

Ansible 2.4 Porting Guide

This section discusses the behavioral changes between Ansible 2.3 and Ansible 2.4.

It is intended to assist in updating your playbooks, plugins and other parts of your Ansible infrastructure so they will work with this version of Ansible.

We suggest you read this page along with [Ansible Changelog for 2.4](#) to understand what updates you may need to make.

This document is part of a collection on porting. The complete list of porting guides can be found at [ref: porting guides <porting_guides>](#).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel][docsite][rst][porting_guides]porting_guide_2.4.rst, line 15); [backlink](#)

Unknown interpreted text role "ref".

Topics

- [Python version](#)
- [Inventory](#)
 - [Initial playbook relative group_vars and host_vars](#)
- [Deprecated](#)
 - [Specifying Inventory sources](#)
 - [Use of multiple tags](#)
 - [Other caveats](#)
- [Modules](#)
 - [Modules removed](#)
 - [Deprecation notices](#)
 - [Noteworthy module changes](#)
- [Plugins](#)
 - [Vars plugin changes](#)
 - [Inventory plugins](#)
 - [Callback plugins](#)
 - [Template lookup plugin: Escaping Strings](#)
- [Tests](#)
 - [Tests succeeded/failed](#)
- [Networking](#)
 - [Persistent Connection](#)
- [Configuration](#)

Python version

Ansible will not support Python 2.4 or 2.5 on the target hosts anymore. Going forward, Python 2.6+ will be required on targets, as already is the case on the controller.

Inventory

Inventory has been refactored to be implemented via plugins and now allows for multiple sources. This change is mostly transparent to users.

One exception is the `inventory_dir`, which is now a host variable; previously it could only have one value so it was set globally. This means you can no longer use it early in plays to determine `hosts`: or similar keywords. This also changes the behaviour of `add_hosts` and the implicit localhost; because they no longer automatically inherit the global value, they default to `None`. See the module documentation for more information.

The `inventory_file` remains mostly unchanged, as it was always host specific.

Since there is no longer a single inventory, the 'implicit localhost' doesn't get either of these variables defined.

A bug was fixed with the inventory path/directory, which was defaulting to the current working directory. This caused `group_vars` and `host_vars` to be picked up from the current working directory instead of just adjacent to the playbook or inventory directory when a host list (comma separated host names) was provided as inventory.

Initial playbook relative group_vars and host_vars

In Ansible versions prior to 2.4, the inventory system would maintain the context of the initial playbook that was executed. This

allowed successively included playbooks from other directories to inherit `group_vars` and `host_vars` placed relative to the top level playbook file.

Due to some behavioral inconsistencies, this functionality will not be included in the new inventory system starting with Ansible version 2.4.

Similar functionality can still be achieved by using `vars_files`, `include_vars`, or `group_vars` and `host_vars` placed relative to the inventory file.

Deprecated

Specifying Inventory sources

Use of `--inventory-file` on the command line is now deprecated. Use `--inventory` or `-i`. The associated ini configuration key, `hostfile`, and environment variable, `ANSIBLE_HOSTS`, are also deprecated. Replace them with the configuration key `inventory` and environment variable `envvar:ANSIBLE_INVENTORY`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 57); [backlink](#)
Unknown interpreted text role "envvar".

Use of multiple tags

Specifying `--tags` (or `--skip-tags`) multiple times on the command line currently leads to the last one overriding all the previous ones. This behavior is deprecated. In the future, if you specify `--tags` multiple times the tags will be merged together. From now on, using `--tags` multiple times on one command line will emit a deprecation warning. Setting the `merge_multiple_cli_tags` option to `True` in the `ansible.cfg` file will enable the new behavior.

In 2.4, the default has change to merge the tags. You can enable the old overwriting behavior via the config option.

In 2.5, multiple `--tags` options will be merged with no way to go back to the old behavior.

Other caveats

No major changes in this version.

Modules

Major changes in popular modules are detailed here

- The `ref:win_shell <win_shell_module>` and `ref:win_command <win_command_module>` modules now properly preserve quoted arguments in the command-line. Tasks that attempted to work around the issue by adding extra quotes/escaping may need to be reworked to remove the superfluous escaping. See [Issue 23019](#) for additional detail.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 81); [backlink](#)
Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 81); [backlink](#)
Unknown interpreted text role "ref".

Modules removed

The following modules no longer exist:

- None

Deprecation notices

The following modules will be removed in Ansible 2.8. Please update your playbooks accordingly.

- azure, use `ref:azure_rm_virtualmachine <azure_rm_virtualmachine_module>`, which uses the new Resource Manager SDK.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 95); [backlink](#)

Unknown interpreted text role "ref".

- `win_msi`, use `ref:win_package <win_package_module>` instead

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 96); [backlink](#)

Unknown interpreted text role "ref".

Noteworthy module changes

- The `ref:win_get_url <win_get_url_module>` module has the dictionary `win_get_url` in its results deprecated, its content is now also available directly in the resulting output, like other modules. This dictionary will be removed in Ansible 2.8.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 101); [backlink](#)

Unknown interpreted text role "ref".

- The `ref:win_unzip <win_unzip_module>` module no longer includes the dictionary `win_unzip` in its results; the contents are now included directly in the resulting output, like other modules.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 102); [backlink](#)

Unknown interpreted text role "ref".

- The `ref:win_package <win_package_module>` module return values `exit_code` and `restart_required` have been deprecated in favor of `rc` and `reboot_required` respectively. The deprecated return values will be removed in Ansible 2.6.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 103); [backlink](#)

Unknown interpreted text role "ref".

Plugins

A new way to configure and document plugins has been introduced. This does not require changes to existing setups but developers should start adapting to the new infrastructure now. More details will be available in the developer documentation for each plugin type.

Vars plugin changes

There have been many changes to the implementation of vars plugins, but both users and developers should not need to change anything to keep current setups working. Developers should consider changing their plugins take advantage of new features.

The most notable difference to users is that vars plugins now get invoked on demand instead of at inventory build time. This should make them more efficient for large inventories, especially when using a subset of the hosts.

Note

- This also creates a difference with `group/host_vars` when using them adjacent to playbooks. Before, the 'first' playbook loaded determined the variables; now the 'current' playbook does. We are looking to fix this soon, since 'all playbooks' in the path should be considered for variable loading.
- In 2.4.1 we added a toggle to allow you to control this behaviour, 'top' will be the pre 2.4, 'bottom' will use the current playbook hosting the task and 'all' will use them all from top to bottom.

Inventory plugins

Developers should start migrating from hardcoded inventory with dynamic inventory scripts to the new Inventory Plugins. The scripts will still work via the `script` inventory plugin but Ansible development efforts will now concentrate on writing plugins rather than enhancing existing scripts.

Both users and developers should look into the new plugins because they are intended to alleviate the need for many of the hacks and workarounds found in the dynamic inventory scripts.

Callback plugins

Users:

- Callbacks are now using the new configuration system. Users should not need to change anything as the old system still works, but you might see a deprecation notice if any callbacks used are not inheriting from the built in classes. Developers need to update them as stated below.

Developers:

- If your callback does not inherit from `CallbackBase` (directly or indirectly via another callback), it will still work, but issue a deprecation notice. To avoid this and ensure it works in the future change it to inherit from `CallbackBase` so it has the new options handling methods and properties. You can also implement the new options handling methods and properties but that won't automatically inherit changes added in the future. You can look at `CallbackBase` itself and/or `AnsiblePlugin` for details.
- Any callbacks inheriting from other callbacks might need to also be updated to contain the same documented options as the parent or the options won't be available. This is noted in the developer guide.

Template lookup plugin: Escaping Strings

Prior to Ansible 2.4, backslashes in strings passed to the template lookup plugin would be escaped automatically. In 2.4, users are responsible for escaping backslashes themselves. This change brings the template lookup plugin inline with the template module so that the same backslash escaping rules apply to both.

If you have a template lookup like this:

```
- debug:
  msg: '{{ lookup("template", "template.j2") }}'
```

OLD In Ansible 2.3 (and earlier) `:file:'template.j2'` would look like this:

```
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-
devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst]
[porting_guides]porting_guide_2.4.rst, line 160); backlink
Unknown interpreted text role "file".
```

```
{{ "name surname" | regex_replace("^[^\s]+\s+(.*)", "\1") }}
```

NEW In Ansible 2.4 it should be changed to look like this:

```
{{ "name surname" | regex_replace("^[^\\s]+\\s+(.*)", "\\1") }}
```

Tests

Tests succeeded/failed

Prior to Ansible version 2.4, a task return code of `rc` would override a return code of `failed`. In version 2.4, both `rc` and `failed` are used to calculate the state of the task. Because of this, test plugins `succeeded/failed` have also been changed. This means that overriding a task failure with `failed_when: no` will result in `succeeded/failed` returning `True/False`. For example:

```
- command: /bin/false
  register: result
  failed_when: no

- debug:
  msg: 'This is printed on 2.3'
  when: result|failed

- debug:
  msg: 'This is printed on 2.4'
  when: result|succeeded

- debug:
  msg: 'This is always printed'
  when: result.rc != 0
```

As we can see from the example above, in Ansible 2.3 `succeeded/failed` only checked the value of `rc`.

Networking

There have been a number of changes to how Networking Modules operate.

Playbooks should still use `connection: local`.

Persistent Connection

The configuration variables `connection_retries` and `connect_interval` which were added in Ansible 2.3 are now deprecated. For Ansible 2.4 and later use `connection_retry_timeout`.

To control timeouts use `command_timeout` rather than the previous top level `timeout` variable under `[default]`

See [ref: Ansible Network debug guide <network_debug_troubleshooting>](#) for more information.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\porting_guides\[ansible-devel] [docs] [docsite] [rst] [porting_guides]porting_guide_2.4.rst, line 212); [backlink](#)

Unknown interpreted text role "ref".

Configuration

The configuration system has had some major changes. Users should be unaffected except for the following:

- All relative paths defined are relative to the *ansible.cfg* file itself. Previously they varied by setting. The new behavior should be more predictable.
- A new macro `{{CWD}}` is available for paths, which will make paths relative to the 'current working directory', this is unsafe but some users really want to rely on this behaviour.

Developers that were working directly with the previous API should revisit their usage as some methods (for example, `get_config`) were kept for backwards compatibility but will warn users that the function has been deprecated.

The new configuration has been designed to minimize the need for code changes in core for new plugins. The plugins just need to document their settings and the configuration system will use the documentation to provide what they need. This is still a work in progress; currently only 'callback' and 'connection' plugins support this. More details will be added to the specific plugin developer guides.