

# Ansible Hands-On Project - Explained

This project demonstrates 4 core features of Ansible: Handlers, Async tasks, Run\_once with Delegate\_to, and Register with Debug. Below is a breakdown of the project structure and explanation of each playbook.

## Project Structure

```
.
├── ansible.cfg
├── inventory.ini
├── playbooks
│   ├── async_db_dump.yaml
│   ├── handler_hosts.yaml
│   ├── register_patch_check.yaml
│   └── runonce_delegate_artifact.yaml
└── templates
    └── nginx.conf
```

### 1. `ansible.cfg`

```
[defaults]
inventory = inventory.ini
host_key_checking = False
private_key_file = /home/ubuntu/.ssh/mypem.pem
retry_files_enabled = False
```

- Defines inventory file and disables strict host key checking.
- Uses custom private key for EC2 connection.

### 2. `inventory.ini`

```
[qa]
qa1 ansible_host=13.126.50.145 ansible_user=ubuntu
qa2 ansible_host=13.203.75.197 ansible_user=ubuntu
```

- Defines two QA target nodes.

### 3. `handler_hosts.yaml` - Handlers & Notify

```
- name: Add /etc/hosts entry and restart NGINX if config changes
  hosts: qa
  become: yes

  tasks:
    - name: Install NGINX if not present
      apt:
        name: nginx
        state: present
        when: ansible_os_family == "Debian"

    - name: Add internal repo entry to /etc/hosts
      lineinfile:
        path: /etc/hosts
        line: "10.10.10.10 internal.repo"
        state: present
        notify: Restart NGINX

    - name: Deploy NGINX config
      template:
        src: ../templates/nginx.conf
        dest: /etc/nginx/nginx.conf
        notify: Restart NGINX

  handlers:
    - name: Restart NGINX
      service:
        name: nginx
        state: restarted
```

#### Purpose:

- Only restarts nginx if `/etc/hosts` or nginx config changes.

### 4. `async_db_dump.yaml` - Async & Poll

```
- name: Run DB backup asynchronously
  hosts: qa
  become: true

  tasks:
    - name: Start DB dump (fake command)
      shell: "sleep 30 && echo 'backup complete' > /tmp/db_backup.txt"
      async: 60
      poll: 0
```

```

    register: db_job

- name: Wait for backup to finish
  async_status:
    jid: "{{ db_job.ansible_job_id }}"
    register: backup_status
    until: backup_status.finished
    retries: 5
    delay: 10

- name: Show DB backup result
  debug:
    var: backup_status

```

### Purpose:

- Demonstrates non-blocking task using `async` and polling for job result.

## 5. `runonce_delegate_artifact.yaml` - `run_once` & `delegate_to`

```

- name: Download artifact once and distribute
  hosts: qa
  become: yes
  vars:
    artifact_path: "/tmp/app.war"

  tasks:
    - name: Ensure mock WAR file exists (on localhost)
      delegate_to: localhost
      run_once: true
      copy:
        content: "Dummy WAR content for testing"
        dest: "{{ artifact_path }}"

    - name: Copy WAR file to all QA nodes
      copy:
        src: "{{ artifact_path }}"
        dest: "/opt/app.war"
        owner: root
        group: root
        mode: '0644'

```

### Purpose:

- Generates the artifact only once on the control node and distributes it to all QA hosts.

## 6. `register_patch_check.yaml` - Register & Debug

```
- name: Check Memory Usage and Warn if Low
  hosts: qa
  become: yes
  tasks:
    - name: Get memory status
      command: free -m
      register: mem_status

    - name: Show memory output
      debug:
        msg: "{{ mem_status.stdout_lines }}"

    - name: Extract available memory in MB
      shell: free -m | awk '/Mem:/ { print $7 }'
      register: available_mem
      changed_when: false

    - name: Print available memory
      debug:
        msg: "Available memory: {{ available_mem.stdout }} MB"

    - name: Warn if memory is low
      debug:
        msg: "\u26a0\ufe0f WARNING: Low Memory! Only {{
available_mem.stdout }} MB available."
      when: available_mem.stdout|int < 500
```

### Purpose:

- Extracts available memory, logs it, and alerts if below 500MB.

## 7. `templates/nginx.conf`

```
# Simple NGINX configuration deployed by Ansible
user www-data;
worker_processes auto;
pid /run/nginx.pid;

events {
    worker_connections 512;
}

http {
    include mime.types;
    default_type application/octet-stream;
```

```

sendfile          on;
keepalive_timeout 65;

server {
    listen 80;
    server_name {{ inventory_hostname }};

    location / {
        root /var/www/html;
        index index.html index.htm;
    }
}

```

### Purpose:

- Templated nginx config that adapts `server_name` to inventory hostname.

## ✓ Summary Table

Feature	Example File	Key Task	Output
Handlers & Notify	handler_hosts.yaml	Change hosts file or config → restart nginx	Restart only on change
Async & Poll	async_db_dump.yaml	Background task using shell + poll	Backup finished result logged
run_once & delegate	runonce_delegate_artifact.yaml	Only localhost creates artifact	All QA get file, 1-time copy
Register & Debug	register_patch_check.yaml	Capture and conditionally alert	Memory checked + warning shown