Ansible Navigator

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

Session - 26 Agenda:
1. Ansible Navigator

Ansible Navigator:
ansible-navigator is a command-based tool for creating, reviewing, and troubleshooting Ansible content, including inventories, playbooks, and collections.

Ansible Navigator (ansible-navigator) is a seriously powerful text user interface (TUI) tool that offers new and even better ways to interact with your Ansible environment and content, a tool which both owes a debt to the lasting legacies of ansible-playbook and its related programs, while also delivering much-needed new features (and feature consolidations) that have been leaving the traditional Ansible CLI toolbox a bit lacking in recent times: especially as the Red Hat Ansible Automation Platform (AAP) codebase and ecosystem continue to evolve and improve rapidly for AAP customers.

Both Ansible Navigator and ansible-playbook are fundamentally CLI tools that you can use to interact with Ansible, but they actually differ from each other in their approach and features.

Comparison Between ansible-playbook & Ansible Navigator:
1. Main usage comparison:
ansible-playbook tool is used primarily for executing Ansible playbooks.
Ansible Navigator is considered a more comprehensive tool designed to explore, manage, and execute Ansible content, including playbooks, roles, and collections, while also allowing you to use the exact same execution environment (EE) images you do in AAP.

2. UI comparison:
ansible-playbook tool has a traditional CLI interface, where you run playbooks by passing flags and specifying playbook files.
Ansible Navigator features a (default) interactive mode, offering a text user interface (TUI) that allows you to navigate through you Ansible content and discover available actions. It also allows you to run your playbooks in non-interactive (aka stdout) mode.

3. Execution Environment support comparison:
ansible-playbook tool has to run playbooks via some type of local Python environment (e.g., using your system Python or a Python virtualenv).
Ansible Navigator supports execution environments, which are containerized environments that isolate Ansible execution. This ensures a consistent environment for running playbooks and reduces the risk of conflicts between dependencies.

4. Extensibility:
ansible-playbook tool focuses on executing playbooks, and you sometimes end up having to use other ansible-* tools to perform other pretty common related tasks.
Ansible Navigator provides additional features, such as browsing documentation, visualizing inventory, and managing collections, all of which makes it a more versatile tool for working with Ansible content.
Ansible Navigator was really created to address some limitations of the traditional Ansible CLI tools, and it provides you with a more efficient way to work with all things Ansible:

Features of Ansible Navigator:
1. Improves the user experience by providing a more interactive look and feel.
2. Enhances the average discoverability of Ansible content and available actions.
3. Simplifies the usage of execution environments, allowing you to work with containerized Ansible instances. This point is highly relevant for AAP customers, as there is no equivalent functionality available to you with ansible-playbook.
4. To go a bit deeper on discoverability, this means being able to effortlessly explore and find Ansible

Ansible Navigator

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

content, actions, and information, all within the Ansible Navigator TUI.

Some examples in practice:

1. *Exploring playbooks and roles:* You can browse through your playbooks and roles, making it easier to find specific automation tasks or examine the structure of your Ansible content.
2. *Browsing collections:* Ansible Navigator allows you to explore Ansible collections. You can navigate through the collections, view the included content, and see ways to use them in your own automations.
3. *Viewing documentation:* Ansible Navigator enables you to view documentation for modules, roles, and plugins, all within in the TUI.
4. *Visualizing inventory:* You can easily visualize your inventory in Ansible Navigator, which provides an organized view of hosts and groups. You'll find that this makes it easier to discover the structure of the inventory and understand how hosts and groups relate to each other.
5. *Listing available actions:* When using the (default) interactive mode, Ansible Navigator displays a list of available actions that you can take, such as running a playbook, examining a role, or exploring a collection. Inarguably, this feature enhances increases your awareness of "the art of the possible" for the particular Ansible content you are working with, by providing an at-your-fingertips overview of possible actions and guiding you in your interactions with custom code especially.

Steps To Install, Configure & Use Ansible Navigator:

First of all, we have to install ansible-core, python-pip and podman packages in our ansible server in order to use ansible navigator.

$ sudo dnf install -y ansible-core python-pip podman

Now, we can execute pip3 install command to install the ansible navigator.

$ pip3 install ansible-navigator --user

Ansible navigator requires container environment in order to work. We can list the available container images by running podman images command.

$ podman images

Now, execute ansible-navigator command to start the ansible navigator. First time it will download the ansible navigator container image and its execution environment.

$ ansible-navigator

Again, execute the podman images command to see the ansible navigator container image in your machine.

$ podman images

Additionally, you can also execute the ansible navigator images command to see the ansible navigator container image.

$ ansible-navigator images

Your ansible navigator is now ready for the use.

Working With Ansible Navigator:

To see the version of ansible navigator, execute the below command:

$ ansible-navigator --version

To get the help on ansible navigator, execute the below command:

$ ansible-navigator --help

Let's create an ansible playbook to perform the ping test with the managed nodes. This playbook will be used to learn the fundamentals of ansible navigator.

$ vim ping.yml

Ansible Navigator

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
---
- name: Ping Test Playbook
  hosts: all
  tasks:
    - name: Testing ping
      ansible.builtin.ping:
      register: result
    - name: Print ping result
      ansible.builtin.debug:
        msg: "{{ result }}"
...
```

Check the playbook for the syntax errors.
$ ansible-playbook ping.yml --syntax-check

Execute the playbook.
$ ansible-playbook ping.yml

Create an ansible navigator inventory file.
$ vim inventory

```
[Ubuntu]
node2

[RHEL]
node1
node3

[dev]
node1
node2
```

Create an ansible navigator configuration file.
$ vim ansible-navigator.yml

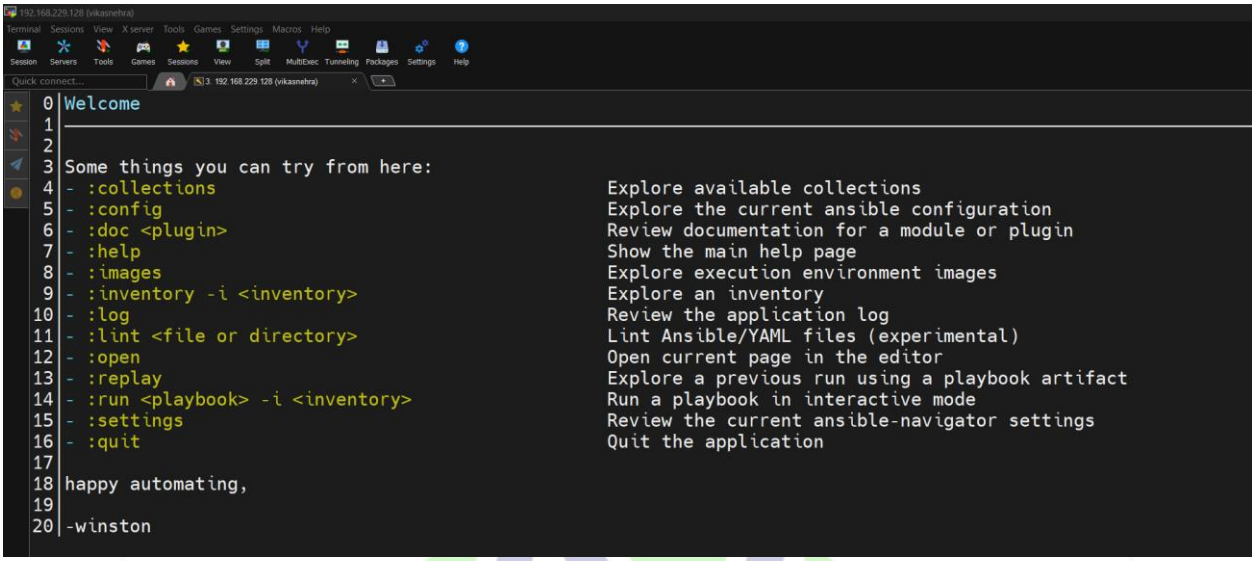
```
---
ansible-navigator:
  ansible:
    inventory:
      entries:
      - inventory
  execution-environment:
    container-engine: podman
    enabled: false
    image: quay.io/ansible/ansible-navigator-demo-ee:latest
  logging:
    level: debug
  playbook-artifact:
    enable: true
    save-as: playbook-artifacts/{playbook_name}-artifact-{time_stamp}.json
# mode: stdout
...
```

Common Ansible-navigator commands:
ansible-navigator: Launches Ansible-navigator in interactive mode, providing a text user interface (TUI) to explore your Ansible content. (and ansible-navigator run ping.yml executes that playbook specifically.)

Ansible Navigator

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

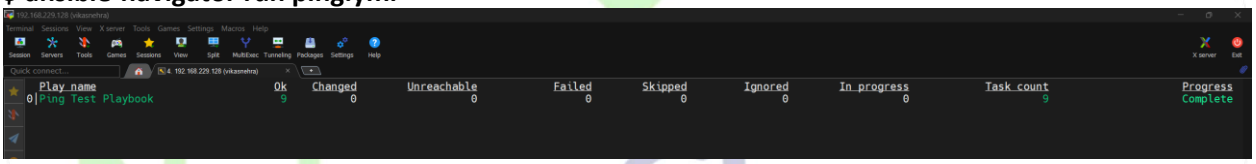
$ ansible-navigator



Here ansible navigator is in TUI mode and it waiting for the inputs from the user. To open any of these items just mention there :item_name and press enter (e.g. to open collections write there :collections). In the list displayed on the next page just press the number against the item in the list to open it and see the details. If you want to go back and to exit the ansible-navigator TUI press the esc key on keyboard.

ansible-navigator run ping.yml: executes that playbook specifically.

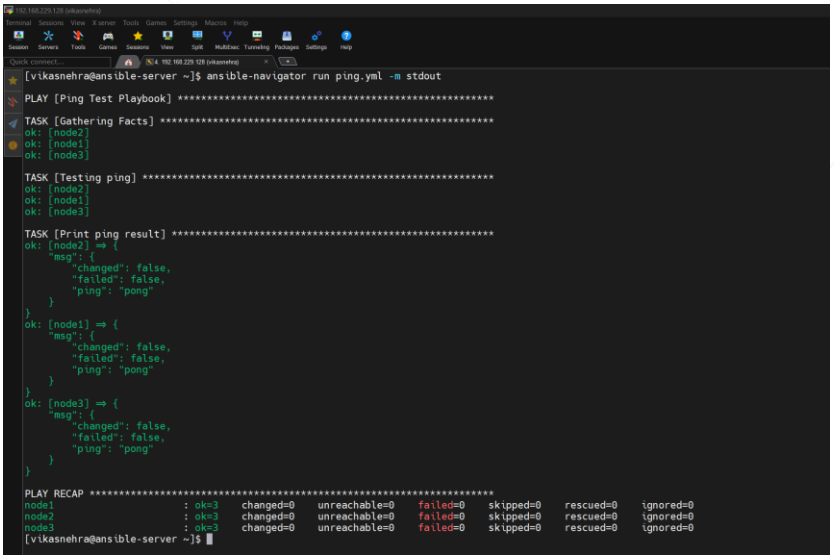
$ ansible-navigator run ping.yml



It will not show you the output by default, however you can press 0 there to see the execution process tasks statuses and the results. On the next page just press the number against the item in the list to open it and see the details. If you want to go back and to exit the ansible-navigator TUI press the esc key on keyboard.

ansible-navigator run ping.yml -m stdout: Executes the specified playbook in non-interactive (stdout) mode, displaying the output directly in the terminal more like ansible-playbook would.

$ ansible-navigator run ping.yml -m stdout



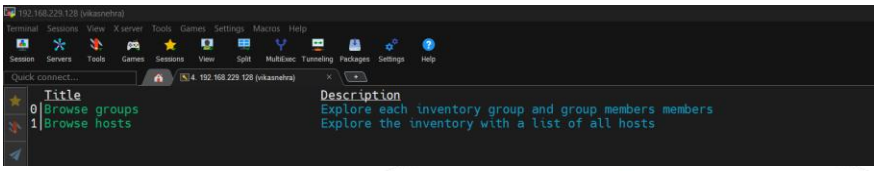
Ansible Navigator

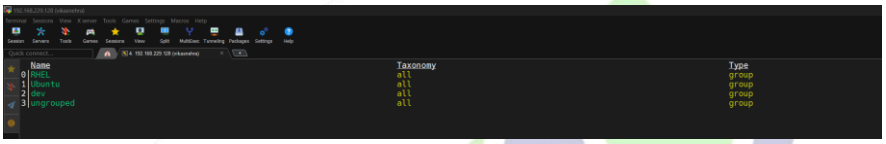
By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

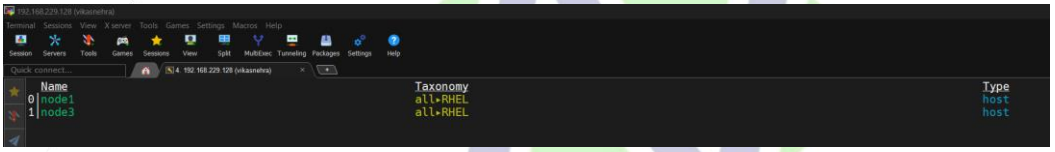
It will show the execution like ansible-playbook command.

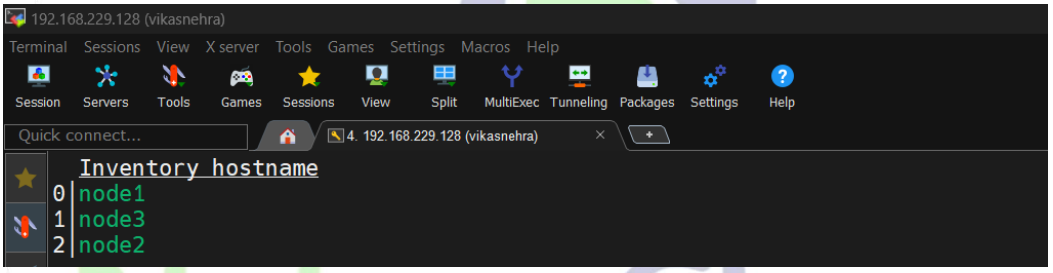
ansible-navigator inventory -i inventory: Visualizes the specified inventory file, showing an organized view of hosts and groups.

$ ansible-navigator inventory -i /etc/ansible/hosts



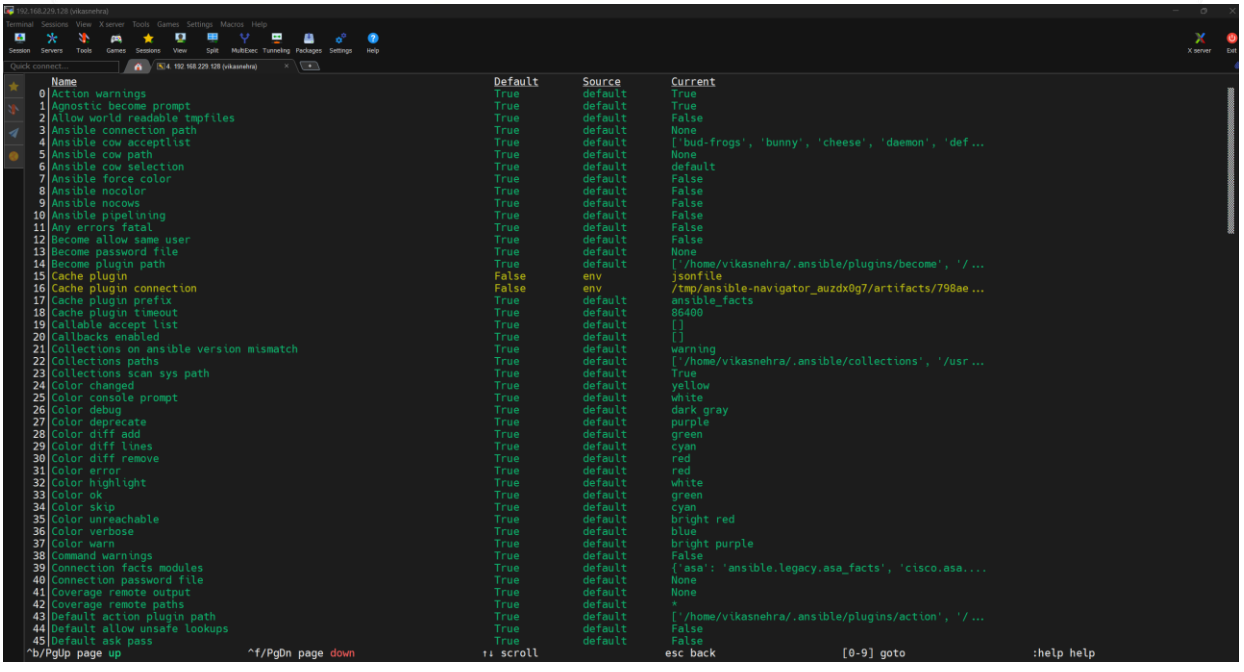






ansible-navigator config: Displays the current Ansible configuration settings, including values from environment variables, configuration files, and defaults.

$ ansible-navigator config

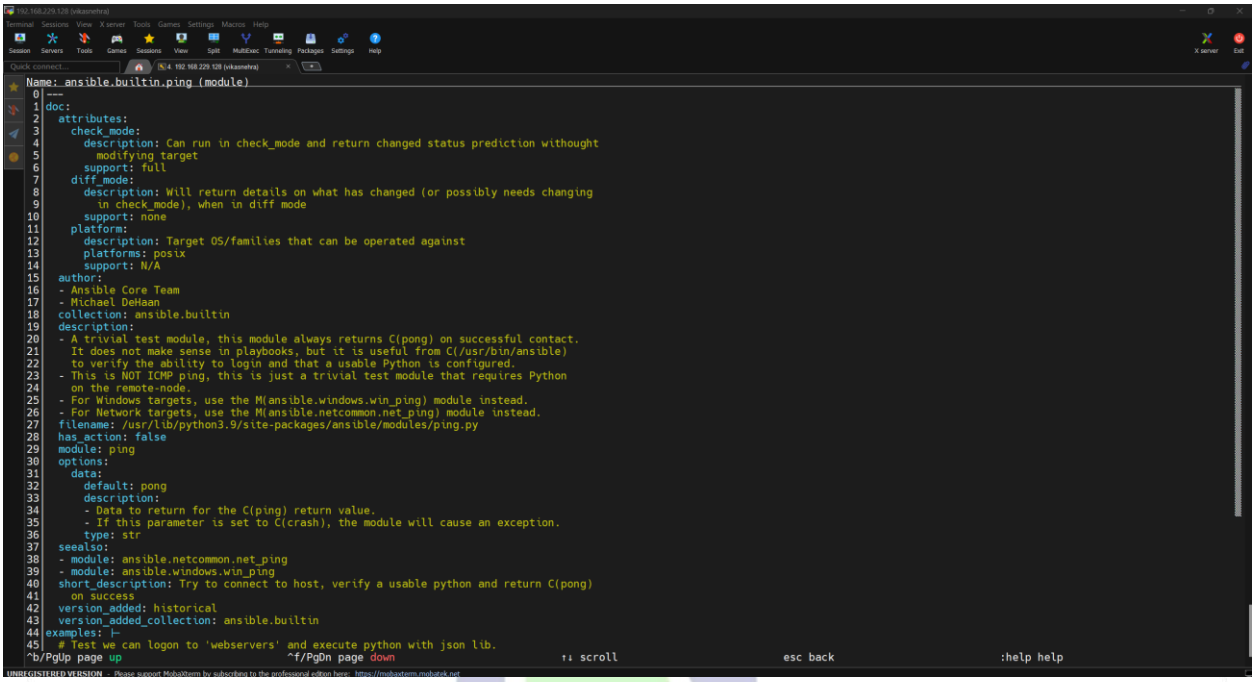


Ansible Navigator

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

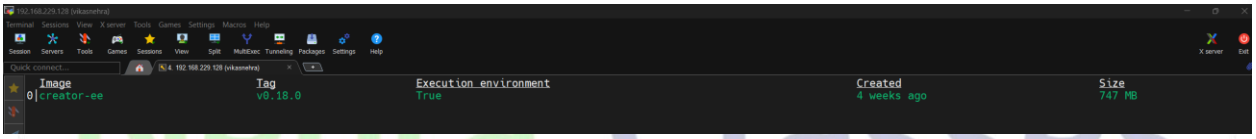
ansible-navigator doc ansible.builtin.ping: Shows the documentation for the specified module, including a description, available parameters, and usage examples.

$ ansible-navigator doc ansible.builtin.ping



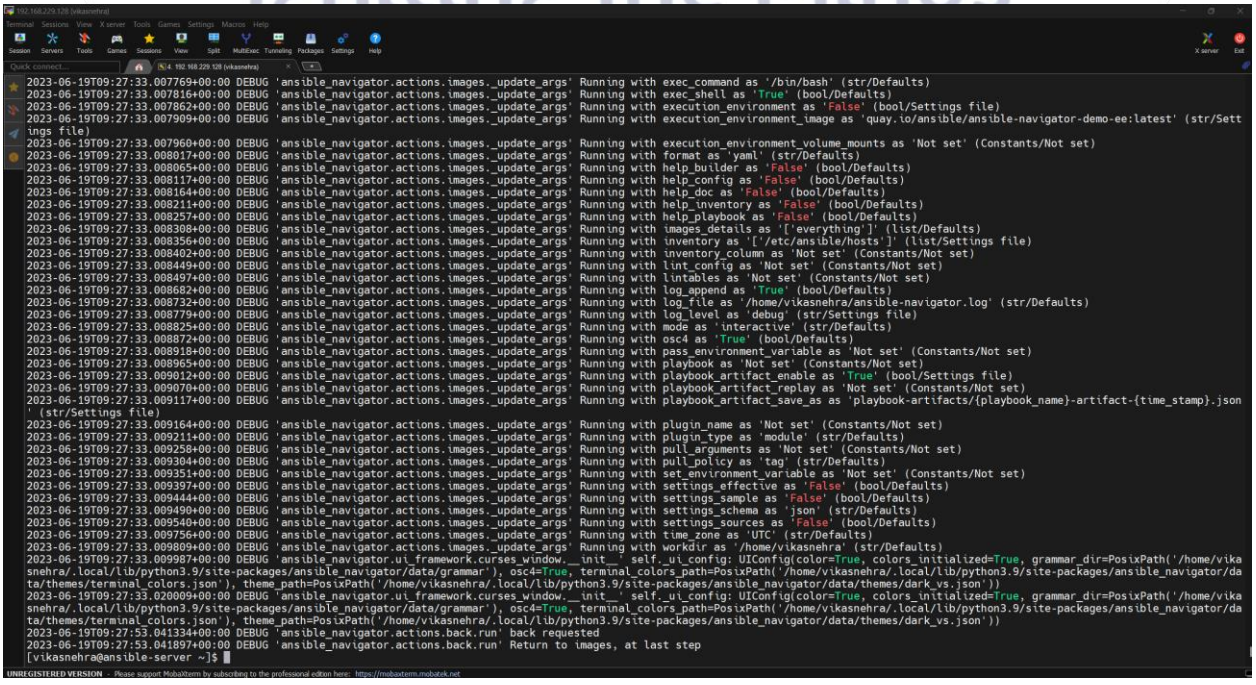
ansible-navigator images: Lists the available container images for execution environments, along with their details and status.

$ ansible-navigator images



cat ~/ansible-navigator.log: Displays the Ansible Navigator log, which contains information about the tool's operation, including any errors or warnings.

$ cat ~/ansible-navigator.log

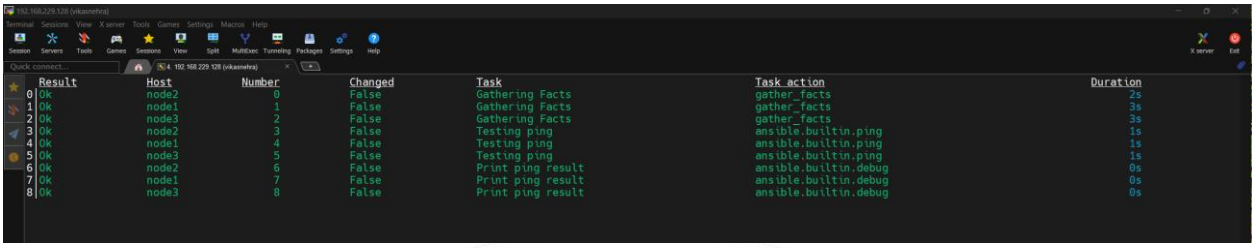


Ansible Navigator

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

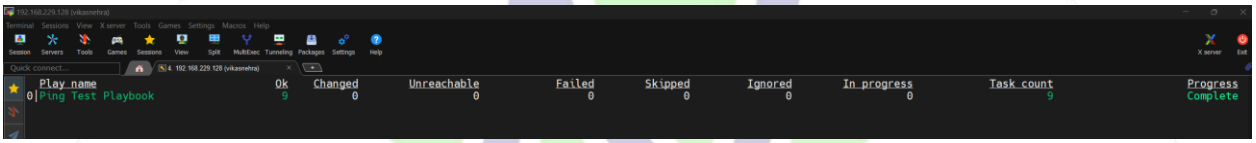
ansible-navigator replay playbook-artifacts/ping-artifact-2023-06-19T09\:19\:59.307330+00\:00.json: "Replays" a previous ansible-navigator "run".

$ ansible-navigator replay playbook-artifacts/ping-artifact-2023-06-19T09\:19\:59.307330+00\:00.json



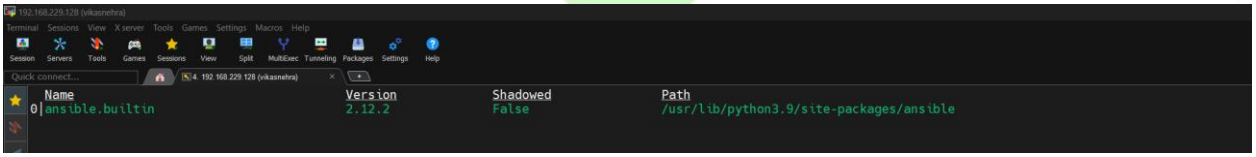
ansible-navigator run ping.yml -i /etc/ansible/hosts --eei ansible-navigator-demo-ee: Specifies both an inventory and execution environment in a "run".

$ ansible-navigator run ping.yml -i /etc/ansible/hosts --eei ansible-navigator-demo-ee

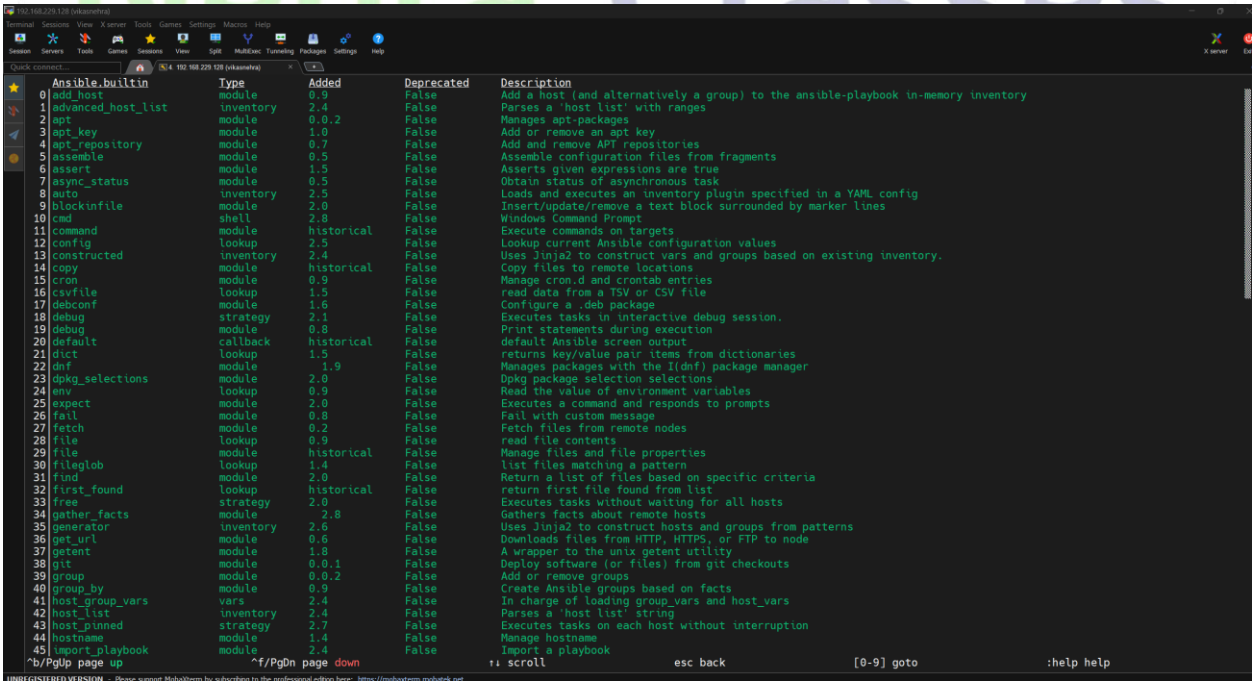


ansible-navigator collections: Lists all installed Ansible collections, including their name, version, and installation path.

$ ansible-navigator collections



Press 0 to see the all builtin collections list.



To open any item from this list just press the same number key on your keyboard.