Support For Retrying BaseOs Upgrade

Introduction

Sometimes, a BaseOs upgrade may fail because of transient error conditions. In general EVE provides for eventual consistency, where it will retry operations after a failure (in case the failure condition has gone away). However, a BaseOS update with associated reboot is quite disruptive and going in a loop repeating this even more so. Hence it makes sense to require some user intervention before a failed BaseOs Upgrade is retried.

Currently, Device Config API doesn't have a retry mechanism. Controller has to first remove the baseos configuration, wait for the device to sync-up, then reconfigure the BaseOs again. This is not very userfriendly.

This document describes the support to retry a failed BaseOs upgrade.

Proposed Solution

Introduce a new command "baseos_upgrade_retry" for devices.

EVE API

diff --git a/api/proto/config/devconfig.proto b/api/proto/config/devconfig.proto

index c58376ab7..d21531336 100644

--- a/api/proto/config/devconfig.proto

+++ b/api/proto/config/devconfig.proto

@@ -83,6 +83,19 @@ message EdgeDevConfig {

   // if we set new epoch, EVE sends all info messages to controller

   // it captures when a new controller takes over and needs all the info be resent

   int64 controller_epoch = 25;

+

+ // Retry the BaseOs upgrade for the configured image ONLY if the image

+ // upgrade has failed.

+ // 1) If the currently configured image is in FAILED state in the other

+ // partition, retry the image upgrade.

+ // 2) If the currently configured image is already active and

+ // fully installed (PartitionState = UPDATED), Do nothing. Just update the

+ // baseos_upgrade_counter in Info message.

+ // 3) If the currently configured image is same as active image, but status is NOT

+ // yet UPDATED, or if the upgrade to the currently configured image is in progress,

+ // wait till the upgrade concludes (Success / Error+rollback) - then trigger the

+ // retry.

+ DeviceOpsCmd baseos_upgrade = 26;

 }

diff --git a/api/proto/info/info.proto b/api/proto/info/info.proto

index cdf33db2f..06d0addf4 100644

--- a/api/proto/info/info.proto

+++ b/api/proto/info/info.proto

@@ -347,6 +347,14 @@ message ZInfoDevice {

```
   // Information about hardware capabilities

   Capabilities capabilities = 42;

+

+  // BaseOsUpgrade Retry Counter. This must be updated only when:

+  // 1) if the configured BaseOs partition is set to UPDATED, mirror

+  // the current value of baseOs_upgrade.counter

+  // 2) At the start if a BaseOs upgrade (either from a partition in error state

+  // or from UPDATED state of another version), copy over current d

+  // deviceConfig.baseOs_upgrade_counter

+  uint32 baseOs_upgrade_counter = 43;

 }
```

Note: Even in case of No-Op for upgrade_retry, the device sends an Info message to the controller to update its baseos_upgrade.counter.

EVE Support

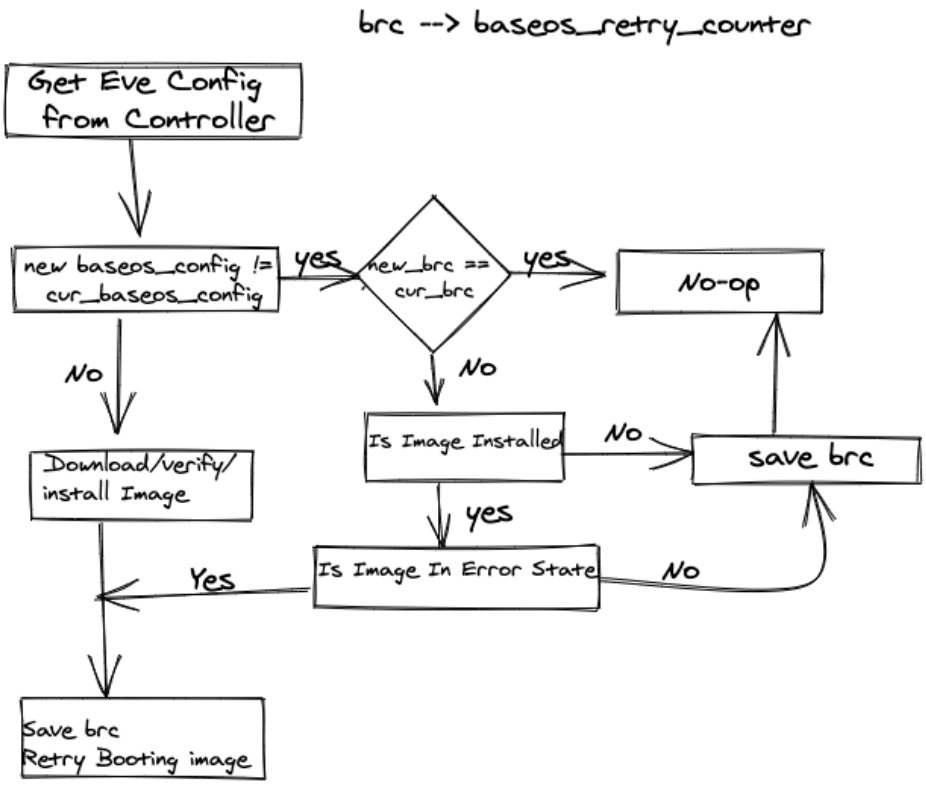
1. Eve Needs to store the status of each partition (Success / Failure) - This is done currently
2. If the baseos_upgrade.counter != local_baseos_upgrade_counter:
    a. Initiate Upgrade State Machine.
    b. Let the State Machine determine if an upgrade is needed or not
        i. For Example, the configured image is already the active image
        ii. An upgrade is already in progress.
    c. BaseOs Upgrade Statemachine update the local_baseos_upgrade_counter at the end of the StateMachine sequence.

Another way to think about this - the end result must be:

1. If Configured Active Image == Currently running image - do nothing. Just update local_baseos_upgrade_counter
2. If Upgrade currently in progress:
    a. If Target image == Currently configured image - wait for upgrade to complete and update local_baseos_upgrade_counter
    b. If Target Image != Currently Configured Image - Abort upgrade if possible. Restart Upgrade process to upgrade to Currently Configured Image.
3. If Currently Configured Image in Error state - restart the upgrade process.

Note: Currently - Eve automatically retries Download / verify / install errors

Control Flow:



Some Scenarios for Implementation

1. Configured EVE image in Error State, Fall Back image Currently Active.

This is the regular scenario for the feature:

1. Device Currently running 6.1.0 (Partition IMGA) (baseos_upgrade.counter = 1)
2. Controller configured 6.4.0 as the active image. (baseos_upgrade.counter = 1)
3. Eve receives new config, downloads and verifies the Image, and starts upgrading to 6.4.0 (IMGB)
4. Even boots up 6.4.0 and enters the Testing Phase.
5. Testing for 6.4.0 Fails and Eve Falls back to running 6.1.0
6. User triggers retry (baseos_upgrade.counter = 2)
7. As indicated in Control Flow section, this triggers booting into 6.4.0 again

2. BaseOs Using Volumes

BaseOs config still only has only Active image name. When Volume support is added, BaseOs Config can actually be simplified even more.

Currently, BaseOsConfig is an array (We limit it to 2 entries). Instead, with volume support, this can be limited to just a reference to a volume. the volume and their details are specified separately. BaseOsConfig will just point to the volume reference for the Active Baseos Version.

BaseOs Using Volumes with Current Partition Scheme:

Currently, Eve has the concept of partitions. Currently, it supports two partitions:

1. Currently running Image Partition (Lets say - PART-USED)
2. Previous Image booted Partition (PART-UNUSED)

Any new image installed in PART-UNUSED. It will also carry if that image was successful or errored. This is all is needed to support this feature.

In the case where Controller supports multiple volumes, and hence multiple partitions, each partition needs to maintain the state of the last Image result - SUCCESS / ERROR / UNKNOWN

This is used to answer the question "is_image_in_error_state()"