



Ansible Playbooks Execution over Server Automation Infrastructure

Consolidate and simplify automation management in Server Automation infrastructure by having a unified Ansible deployment, common inventory, and proven, scaled RBAC-permitted execution.

Server Automation at a Glance:

- **Reduce management complexity:**
 Consolidate automation management over trusted Server Automation infrastructure
- **Centralize Playbooks management:**
 A single place to create, edit and execute Playbooks with built-in version control
- **Extend the value of Server Automation :**
 Streamline installation and upgrade of Ansible and execution of Playbooks with no additional OS accounts, network changes or credential management required

Ansible Playbooks offer accessible, reusable content. They help automate tasks such as configuration management, application deployment, and provisioning, and offer IaC (infrastructure as code), version control and more. But are your teams getting the full benefit they expected and need? How do your infrastructure admins cope with the incremental increase of Ansible deployments, and how do they manage disparate servers' inventories and solve network connectivity? How do your IT Ops execute their and DevOps' developed Playbooks at scale or in isolated environments? And is your SecOps burdened creating additional OS accounts, managing and enforcing credentials rotation policies?

With Server Automation, you can now take advantage of the missing capabilities as it treats Ansible Playbooks like its own native content delivering more value at no extra cost.

Server Automation infrastructure and policy-based management extend application delivery capabilities, configuration management and task automation to Ansible Playbooks. This allows Ansible users to consolidate and simplify automation management in Server Automation infrastructure by having a unified Ansible deployment, a common inventory, and proven, scaled RBAC-permitted execution.

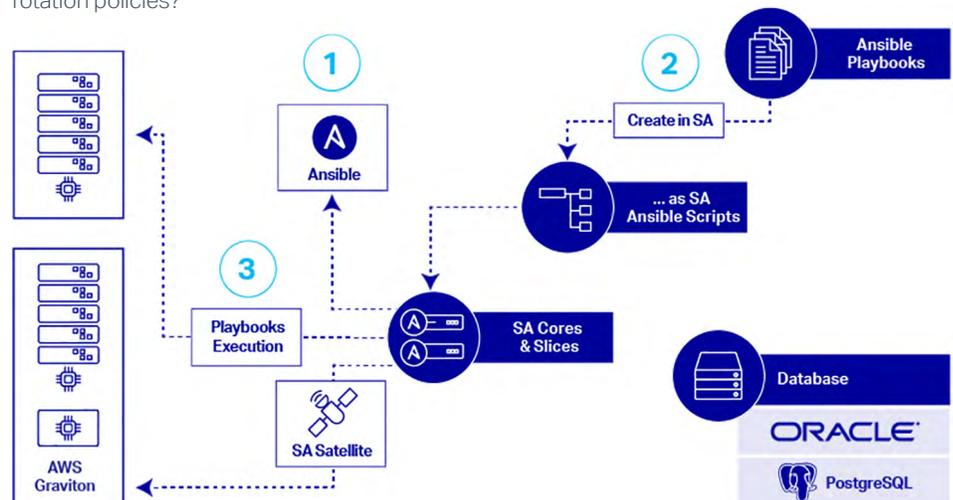


Figure 1. Three phases for executing Ansible Playbooks over Server Automation infrastructure: 1. Install Ansible, 2. Create Playbook, 3. Execute Playbook

Standardize, consolidate, and automate your server operations at scale and across hybrid data centers with a complete and heterogeneous lifecycle management solution. OpenText Server Automation centralizes OS provisioning, security patch management, audit and compliance, configuration management, script execution and access across Windows®, Linux, and UNIX® platforms.

Connect with Us



INSTALL: Streamlined Installation and Upgrade of Ansible

Server Automation provides mesh-wide online/offline Ansible installation or upgrade inside the OGFS (OpenText Global File System) of each slice throughout the mesh. It automates this process using the new Ansible Manager APX in an online mode where a direct internet connectivity exists or through a proxy, as well as an offline mode using pre-staged binaries. You can also install through the same APX collections or roles required during the Playbooks execution.

CREATE: Centralize and Simplify Playbooks Management

Within the Server Automation Library, the Ansible category provides a single place to create, edit and execute Playbooks. From the same UI you can rename, export or delete the Playbook—any changes to the Playbook will be saved as a new version. Built-in version control allows you to easily identify errors, swap between versions, and flag the current—out of all available—version to be used for execution.

EXECUTE: No Additional OS Accounts, Network Changes or Credential Management Required

In order to execute the Playbooks, Server Automation provides an Ansible Runner APX that requires a space separated list of device ID's along with the Playbook ID. The user that performs the Playbooks execution on the targets can be overwritten as well. A debug mode can also be enabled. The entire communication is facilitated by Server Automation through the agents deployed on each target, implying no network changes, additional OS accounts created, nor credential management required.

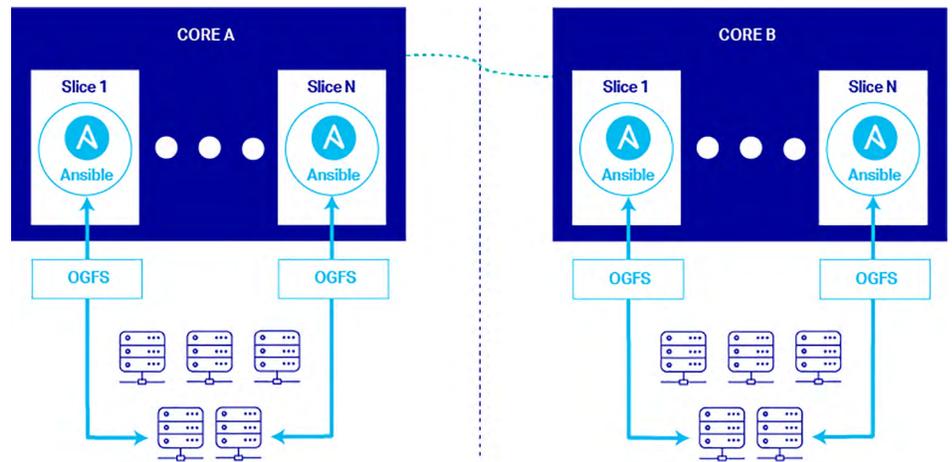
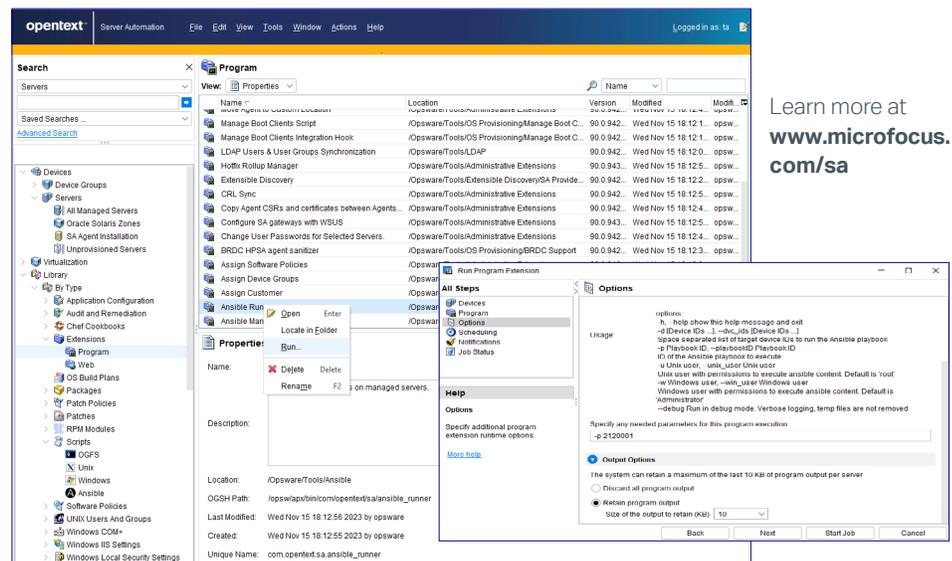


Figure 2. Ansible installed inside the OGFS (OpenText Global File System)



Learn more at www.microfocus.com/sa

Figure 3. Execution through Ansible Runner APX