ansible-lint Ansible linter

START HERE

ansible-lint
Lint everything found

ansible-lint *<playbook.yml>*
Lint one file

ansible-lint *<dir>*
Lint a folder

ansible-lint --fix
Auto-fix what it can

ansible-lint --profile *<name>*
Pick a strictness

ansible-lint -L
List all rules

ansible-lint -x *<rule>*
Skip a rule

ansible-lint -f json
Machine output

min loads at all → **basic** style, syntax → **moderate** readability → **safety** deterministic → **shared** for Galaxy → **production** for AAP

Profiles are cumulative: each one includes the rules of the previous

CONCEPTS

Without paths it auto-detects the project from the git root and lints playbooks, roles and collections it finds.

Rules can have sub-ids such as fqcn[action-core]; skipping or noqa-ing the bare id covers all of them.

Exit code 2 means violations, and warn-list rules still show but do not fail the run (unless --strict).

RUN IT

ansible-lint
Lint the project (auto-detect)

ansible-lint *<file> <dir>*
Lint given files or folders

ansible-lint --fix
Apply all safe auto-fixes
Also reformats YAML. Commit first so you can review the diff.

ansible-lint --fix=*<rule>*,*<tag>*
Fix only chosen rules or tags

ansible-lint --profile *<name>*
Check a profile's rule set

ansible-lint --exclude *<path>*
Skip files or folders

ansible-lint -c *<file>*
Use a specific config file

ansible-lint --offline
No requirement installs or schema refresh

ansible-lint -s
Fail on warnings too

ansible-lint -r *<dir>* **-R**
Add custom rules, keep built-ins

ansible-lint --version
Show version and ansible-core

RULES AND PROFILES

ansible-lint -L
List all rules

ansible-lint -L -f brief
Rule ids with short text

ansible-lint -T
List tags and their rules

ansible-lint -P
Describe the profiles

RULES AND PROFILES (cont.)

ansible-lint -x *<rule>*
Skip rules or tags

ansible-lint -w *<rule>*
Only warn for these rules

ansible-lint -t *<rule>*
Check only these rules or tags

ansible-lint --enable-list *<tag>*
Turn on opt-in rules
Rules tagged opt-in (e.g. no-log-password) are off by default.

ansible-lint --generate-ignore
Write .ansible-lint-ignore baseline

ansible-lint -i *<file>*
Use a specific ignore file

OUTPUT

ansible-lint -f *<format>*
brief, full, md, pep8, quiet, json, sarif

ansible-lint -f json
Code Climate JSON on stdout

ansible-lint -f pep8
One line per finding

ansible-lint -f sarif --sarif-file *<out>*
SARIF file for code scanning

ansible-lint -q
Quieter output

ansible-lint -v / -vv
More verbose output

ansible-lint --show-relpath
Paths relative to CWD

ansible-lint --nocolor
Plain output (NO_COLOR=1)

ansible-lint --force-color
Colour even when piped

INLINE SKIPS

 # noqa: *<rule>*
Skip a rule on this line

 # noqa: *<rule1> <rule2>*
Skip several rules on a line

tags: [skip_ansible_lint]
Skip a whole task (legacy)

ANSIBLE_LINT_NODEPS=1
Skip installing dependencies

INLINE SKIPS (cont.)

Put the noqa comment on the line the violation is reported on; for a play or task name, that is the name line.

CONFIG FILE

.ansible-lint
Default config, YAML in project root

profile: production
Default profile

exclude_paths: [*<path>***]**
Paths never linted

skip_list: [*<rule>***]**
Rules never reported

warn_list: [*<rule>***]**
Rules reported as warnings

enable_list: [*<tag>***]**
Opt-in rules to activate

offline: true
Never fetch requirements or schemas

strict: true
Warnings fail the run

var_naming_pattern: *<regex>*
Pattern for variable names

kinds:
Map path globs to file kinds

mock_modules: [*<name>***]**
Pretend a missing module exists

mock_roles: [*<name>***]**
Pretend a missing role exists

.ansible-lint-ignore
Per-file rule ignores, one per line
Format: '<path> <rule>'. Created by --generate-ignore.

write_list: [*<rule>***]**
Rules --fix may rewrite

COMMON RULES

fqcn
Use fully qualified module names

name[missing]
Tasks need a name

name[casing]
Names start uppercase

yaml
YAML style (via yamllint)

no-changed-when
command/shell need changed_when

COMMON RULES (cont.)

risky-file-permissions
Set mode on file-creating modules

command-instead-of-module
Use a module instead of curl or rm

package-latest
Avoid state: latest

var-naming
Variable naming conventions

risky-shell-pipe
Set pipefail when piping in shell

key-order
name first in a task

jinja[spacing]
Spaces inside {{ }}

no-handler
Use a handler for changed-based tasks

ignore-errors
Prefer failed_when over ignore_errors

schema
File structure fails its schema

syntax-check
ansible-playbook --syntax-check fails

RECIPES

Project config

```
# .ansible-lint
profile: safety
exclude_paths:
  - .cache/
  - molecule/
skip_list:
  - yaml[line-length]
warn_list:
  - experimental
```

Adopt in a legacy repo

```
ansible-lint --generate-ignore
# commit .ansible-lint-ignore,
then only new
# violations fail the build
```

Pre-commit hook

```
# .pre-commit-config.yaml
repos:
  - repo: https://github.com/
ansible/ansible-lint
    rev: <tag>
    hooks:
      - id: ansible-lint
```

GitHub Actions step

```
- uses: actions/checkout@v6
- uses: ansible/ansible-
lint@<tag>
  with:
    args: --profile production
```