

Ansible Jinja2 Templates - Template Module

In Ansible, **Jinja2 templates** allow you to create **dynamic configuration files** using variables, conditionals, and loops. These templates typically use the `.j2` file extension and are processed using the Jinja2 templating engine (the same engine used in Flask and other Python tools).

What Is a Jinja2 Template?

A Jinja2 template is a **text file with embedded variables and logic**, which Ansible renders into a final file during execution.

Example Use Case:

Creating a dynamic Nginx config based on host variables.

Basic Syntax

Variables

```
server_name {{ inventory_hostname }};
```

Loops

```
{% for site in sites %}
server {
    listen 80;
    server_name {{ site.name }};
    root {{ site.root }};
}
{% endfor %}
```

❓ Conditionals

```
{% if enable_ssl %}  
listen 443 ssl;  
{% else %}  
listen 80;  
{% endif %}
```

📁 Real Example: Using Templates in Ansible

1. Create a Template File

📄 `nginx.conf.j2`:

```
server {  
    listen {{ port }};  
    server_name {{ server_name }};  
  
    location / {  
        proxy_pass http://{{ proxy_target }};  
    }  
}
```

2. Playbook to Use Template

📄 `nginx-playbook.yml`:

```
---  
- name: Deploy Nginx Config  
  hosts: web  
  vars:  
    port: 80  
    server_name: example.com  
    proxy_target: 127.0.0.1:3000
```

tasks:

- name: Generate nginx config from template
template:
src: nginx.conf.j2
dest: /etc/nginx/sites-available/default

3. Run the Playbook

```
ansible-playbook -i inventory.ini nginx-playbook.yml
```

Common Jinja2 Filters

Filter	Description	Example
default	Sets a default value	`{{ var
lower	Converts string to lowercase	`{{ name
replace	Replaces substrings	`{{ path
join	Joins list into string	`{{ mylist
length	Gets length of list or string	`{{ mylist

Use Case Example: /etc/hosts Generator

 hosts.j2:

127.0.0.1 localhost

```
{% for host in groups['all'] %}  
{{ hostvars[host]['ansible_host'] }} {{ host }}  
{% endfor %}
```

 Playbook:

```
- hosts: localhost
  tasks:
    - template:
        src: hosts.j2
        dest: /tmp/hosts
```

 **Summary**

-  Jinja2 templates make config files dynamic.
 -  Support variables, conditionals, loops, and filters.
 -  Used via the `template` module in Ansible.
-