



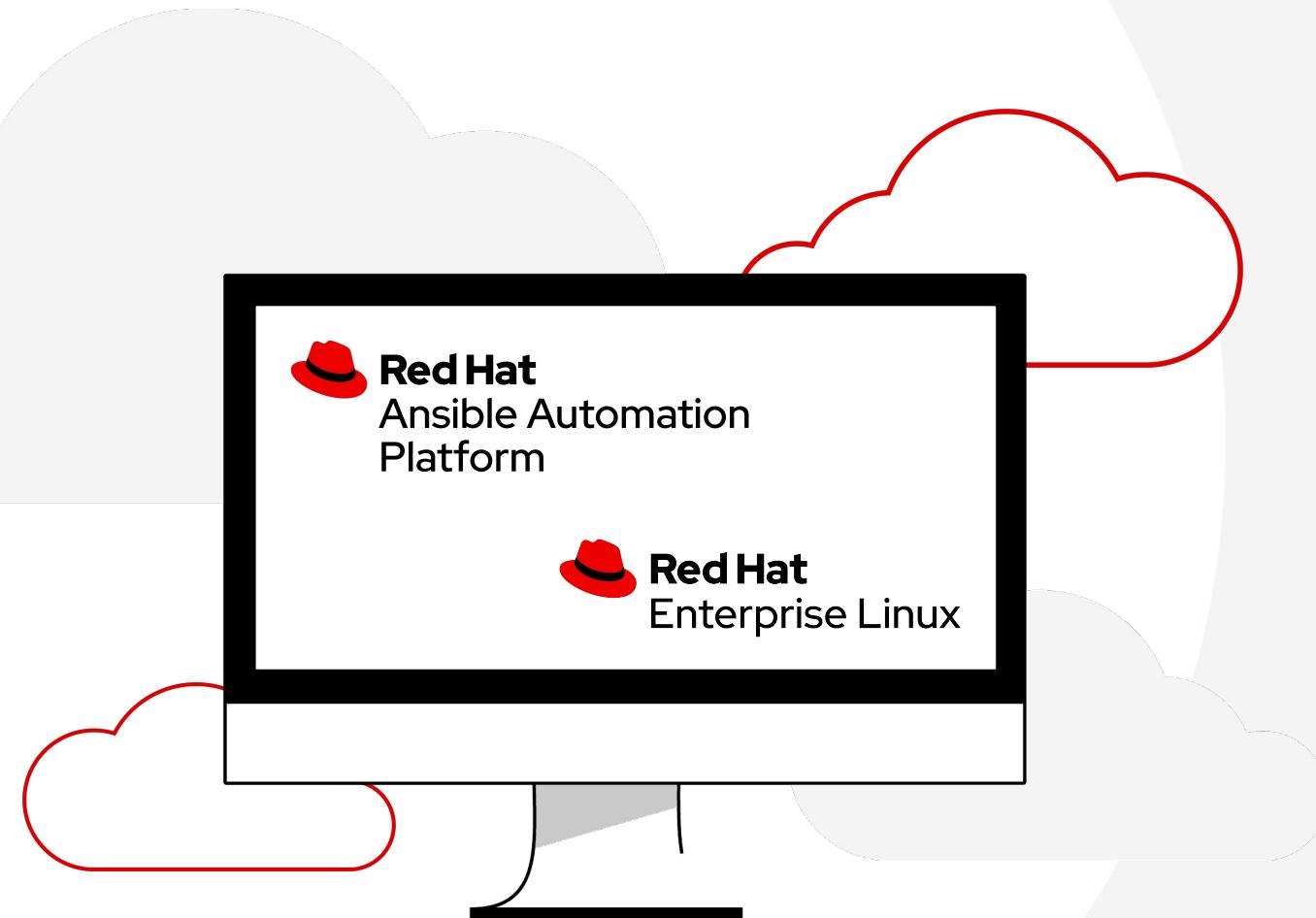
RHEL In-place Upgrade Automation Workshop

A comprehensive approach for automating in-place upgrades of Red Hat Enterprise Linux



What you will learn

- ▶ Key features of the upgrade automation approach
- ▶ Workshop setup and walkthrough
- ▶ Generating pre-upgrade reports
- ▶ Automating recommended remediations
- ▶ Understand potential application impacts
- ▶ How upgrade from RHEL 7 and RHEL 8
- ▶ The importance of snapshot/rollback
- ▶ Ansible roles available to help you get started



Introduction

Topics Covered:

- Key features of the upgrade automation approach
- Overview of the workshop lab environment



Red Hat
Ansible Automation
Platform



Red Hat
Enterprise Linux



**Automation happens when
one person meets a problem
they never want to solve again**

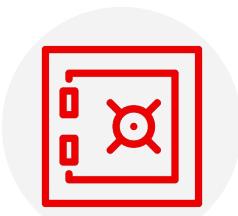
RHEL In-place Upgrade Automation

Key Features to Succeed at Scale



Automate Everything

Make RHEL upgrades a push-button service easily consumable by Ops and App teams



Snapshot/rollback

Eliminate application impact risk so there is never an excuse not to upgrade



Custom Modules

Automate remediations to deal with your standard tools, agents and middleware

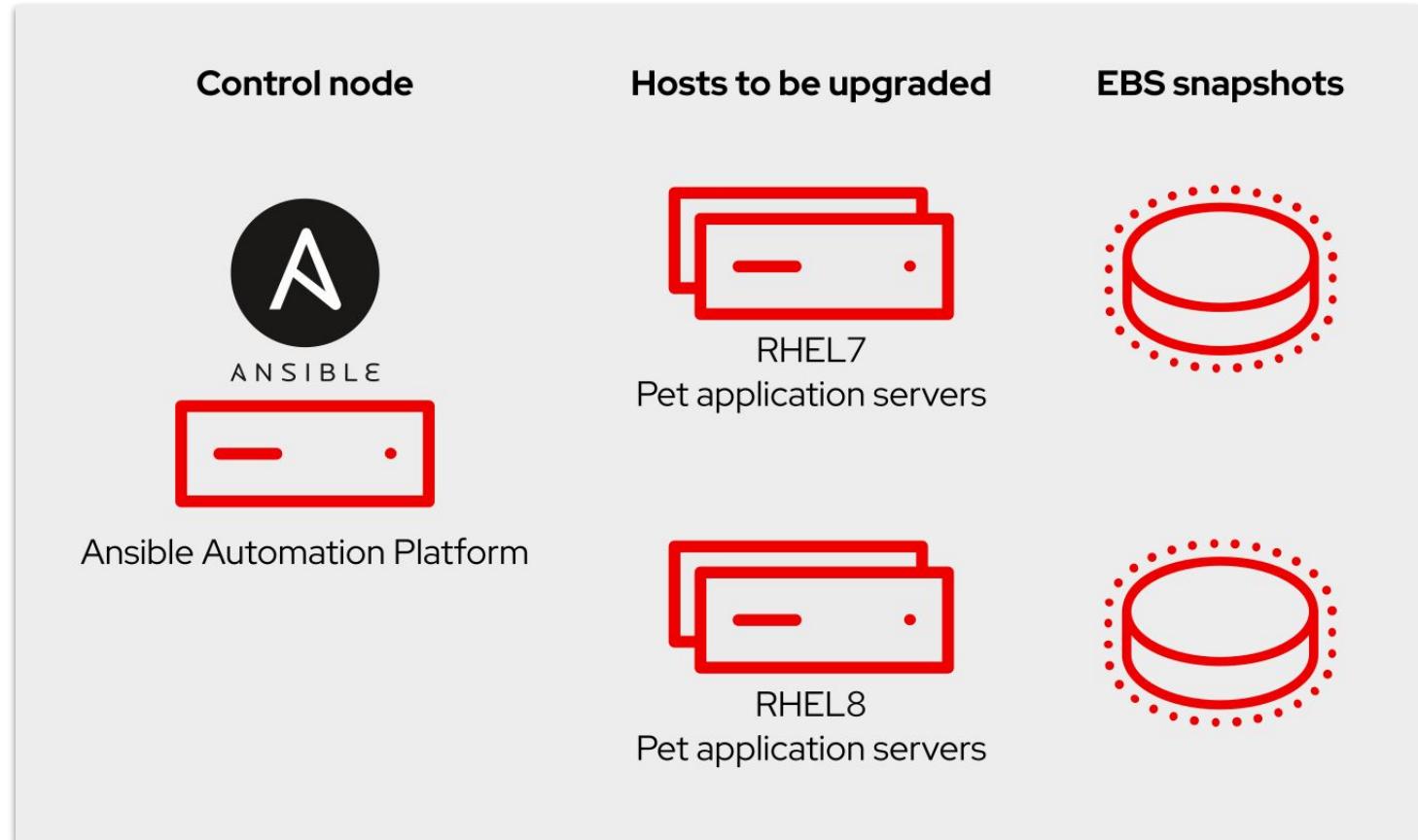


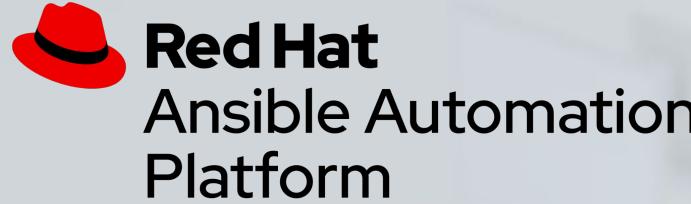
Reporting Dashboard

Visualize pre-upgrade results and track progress of upgrades completed across the estate

Workshop Lab Environment

- ▶ The workshop lab environment includes a number of RHEL cloud instances
- ▶ One instance is dedicated to hosting AAP and launches playbook and workflow jobs
- ▶ The jobs are executed perform in-place upgrades on the remaining RHEL hosts
- ▶ The automation uses Amazon EBS for the snapshot/rollback capability





Lab Time - Workshop Lab Environment

Complete exercise 1.1 in your lab environment now



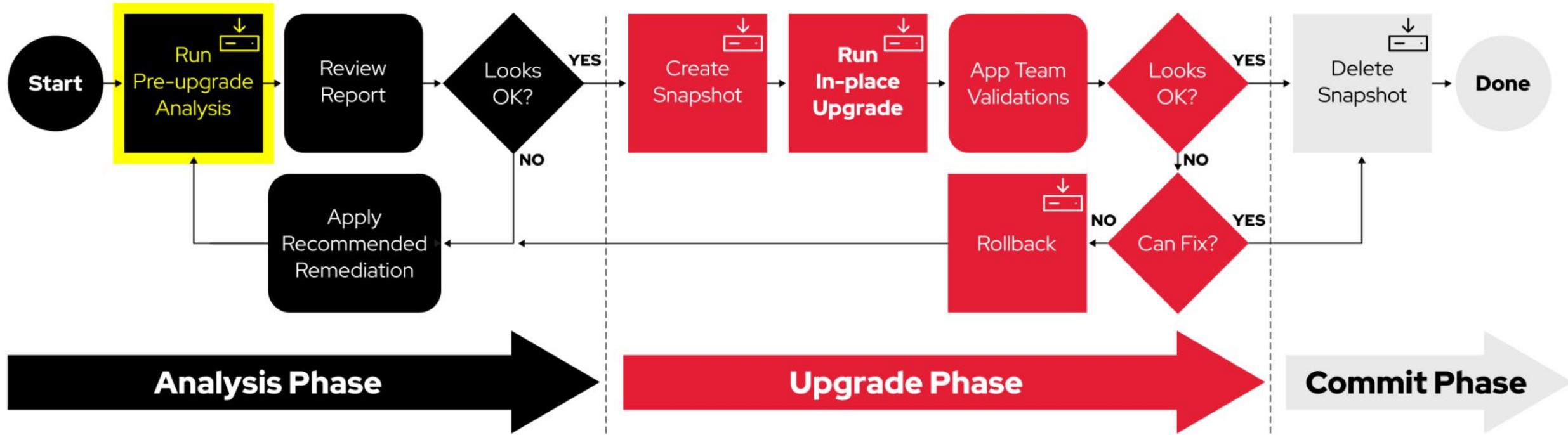
Section 1

Pre-upgrade Analysis

 **Red Hat**
Ansible Automation
Platform

 **Red Hat**
Enterprise Linux

RHEL In-place Upgrade Automation Workflow



ⓘ Note

The icon indicates workflow steps that are automated by Ansible playbooks.

Launching the Analysis Job Template

- ▶ Job templates are used to launch automation jobs from AAP
- ▶ We'll use the Analysis job template to generate pre-upgrade reports for all of the RHEL pet app servers in our lab

The screenshot shows the Red Hat Ansible Automation Platform (AAP) web interface. The left sidebar has a dark theme with white text and includes links for Views (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals), Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration (Credential Types). The main content area is titled "Templates" and lists 14 items. The first item, "AUTO / 01 Analysis", is highlighted with a cursor icon pointing at it. The table columns are Name, Type, Organization, Last Ran, and Actions. The "Actions" column contains icons for edit, delete, and more. The "Last Ran" column shows the date and time of the last execution: "4/25/2023, 6:17:33 AM".

Name	Type	Organization	Last Ran	Actions
AUTO / 01 Analysis	Job Template	Default	4/25/2023, 6:17:33 AM	
AUTO / 02 Upgrade	Workflow Job Template	Default		
AUTO / 03 Rollback	Job Template	Default		
AUTO / 04 Commit	Job Template	Default		
FIX / Patch OS to latest	Job Template	Default		
FIX / Remediate	Job	Default		

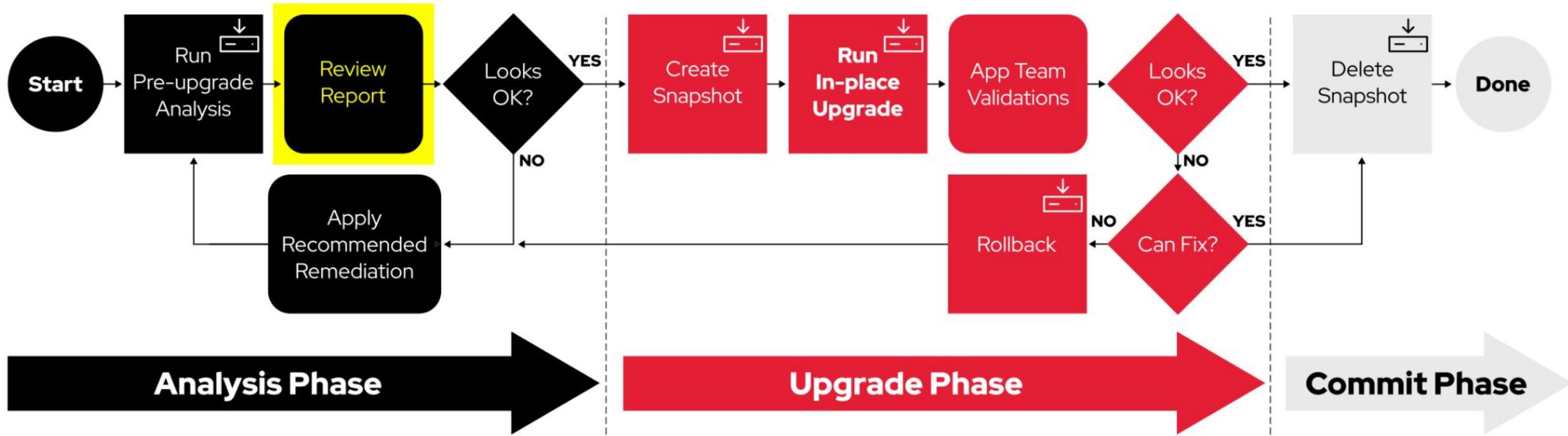


Lab Time - Run Pre-upgrade Jobs

Complete exercise 1.2 in your lab environment now



RHEL In-place Upgrade Automation Workflow



ⓘ Note

The icon indicates workflow steps that are automated by Ansible playbooks.

Example Pre-upgrade Report

- ▶ When inhibitor risk findings are reported, the RHEL upgrade is blocked and can't proceed without first resolving the issue.
- ▶ Other high risk findings are no big deal as we'll learn at the end of the next exercise.

The screenshot shows a web-based interface for the Ansible Automation Platform. On the left, a sidebar menu includes options like System, Logs, Networking, Accounts, Services, Tools, Diagnostic Reports, Kernel Dump, SELinux, Subscriptions, Terminal, and Upgrade Report. The Upgrade Report option is currently selected. The main content area displays a table titled "Upgrade Report for: strong-hyena.example.com". The table has columns for Title, Risk Factor, Description, Tags, and Time. There are 11 rows of data, each representing a different finding. Some findings are marked as "Inhibitor" (indicated by a red circle icon). The findings include:

Title	Risk Factor	Description	Tags	Time
Packages available in excluded repositories will not be installed	High		repository	25.04.2023 15:36:45
Difference in Python versions and support in RHEL 8	High	Remediation hint Links	python	25.04.2023 15:36:45
Possible problems with remote login using root account	High	Inhibitor Remediation hint	authentication security network services	25.04.2023 15:36:48
GRUB core will be updated during upgrade	High		boot	25.04.2023 15:36:51
Missing required answers in the answer file	High	Inhibitor Remediation hint Remediation command		25.04.2023 15:36:52
Usage of deprecated Model "RequiredTargetUserspacePackages" at /usr/share/leapp-repository/repositories/system_upgrade/common/actors/cloud/checkrhui/actor.py:113	High			25.04.2023 15:36:30
SELinux will be set to permissive mode	Low	Remediation hint	selinux security	25.04.2023 15:36:46
Postfix has incompatible changes in the next major version	Low		services email	25.04.2023 15:36:48
chrony using non-default configuration	Low		services time management	25.04.2023 15:36:50
Grep has incompatible changes in the next major version	Low	Remediation hint	tools	25.04.2023 15:36:51

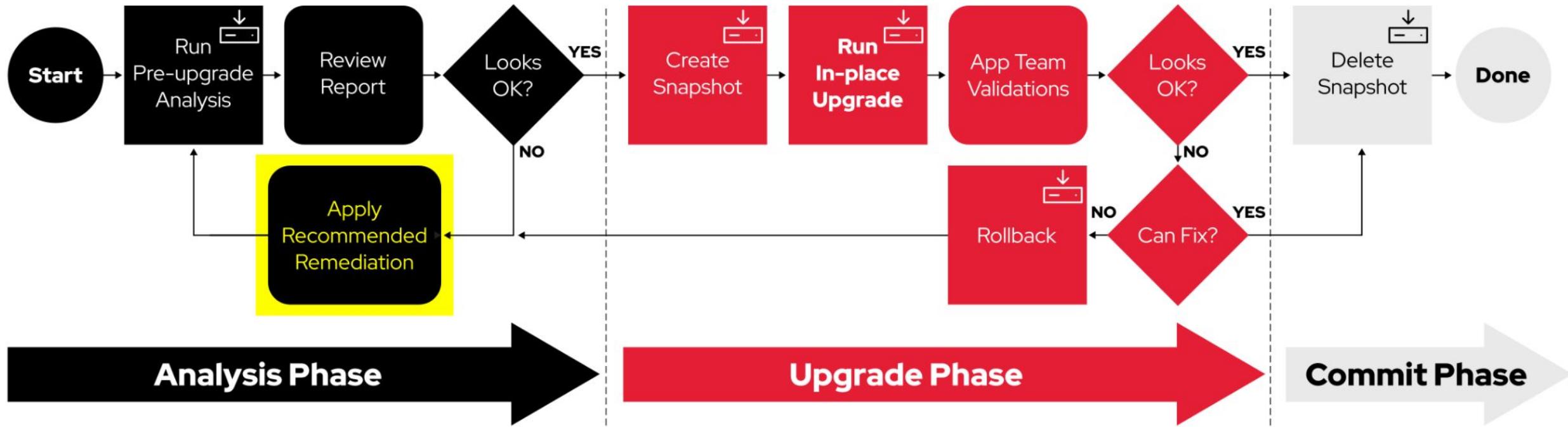


Lab Time - Review Pre-upgrade Reports

Complete exercise 1.3 in your lab environment now



RHEL In-place Upgrade Automation Workflow



ⓘ Note

The icon indicates workflow steps that are automated by Ansible playbooks.

Automating Remediation of Inhibitor Findings

- ▶ Using an Ansible playbook is the best way to automate remediations for inhibitors identified in the pre-upgrade reports.
- ▶ The `Configure sshd` task addresses the "Possible problems with remote login using root account" inhibitor.
- ▶ The `Remove pam_pkcs11 module` task will update the Leapp answer file to solve our other inhibitor.
- ▶ After running the remediation playbook job, we'll generate a fresh pre-upgrade report and see no more inhibitors.

```
- name: Configure sshd
  ansible.builtin.lineinfile:
    path: "/etc/ssh/sshd_config"
    regex: "^(#)?{{ item.key }}"
    line: "{{ item.key }} {{ item.value }}"
    state: present
  loop:
    - {key: "PermitRootLogin", value: "prohibit-password"}
    - {key: "PasswordAuthentication", value: "no"}
  notify:
    - Restart sshd

- name: Remove pam_pkcs11 module
  ansible.builtin.shell: |
    set -o pipefail
    leapp answer --section
    remove_pam_pkcs11_module_check.confirm=True
  args:
    executable: /bin/bash
```



Lab Time - Perform Recommended Remediations

Complete exercise 1.4 in your lab environment now



Example Leapp Custom Actor

- ▶ A custom actor can implement pre-upgrade checks specific to your enterprise.
- ▶ Here we see an inhibitor raised for failed compliance with an imaginary organization's "reboot hygiene" policy.
- ▶ Continue to the next exercise to learn more about custom actors and other ways to tailor automation to your needs.

The screenshot shows a browser window titled "Upgrade Report - ec2-user" with the URL <https://student2.s4v0dev04.sandbox154.opentlc.com:9090/@well-salmon/leapp#/report/f761a08f-cead-4d5b-a796-278faafdb0a>. The page displays an "Upgrade Report for: well-salmon.example.com". On the left, a sidebar menu includes "ec2-user@well-salmon", "Search", and various system management options like System, Logs, Networking, Accounts, Services, Tools, Diagnostic Reports, Kernel Dump, SELinux, Subscriptions, Terminal, and Upgrade Report (which is selected). The main content area shows a table of upgrade findings:

Check	Risk Factor
Packages available in excluded repositories will not be installed	High
Modified files under /boot detected	High
Difference in Python versions and support in RHEL 8	High
Packages not signed by Red Hat found on the system	High
GRUB core will be updated during upgrade	High
Usage of deprecated Model "RequiredTargetUserspacePackages" at /usr/share/leapp-repository/repositories/system_upgrade/common/actors/cloud/checkrhui/actor.py:113	High
Module pam_pkcs11 will be removed from PAM configuration	Medium
chrony using non-default configuration	Low
SELinux will be set to permissive mode	Low
Postfix has incompatible changes in the next major version	Low

The "Modified files under /boot detected" row is highlighted with a blue background and a hand cursor icon. The "Risk factor" column uses a color-coded scale from red (High) to blue (Low). To the right of the table, a "Detail" panel provides more information about the "Modified files under /boot detected" issue, including a summary, remediations (which advise a reboot), and related resources (linking to "/boot/policy-violation").



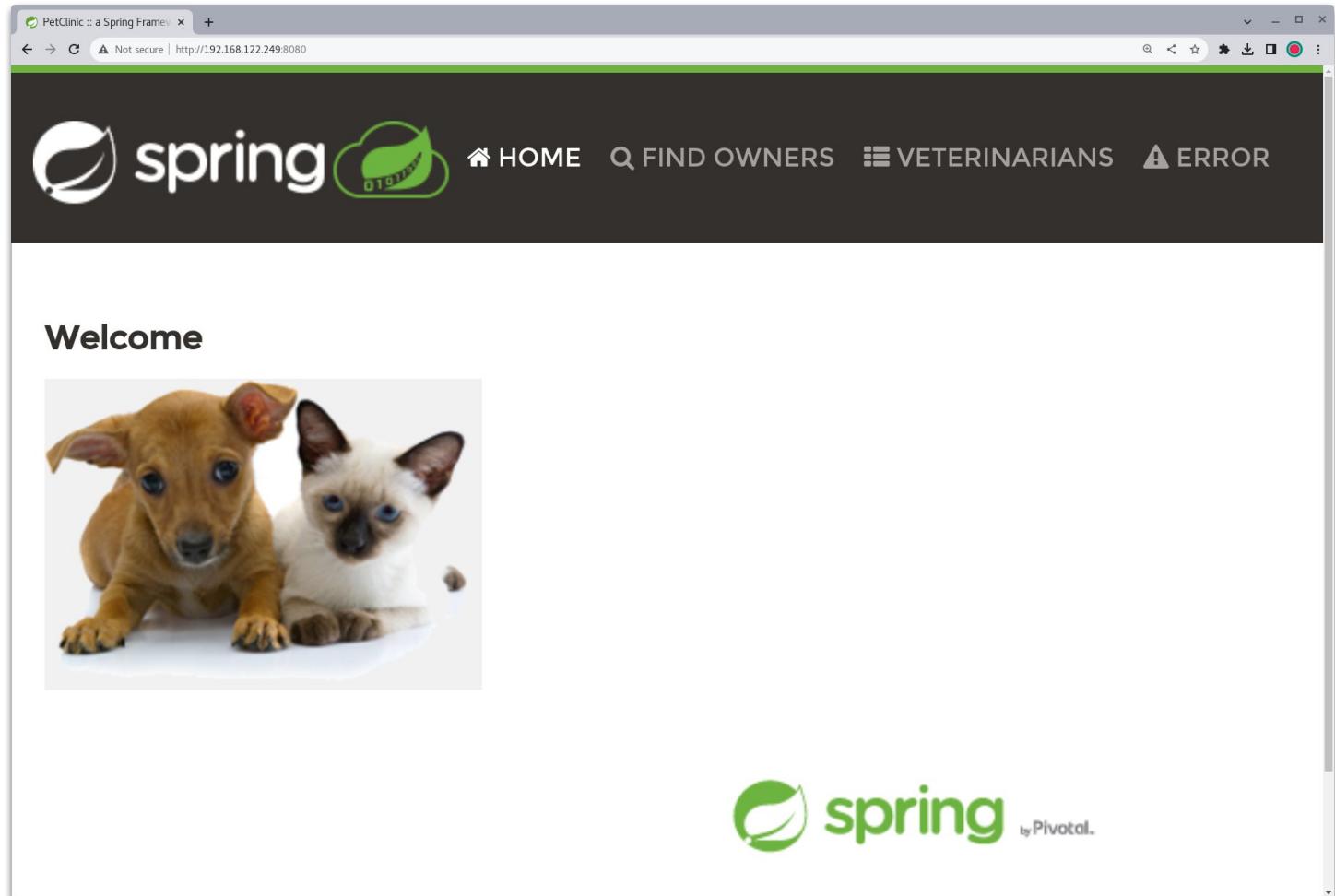
Lab Time - Custom Modules

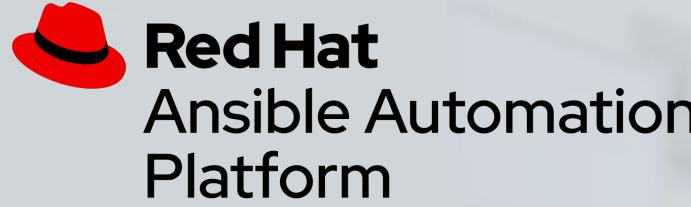
Complete exercise 1.5 in your lab environment now



Deploy a Pet Application

- ▶ To demonstrate how in-place upgrades and rollbacks might impact business applications, we'll install a sample app.
- ▶ The Spring Pet Clinic Sample Application will be installed to use a third-party JDK runtime and a local MariaDB database.
- ▶ Will the application lead to more findings on the pre-upgrade report? Will it still function correctly after we perform the RHEL upgrade? Continue to the next exercise to find out!





Lab Time - Deploy a Pet App

Complete exercise 1.6 in your lab environment now



Section 2

RHEL In-place Upgrade

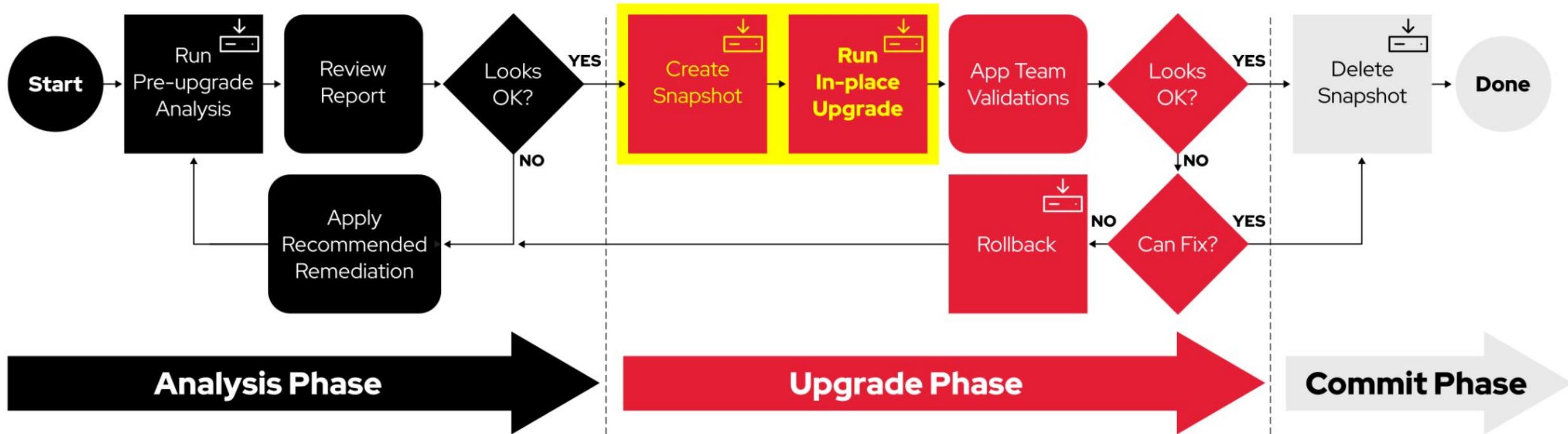


Red Hat
Ansible Automation
Platform



Red Hat
Enterprise Linux

RHEL In-place Upgrade Automation Workflow



i Note

The icon indicates workflow steps that are automated by Ansible playbooks.

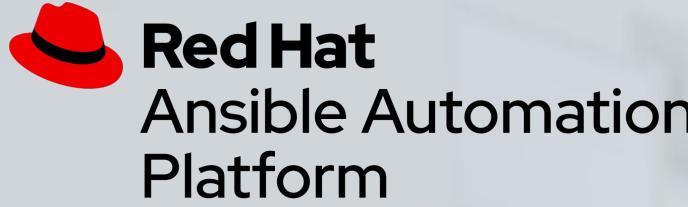
Launching the Upgrade Workflow Job Template

- ▶ A workflow job template in AAP is used to run the snapshot and upgrade playbooks back-to-back in a single job.
- ▶ This job will take about 20 minutes to finish the upgrades on all the RHEL hosts in our lab. While we are waiting, we'll read up to learn more about how Leapp works.

The screenshot shows the Red Hat Ansible Automation Platform interface. The left sidebar has sections for Views (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals), Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration (Credential Types). The main content area is titled 'Details' for the template 'AUTO / 02 Upgrade'. It shows basic information: Name (AUTO / 02 Upgrade), Organization (Default), Job Type (Workflow Job Template), Inventory (Workshop Inventory), Created (5/8/2023, 1:45:49 PM by admin), and Modified (5/8/2023, 1:45:49 PM by admin). Below this is a 'Variables' section with YAML and JSON tabs, displaying the following variable definition:

```
1 -> {
2   "leapp_upgrade_type": "connected",
3   "leapp_repos_enabled": []
4 }
```

At the bottom are 'Edit', 'Launch', and 'Delete' buttons. A hand cursor is hovering over the 'Launch' button.



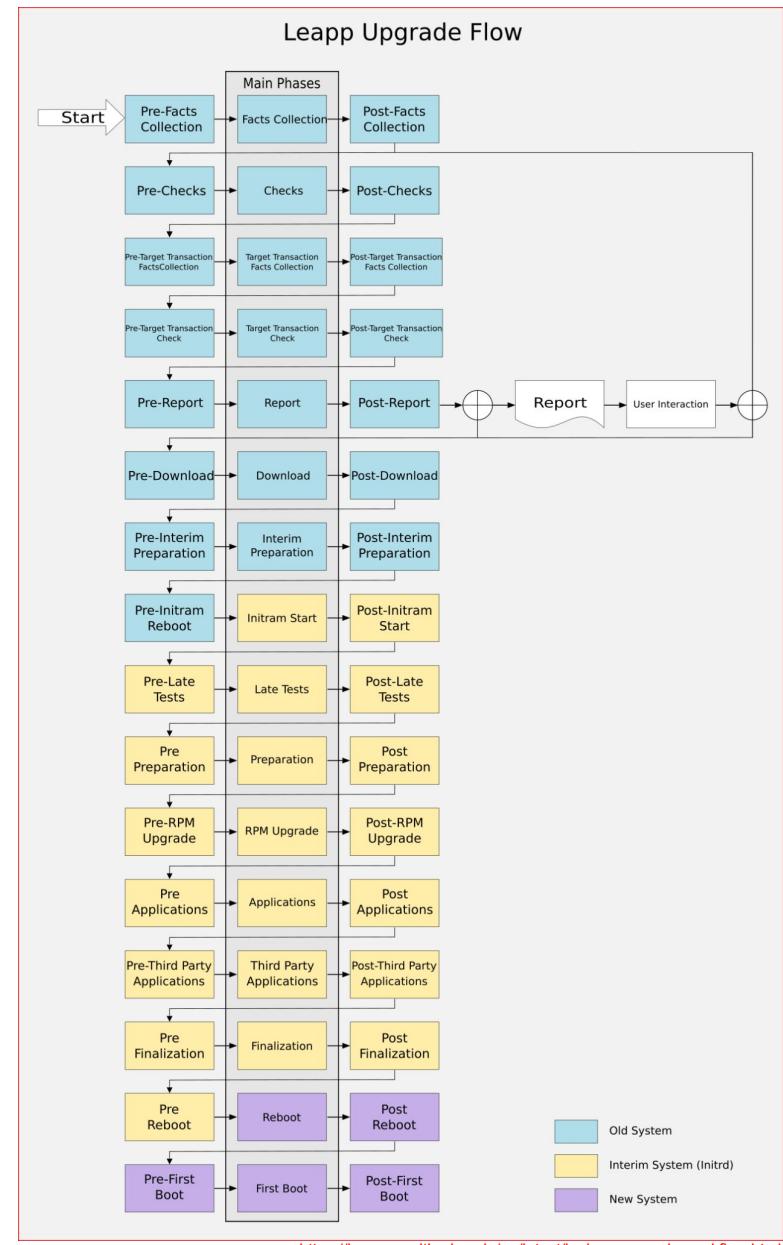
Lab Time - Run OS Upgrade Jobs

Complete exercise 2.1 in your lab environment now



Leapp Framework Overview

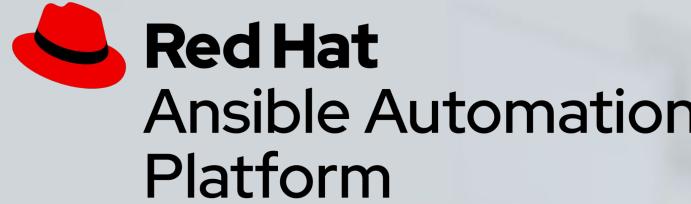
- ▶ Keep in mind that the Leapp framework is responsible only for upgrading the RHEL OS packages. Additional tasks required for upgrading your standard agents, tools, middleware, etc., need to be included in the upgrade playbooks you develop to deal with the specific requirements of your organization's environment.
- ▶ The Leapp framework performs the RHEL in-place upgrade by following a sequence of phases as shown in this flowchart found in the Leapp developer docs. The phases of the RHEL in-place upgrade are implemented in modules known as Leapp actors.
- ▶ Phases under the Old System group run under the existing RHEL installed version. The Interim System phases starts when the host reboots to an upgrade environment under which the network and other services are not started. It is at this time that all RHEL packages are upgraded. Once all the packages are upgraded, another reboot brings the host up under the new RHEL major version and the FirstBoot phase starts. This final phase runs a few post-upgrade actors that require network access and then the upgrade is done.



<https://leapp.readthedocs.io/en/latest/inplace-upgrade-workflow.html>

Different Snapshot Options Compared

Snapshot type	Works with	Benefits	Drawbacks
LVM	<ul style="list-style-type: none">▶ Bare metal▶ On-prem VMs▶ Cloud*	<ul style="list-style-type: none">▶ No external API access required▶ Scope can be just OS or everything	<ul style="list-style-type: none">▶ Free space required in volume group▶ Snapshots can run out of space if not sized correctly▶ Automation must backup and restore /boot separately
VMware	<ul style="list-style-type: none">▶ On-prem VMs (ESX)	<ul style="list-style-type: none">▶ Simple and reliable▶ Scope includes everything	<ul style="list-style-type: none">▶ Doesn't support bare metal, etc.▶ Using VMware snapshot for over 3 days is discouraged▶ Getting API access can be difficult▶ No free space in datastores because of overcommitment▶ Everything scope might be too much
Amazon EBS	<ul style="list-style-type: none">▶ Amazon EC2	<ul style="list-style-type: none">▶ Simple and reliable▶ Unlimited storage capacity▶ Scope can be just OS or everything	<ul style="list-style-type: none">▶ Only works on AWS
Break Mirror	<ul style="list-style-type: none">▶ Bare metal	<ul style="list-style-type: none">▶ Alternative to LVM for servers with hardware RAID	<ul style="list-style-type: none">▶ Significant development and testing effort required▶ RAID and Redfish API standards vary across different vendors and hardware models
ReaR	<ul style="list-style-type: none">▶ Bare metal▶ On-prem VMs	<ul style="list-style-type: none">▶ Method of last resort if no snapshot options will work	<ul style="list-style-type: none">▶ Not really a snapshot, but does offer boot ISO full recovery capability



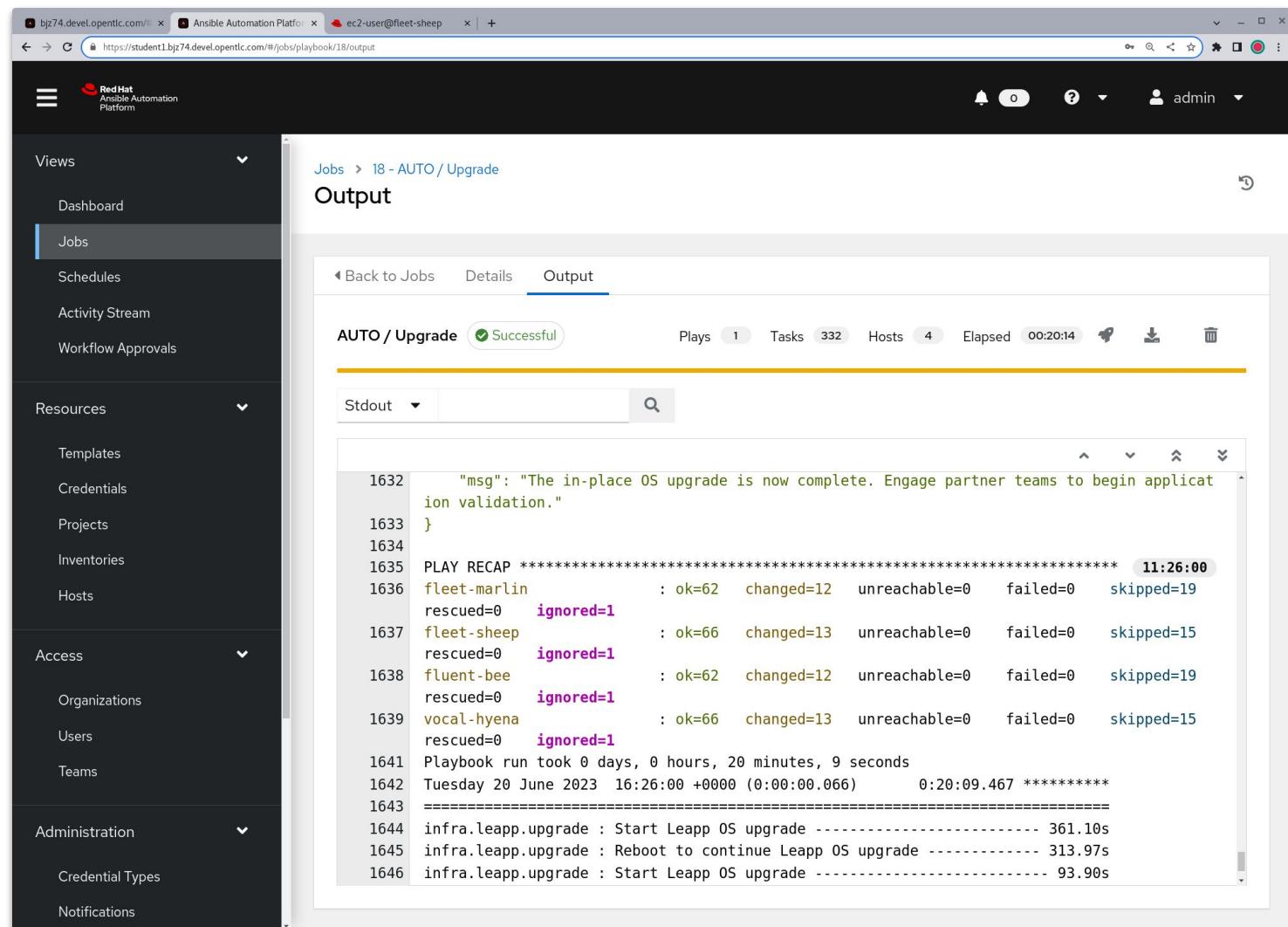
Lab Time - Let's Talk About Snapshots

Complete exercise 2.2 in your lab environment now



Checking Upgrade Job Log Output

- ▶ After the upgrade playbook job finishes, review the play recap in the log output.
- ▶ All hosts should show failed=0 indicating that the upgrades are completed.
- ▶ The application teams can now start their validation and acceptance testing.



The screenshot shows the Red Hat Ansible Automation Platform web interface. The left sidebar has sections for Views (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals), Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration (Credential Types, Notifications). The main content area shows a job details page for '18 - AUTO / Upgrade'. The 'Output' tab is selected, showing the log output. The log output indicates a successful upgrade with the following key lines:

```
1632      "msg": "The in-place OS upgrade is now complete. Engage partner teams to begin application validation."
1633  }
1634
1635 PLAY RECAP ****
1636 fleet-marlin : ok=62  changed=12  unreachable=0  failed=0  skipped=19
1637 fleet-sheep : ok=66  changed=13  unreachable=0  failed=0  skipped=15
1638 fluent-bee  : ok=62  changed=12  unreachable=0  failed=0  skipped=19
1639 vocal-hyena : ok=66  changed=13  unreachable=0  failed=0  skipped=15
1640 rescued=0  ignored=1
1641 Playbook run took 0 days, 0 hours, 20 minutes, 9 seconds
1642 Tuesday 20 June 2023 16:26:00 +0000 (0:00:00.066) 0:20:09.467 ****
1643 =====
1644 infra.leapp.upgrade : Start Leapp OS upgrade ----- 361.10s
1645 infra.leapp.upgrade : Reboot to continue Leapp OS upgrade ----- 313.97s
1646 infra.leapp.upgrade : Start Leapp OS upgrade ----- 93.90s
```

Checking the RHEL and Kernel Versions

- ▶ You can refresh the RHEL Web Console system overview page to show the upgraded RHEL version.

The screenshot shows the Red Hat Enterprise Linux Web Console interface. At the top, there's a header bar with the user information "ec2-user@fluent-bee", "Administrative access", and a "Help" button. Below the header is a search bar with the placeholder "Search". The main content area has a title "fluent-bee running Red Hat Enterprise Linux 9.2 (Plow)" where "Red Hat Enterprise Linux 9.2 (Plow)" is highlighted in yellow. There are three tabs: "System" (disabled), "Overview" (selected and highlighted in blue), and "Logs". On the right side, there's a "Health" section with a green checkmark and the text "System is up to date".

- ▶ If you prefer the shell prompt, try these commands:

```
cat /etc/redhat-release  
uname -r
```

```
[ec2-user@fluent-bee ~]$ cat /etc/redhat-release  
Red Hat Enterprise Linux release 9.2 (Plow)  
[ec2-user@fluent-bee ~]$ uname -r  
5.14.0-284.11.1.el9_2.x86_64  
[ec2-user@fluent-bee ~]$ █
```

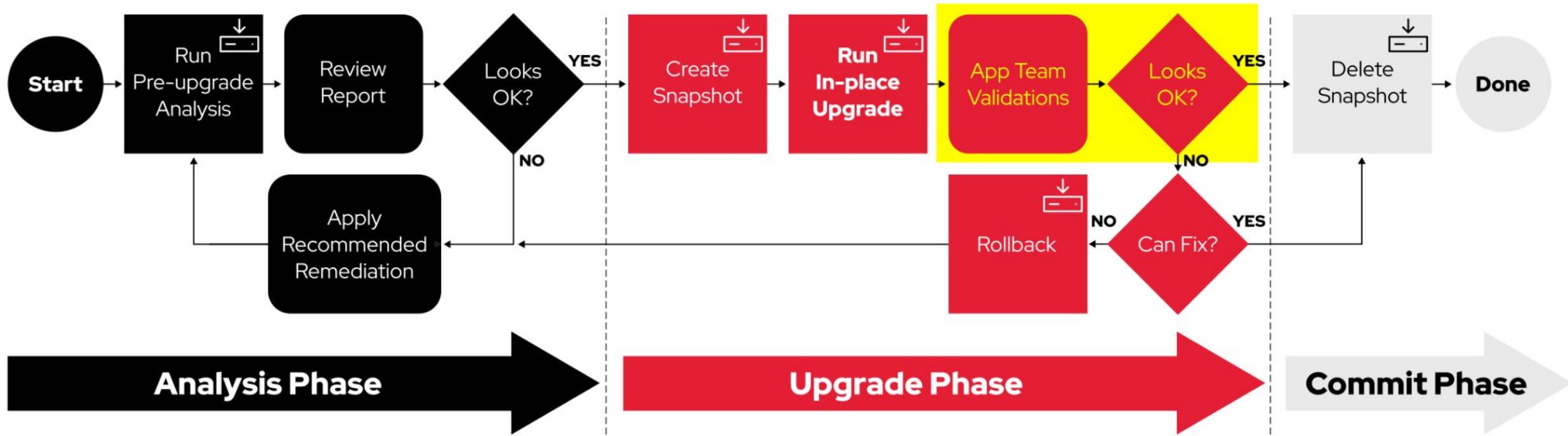


Lab Time - Check if the Upgrades Worked

Complete exercise 2.3 in your lab environment now



RHEL In-place Upgrade Automation Workflow



ⓘ Note

The icon indicates workflow steps that are automated by Ansible playbooks.



Lab Time - How is the Pet App Doing?

Complete exercise 2.4 in your lab environment now



Section 3

Rolling Back

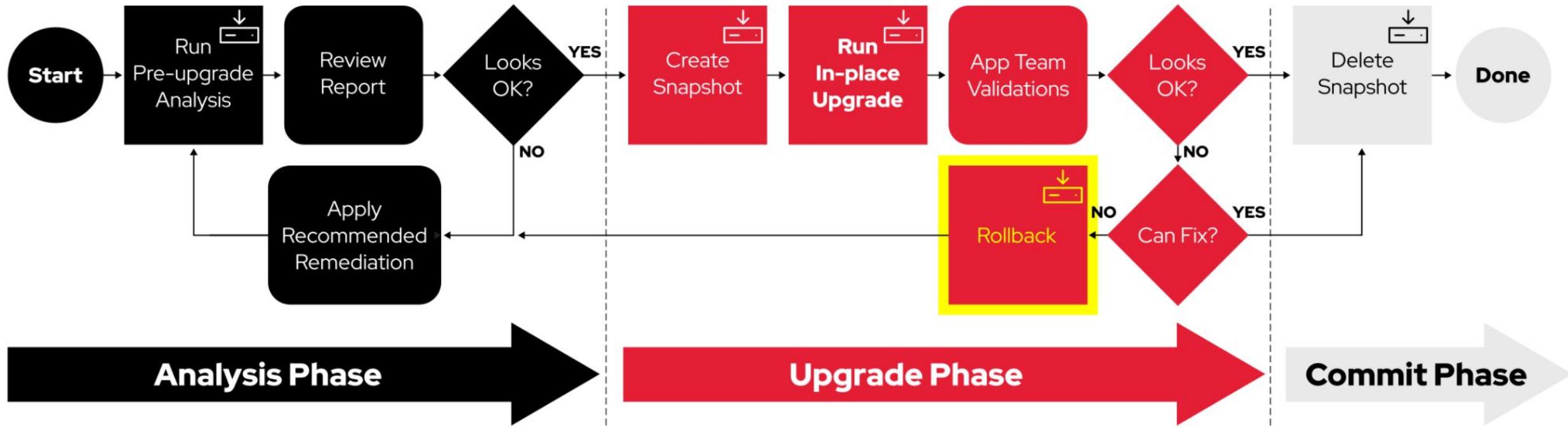


Red Hat
Ansible Automation
Platform



Red Hat
Enterprise Linux

RHEL In-place Upgrade Automation Workflow



i Note

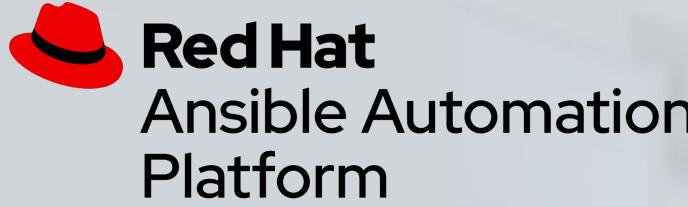
The icon indicates workflow steps that are automated by Ansible playbooks.

Simulate a Failed Upgrade or Application Impact

- ▶ To help demonstrate the effectiveness of rolling back, we're going to intentionally mess up one of our upgraded hosts.
- ▶ Get your hammer ready. The next exercise offers a number of ideas for different ways to destroy your host.



youngthousands from usa, CC BY 2.0



Lab Time - Trash the Instance

Complete exercise 3.1 in your lab environment now



Rolling Back the Upgrade

- ▶ The rollback playbook uses the snapshot that was automatically created by the upgrade workflow job and reverts an instance back to its previous state.
- ▶ This job completed in just under 2 minutes!
- ▶ After rolling back, we see the RHEL and kernel versions are the same as before we upgraded.

The screenshot shows the Red Hat Ansible Automation Platform web interface. On the left, a sidebar navigation includes 'Views' (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals), 'Resources' (Templates, Credentials, Projects, Inventories, Hosts), 'Access' (Organizations, Users, Teams), and 'Administration' (Credential Types, Notifications). The main area is titled 'Jobs > 16 - AUTO / 03 Rollback' and shows a status of 'Successful'. It provides details like Plays (1), Tasks (35), Hosts (1), and Elapsed time (00:01:57). The 'Output' tab is selected, displaying the command-line logs of the Ansible playbook execution. The logs show various tasks being run on an EC2 instance, including stopping instances, creating EBS volumes from snapshots, and attaching them. The final log entry shows the creation of a new EBS volume with a duration of 0.17s.

```
[student@ansible-1 ~]$ ssh causal-shark
Last login: Wed Jun 28 01:13:40 2023 from ansible-1.example.com
[ec2-user@causal-shark ~]$ cat /etc/redhat-release
Red Hat Enterprise Linux Server release 7.9 (Maipo)
[ec2-user@causal-shark ~]$ uname -r
3.10.0-1160.92.1.el7.x86_64
[ec2-user@causal-shark ~]$
```



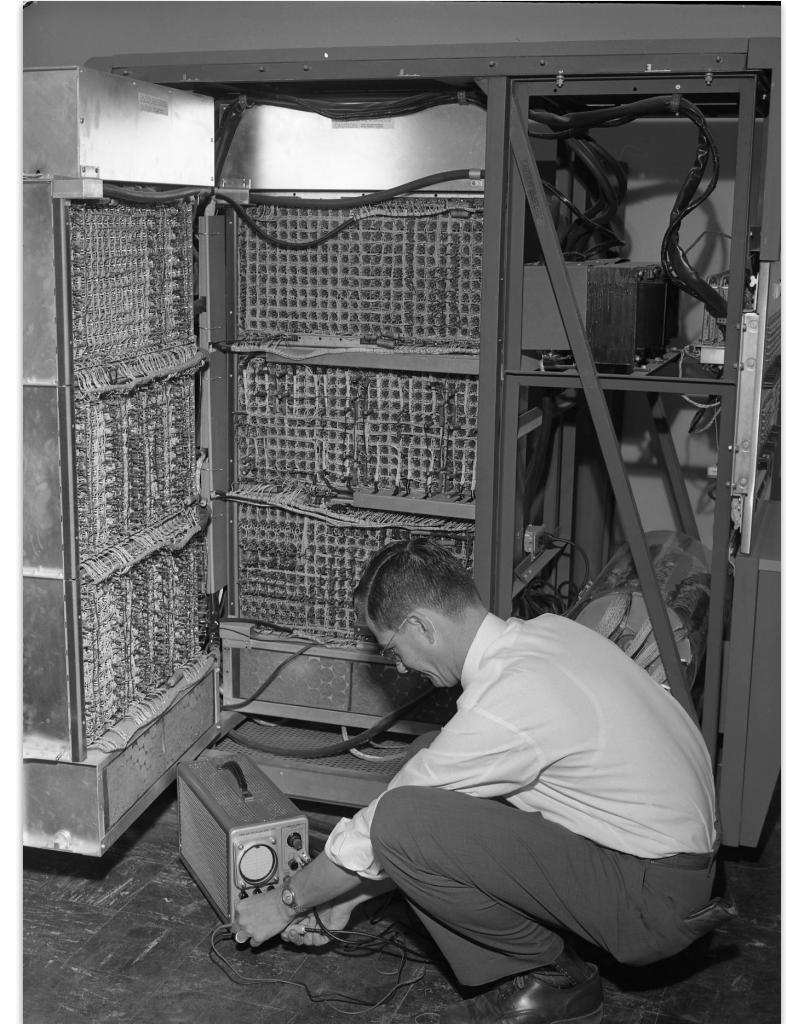
Lab Time - Run Rollback Job

Complete exercise 3.2 in your lab environment now



Is Everything Working Again?

- ▶ After rolling back, repeat the observations we made on our host after the upgrade with the expectation that everything is back as it was before the upgrade.
- ▶ Look for any app data you added or modified after the upgrade and you will find that all those changes are lost. What does this tell us about the snapshot scope implemented by our rollback playbook?
- ▶ In the next exercise, we will assess our state after rolling back and consider next steps.



Cushing Memorial Library and Archives, Texas A&M, CC BY 2.0



Lab Time - Check if Upgrade Undone

Complete exercise 3.3 in your lab environment now



Congratulations!

- ▶ You have reached the end of the workshop. You are now armed with the knowledge needed to start developing an automation solution to help your organization manage RHEL upgrades at scale.
- ▶ The workshop lab environment is now yours to play with. Dream up your own ideas for additional learning and experimentation. Remember you can upgrade and roll back as often as you like. Rinse and repeat!



McElspeth



Lab Time - Rinse and Repeat

Complete exercise 3.4 in your lab environment now



Checkout the Code

- ▶ All of the Ansible roles and playbooks used in this workshop are maintained in open source repositories.
- ▶ Take some time to review the code and get engaged with the communities supporting these resources.

[redhat-cop/infra.leapp](#)

The `infra.leapp` collection provides the Ansible role that generates the pre-upgrade reports and another that is used to perform the RHEL upgrades. This collection uses the Leapp framework for upgrades from RHEL 7 and later, but also supports upgrading from RHEL 6 using the older Red Hat Upgrade Tool. The collection is published on Ansible Galaxy [here](#) and also available from Ansible Automation Hub validated content [here](#). If you are planning to do RHEL in-place upgrades for your organization, these roles will help you quickly roll out proof-of-concept automation and start upgrading.

[oamg/leapp-supplements](#)

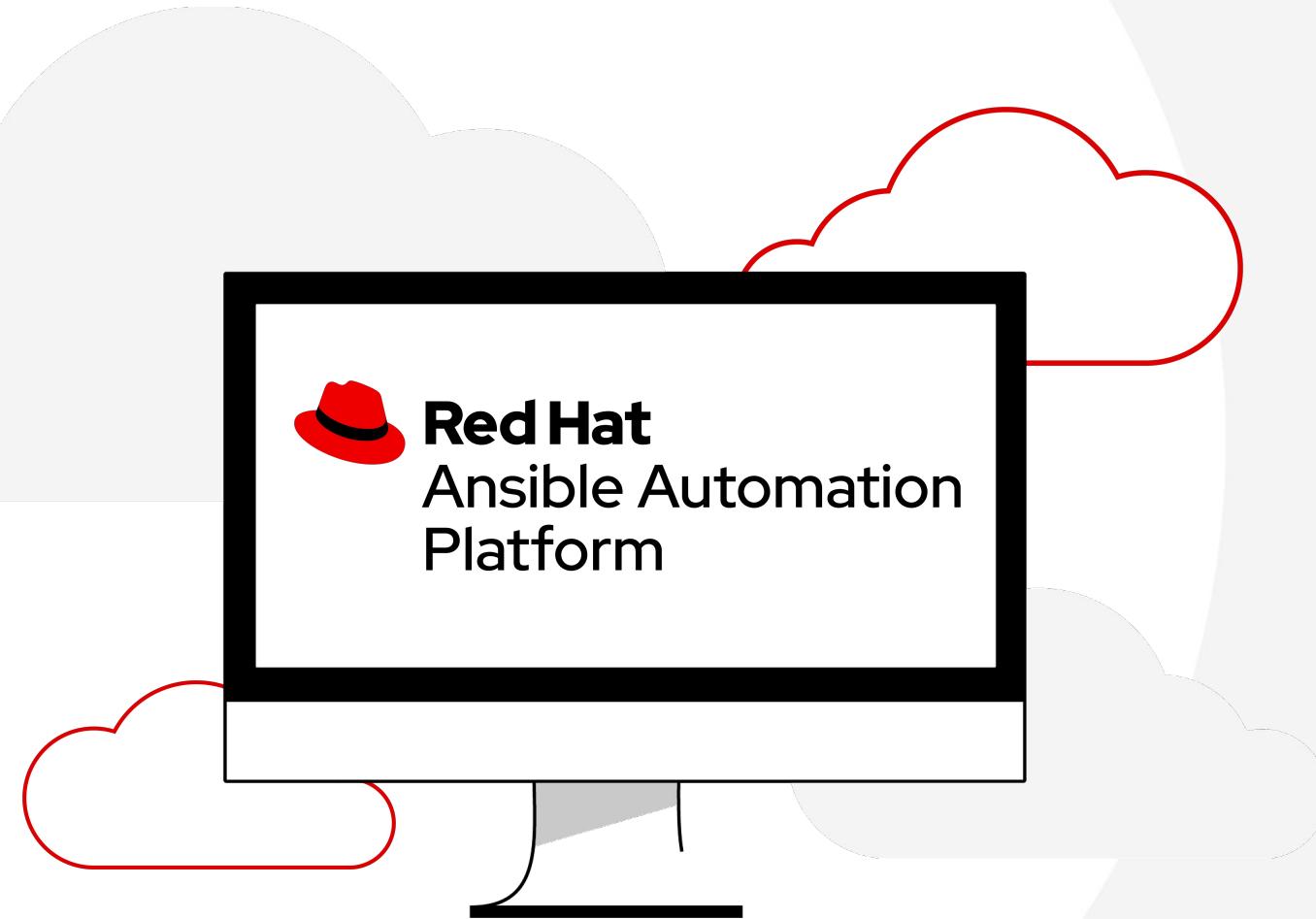
Leapp Supplements is a repository of example Leapp custom actors. The `CheckRebootHygiene` actor that was demonstrated in the optional [Custom Pre-upgrade Checks](#) exercise is maintained here. There is also a Makefile and RPM spec file that can be used to build packages for installing your Leapp custom actors.

[redhat-partner-tech/leapp-project](#)

This is where you will find all of the AAP job templates and Ansible playbooks included in the workshop. You can also explore the infrastructure as code (IaC) magic that is used to provision the workshop lab environment.

[swapdisk/snapshot](#)

Here you will find work in progress on a new Ansible role for managing snapshot sets using LVM. If you are interested in automating LVM snapshots as explained in the [Let's Talk About Snapshots](#) exercise, connect with the authors of this project to get in on the action.



Where to go next

Learn more

- ▶ [Workshops](#)
- ▶ [Documents](#)
- ▶ [Youtube](#)
- ▶ [Twitter](#)

Get started

- ▶ [Evals](#)
- ▶ [cloud.redhat.com](#)

Get serious

- ▶ [Red Hat Automation Adoption Journey](#)
- ▶ [Red Hat Training](#)
- ▶ [Red Hat Consulting](#)

Thank you



linkedin.com/company/red-hat



youtube.com/AnsibleAutomation



facebook.com/ansibleautomation



twitter.com/ansible



github.com/ansible