**Handlers: Running Operations On Change**

As we’ve mentioned, modules should be idempotent and can relay when they have made a change on the remote system. Playbooks recognize this and have a basic event system that can be used to respond to change.

These ‘notify’ actions are triggered at the end of each block of tasks in a play, and will only be triggered once even if notified by multiple different tasks.

For instance, multiple resources may indicate that apache needs to be restarted because they have changed a config file, but apache will only be bounced once to avoid unnecessary restarts.

Here’s an example of restarting two services when the contents of a file change, but only if the file changes:

- name: template configuration file

template: src=template.j2 dest=/etc/foo.conf

notify:

- restart memcached

- restart apache

The things listed in the notify section of a task are called handlers.

Handlers are lists of tasks, not really any different from regular tasks, that are referenced by a globally unique name, and are notified by notifiers. If nothing notifies a handler, it will not run. Regardless of how many tasks notify a handler, it will run only once, after all of the tasks complete in a particular play.

Here’s an example handlers section:

handlers:

- name: restart memcached

service: name=memcached state=restarted

- name: restart apache

service: name=apache state=restarted

As of Ansible 2.2, handlers can also “listen” to generic topics, and tasks can notify those topics as follows:

handlers:

- name: restart memcached

service: name=memcached state=restarted

listen: "restart web services"

- name: restart apache

service: name=apache state=restarted

listen: "restart web services"

tasks:

- name: restart everything

command: echo "this task will restart the web services"

notify: "restart web services"

This use makes it much easier to trigger multiple handlers. It also decouples handlers from their names, making it easier to share handlers among playbooks and roles (especially when using 3rd party roles from a shared source like Galaxy).