

orphan:

## Testing using the Legacy Integration system

### Topics

- [Testing using the Legacy Integration system](#)
  - [Running Cloud Tests](#)
  - [Other configuration](#)
  - [IAM policies for AWS](#)
    - [testing-iam-policy.json.j2](#)
    - [Other Definitions required](#)
  - [Running Tests](#)

This page details how to run the integration tests that haven't been ported to the new `ansible-test` framework.

The following areas are still tested using the `legacy make tests` command:

- amazon (some)
- azure
- cloudflare
- cloudscale
- cloudstack
- consul
- exoscale
- gce
- jenkins
- rackspace

Over time the above list will be reduced as tests are ported to the `ansible-test` framework.

### Running Cloud Tests

Cloud tests exercise capabilities of cloud modules (for example, `ec2_key`). These are not 'tests run in the cloud' so much as tests that use the cloud modules and are organized by cloud provider.

Some AWS tests may use environment variables. It is recommended to either unset any AWS environment variables( such as `AWS_DEFAULT_PROFILE`, `AWS_SECRET_ACCESS_KEY`, and so on) or be sure that the environment variables match the credentials provided in `credentials.yml` to ensure the tests run with consistency to their full capability on the expected account. See [AWS CLI docs](#) for information on creating a profile.

Subsets of tests may be run by `#commenting` out unnecessary roles in the appropriate playbook, such as `test/integration/amazon.yml`.

In order to run cloud tests, you must provide access credentials in a file named `credentials.yml`. A sample credentials file named `credentials.template` is available for syntax help.

Provide cloud credentials:

```
cp credentials.template credentials.yml
${EDITOR:-vi} credentials.yml
```

### Other configuration

In order to run some tests, you must provide access credentials in a file named `credentials.yml`. A sample credentials file named `credentials.template` is available for syntax help.

### IAM policies for AWS

In order to run the tests in an AWS account ansible needs fairly wide ranging powers which can be provided to a dedicated user or temporary credentials using a specific policy configured in the AWS account.

#### `testing-iam-policy.json.j2`

The `testing-iam-policy.json.j2` file contains a policy which can be given to the user running the tests to give close to minimum rights required to run the tests. Please note that this does not fully restrict the user; The user has wide privileges for viewing account definitions and is also able to manage some resources that are not related to testing (for example, AWS lambdas with different names) primarily due to the limitations of the Amazon ARN notation. At the very least the policy limits the user to one region, however tests should not be run in a primary production account in any case.

### Other Definitions required

Apart from installing the policy and giving it to the user identity running the tests, a lambda role `ansible_integration_tests` has to be created which has lambda basic execution privileges.

## Running Tests

The tests are invoked via a `Makefile`.

If you haven't already got Ansible available use the local checkout by doing:

```
source hacking/env-setup
```

Run the tests by doing:

```
cd test/integration/  
# TARGET is the name of the test from the list at the top of this page  
#make TARGET  
# for example  
make amazon  
# To run all cloud tests you can do:  
make cloud
```

### Warning

Possible cost of running cloud tests

Running cloud integration tests will create and destroy cloud resources. Running these tests may result in additional fees associated with your cloud account. Care is taken to ensure that created resources are removed. However, it is advisable to inspect your AWS console to ensure no unexpected resources are running.