Automatically upgrade the node image

- Overview
- Problem
- Solution
- Pre-requisites
- Procedure
- Steps to create Change Request
  - Create Change Request
- Check if your node pool is on the latest node image
- Upgrade node images with node surge
- Gitlab CI to Automatically update node / nodepool
  - Create CI Pipeline
  - Create Scheduled Build in Gitlab
- Related articles

Overview

Platform	**Azure**
Owner of this SOP:	Fully Managed POD-B
Cloud Services	AKS
Author	@ Nilesh Kumar

Problem

- The node needs to be kept up to date with newest OS and runtime updates.

> *Note: Linux node images are updated weekly, and Windows node images updated monthly*

Solution

Cloud Services operators will collect all the requirements and update the associated record as required. In most cases, the record will be an Change Task (which may be assigned or manually created).

Pre-requisites

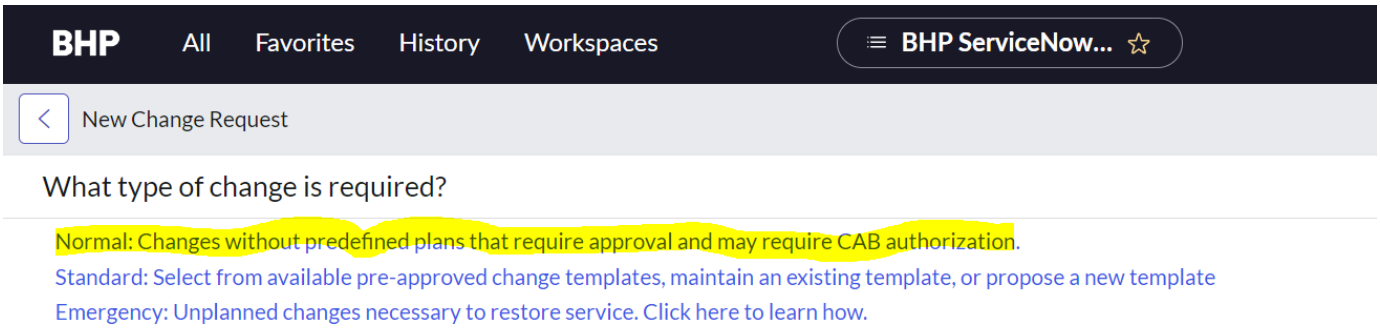
- The operator needs to have access to the **Cloud Factory** ServiceNow Dashboard. If this access is missing raise a request for Access to ServiceNow and request the group “Cloud Factory - Dashboard Access”.
- Must be a member of the “Cloud Svcs- CloudOne” assignment group to work on ServiceNow tickets. As well access to the required report in this SOP. Use the same Access to ServiceNow request to gain this group if needed.
- The operator needs to be part of **OperationsCloudOneAccess** role in AWS & **CloudOne Operator** role in Azure . If this is missing raise a Manage Cloud Access request to get this custom role.

Procedure

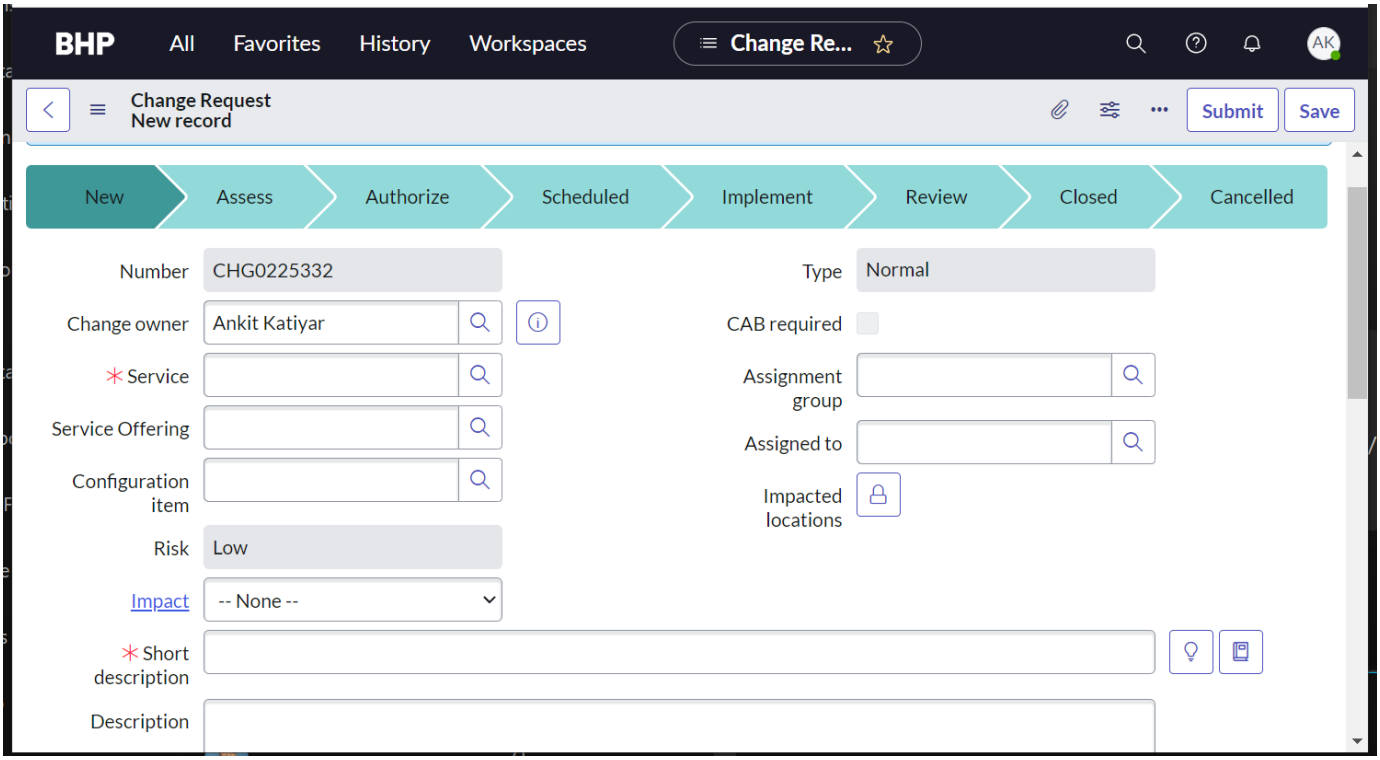
Steps to create Change Request

Create Change Request

On *SNOW* Tab, click *All* and choose *create new* under *change*. Then click on Normal: Changes without predefined plans that require approval and may require CAB authorization.



After this fill below form



Check if your node pool is on the latest node image

You can see what is the latest node image version available for your node pool with the following command:

Azure CLI

```
az aks nodepool get-upgrades \
    --nodepool-name mynodepool \
    --cluster-name myAKSCluster \
    --resource-group myResourceGroup
```

In the output you can see the `latestNodeImageVersion` like on the example below:

Output

```
{
  "id": "/subscriptions/XXXX-XXX-XXX-XXX-XXXXX/resourcegroups
/myResourceGroup/providers/Microsoft.ContainerService/managedClusters
/myAKSCluster/agentPools/nodepool1/upgradeProfiles/default",
  "kubernetesVersion": "1.17.11",
  "latestNodeImageVersion": "AKSUbuntu-1604-2020.10.28",
  "name": "default",
  "osType": "Linux",
  "resourceGroup": "myResourceGroup",
  "type": "Microsoft.ContainerService/managedClusters/agentPools
/upgradeProfiles",
  "upgrades": null
}
```

Upgrade node images with node surge

To speed up the node image upgrade process, you can upgrade your node images using a customizable node surge value. By default, AKS uses one additional node to configure upgrades.

If you'd like to increase the speed of upgrades, use the `--max-surge` value to configure the number of nodes to be used for upgrades so they complete faster. To learn more about the trade-offs of various `--max-surge` settings, see Customize node surge upgrade.

The following command sets the max surge value for performing a node image upgrade:

Azure CLI

```
az aks nodepool update \
    --resource-group myResourceGroup \
    --cluster-name myAKSCluster \
    --name mynodepool \
    --max-surge 33% \
    --no-wait
```

Gitlab CI to Automatically update node / nodepool

Create CI Pipeline

In the repo create the following file **.gitlab-ci.yml**

```
default:
  tags:
    - shared-azure-gitlab-runner-shell

stages:
  - updateNodes

before_script:
  - chmod -R +x ./scripts/*.sh

update_job:
  stage: updateNodes
  script:
    - ./scripts/env.sh
    - ./scripts/fetch_secrets.sh
    -  az aks upgrade -g $resourceGroupName -n $aksClusterName --node-
image-only --yes
```

The above code snippet is just a sample of the command you would execute to trigger the node image update. The flag `--node-image-only` is very important to note as this will update the node image but leave the Kubernetes version unchanged.

> Note
>
> To upgrade a single node pool instead of all node pools on the cluster, add the `--name` parameter to the `az aks nodepool upgrade` command to specify the node pool name. For example:

Azure CLI

```
az aks nodepool upgrade -g {resourceGroupName} --cluster-name
{aksClusterName} --name {{nodePoolName}} --node-image-only
```

The `env.sh` can be used to set environment variables or you can use the variables section in the **gitlab-ci.yml**

```
variables:
  resourceGroupName: test-rg
```

The `fetch_secrets.sh` is a script you can use to grab the necessary credentials for your environment.

Create Scheduled Build in Gitlab

To add a pipeline schedule:

1. On the top bar, select **Main menu > Projects** and find your project.
2. On the left sidebar, select **CI/CD > Schedules**.
3. Select **New schedule** and fill in the form.
   - **Interval Pattern**: Select one of the preconfigured intervals, or enter a custom interval in cron notation. You can use any cron value, but scheduled pipelines cannot run more frequently than the instance's maximum scheduled pipeline frequency.
   - **Target branch or tag**: Select the branch or tag for the pipeline.

- **Variables**: Add any number of CI/CD variables to the schedule. These variables are available only when the scheduled pipeline runs, and not in any other pipeline run.

Related articles

- Upgrade an Azure Kubernetes Service (AKS) cluster - Azure Kubernetes Service
- https://github.com/Azure/AKS/releases
- https://docs.gitlab.com/ee/ci/pipelines/schedules.html