

Ansible 101

Missing pieces

Manuel Tiago Pereira

Conditionals

- Execution flow control, as in a ordinary programming language
- Evaluated based on variables and facts

when statement

- Execute if the Jinja2 expression returns `True`
- Can be applied to tasks, includes and roles

Conditions

- `==`, `!=` and `is`
- `and`, `or`, `not`

Useful filters

- `success` and `failed`
- `changed` and `skipped`
- `bool` and other casts

Loops

- Repeat execution of a task
- (Mostly) used with iterators
- Can be used together with the `when` statement

`with_items`

- Iterate over a list of items, using the iterator `item`
- The items can be linear or complex
- The items can be hardcoded or defined in a variable/fact

`with_nested`

- Nested loops over multiple lists of items
- Same logic as nested `for` loops in programming languages
- Access items with `item[0]`, `item[1]`, etc

`with_dict`

- Iterator over a dictionary
- Access keys with `item.key`
- Access values with `item.value.field_name`

`with_subelements`

- Iterator over a list of hashes, then on a list
- Access the hash item with `item.0` and the list item with `item.1`
- Other values of the hash item accessible with the usual dot syntax

Many others available

- Check http://docs.ansible.com/ansible/playbooks_loops.html for more
- Reflect on why you're needing to use complex loops!

You can also write your own

- Check http://docs.ansible.com/ansible/developing_plugins.html

A note on register

- A `registered` task with a loop will result in a different variable structure
- Use `registered_var.results` list for getting the multiple results

Templates

- Jinja2 templates with access to variables defined on Ansible
- Useful when `copy` is too much and `lineinfile` too little
 - Spoiler alert: We'll use them most of the time
- Avoid duplicated config files
- If's and loops for very dynamic templates

Noteworthy parameters

- `validate`: Run a command for config validation (requires passing the `%s` arg)
- `backup`: Backup original file to the passed name
- `force`: Defaults to `yes`. If set to `no`, only applies template if original is missing

Template-only variables

- `ansible_managed`: A note for sneaky admins locally editing files (AKA your future self ;-))
- `template_{host,uid,path,fullpath,run_date}`: Info on the Ansible's host, should not be needed

Example - template

```
{% set rails_site = item %}  
{% set rails_prestartprotocol = 'https' if rails_site.ssl is defined and rails_site.ssl  
PassengerPreStart {{ rails_prestartprotocol }}://{{ rails_site.vhostname }}:{{ rails_si  
PassengerSpawnMethod {{ rails_spawnmethod | default(smart) }}  
{% if rails_environment == "production" %}  
PassengerFriendlyErrorPages off  
{% else %}  
PassengerFriendlyErrorPages on  
{% endif %}
```

Example - variables

```
rails_environment: production

rails_sites:
- name: backoffice
  vhostname: backoffice.helabs.com
  port: 443
  ssl: true
```

Lookup plugins

- Another method for getting values from the system and external sources
- Lookups available for files, env, etcd and many more
- Check http://docs.ansible.com/ansible/playbooks_lookups.html

Tags

- Used for filtering the tasks that will be executed
- Filter is statically defined, per execution
- Tags can be applied to single tasks or to includes and roles
- Always namespace your tag names, for instance, `nginx_install`

Ansible Vault

- A method for storing secrets safely in your repository
- No-frills, transparent access to encrypted variables
- `ansible-vault` is a CRUD tool for encrypted YAML files
- Path for the password file/script on `ansible.cfg`'s `default` section
- Define wrapper variables for the encrypted files

Useful links

- Loops:
 - http://docs.ansible.com/ansible/playbooks_loops.html
 - <https://github.com/lorin/ansible-quickref#return-value-of-a-loop>
- Lookups: http://docs.ansible.com/ansible/playbooks_lookups.html
- Vault: http://docs.ansible.com/ansible/playbooks_vault.html
- Filters:
 - http://docs.ansible.com/ansible/playbooks_filters.html
 - <https://github.com/mitsuhiko/jinja2/blob/master/jinja2/filters.py>
 - <https://github.com/ansible/ansible/blob/devel/lib/ansible/plugins/fi>