

- 1- <https://github.com/CiscoDevNet/ansible-aci>
 1. <https://github.com/CiscoDevNet/ansible-aci/issues/598>
Network object not working as expected → Broken dependency chain and idempotency for creation and checking the expected outcome and adding 'debug' module.
 2. <https://github.com/CiscoDevNet/ansible-aci/pull/595>
A 'direct link' for a dependency file has been used that got outdated → failed playbook → updated the link.
 3. <https://github.com/CiscoDevNet/ansible-aci/issues/28>
A 'direct path' for a dependency file has been used that can get changed → failed playbook → updated the link.
- 2- <https://github.com/alvistack/ansible-collection-kubernetes>
 1. <https://github.com/alvistack/ansible-collection-kubernetes/issues/11>
A dependency installation task is failing the playbook and not working as expected → Broken dependency chain and idempotency check needed.
- 3- <https://github.com/linux-system-roles/linux-system-roles.github.io>
 1. <https://github.com/linux-system-roles/linux-system-roles.github.io/issues/70>
The error was caused from the outdated version of a dependency
Also the absolute path used in dependency files caused error.
- 4- <https://github.com/ansible-collections/cisco.nxos>
 1. <https://github.com/ansible-collections/cisco.nxos/issues/801>
 2. <https://github.com/ansible-collections/cisco.nxos/issues/803>
The task does not check the idempotency when writing in a configuration file.
Task used absolute path for a dependency file which later changed and resulted in the Broken dependency chain and task failure.
 3. <https://github.com/ansible-collections/cisco.nxos/issues/698>
Does not check the idempotency when creating/deleting accounts and users.
Task needs to sanity check the system to know the changes are in the desired state.
 4. <https://github.com/ansible-collections/cisco.nxos/issues/787>
Invalid commands should show the failure and they can break idempotency and cause Broken dependency chain.
 5. <https://github.com/ansible-collections/cisco.nxos/issues/542>
Usage of a 'command' module is failing the task.
Should provide necessary failing comments and state → unguarded operation

6. <https://github.com/ansible-collections/cisco.nxos/issues/420>
The task should have 'gather_facts' component to avoid issues and further errors
Avoid Broken dependency chain
 7. <https://github.com/ansible-collections/cisco.nxos/issues/408>
Task does not check for idempotency. Have to apply more checking components.
Does not use the idempotency safe component when writing a file.
 8. <https://github.com/ansible-collections/cisco.nxos/issues/391>
Ignoring a task and not checking the execution can cause failure and Broken dependency chain.
- 5- <https://github.com/oravirt/ansible-oracle>
1. <https://github.com/oravirt/ansible-oracle/issues/335>
One of the packages/ dependencies became updated but the script worked with the outdated version.
 2. <https://github.com/oravirt/ansible-oracle/issues/327>
The need of having 'assert' to check if the changes are applied to prevent issue.
Avoid having broken dependency chain by using assert.
 3. <https://github.com/oravirt/ansible-oracle/issues/265>
Using systemd caused failure in the execution of the task. Should have applied more checking components to avoid issues.
- 6- <https://github.com/cisco-en-programmability/dnacenter-ansible>
1. <https://github.com/cisco-en-programmability/dnacenter-ansible/issues/103>
Hardcoded configuration parameters that can cause failure once changed or be different when running on different environments.
 2. <https://github.com/cisco-en-programmability/dnacenter-ansible/issues/102>
Hardcoded configuration parameters that can cause failure once changed or be different when running on different environments.
 3. <https://github.com/cisco-en-programmability/dnacenter-ansible/issues/62>
Should perform the file update with a safe ansible module.
Used absolute path for a directory to download dependencies that can lead to issues when it's not the same as other machines.
- 7- <https://github.com/fortinet-ansible-dev/ansible-galaxy-fortimanager-collection>
1. <https://github.com/fortinet-ansible-dev/ansible-galaxy-fortimanager-collection/issues/79>
A 'direct link' for a dependency file has been used that got outdated → failed playbook → updated the link.

2. <https://github.com/fortinet-ansible-dev/ansible-galaxy-fortimanager-collection/issues/74>
Hardcoded configuration parameters and authentication credentials caused failure because of different versions of a dependency installed by the script.
Also has to ensure the same stable version is installed with all the executions.
 3. <https://github.com/fortinet-ansible-dev/ansible-galaxy-fortimanager-collection/issues/18>
The missing 'state' component caused failure in task execution. Having this checking component will ensure the desired state of the machine after execution.
- 8- <https://github.com/ansible-collections/cisco.asa>
1. <https://github.com/ansible-collections/cisco.asa/issues/196>
The task is using a module that is not functioning as expected as in the previous versions of the package before updating the package version. Developers should check the package before updating in the script.
 2. <https://github.com/ansible-collections/cisco.asa/issues/187>
Os operations (changing the ACL for a file) has been done without applying proper confirmation for the changes in the task and without using the correct ansible attributes.
 3. <https://github.com/ansible-collections/cisco.asa/issues/197>
The task has a version specific installation of a dependency that got outdated and caused other tasks to fail.
 4. <https://github.com/ansible-collections/cisco.asa/issues/195>
2 versions of the dependencies are not compatible with each other after updating.
 5. <https://github.com/ansible-collections/cisco.asa/issues/166>
Did not check the desired state of the machine after creating a group object. Should have used 'state' or assert.