Ansible Workshop - Exercises

Automation Platform

Learn to manage and run your Ansible content in AAP.

4 - Surveys

Objective

Demonstrate the use of Ansible Automation controller survey feature. Surveys set extra variables for the playbook similar to 'Prompt for Extra Variables' does, but in a user-friendly question and answer way. Surveys also allow for validation of user input.

Guide

You have installed Apache on all hosts in the job you just ran. Now we're going to extend on this:

- Use a proper role that has a Jinja2 template to deploy an `index.html` file.
- Create a job **Template** with a survey to collect the values for the `index.html` template.
- Launch the job **Template**

Additionally, the role will make sure that the Apache configuration is properly set up for this exercise.

> 🔥 **Tip**
>
> The survey feature only provides a simple query for data - it does not support four-eye principles, queries based on dynamic data or nested menus.

The Apache-configuration Role

The playbook and the role with the Jinja2 template already exist in the Github repository https://github.com/ansible/workshop-examples in the directory `rhel/apache`.

Head over to the Github UI and have a look at the content: the playbook `apache_role_install.yml` merely references the role. The role can be found in the `roles/role_apache` subdirectory.

- Inside the role, note the two variables in the `templates/index.html.j2` template file marked by `{{…}}`.
- Notice the tasks in `tasks/main.yml` that deploy the file from the template.

What is this playbook doing? It creates a file (**dest**) on the managed hosts from the template (**src**).

The role deploys a static configuration for Apache. This is to make sure that all changes done in the previous chapters are overwritten and your examples work properly.

Because the playbook and role is located in the same Github repo as the `apache_install.yml` playbook you don't have to configure a new project for this exercise.

Create a Template with a Survey

Now you create a new Template that includes a survey.

Create Template

- Go to **Automation Execution** → **Templates**, click the **Create template** button, and choose **Create job template**.

- Fill out the following information:

Parameter	Value
Name	`Create index.html`
Job Type	`Run`
Inventory	`Workshop Inventory`
Project	`Workshop Project`
Execution Environment	`Default execution environment`
Playbook	`rhel/apache/apache_role_install.yml`
Credentials	`Workshop Credential`
Limit	`web`
Options	☑ Privilege Escalation

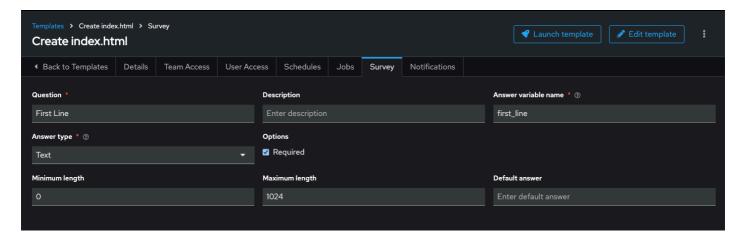- Click **Create job template**.

> ⚠ **Warning**
>
> **Do not run the template yet!**

Add the Survey

- In the Template, click the **Survey** tab, then click **Create survey question**.

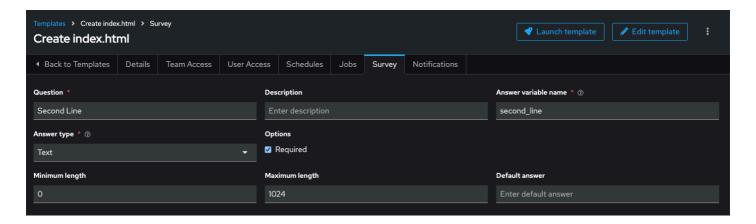- Fill out the following information:

Parameter	Value
Question	`First Line`
Answer Variable Name	`first_line`

Answer Type	Text

- Click **Create survey question**.
- Click **Create survey question** to create a second **Survey Question**

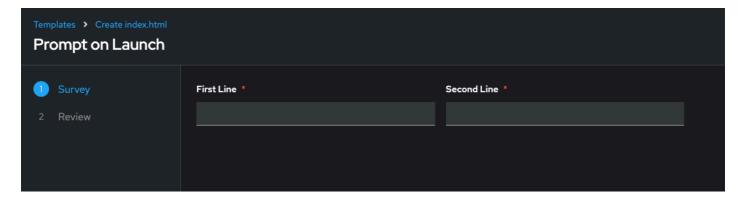| Parameter	Value
Question	Second Line
Answer Variable Name	second_line
Answer Type	Text

- Click **Create survey question**.
- Enable the survey by toggling the **Survey disabled** button to the **On** posiiton.

Launch the Template

Now, launch the **Create index.html** job template by clicking the **Launch template** button.

Before the actual launch the survey will ask for **First Line** and **Second Line**. Fill in some text and click **Next**. The **Preview** window shows the values, if all is good run the Job by clicking **Finish**.

After the job has completed, check the Apache homepage. In the SSH console on the control host, execute `curl` against `node1`:

```
[student1@ansible-1 ~]$ curl http://node1
<body>
<h1>Apache is running fine</h1>
<h1>This is survey field "First Line": line one</h1>
<h1>This is survey field "Second Line": line two</h1>
</body>
```

Note how the two variables where used by the playbook to create the content of the `index.html` file.