

Callback plugins

- [Example callback plugins](#)
- [Enabling callback plugins](#)
- [Setting a callback plugin for `ansible-playbook`](#)
- [Setting a callback plugin for ad hoc commands](#)
- [Plugin list](#)

Callback plugins enable adding new behaviors to Ansible when responding to events. By default, callback plugins control most of the output you see when running the command line programs, but can also be used to add additional output, integrate with other tools and marshal the events to a storage backend. If necessary, you can [ref: create custom callback plugins <developing_callbacks>](#).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 10); [backlink](#)

Unknown interpreted text role "ref".

Example callback plugins

The [ref: log_plays <log_plays_callback>](#) callback is an example of how to record playbook events to a log file, and the [ref: mail <mail_callback>](#) callback sends email on playbook failures.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 17); [backlink](#)

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 17); [backlink](#)

Unknown interpreted text role "ref".

The [ref: say <say_callback>](#) callback responds with computer synthesized speech in relation to playbook events.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 19); [backlink](#)

Unknown interpreted text role "ref".

Enabling callback plugins

You can activate a custom callback by either dropping it into a `callback_plugins` directory adjacent to your play, inside a role, or by putting it in one of the callback directory sources configured in [ref: ansible.cfg <ansible_configuration_settings>](#).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 26); [backlink](#)

Unknown interpreted text role "ref".

Plugins are loaded in alphanumeric order. For example, a plugin implemented in a file named `1_first.py` would run before a plugin file named `2_second.py`.

Most callbacks shipped with Ansible are disabled by default and need to be enabled in your [ref: ansible.cfg <ansible_configuration_settings>](#) file in order to function. For example:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 30); [backlink](#)

Unknown interpreted text role "ref".

```
#callbacks_enabled = timer, mail, profile_roles, collection_namespace.collection_name.custom_callback
```

Setting a callback plugin for ansible-playbook

You can only have one plugin be the main manager of your console output. If you want to replace the default, you should define `CALLBACK_TYPE = stdout` in the subclass and then configure the stdout plugin in [ref`ansible.cfg`](#) `<ansible_configuration_settings>`. For example:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 39); [backlink](#)

Unknown interpreted text role "ref".

```
stdout_callback = dense
```

or for my custom callback:

```
stdout_callback = mycallback
```

This only affects [ref`ansible-playbook`](#) by default.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 51); [backlink](#)

Unknown interpreted text role "ref".

Setting a callback plugin for ad hoc commands

The [ref`ansible`](#) ad hoc command specifically uses a different callback plugin for stdout, so there is an extra setting in [ref`ansible_configuration_settings`](#) you need to add to use the stdout callback defined above:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 56); [backlink](#)

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 56); [backlink](#)

Unknown interpreted text role "ref".

```
[defaults]
bin_ansible_callbacks=True
```

You can also set this as an environment variable:

```
export ANSIBLE_LOAD_CALLBACK_PLUGINS=1
```

Plugin list

You can use `ansible-doc -t callback -l` to see the list of available plugins. Use `ansible-doc -t callback <plugin name>` to see specific documents and examples.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\ansible-devel\docs\docsite\rst\plugins\[ansible-devel] [docs] [docsite] [rst] [plugins] callback.rst, line 78)

Unknown directive type "seealso".

```
.. seealso::
```

```
:ref:`action_plugins`
```

```
    Action plugins
:ref:`cache_plugins`
    Cache plugins
:ref:`connection_plugins`
    Connection plugins
:ref:`inventory_plugins`
    Inventory plugins
:ref:`shell_plugins`
    Shell plugins
:ref:`strategy_plugins`
    Strategy plugins
:ref:`vars_plugins`
    Vars plugins
`User Mailing List <https://groups.google.com/forum/#!forum/ansible-devel>`_
    Have a question? Stop by the google group!
:ref:`communication_irc`
    How to join Ansible chat channels
```