Agent - "ChangeFreeze" state

Status: In Progress

Sponsor User: *Mathis Moder*

Date of Submission: 07 Mar 2023

Submitted by: John Walicki

Affiliation(s): IBM

<Please fill out the above fields, and the Overview, Design and User Experience sections below for an initial review of the proposed feature.>

Scope and Signoff: *(to be filled out by Chair)*

Overview

For specific edge-nodes (e.g. moving cars or other critical equipment), a new Safety Critical "Change Freeze" mode should be introduced. The Agent will continue to run any active agreements / workloads but will not download/start new services/cancel existing services

Goal: prevent workloads from changing while the edge node is in a "safety critical state".

Design

The changefreeze state will be initiated with a call to the agent's /node/configstate api. If agent has one of the following conditions, it cannot enter a changefreeze state and the api will respond with a rejection.

- node is not currently in "configured" state
- node has an nmp in the initiated state
- waiting for a particular proposal (received an agreement cancel for policy upgrade/downgrade or userinput change)
- node has an unfinalized agreement
- mms model download in progress

In this case, it is the responsibility of the entity calling the api to retry the refused call after a period of time.

The ability to reject the state change allows the agent to ensure it's agreements are in the "correct" state before freezing. The 4th bullet above assigns a new meaning to the TerminatedReason field that is stored in all cancelled agreements. To motivate this change, consider a node running a service when a newer version of the service is added to the deployment policy. The agbot will cancel the current agreement. If at this point, the changefreeze state is enabled, the node will not get the proposal for the upgraded service and will be frozen in an incorrect state. Instead, when the node api call is made to enable changefreeze, the node should look at recently cancelled agreements and see that there is an agreement cancelled for an upgrade and no new proposal has been received for this policy yet. Therefore the node should reject the changefreeze state as not node is likely not in the intended state. An agreement is defined as recently cancelled here if its AgreementProtocolTerminatedTime is after the changes worker's lastHeartbeat.

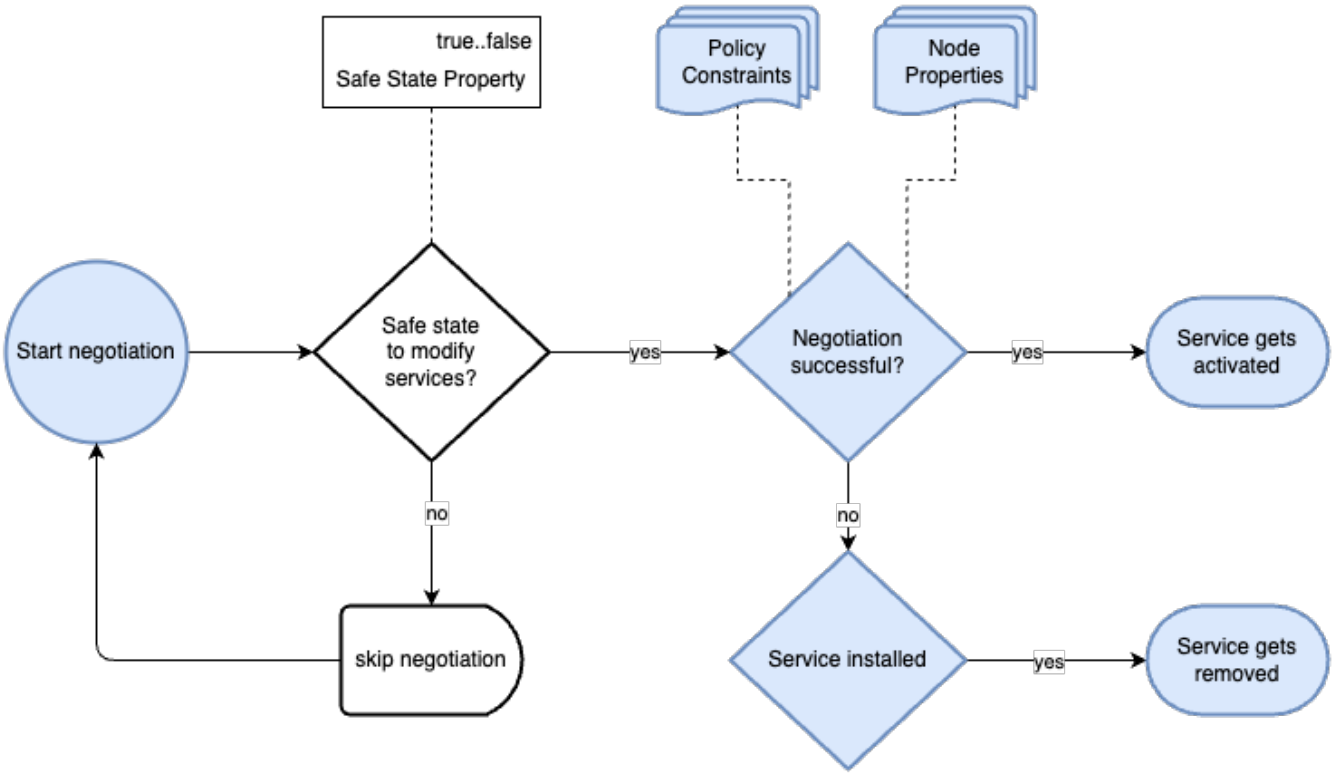
If the node determines that it can enter the changefreeze state, the agent will update its resource in the exchange to reflect the changefreeze state.

Once the agent enters the changefreeze state the agent's behavior will change in the following ways:

- set maximum changes from exchange to 0 (heartbeat but don't poll for changes)
- do not give the cron job any updates to perform
- do not allow any agent upgrades to proceed
- attempt to restart services if containers are missing (instead of cancelling the agreement)

The edge sync service on the node will continue to run, but communication with CSS will be suspended.

To end the changefreeze state the agent's /node/configstate api is called to set the state back to "configured". After this occurs, the agent may have missed changes or updates to its agreements, since relevant changes may have expired and been dropped from the exchange table the agent needs to run its exchange sync protocol. The agent should also send AgreementVerifications for each of its agreements. Given the lifespan of messages in the exchange, the agreementverification message needs to be modified to include the time of last update for secrets and policies. If the last update time in the agent's agreement is before the agbot's, then the agbot will resend the missed updates.



Possibly send heartbeats but not accept node property updates or changes

Possibly allow geofencing information updates? Where the edge node is located might be important to know. Aha "The car is on the driveway, geofenced at home" is an important clue that might allow the agent to trigger changes to workloads. If the car is at the supermarket, not a good idea.

Governance should restart the agreement, if it dies unexpectedly - tricky?

node health state ?

HA node groups need to skip over nodes that are in ChangeFreeze state. This is orthogonal to the reason for a HA group. Unsupported configuration.

Let the external change "The car is in park and the GPS knows that the car is "home" - Call the API to change out of ChangeFreeze state".

The agent never decides for itself that it out of ChangeFreeze state

Build a "Agent Config State" API

If a secret changes, the agbot sends a message of a change, if the agent doesn't see or handle that message, what happens? Max? Would the agreement get cancelled if the agent doesn't reply?

MMS handling of agents in ChangeFreeze status -

ESS should also go into ChangeFreeze state as well. It should not look for model updates while the edge node is in changefreeze state.

Node Management- behavior?

User Experience

As a node owner, I want to want to "freeze" the services of the node until i decide they can be changed again / after a defined timeout expires.

As a service deployer, I want to have feedback about this "frozen" state of the node.

As an admin, I want to be able to unfreeze the node remotely via cli.

Command Line Interface

<Describe any changes to the hzn CLI, including before and after command examples for clarity. Include which users will use the changed CLI. This section should flow very naturally from the User Experience section.>

hzn node freeze - enable the changefreeze state

hzn node unfreeze - disable the changefreeze state

External Components

<Describe any new or changed interactions with components that are not the agent or the management hub.>

Affected Components

<List all of the internal components (agent, MMS, Exchange, etc) which need to be updated to support the proposed feature. Include a link to the github epic for this feature (and the epic should contain the github issues for each component).>

AgBot

Security

<Describe any related security aspects of the solution. Think about security of components interacting with each other, users interacting with the system, components interacting with external systems, permissions of users or components>

APIs

<Describe and new/changed/deprecated APIs, including before and after snippets for clarity. Include which components or users will use the APIs.>

When the agent comes out of ChangeFreeze state, the agbot should numerate through a list of its BasicAgreementVerification() s.

Build, Install, Packaging

<Describe any changes to the way any component of the system is built (e.g. agent packages, containers, etc), installed (operators, manual install, batch install, SDO), configured, and deployed (consider the hub and edge nodes).>

Documentation Notes

<Describe the aspects of documentation that will be new/changed/updated. Be sure to indicate if this is new or changed doc, the impacted artifacts (e.g. technical doc, website, etc) and links to the related doc issue(s) in github.>

Test

<Summarize new automated tests that need to be added in support of this feature, and describe any special test requirements that you can foresee.>