Red Hat Ansible Automation Platform 2.3

Upgrades Engagement Journey

For Hong Kong Jockey Club

# Preface

## Confidentiality, Copyright, and Disclaimer

This is a Customer-facing document between Red Hat, Inc. and Hong Kong Jockey Club (“HKJC”).

Copyright ©2023 Red Hat, Inc. All Rights Reserved. No part of the work covered by the copyright herein may be reproduced or used in any form or by any means – graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems without permission in writing from Red Hat except as is required to share this information as provided with the aforementioned confidential parties.

This document is not a quote and does not include any binding commitments by Red Hat. If acceptable, a formal quote can be issued upon request, which will include the scope of work, cost, and any customer requirements as necessary.

## Trademarks

Trademarked names may appear throughout this document. Rather than list the names and entities that own the trademarks or insert a trademark symbol with each mention of the trademarked name, the names are used only for editorial purposes and to the benefit of the trademark owner with no intention of infringing upon that trademark.

## Audience

This document is intended for Client technical staff responsible for the environment.

## Additional Background and Related Documents

This document does not contain step by step details of installation or other tasks, as they are covered in the relevant documentation on <http://access.redhat.com/>.

Links to the appropriate documents will be made when required.

## Scripts and playbooks

Any scripts provided are being provided as-is, without any form of support or warranty.

All provided scripts can be modified by the customer at will.

# 

# Version history

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Contributor** | **Role** | **Description** |
| 1.0 | 2023-06-14 | Raymond Lui | Red Hat Consultant | **First Version of the document** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table of Contents

[**Preface**](#_gzdt39j8e6nr) **2**

[Confidentiality, Copyright, and Disclaimer](#_x29dipxs95a0) 2

[Trademarks](#_k406uvkp5ri6) 2

[Audience](#_9pxxylzgrkin) 2

[Additional Background and Related Documents](#_jzcw3szhsm95) 2

[Scripts and playbooks](#_ucfa9zqebjg2) 2

[**Version history**](#_b8aixy9oaoqn) **3**

[**1. Introduction**](#_cgvwxcn0xpcw) **5**

[1.1. Purpose](#_1gj7vt677mmz) 5

[1.2. Staffing](#_fm2bhgdm5gik) 5

[1.3. Terms and acronyms](#_4wwd1cu6ybzy) 6

[**2. UPGRADES PROCEDURE**](#_eyeft0vuoouz) **7**

[2.1 Pre-Upgrades Steps](#_3p7014z5tfor) 7

[2.2. Run backup before upgrades (if no offline VM snapshot taken)](#_wo4sus4xq9ue) 8

[2.3 Run Ansible Tower Upgrades](#_6ope855q93v0) 8

[2.4 Create and Apply New Ansible Tower License Manifest](#_1tsdbl4ft283) 9

# Introduction

## Purpose

The purpose of this document is to provide the engagement Journey of the upgrades of Red Hat Ansible Tower, from version 3.8.4 to Ansible Automation Platform 2.3, for HKJC.

## Staffing

*Customer project team*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Function | E-mail address | Phone number |
| Alex Wong | Infrastructure administrator | alex.cs.wong@hkjc.org.hk | +852 9722 09193 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

*Red Hat project team*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Function | E-mail address | Phone number |
| Raymond Lui | Consultant | yelui@redhat.com | +852 96027743 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 

## Terms and acronyms

The table below provides a glossary of the terms and acronyms used within this document.

|  |  |
| --- | --- |
| Acronym | Description |
| HKJC | Hong Kong Jockey Club |
| RH | Red Hat, Inc |
| RHAT | Red Hat Ansible Tower |
| RHEL | Red Hat Enterprise Linux |
| AD | Active Directory |
| API | Application Programming Interface |
| CA | Certificate Authority |
| DC | Data Centre |
| DNS | Domain Name System |
| DHCP | Dynamic Host Configuration Protocol |
| FQDN | Fully Qualified Domain Name |
| Guest | Also see “VM”. This is virtual machine running on a Host. |
| HA | High-Availability or Highly-Available |
| Host | The physical hardware or the logical OS which runs virtualisation technology allowing one or more Guest OS’s to run on the hardware owned by the Host |
| L2 | Layer 2, part of the TCI/IP Network Stack |
| L3 | Layer 3, part of the TCI/IP Network Stack |
| NAT | Network Address Translation |
| NIC | Network Interface Card. References a virtual or a physical port allowing network access and interface to a Host or Guest VM. |
| NTP | Network Time Protocol |
| OS | Operating System |
| QA | Quality Assurance |
| SAN | Storage Area Network |
| SSL | Secure Sockets Layer |
| TLS | Transport Layer Security |
| VLAN | Virtual LAN is a networking virtualisation technology |
| VM | Virtual machine, in VMware terms, synonymous with “Guest” or “ VM Guest” |

# 

# Upgrades Summary

## Upgrade steps summary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Name | Site | Category | Action |
| 1 | qcstcfgp01 | ST | Ansible Tower | Upgrade from 3.8.4 to 3.8.6 |
| 2 | qcstcfgp01 | ST | Ansible Tower | Backup Ansible Tower |
| 3 | qccfgsatwin02 | HV | RHEL | Upgrade from RHEL 7.7 to RHEL 7.9 |
| 4 | qccfgsatwin02 | HV | RHEL | Upgrade from RHEL 7.9 to RHEL 8.8 |
| 5 | qchvcfgp01 | HV | RHEL | Fresh Install RHEL 8.8 |
| 6 | qchvcfgp01 + qccfgsatwin02 | HV | Ansible Tower | Fresh Install Ansible Tower 3.8.6 |
| 7 | qchvcfgp01 | HV | Ansible Tower | Restore from ST Ansible Tower 3.8.6 Backup |
| 8 | qchvcfgp01 + qccfgsatwin02 | HV | AAP2 | Upgrade from Ansible Tower 3.8.6 to AAP 2.2 |
| 9 | qccfgsatwin02 | HV | RHEL | Upgrade from RHEL 8.8 to RHEL 9.1 |
| 10 | qchvcfgp01 | HV | RHEL | Upgrade from RHEL 8.8 to RHEL 9.1 |
| 11 | qchvcfgp01 + qccfgsatwin02 | HV | AAP2 | Upgrade from AAP 2.2 to AAP 2.3 |

## Upgrade sources

rhel7.9-x86\_64-dvd.iso

rhel8.8-x86\_64-dvd.iso

ansible-tower-setup-bundle-3.8.6-2.tar.gz

ansible-automation-platform-setup-bundle-2.2.2-1.tar.gz

ansible-automation-platform-setup-bundle-2.3.2-1.tar.gz

leapp-software.tar.gz

## Ports Need To Be Opened In The Firewall For Ansible Automation Platform 2 Services

|  |  |  |
| --- | --- | --- |
| From | To | Port |
| Client | controller | 443/tcp |
| Client | controller | 80/tcp |
| Client | controller | 22/tcp |
| controller | execution node | 27199/tcp |
| controller | execution node | 22/tcp |

Ref: https://access.redhat.com/solutions/6756251

# Upgrades Procedure

## Upgrade Ansible Tower in ST from 3.8.4 to 3.8.6

### Pre-Upgrades Steps

1. Upload the ***ansible-tower-setup-bundle-3.8.6-2.tar.gz*** to Ansible Tower.
2. Login Ansible Tower, as root user.
3. Set umask to 0022.

|  |
| --- |
| # umask 0022 |

1. Extract the setup bundle to /opt/ansible.

(Please make sure / has enough spaces.)

|  |
| --- |
| # cd /opt/ansible  # tar zxvf <path of setup bundle>/ansible-tower-setup-bundle-3.8.6-2.tar.gz |

1. Backup the original inventory file.

|  |
| --- |
| # cd ansible-tower-setup-bundle-3.8.6-2/  # cp inventory inventory.org |

1. Copy the Ansible Tower 3.8.4 inventory file to 3.8.6 folder.

|  |
| --- |
| # cp ../ansible-tower-setup-bundle-3.8.4-1/inventory . |

1. Edit the inventory file, as follows:

(Also make sure the admin password and PostgreSQL password are set.)

|  |
| --- |
| # vi inventory  …  …  …  …  [all:vars]  ## Upgrades Tower with upgrade Ansible engine  upgrade\_ansible\_with\_tower=true  ….  …. |

1. Check if Ansible Tower’s License is still in compliance/valid.

|  |
| --- |
| Login into Ansible Tower WebUI, as **admin** user  **Settings** -> **License**  See if it is still in valid license and check expires on date. Upload the valid license before running upgrades. |

### Run backup before upgrades (if no offline VM snapshot taken)

1. Change the backup destination folder, to avoid out of space on / volume.

|  |
| --- |
| # cd /opt/ansible/ansible-tower-setup-bundle-3.8.4-1  # vi /opt/ansible/ansible-tower-setup-bundle-3.8.4-1/roles/backup/defaults/main.yml  ---  backup\_dir: /var/backups/tower/  #backup\_dest: "{{ playbook\_dir }}/"  backup\_dest: /var/backups/ |

1. Run Ansible Tower backup.

|  |
| --- |
| # /opt/ansible/ansible-tower-setup-bundle-3.8.4-1/setup.sh -b |

### Run Ansible Tower Upgrades

1. Run Ansible Tower upgrades.

|  |
| --- |
| # /opt/ansible/ansible-tower-setup-bundle-3.8.6-1/setup.sh |

1. After Ansible Tower upgrades successfully, please restart the services once.

|  |
| --- |
| # ansible-tower-service restart |

1. Check Ansible tower services.

|  |
| --- |
| # ansible-tower-service status |

1. Go to Ansible Tower API ping page to see the status:

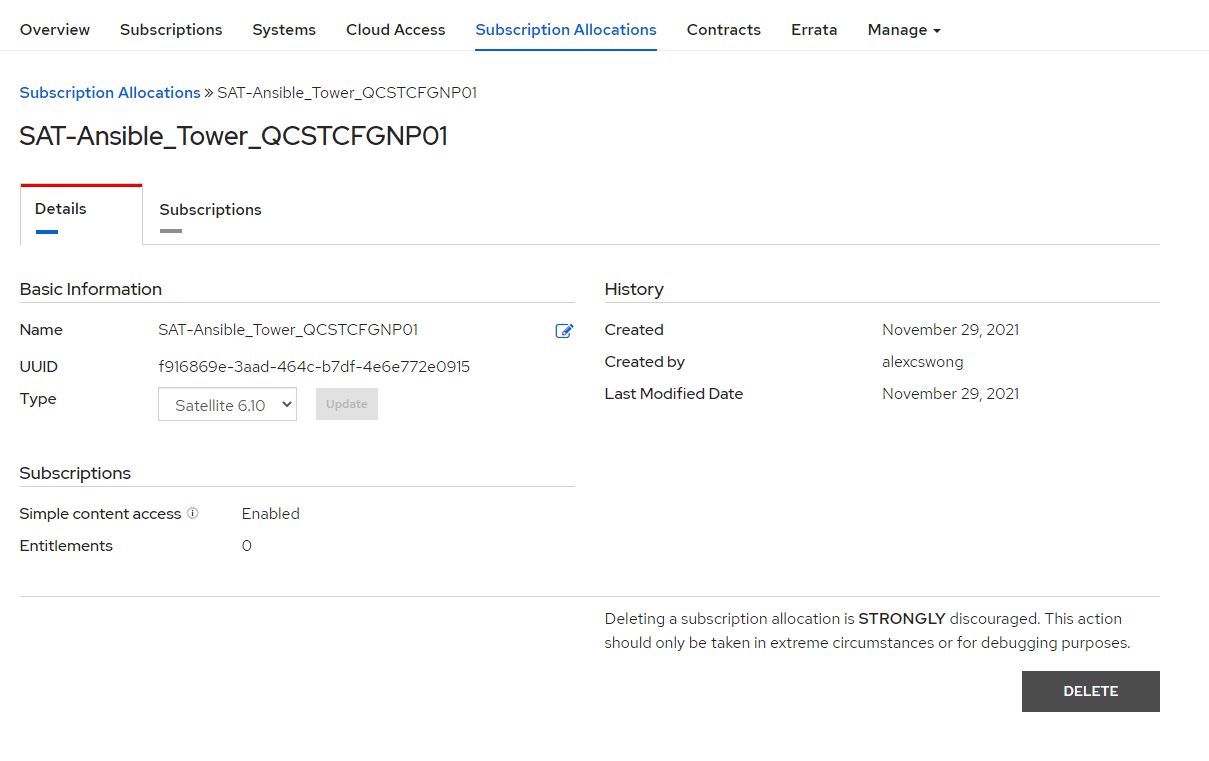
|  |
| --- |
| Go to Ansible Tower API ping page:  https://<ip address>/api/v2/ping  Tower and Isolated node should be shown in the page. |

### Create and Apply New Ansible Tower License Manifest

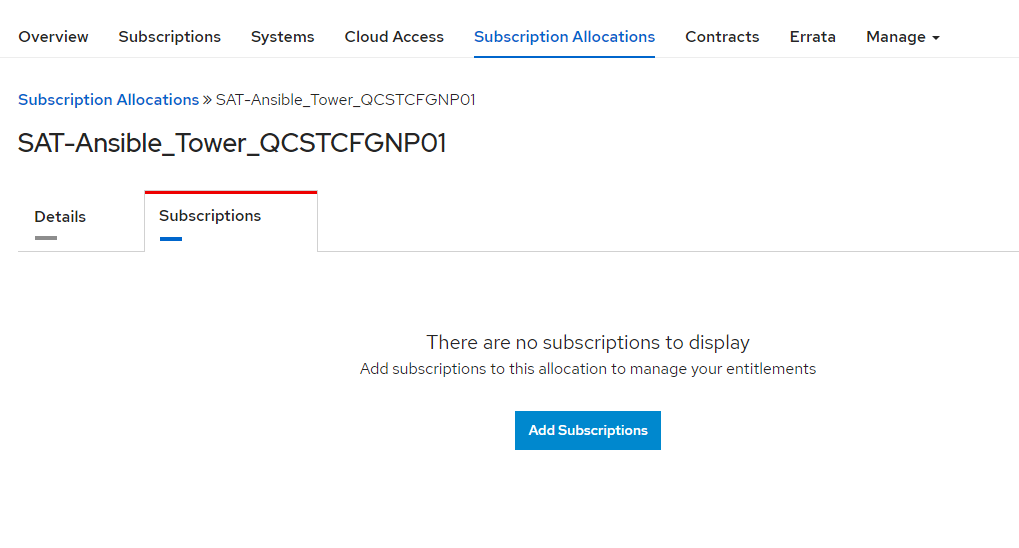
1. Open the browser, and go to <http://access.redhat.com>.
2. Login with “Manage Subscription” permission user.
3. Click to **Subscriptions** on Top right corner:

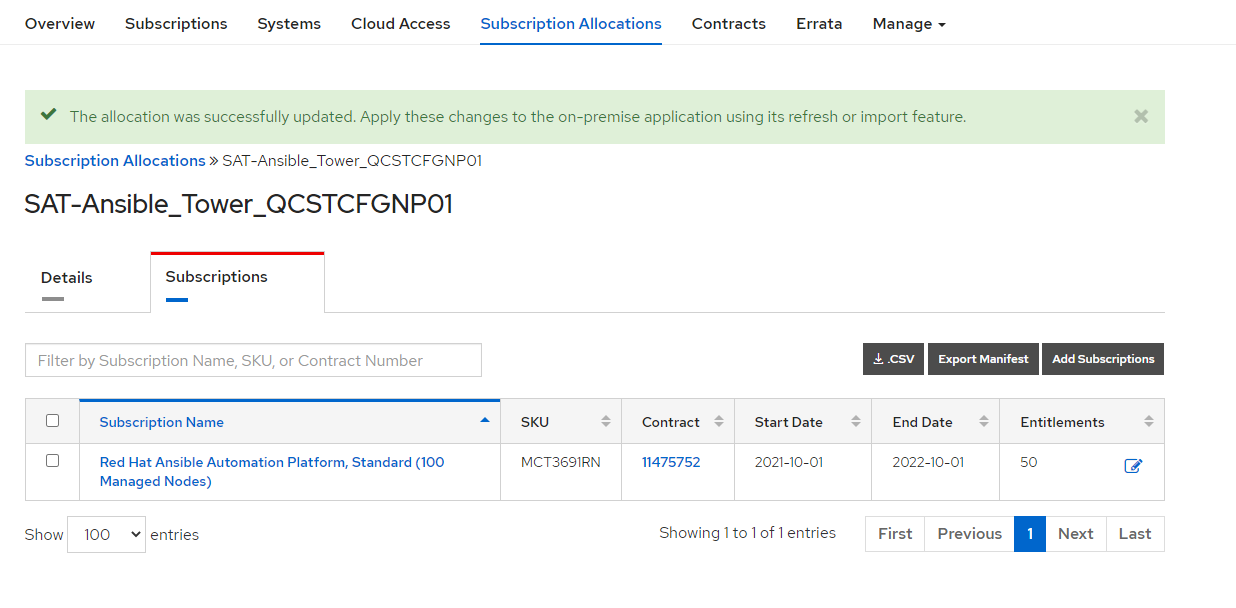


1. Click on the **Subscription Allocations** tab, then create the new allocation (choose the type as Satellite 6.10)



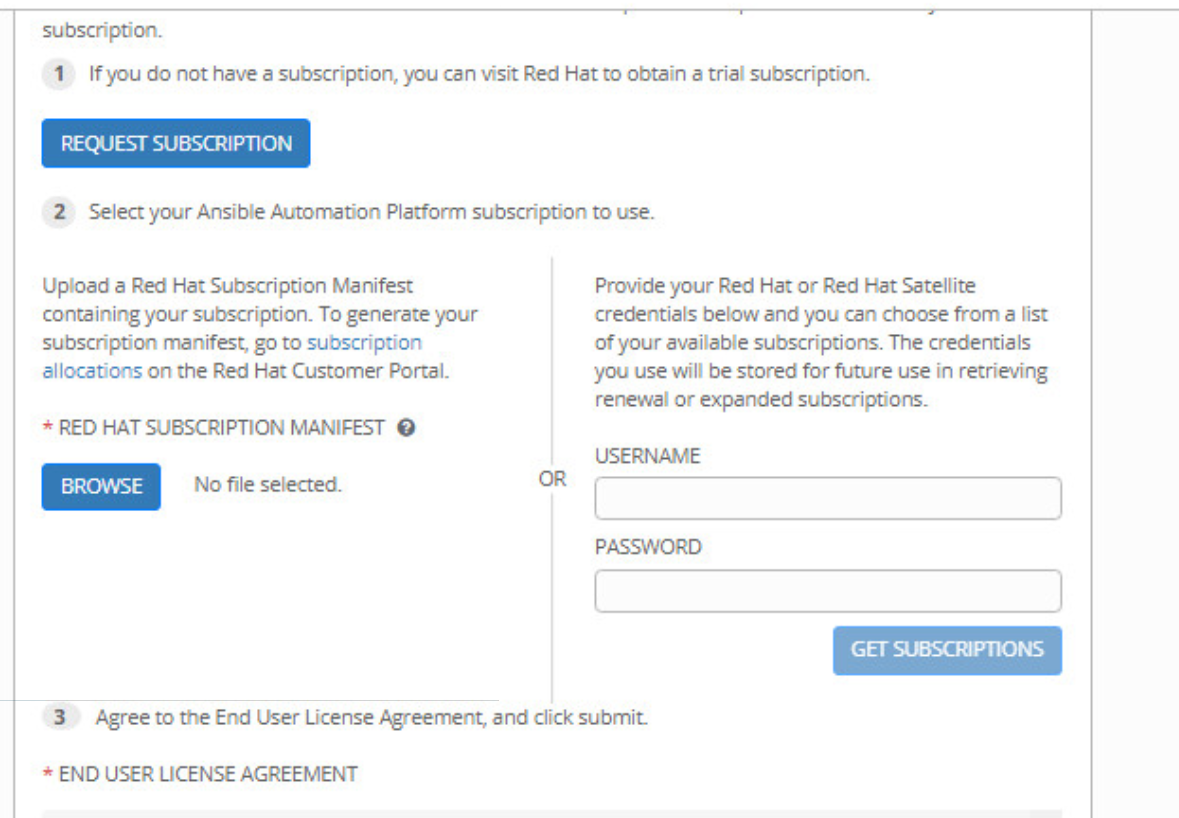
1. Add the **Ansible Automation Platform (Manged Nodes) Subscription**, and allocate the quantity.

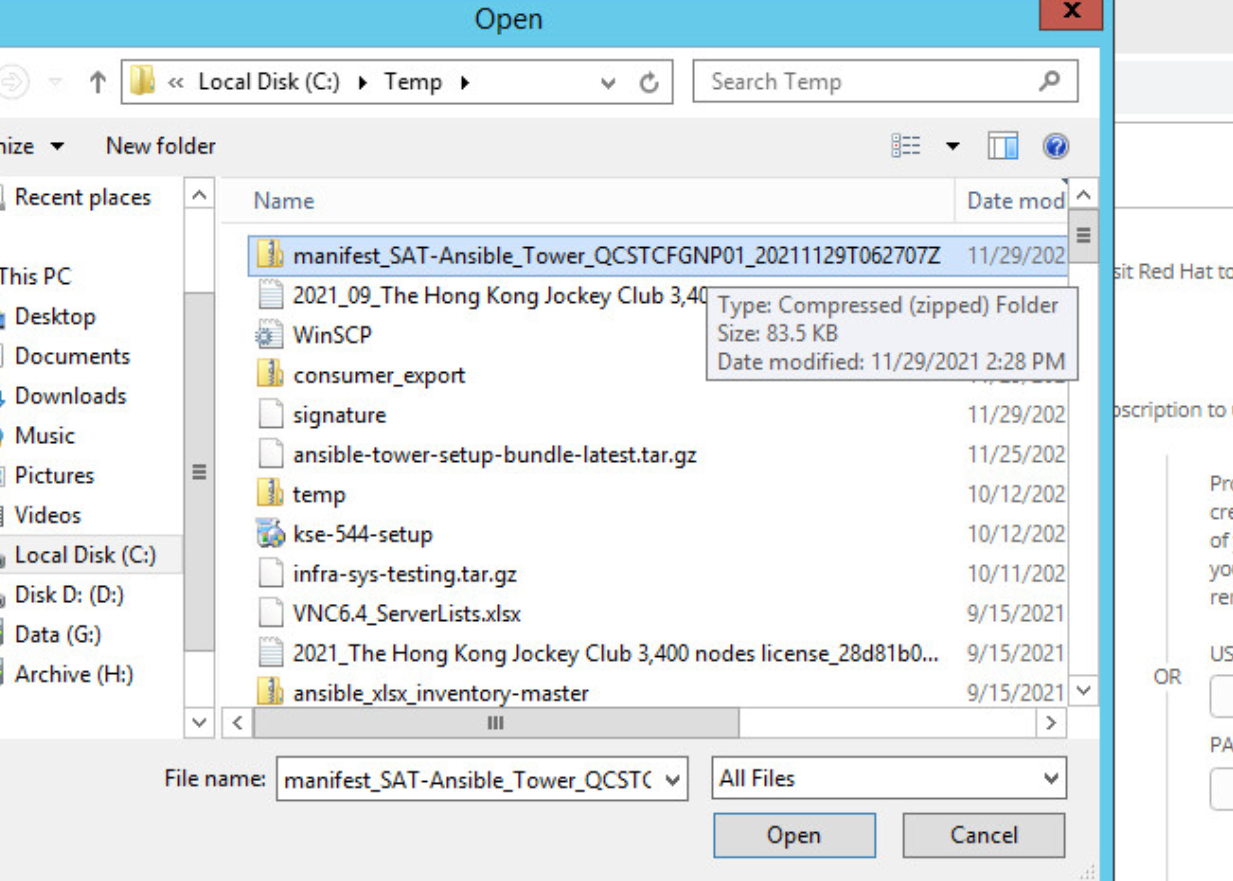




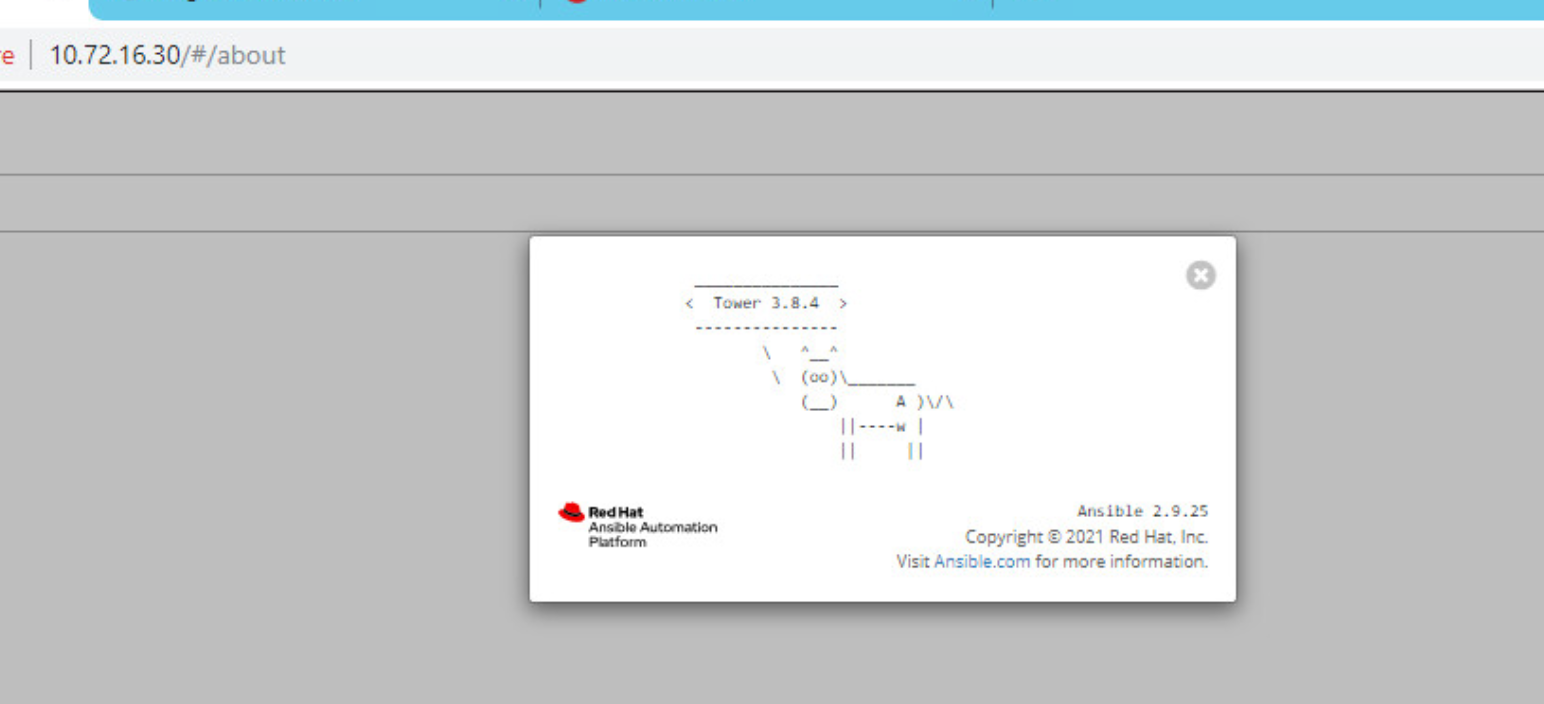
Click **Export Manifest** to download the manifest zip file to the local PC.

1. Login to Ansible Tower WebUI, and upload the Manifest zip file.





1. Tick to agree to the agreement, and then OK to apply the License Manifest.
2. Ansible Tower Dashboard should be now shown.
3. Click on the **i** button on the Top right to check the Ansible Tower and Ansible Version.



## Backup Ansible Tower 3.8.6 in ST

### Run backup for Ansible Tower

1. Change the backup destination folder, to avoid out of space on / volume.

|  |
| --- |
| # cd /opt/ansible/ansible-tower-setup-bundle-3.8.6-2  # vi /opt/ansible/ansible-tower-setup-bundle-3.8.6-2/roles/backup/defaults/main.yml  ---  backup\_dir: /var/backups/tower/  #backup\_dest: "{{ playbook\_dir }}/"  backup\_dest: /var/backups/ |

1. Run Ansible Tower backup.

|  |
| --- |
| # /opt/ansible/ansible-tower-setup-bundle-3.8.6-2/setup.sh -b |

1. Verify backup file and this file will be used for restore Ansible Tower in HV.

|  |
| --- |
| # cd /var/backups  # ls  tower-backup-latest.tar.gz |

## Upgrade from RHEL7.7 to RHEL 7.9

### Pre-Upgrades Steps

1. Attach ISO file rhel7.9-x86\_64-dvd.iso in qccfgsatwin02 in vCenter
2. Create directory

|  |
| --- |
| # mkdir /media/rheldvd |

1. Mount dvd

|  |
| --- |
| # mount /dev/sr0 /media/rheldvd |

1. Config repo file

|  |
| --- |
| # vi /etc/yum.repos.d/rhel79dvd.repo |
| [InstallMedia]  name=DVD for Red Hat Enterprise Linux 7.9 Server  baseurl=file:///media/rheldvd/  gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release  gpgcheck=1  enabled=1 |

1. Clear the cache and check whether you can get the packages list from the DVD repo

|  |
| --- |
| # yum clean all  # yum repolist enabled |

1. Run below command to update and reboot

|  |
| --- |
| # yum update  # reboot |

1. Verify that the current OS version is Red Hat Enterprise Linux 7.9

|  |
| --- |
| # cat /etc/redhat-release |

1. Check the OS kernel version

|  |
| --- |
| # uname -r |

## Upgrade from RHEL7.9 to RHEL 8.8

### Prepare of Hosting the repositories (Example using qccfgsatwin01)

1. Source file for leapp upgrade:

* leapp-software.tar.gz

1. Install and enable the Apache httpd service

|  |
| --- |
| # yum install httpd -y  # systemctl enable --now httpd |

1. Ensure that TCP network port 80 is open

|  |
| --- |
| # iptables -S  -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT |

1. Create sub-directory in HTTP document root.

|  |
| --- |
| # mkdir /var/www/html/rhel88 |

1. Attach ISO file rhel8.8-x86\_64-dvd.iso in qccfgsatwin01 in vCenter
2. Mount the RHEL 8.8 ISO

|  |
| --- |
| # mount /dev/sr0 /var/www/html/rhel88 |

1. Ensure the YUM repository metadata is being served by Apache

|  |
| --- |
| # curl http://localhost/rhel88/BaseOS/repodata/repomd.xml |

1. Copy the leapp-software to another sub-directory in the HTTP document root

|  |
| --- |
| # tar -xvf leapp-software.tar.gz -C /var/www/html/ |

### Config the machine to be in-place upgraded (qccfgsatwin02)

1. Define the repository for the Leapp software. Replace "IPADDR" by the IP address or host name of the above Apache server

|  |
| --- |
| # vi /etc/yum.repos.d/leapp.repo |
| [leapp-software]  name=rhel-7-leapp-software  baseurl=http://IPADDR/leapp-software  enabled=1  gpgcheck=0 |

1. Install the Leapp software

|  |
| --- |
| # yum -y install leapp-upgrade |

1. Define the custom repositories. Replace "IPADDR" by the IP address or host name of the above Apache server

|  |
| --- |
| # vi /etc/leapp/files/leapp\_upgrade\_repositories.repo |
| [BaseOS]  name=rhel-8-for-x86\_64-baseos-rpms  baseurl=http://IPADDR/rhel88/BaseOS  [AppStream]  name=rhel-8-for-x86\_64-appstream-rpms  baseurl=http://IPADDR/rhel88/AppStream |

1. Place the Leapp metadata on the system & expand it. Replace "IPADDR" by the IP address or host name of the above Apache server

|  |
| --- |
| # curl -O http://IPADDR/leapp-software/leapp-data-22.tar.gz  # tar -xf leapp-data-22.tar.gz -C /etc/leapp/files  # rm -f leapp-data-22.tar.gz |

1. Try a preupgrade

|  |
| --- |
| # leapp preupgrade --no-rhsm --enablerepo BaseOS --enablerepo AppStream |

1. Read the leapp report and fix any issues encountered

Solution 1:

Leapp preupgrade getting "Inhibitor: Detected loaded kernel drivers which have been removed in RHEL 8. Upgrade cannot proceed."

Ref doc: https://access.redhat.com/solutions/6971716

Solution 2:

leapp upgrade fails on kernel-devel packages

Ref doc: https://access.redhat.com/solutions/4723671

Solution 3:

Update answer file

|  |
| --- |
| # leapp answer --section remove\_pam\_pkcs11\_module\_check.confirm=True |

Solution 4:

If running leapp with idna version errors, remove idna-2.4 files

|  |
| --- |
| # rm /usr/lib/python2.7/site-packages/idna\* -rf |

Install idna-2.6

|  |
| --- |
| # tar -xvf idna-2.6.tar.gz -C /usr/lib/python2.7/site-packages/ |

Solution 5:

leapp preupgrade command is failing with python traceback ImportError: cannot import name UnrewindableBodyError

Ref: https://access.redhat.com/solutions/6971428

Solution 6:

Leapp preupgrade failed with "pkg\_resources.DistributionNotFound: urllib3>=1.21.1,<1.24"

Ref: https://access.redhat.com/solutions/6644591

1. Run the upgrade

|  |
| --- |
| # leapp upgrade --no-rhsm --enablerepo BaseOS --enablerepo AppStream |

1. Reboot

|  |
| --- |
| # reboot |

1. Verify that the current OS version is Red Hat Enterprise Linux 8

|  |
| --- |
| # cat /etc/redhat-release |

1. Check the OS kernel version

|  |
| --- |
| # uname -r |

1. Remove any remaining Leapp packages from the exclude list in the /etc/dnf/dnf.conf

|  |
| --- |
| # yum config-manager --save --setopt exclude='' |

1. Remove remaining RHEL 7 packages, Determine old kernel versions

|  |
| --- |
| # cd /lib/modules && ls -d \*.el7\* |

1. Remove weak modules from the old kernel. If you have multiple old kernels, repeat the following step for each kernel:

|  |
| --- |
| # [ -x /usr/sbin/weak-modules ] && /usr/sbin/weak-modules --remove-kernel <version> |

1. Remove the old kernel from the boot loader entry. If you have multiple old kernels, repeat this step for each kernel:

|  |
| --- |
| # /bin/kernel-install remove <version> /lib/modules/<version>/vmlinuz |

1. Locate and Remove remaining RHEL 7 packages

|  |
| --- |
| # rpm -qa | grep -e '\.el[67]' | grep -vE '^(gpg-pubkey|libmodulemd|katello-ca-consumer)' | sort |

1. Remove remaining Leapp dependency packages

|  |
| --- |
| # yum remove leapp-deps-el8 leapp-repository-deps-el8 |

1. Remove any remaining empty directories

|  |
| --- |
| # rm -r /lib/modules/\*el7\* |

1. Replace the old rescue kernel and initial RAM disk with the current kernel and disk. Remove the existing rescue kernel and initial RAM disk:

|  |
| --- |
| # rm /boot/vmlinuz-\*rescue\* /boot/initramfs-\*rescue\* |

1. Reinstall the current kernel to recover the rescue kernel and related initial RAM disk:

|  |
| --- |
| # cp /etc/leapp/files/leapp\_upgrade\_repositories.repo /etc/yum.repos.d/  # vi /etc/yum.repos.d/ |
| [BaseOS]  name=rhel-8-for-x86\_64-baseos-rpms  baseurl=http:// IPADDR /rheldvd/BaseOS  enabled=1  [AppStream]  name=rhel-8-for-x86\_64-appstream-rpms  baseurl=http:// IPADDR /rheldvd/AppStream  enabled=1 |
| dnf reinstall -y kernel-core-$(uname -r) |

1. Verify that the old kernels have been removed from the bootloader entry:

|  |
| --- |
| # grubby --info=ALL | grep "\.el7" || echo "Old kernels are not present in the bootloader." |

1. Verify that the previously removed rescue kernel and rescue initial RAM disk files have been created for the current kernel:

|  |
| --- |
| # ls /boot/vmlinuz-\*rescue\* /boot/initramfs-\*rescue\* |
| # lsinitrd /boot/initramfs-\*rescue\*.img | grep -qm1 "$(uname -r)/kernel/" && echo "OK" || echo "FAIL" |

1. Verify the rescue boot entry refers to the existing rescue files. See the grubby output

|  |
| --- |
| # grubby --info $(ls /boot/vmlinuz-\*rescue\*) |

1. Open the /etc/selinux/config

|  |
| --- |
| # vi /etc/selinux/config |

Configure the SELINUX=enforcing option:

|  |
| --- |
| # This file controls the state of SELinux on the system.  # SELINUX= can take one of these three values:  # enforcing - SELinux security policy is enforced.  # permissive - SELinux prints warnings instead of enforcing.  # disabled - No SELinux policy is loaded.  SELINUX=enforcing  # SELINUXTYPE= can take one of these two values:  # targeted - Targeted processes are protected,  # mls - Multi Level Security protection.  SELINUXTYPE=targeted |

1. Save the change, and restart the system:

|  |
| --- |
| # reboot |

## Fresh Install RHEL 8

### Installation Steps

1. Attach ISO file rhel8.8-x86\_64-dvd.iso in qchvcfgp01 in vCenter

## Fresh Install Ansible Tower 3.8.6

### Installation Steps

1. Extract the setup bundle to /opt/ansible.

(Please make sure / has enough spaces.)

|  |
| --- |
| # cd /opt/ansible  # tar zxvf <path of setup bundle>/ansible-tower-setup-bundle-3.8.6-2.tar.gz |

1. Copy the ST Ansible Tower 3.8.6 inventory file to HV Ansible Tower 3.8.6. And copy to setup bundler directory to replace the default inventory file.

|  |
| --- |
| # cp <inventory from ST Ansible Tower> /opt/ansible/ansible-tower-setup-bundle-3.8.6-2/inventory |

1. Update hostname

|  |
| --- |
| # hostnamectl set-hostname <hostname.fqdn> |

1. Create user “redhat”

|  |
| --- |
| # useradd redhat |

1. Assign password to user “redhat”

|  |
| --- |
| # passwd redhat |

1. Assign sudo right to user “redhat”

Add the entry line as below.

|  |
| --- |
| # visudo  .  .  .  # wheel ALL=(ALL) NOPASSWD: ALL  redhat ALL=(ALL) NOPASSWD: ALL  .  .  . |

1. Change the setup bundler owner to redhat user

|  |
| --- |
| # chown -R redhat: /opt/ansible/ansible-tower-setup-bundle-3.8.6-2 |

1. Change to user “Redhat”

|  |
| --- |
| # su - redhat |

1. Generate ssh-key

|  |
| --- |
| $ ssh-keygen |

1. Copy ssh key to isolated node (qccfgsatwin02)

|  |
| --- |
| $ ssh-copy-id redhat@192.168.145.17 |

1. Run Ansible Tower Installation.

|  |
| --- |
| $ ANSIBLE\_SUDO=True /opt/ansible/ansible-tower-setup-bundle-3.8.6-2/setup.sh |

1. Check Ansible tower services.

|  |
| --- |
| # ansible-tower-service status |

1. Go to Ansible Tower API ping page to see the status:

|  |
| --- |
| Go to Ansible Tower API ping page:  https://<ip address>/api/v2/ping  Tower and Isolated node should be shown in the page. |

## Restore from ST Ansible Tower 3.8.6 Backup

### Restore Steps

1. Copy the ST Ansible Tower backup file to HV Ansible Tower
2. Change to user “redhat”

|  |
| --- |
| # su - redhat |

1. Restore the backup

(Please make sure / has enough spaces.)

|  |
| --- |
| $ cd /opt/ansible/ansible-tower-setup-bundle-3.8.6-2  $ ANSIBLE\_SUDO=True /opt/ansible/ansible-tower-setup-bundle-3.8.6-2/setup.sh -e ‘<path of backup>/tower-backup-latest.tar’ -r |

1. Check Ansible tower services.

|  |
| --- |
| # ansible-tower-service status |

1. Go to Ansible Tower API ping page to see the status:

|  |
| --- |
| Go to Ansible Tower API ping page:  https://<ip address>/api/v2/ping  Tower and Isolated node should be shown in the page. |

## Upgrade from Ansible Tower 3.8.6 to AAP 2.2

### Upgrade steps

1. Change to user “redhat”

|  |
| --- |
| # su - redhat |

1. Extract the setup bundle to /opt/ansible.

(Please make sure / has enough spaces.)

|  |
| --- |
| # cd /opt/ansible  # tar zxvf <path of setup bundle>/ansible-automation-platform-setup-bundle-2.2.2-1.tar.gz |

1. Backup the original inventory file.

|  |
| --- |
| # cd ansible-automation-platform-setup-bundle-2.2.2-1/  # cp inventory inventory.org |

1. Copy the Ansible Tower 3.8.6 inventory file to 2.2.2 folder.

|  |
| --- |
| # cp ../ansible-tower-setup-bundle-3.8.6-2/inventory . |

1. Update the inventory as follow:

|  |
| --- |
| # cat inventory  [automationcontroller]  qchvcfgp01.corpqc.hkjc.com  [automationcontroller:vars]  peers=execution\_nodes  ansible\_connection='local'  [isolated\_group\_sat\_win]  192.168.145.17  [isolated\_group\_sat\_win:vars]  node\_state='iso\_migrate'  # Execution Nodes  # There are two valid node\_types that can be assigned for this group.  # A node\_type=hop implies that the node will forward jobs to an execution node.  # A node\_type=execution implies that the node will be able to run jobs.  # If you do not define the node\_type, it defaults to execution.  #  # hop.example node\_type=hop  # execution.example node\_type=execution  # execution2.example <- this will default to execution  [execution\_nodes]  [execution\_nodes:children]  isolated\_group\_sat\_win  [automationhub]  [automationcatalog]  [database]  # Single Sign-On  # If sso\_redirect\_host is set, that will be used for application to connect to  # SSO for authentication. This must be reachable from client machines.  #  # ssohost.example sso\_redirect\_host=<host/ip>  [sso]  [all:vars]  admin\_password=’xxxxxx’  pg\_host=''  pg\_port=''  pg\_database='awx'  pg\_username='awx'  pg\_password='xxxxxx'  pg\_sslmode='prefer' # set to 'verify-full' for client-side enforced SSL |

1. Run setup.sh to generate new format of inventory file called inventory.new.ini

|  |
| --- |
| # ./setup.sh |

1. Rename the inventory filename to inventory, and update the content as follows:

(Also make sure the admin password and PostgreSQL password are set.)

|  |
| --- |
| [root@QCCFGSATWIN02 redhat]# cat inventory  [all:vars]  admin\_password='xxxxxx'  pg\_host=''  pg\_port=''  pg\_database='awx'  pg\_username='awx'  pg\_password='xxxxxx'  pg\_sslmode='prefer'  registry\_url='registry.redhat.io'  registry\_username=''  registry\_password=''  receptor\_listener\_port=27199  automationhub\_admin\_password=''  automationhub\_pg\_host=''  automationhub\_pg\_port=5432  automationhub\_pg\_database='automationhub'  automationhub\_pg\_username='automationhub'  automationhub\_pg\_password=''  automationhub\_pg\_sslmode='prefer'  automationcatalog\_pg\_host=''  automationcatalog\_pg\_port=5432  automationcatalog\_pg\_database='automationservicescatalog'  automationcatalog\_pg\_username='automationservicescatalog'  automationcatalog\_pg\_password=''  sso\_keystore\_password=''  sso\_console\_admin\_password=''  tower\_package\_name='automation-controller'  tower\_package\_version='4.2.2'  automation\_platform\_version='2.2.2'  minimum\_ansible\_version='2.12'  bundle\_install='true'  setup\_dir='/opt/ansible/ansible-automation-platform-setup-bundle-2.2.2-1/.'  [automationcontroller]  qchvcfgp01.corpqc.hkjc.com  [automationcontroller:vars]  peers='execution\_nodes'  node\_type=control  ansible\_connection='local'  ansible\_system\_capabilities\_enforced='False'  ansible\_system='Linux'  ansible\_kernel='4.18.0-477.10.1.el8\_8.x86\_64'  ansible\_kernel\_version='#1 SMP Wed Apr 5 13:35:01 EDT 2023'  ansible\_machine='x86\_64'  ansible\_python\_version='3.6.8'  ansible\_fqdn='qchvcfgp01.corpqc.hkjc.com'  ansible\_hostname='qchvcfgp01'  ansible\_nodename='qchvcfgp01.corpqc.hkjc.com'  ansible\_domain='corpqc.hkjc.com'  ansible\_userspace\_bits='64'  ansible\_architecture='x86\_64'  ansible\_userspace\_architecture='x86\_64'  ansible\_machine\_id='fe601f9d044f481594584f5f43058163'  ansible\_is\_chroot=False  ansible\_virtualization\_role='guest'  ansible\_virtualization\_type='VMware'  ansible\_ssh\_host\_key\_rsa\_public=''  ansible\_ssh\_host\_key\_rsa\_public\_keytype='ssh-rsa'  ansible\_ssh\_host\_key\_ecdsa\_public='AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBF3GAPZ7av7ses0Mgw7G4LqhQTfxiRsZe0OMugss+4+qw0K3N6Klw5OPT2ss0d3dPPCSN2HUAuufYT9Ip8riaSc='  ansible\_ssh\_host\_key\_ecdsa\_public\_keytype='ecdsa-sha2-nistp256'  ansible\_ssh\_host\_key\_ed25519\_public='AAAAC3NzaC1lZDI1NTE5AAAAIBrra8kSRu3Y7/t4NJ6XUiFPKQau0bB0vm0ky17jSrhY'  ansible\_ssh\_host\_key\_ed25519\_public\_keytype='ssh-ed25519'  ansible\_distribution='RedHat'  ansible\_distribution\_release='Ootpa'  ansible\_distribution\_version='8.8'  ansible\_distribution\_major\_version='8'  ansible\_distribution\_file\_path='/etc/redhat-release'  ansible\_distribution\_file\_variety='RedHat'  ansible\_distribution\_file\_parsed=True  ansible\_distribution\_file\_search\_string='Red Hat'  ansible\_os\_family='RedHat'  ansible\_user\_id='root'  ansible\_user\_uid=0  ansible\_user\_gid=0  ansible\_user\_gecos='root'  ansible\_user\_dir='/root'  ansible\_user\_shell='/bin/bash'  ansible\_real\_user\_id=0  ansible\_effective\_user\_id=0  ansible\_real\_group\_id=0  ansible\_effective\_group\_id=0  ansible\_selinux\_python\_present=True  ansible\_hostnqn=''  ansible\_fips=False  ansible\_iscsi\_iqn=''  ansible\_processor\_count=4  ansible\_processor\_cores=1  ansible\_processor\_threads\_per\_core=1  ansible\_processor\_vcpus=4  ansible\_processor\_nproc=4  ansible\_memtotal\_mb=15761  ansible\_memfree\_mb=526  ansible\_swaptotal\_mb=4095  ansible\_swapfree\_mb=4093  ansible\_bios\_date='11/12/2020'  ansible\_bios\_vendor='Phoenix Technologies LTD'  ansible\_bios\_version='6.00'  ansible\_board\_asset\_tag='NA'  ansible\_board\_name='440BX Desktop Reference Platform'  ansible\_board\_serial='None'  ansible\_board\_vendor='Intel Corporation'  ansible\_board\_version='None'  ansible\_chassis\_asset\_tag='No Asset Tag'  ansible\_chassis\_serial='None'  ansible\_chassis\_vendor='No Enclosure'  ansible\_chassis\_version='N/A'  ansible\_form\_factor='Other'  ansible\_product\_name='VMware Virtual Platform'  ansible\_product\_serial='VMware-42 10 4e 73 62 75 d6 32-8a 4f 7d 16 2c e9 a2 8b'  ansible\_product\_uuid='42104e73-6275-d632-8a4f-7d162ce9a28b'  ansible\_product\_version='None'  ansible\_system\_vendor='VMware, Inc.'  ansible\_uptime\_seconds=4170  ansible\_pkg\_mgr='dnf'  ansible\_service\_mgr='systemd'  module\_setup=True  discovered\_interpreter\_python='/usr/libexec/platform-python'  # in AAP 2.X isolated groups are no longer a special type, and should be renamed to be instance groups  [instance\_group\_sat\_win]  192.168.145.17  [instance\_group\_sat\_win:vars]  # in AAP 2.X Isolated Nodes are converted into Execution Nodes using node\_state=iso\_migrate  node\_state='iso\_migrate'  # In AAP 2.X Execution Nodes have replaced isolated nodes. All of these nodes will be by default  # `node\_type=execution`. You can specify new nodes that cannot execute jobs and are intermediaries  # between your control and execution nodes by adding them to [execution\_nodes] and setting a host var  # `node\_type=hop` on them.  [execution\_nodes]  [execution\_nodes:children]  instance\_group\_sat\_win  [execution\_nodes:vars]  node\_state='iso\_migrate'  node\_type=execution  [aap\_valid\_hosts]  qchvcfgp01.corpqc.hkjc.com |

1. Change the setup bundler owner to redhat user

|  |
| --- |
| # chown -R redhat: /opt/ansible/ansible-automation-platform-setup-bundle-2.2.2-1 |

1. Change to user “Redhat”

|  |
| --- |
| # su - redhat |

1. Run Ansible Tower Installation.

|  |
| --- |
| $ ANSIBLE\_SUDO=True /opt/ansible/ansible-automation-platform-setup-bundle-2.2.2-1/setup.sh |