



# Lab 11 — Handlers & Notifications in Ansible

Author: Sandeep Kumar Sharma

---



## Learning Objectives

In this lab, you will learn: - What handlers are in Ansible - Why handlers are used - How tasks notify handlers - How to restart services only when changes occur - How to build efficient, idempotent automation

---



## Learning Outcomes

After completing this lab, you will: - Understand the purpose of handlers - Trigger handlers using notifications - Restart services only when needed - Write clean and production-ready automation playbooks

---



## What Are Handlers in Ansible?

Handlers are special tasks in Ansible that run **only when notified**.

Typical use cases: - Restarting a service - Reloading configuration - Rebuilding cache

Handlers ensure tasks run only when something changes, improving performance and keeping automation idempotent.

---



## SECTION A — How Handlers Work

Structure:

```
handlers:
  - name: restart apache
    service:
      name: apache2
      state: restarted
```

Tasks notify handlers when they change something:

```
notify: restart apache
```



## SECTION B — Hands-On: Using Handlers with Apache

Create file:

```
nano handlers-example.yml
```

Add:

```
---
- name: Handlers & Notifications demo
  hosts: dev
  become: yes
  gather_facts: yes

  vars:
    apache_pkg: "{{ 'apache2' if ansible_facts['os_family'] == 'Debian' else
'httpd' }}"
    apache_service: "{{ apache_pkg }}"
    apache_conf: "{{ '/etc/apache2/apache2.conf' if ansible_facts['os_family']
== 'Debian' else '/etc/httpd/conf/httpd.conf' }}"

  tasks:
    - name: Install Apache
      package:
        name: "{{ apache_pkg }}"
        state: present

    - name: Update Apache configuration file
      lineinfile:
        path: "{{ apache_conf }}"
        line: "# Managed by Ansible"
        notify: restart apache

  handlers:
    - name: restart apache
      service:
```

```
name: "{{ apache_service }}"
state: restarted
```

Run:

```
ansible-playbook handlers-example.yml
```

### What happens?

- If the config file changes → handler runs
- If no change → handler does NOT run



## SECTION C — Multiple Notifications for Same Handler

A single handler can be triggered by multiple tasks.

Example:

```
notify: restart apache
```

can be added to: - Template deployment task - Lineinfile modification task - File replacement task

Handler still runs **only once** even if triggered multiple times.



## SECTION D — Create a Handler for Nginx (Alternative Example)

Create file:

```
nano nginx-handlers.yml
```

Add:

```

---
- name: Nginx handler example
  hosts: dev
  become: yes

  tasks:
    - name: Ensure Nginx installed
      package:
        name: nginx
        state: present

    - name: Insert custom config
      lineinfile:
        path: /etc/nginx/nginx.conf
        line: "# Managed by Ansible"
      notify: restart nginx

  handlers:
    - name: restart nginx
      service:
        name: nginx
        state: restarted

```

Run:

```
ansible-playbook nginx-handlers.yml
```

---

## Hands-On Checklist

- ☐ Create handler to restart Apache
- ☐ Notify handler when config changes
- ☐ Trigger handler from multiple tasks
- ☐ Use service module inside handlers
- ☐ Verify handler runs only when necessary

---

## Lab Summary

In this lab, you learned: - What handlers are - How notifications trigger handlers - How to restart services only when needed - How to build idempotent, efficient playbooks

Next Lab: 🙌 **Lab 12 — Error Handling** (`ignore_errors`, `failed_when`, `changed_when`)