Ansible-core 2.11 Porting Guide

This section discusses the behavioral changes between `ansible-base` 2.10 and `ansible-core` 2.11.

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

We suggest you read this page along with the ansible-core Changelog for 2.11 to understand what updates you may need to make.

`ansible-core` is mainly of interest for developers and users who only want to use a small, controlled subset of the available collections. Regular users should install Ansible.

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)(docs)(docsite)(rst)(porting_guides)porting_guide_core_2.11.rst`, **line 16);** *backlink*
>
> Unknown interpreted text role "ref".

Contents

Playbook

- The `jinja2_native` setting now does not affect the template module which implicitly returns strings. For the template lookup there is a new argument `jinja2_native` (off by default) to control that functionality. The rest of the Jinja2 expressions still operate based on the `jinja2_native` setting.

Command Line

- The `ansible-galaxy login` command has been removed, as the underlying API it used for GitHub auth has been shut down. Publishing roles or collections to Galaxy with `ansible-galaxy` now requires that a Galaxy API token be passed to the CLI using a token file (default location `~/.ansible/galaxy_token`) or (insecurely) with the `--token` argument to `ansible-galaxy`.

Deprecated

The constant `ansible.module_utils.basic._CHECK_ARGUMENT_TYPES_DISPATCHER` is deprecated. Use :const:`ansible.module_utils.common.parameters.DEFAULT_TYPE_VALIDATORS` 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_core_2.11.rst`, **line 35);** *backlink*
>
> Unknown interpreted text role "const".

Breaking Changes

Changes to `AnsibleModule`

With the move to :class:`ArgumentSpecValidator <ansible.module_utils.common.arg_spec.ArgumentSpecValidator>` for performing argument spec validation, the following private methods in :class:`AnsibleModule <ansible.module_utils.basic.AnsibleModule>` have

been removed:

- `_check_argument_types()`
- `_check_argument_values()`
- `_check_arguments()`
- `_check_mutually_exclusive()` --> :func:`ansible.module_utils.common.validation.check_mutually_exclusive`

- `_check_required_arguments()` --> :func:`ansible.module_utils.common.validation.check_required_arguments`

- `_check_required_by()` --> :func:`ansible.module_utils.common.validation.check_required_by`

- `_check_required_if()` --> :func:`ansible.module_utils.common.validation.check_required_if

- `_check_required_one_of()` --> :func:`ansible.module_utils.common.validation.check_required_one_of

- `_check_required_together()` --> :func:`ansible.module_utils.common.validation.check_required_together`

- `_check_type_bits()` --> :func:`ansible.module_utils.common.validation.check_type_bits`

- `_check_type_bool()` --> :func:`ansible.module_utils.common.validation.check_type_bool`

- `_check_type_bytes()` --> :func:`ansible.module_utils.common.validation.check_type_bytes`

- `_check_type_dict()` --> :func:`ansible.module_utils.common.validation.check_type_dict`

- `_check_type_float()` --> :func:`ansible.module_utils.common.validation.check_type_float`

- `_check_type_int()` --> :func:`ansible.module_utils.common.validation.check_type_int`

- `_check_type_jsonarg()` --> :func:`ansible.module_utils.common.validation.check_type_jsonarg`

(docs)(docsite)(rst)(porting_guides)porting_guide_core_2.11.rst, **line 61**); *backlink*

Unknown interpreted text role "func".

- `_check_type_list()` --> :func:`ansible.module_utils.common.validation.check_type_list`

  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_core_2.11.rst, **line 62**); *backlink*

  Unknown interpreted text role "func".

- `_check_type_path()` --> :func:`ansible.module_utils.common.validation.check_type_path`

  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_core_2.11.rst, **line 63**); *backlink*

  Unknown interpreted text role "func".

- `_check_type_raw()` --> :func:`ansible.module_utils.common.validation.check_type_raw`

  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_core_2.11.rst, **line 64**); *backlink*

  Unknown interpreted text role "func".

- `_check_type_str()` --> :func:`ansible.module_utils.common.validation.check_type_str`

  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_core_2.11.rst, **line 65**); *backlink*

  Unknown interpreted text role "func".

- `_count_terms()` --> :func:`ansible.module_utils.common.validation.count_terms`

  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_core_2.11.rst, **line 66**); *backlink*

  Unknown interpreted text role "func".

- `_get_wanted_type()`
- `_handle_aliases()`
- `_handle_no_log_values()`
- `_handle_options()`
- `_set_defaults()`
- `_set_fallbacks()`

Modules or plugins using these private methods should use the public functions in :mod:`ansible.module_utils.common.validation` or :meth:`ArgumentSpecValidator.validate() <ansible.module_utils.common.arg_spec.ArgumentSpecValidator.validate>` if no public function was listed above.

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_core_2.11.rst, **line 74**); *backlink*

Unknown interpreted text role "mod".

Changes to :mod:`ansible.module_utils.common.parameters`

The following functions in :mod:`ansible.module_utils.common.parameters` are now private and should not be used directly. Use :meth:`ArgumentSpecValidator.validate() <ansible.module_utils.common.arg_spec.ArgumentSpecValidator.validate>` instead.

- `list_no_log_values`
- `list_deprecations`
- `handle_aliases`

Other

- **Upgrading**: If upgrading from `ansible < 2.10` or from `ansible-base` and using pip, you must `pip uninstall ansible` or `pip uninstall ansible-base` before installing `ansible-core` to avoid conflicts.
- Python 3.8 on the controller node is a soft requirement for this release. `ansible-core` 2.11 still works with the same versions of Python that `ansible-base` 2.10 worked with, however 2.11 emits a warning when running on a controller node with a Python version less than 3.8. This warning can be disabled by setting `ANSIBLE_CONTROLLER_PYTHON_WARNING=False` in your environment. `ansible-core` 2.12 will require Python 3.8 or greater.
- The configuration system now validates the `choices` field, so any settings that violate it and were ignored in 2.10 cause an error in 2.11. For example, `ANSIBLE_COLLECTIONS_ON_ANSIBLE_VERSION_MISMATCH=0` now causes an error (valid choices are `ignore`, `warn` or `error`).
- The `ansible-galaxy` command now uses `resolvelib` for resolving dependencies. In most cases this should not make a user-facing difference beyond being more performant, but we note it here for posterity and completeness.
- If you import Python `module_utils` into any modules you maintain, you may now mark the import as optional during the module payload build by wrapping the `import` statement in a `try` or `if` block. This allows modules to use `module_utils` that may not be present in all versions of Ansible or a collection, and to perform arbitrary recovery or fallback actions during module runtime.

Modules

- The `apt_key` module has explicitly defined `file` as mutually exclusive with `data`, `keyserver` and `url`. They cannot be used together anymore.
- The `meta` module now supports tags for user-defined tasks. Set the task's tags to 'always' to maintain the previous behavior. Internal `meta` tasks continue to always run.

Modules removed

The following modules no longer exist:

- No notable changes

Deprecation notices

No notable changes

Noteworthy module changes

- facts - On NetBSD, `ansible_virtualization_type` now tries to report a more accurate result than `xen` when virtualized and not running on Xen.
- facts - Virtualization facts now include `virtualization_tech_guest` and `virtualization_tech_host` keys. These are lists of virtualization technologies that a guest is a part of, or that a host provides, respectively. As an example, if you set up a host to provide both KVM and VirtualBox, both values are included in `virtualization_tech_host`. Similarly, a podman container running on a VM powered by KVM has a `virtualization_tech_guest` of `["kvm", "podman", "container"]`.
- The parameter `filter` type is changed from `string` to `list` in the :ref:`setup <setup_module>` module in order to use more than one filter. Previous behavior (using a `string`) still remains and works as a single filter.

> **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_core_2.11.rst`, line 123); *backlink*
>
> Unknown interpreted text role "ref".

Plugins

- inventory plugins - `CachePluginAdjudicator.flush()` now calls the underlying cache plugin's `flush()` instead of only deleting keys that it knows about. Inventory plugins should use `delete()` to remove any specific keys. As a user, this means that when an inventory plugin calls its `clear_cache()` method, facts could also be flushed from the cache. To work around this, users can configure inventory plugins to use a cache backend that is independent of the facts cache.
- callback plugins - `meta` task execution is now sent to `v2_playbook_on_task_start` like any other task. By default, only explicit meta tasks are sent there. Callback plugins can opt-in to receiving internal, implicitly created tasks to act on those as well, as noted in the plugin development documentation.
- The `choices` are now validated, so plugins that were using incorrect or incomplete choices issue an error in 2.11 if the value provided does not match. This has a simple fix: update the entries in `choices` to match reality.

Porting custom scripts

No notable changes