



链滴

ansible-playbook 之 lookup 插件的使用

作者: [Smiteli](#)

原文链接: <https://ld246.com/article/1585392727097>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

ansible-playbook之lookup插件的使用

官方资料: [链接](#)

一、Using lookup plugins

Lookup plugins can be used anywhere you can use templating in Ansible: in a play, in variable file, or in a Jinja2 template for the [template](#) module.

下面为lookup.yml的内容

```
---
- hosts: localhost
  gather_facts: true
  vars:
    host_name: 55.qiweioa.cn
    file_contents: "{{lookup('file', 'file.txt')}}"
  tasks:
    - name: the contents of file.txt
      debug:
        msg: "{{file_contents}}"
```

[root@VM_129_89_centos playbook]# tree -L 1

```
.
|-- file.txt
|-- hosts
|-- lookup.yml
`-- playbook.yml
```

上面为目录结构, 运行后可获得file.txt里的内容。

Lookups are an integral part of loops. Wherever you see [with_](#), the part after the underscore is the name of a lookup. This is also the reason most lookups output lists and take lists as input; or example, [with_items](#) uses the [items](#) lookup:

```
tasks:
- name: count to 3
  debug: msg={{item}}
  with_items: [1, 2, 3]
```

You can combine lookups with [Filters](#), [Tests](#) and even each other to do some complex data generation and manipulation. For example:

```
tasks:
- name: valid but useless and over complicated chained lookups and filters
  debug: msg="find the answer here:\n{{ lookup('url', 'https://google.com/search?q=' + item
urlencode)|join(' ') }}"
  with_nested:
    - "{{lookup('consul_kv', 'bcs/' + lookup('file', '/the/question') + ', host=localhost, port=200
')|shuffle}}"
    - "{{lookup('sequence', 'end=42 start=2 step=2')|map('log', 4)|list}}"
    - ['a', 'c', 'd', 'c']
```

二、 Invoking lookup plugins with `query`

New in version 2.5.

In Ansible 2.5, a new jinja2 function called `query` was added for invoking lookup plugins. The difference between `lookup` and `query` is largely that `query` will always return a list. The default behavior of `lookup` is to return a string of comma separated values. `lookup` can be explicitly configured to return a list using `wantlist=True`.

This was done primarily to provide an easier and more consistent interface for interacting with the new `loop` keyword, while maintaining backwards compatibility with other uses of `lookup`.

The following examples are equivalent:

```
lookup('dict', dict_variable, wantlist=True)
```

```
query('dict', dict_variable)
```

As demonstrated above the behavior of `wantlist=True` is implicit when using `query`.

Additionally, `q` was introduced as a shortform of `query`:

```
q('dict', dict_variable)
```

三、 Plugin list

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