0px 0px 0px 0px;
        padding: 0px 0px 0px 0px;
      }

      body, html {
        padding: 3px 3px 3px 3px;

        background-color: #D8DBE2;

        font-family: Ubuntu, Verdana, sans-serif;
        font-size: 11pt;
        text-align: center;
      }

      div.main_page {
        position: relative;
        display: table;

        width: 800px;

        margin-bottom: 3px;
        margin-left: auto;
        margin-right: auto;
```

```
padding: 0px 0px 0px 0px;

        border-width: 2px;
        border-color: #212738;
        border-style: solid;

        background-color: #FFFFFF;

        text-align: center;
      }

      div.page_header {
        height: 180px;
        width: 100%;

        background-color: #F5F6F7;
      }

      div.page_header span {
        margin: 15px 0px 0px 50px;

        font-size: 180%;
        font-weight: bold;
      }

      div.page_header img {
        margin: 3px 0px 0px 40px;

        border: 0px 0px 0px;
      }

      div.banner {
        padding: 9px 6px 9px 6px;
        background-color: #E9510E;
        color: #FFFFFF;
        font-weight: bold;
        font-size: 112%;
        text-align: center;
        position: absolute;
        left: 40%;
        bottom: 30px;
        width: 20%;
      }
```

### Vérifier l'utilisateur `webuser`

`id webuser`

```
uid=1001(webuser) gid=1001(webuser) groups=1001(webuser)
```

### UFW actif et restreint

`sudo ufw status verbose`

```
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), deny (routed)
New profiles: skip

To Action From
--
80/tcp ALLOW IN Anywhere
80/tcp (v6) ALLOW IN Anywhere (v6)
```

### Service désactivé : Bluetooth

`systemctl is-active bluetooth`

```
narm@pcn:~/projets/mini-projets/secure-web-deploy$ systemctl is-active bluetooth
inactive
```

## 7. Test de ré-exécution (idempotence)

ansible-playbook -i inventory.ini site.yml --ask-become-pass

Relancer :

```
PLAY [local] *****
****

TASK [Gathering Facts] *****
****
ok: [localhost]

TASK [common : Inclusion des tâches de création utilisateur] *****
****
included: /home/narm/projets/mini-projets/secure-web-deploy/roles/common/tasks/users.yml for localhost

TASK [common : Créer un utilisateur non-root] *****
****
ok: [localhost]

PLAY RECAP *****
localhost : ok=3    changed=0    unreachable=0    failed=0
           skipped=0    rescued=0    ignored=0
```

L'idempotence est bien respectée : les tâches ne s'exécutent pas inutilement si déjà appliquées.

## 8. Conclusion

Ce projet a permis de :

- Déployer automatiquement Apache sur une machine locale
- Créer un utilisateur sécurisé
- Mettre en place un pare-feu restrictif (UFW)
- Désactiver un service inutile
- Appliquer des bonnes pratiques DevSecOps avec Ansible

Ce type de déploiement est utile pour les environnements de production où la sécurité par défaut est essentielle.