HBASE-24620: All new ClusterManager which submits destructive actions through ZK and it's agent which picks up those commands and executes them blindly.

This is the design doc for HBASE-24620 i.e. a new clustermanager which does not require passwordless-ssh to execute destructive actions used in Chaos Monkey.

Context:

Current Chaos Monkey has basically one ClusterManager which is default HBaseClusterManager which uses passwordless-ssh to execute destructive actions, and that sometimes is a blocker if someone wants to integrate chaos as a tool with their service which uses chaos.
Apart from HBaseClusterManager there are two other ClusterManagers

- RestApiClusterManager : A ClusterManager which controls Cloudera Manager clusters via RestApis.
- CoprocClusterManager : A recently added Coproc based ClusterManager which only takes actions against Master and RegionServer processes and those coprocs are itself running within those processes.

Goal:

Our Goal is to add a ClusterManager which can execute destructive actions without passwordless-ssh.

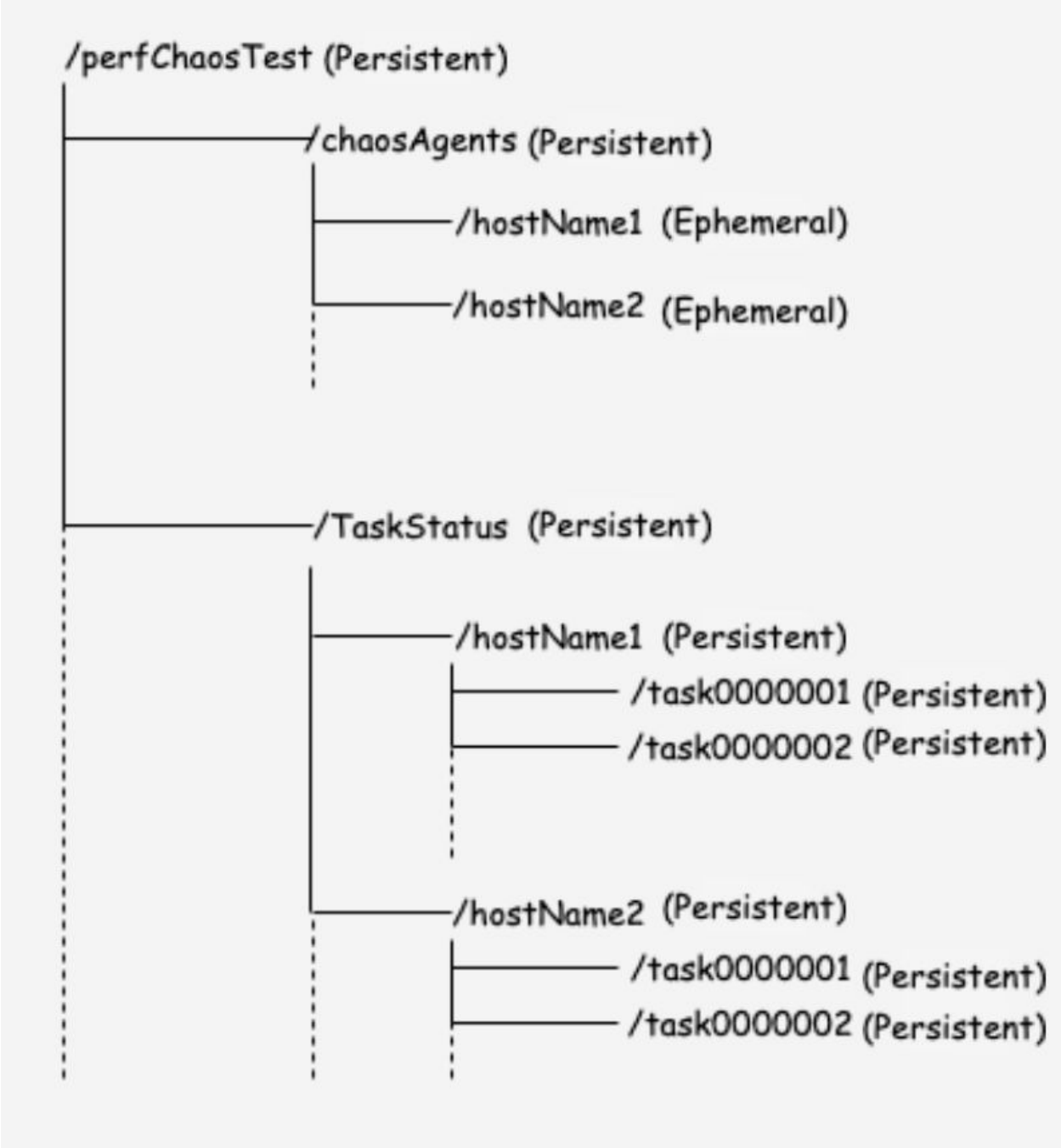
Proposed Design:

Higher Level :

On a Higher Level this new ClusterManager will be using an agent-based architecture, where instead of directly executing commands, it'll be submitting tasks to the ZooKeeper as a ZNode and an Agent will be running on every node which will pick up that submitted task and executes it blindly.
Here we'll be introducing a new small component chaos agent whose function is simply to execute commands which have been assigned to them.

ZNode Directory Structure:

Below is the directory structure that'll be used by ClusterManager :



- Here root directory is /perfChaosTest which is a persistent ZNode.
- It has two children → /chaosAgents and /TaskStatus both Persistent

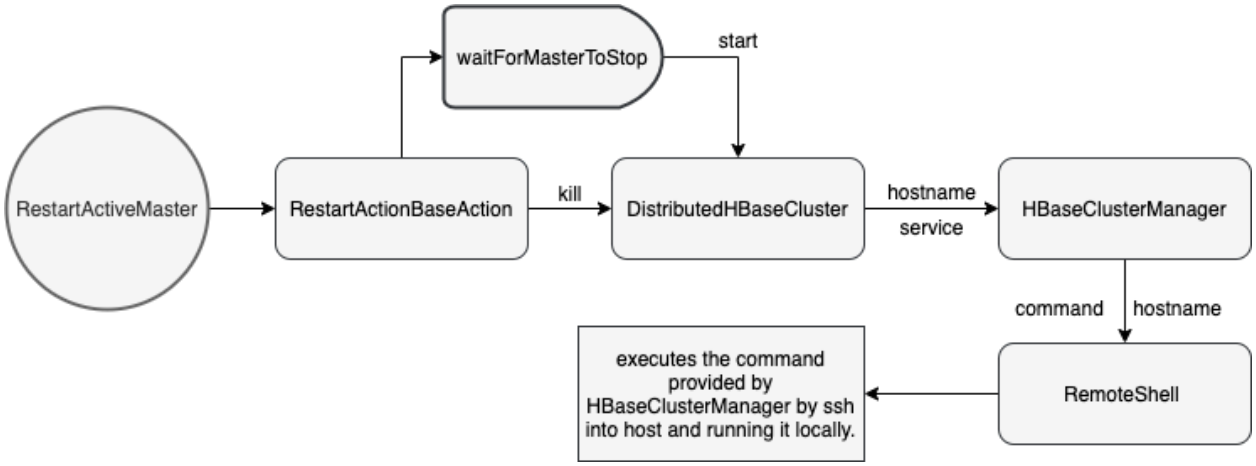
- /chaosAgents is the root ZNode for all the chaosagent's registration, each node when came up will create an ephemeral Znode under this node as a registration process so that the client can tell if the agent is up for command execution or not.
- /TaskStatus is the root ZNode for all the tasks that's been submitted for execution, it contains a list of hostnames (chaos agents) as children and tasks will be submitted as child of the hostname on which the command has to be executed.

Lower Level:

Changes that we are introducing are as follow:

1) Introducing a new ClusterManager (ZNodeClusterManager) and a Client to handle all the ZK interactions (ChaosZKClient).

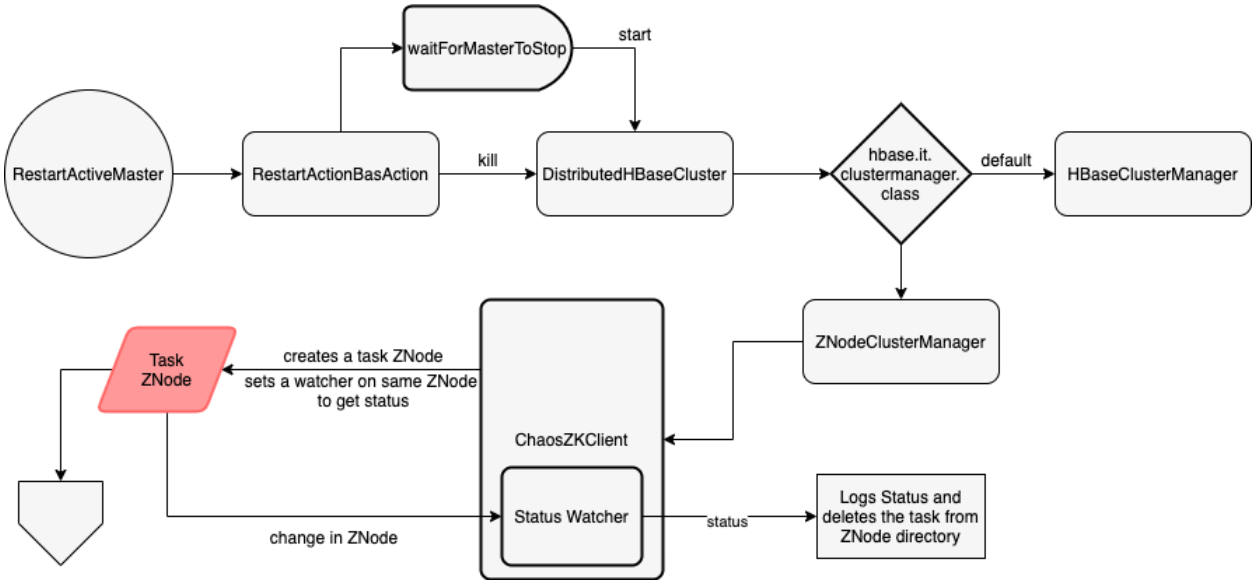
Flow Diagram : Before any change →



Flow Diagram : After change →

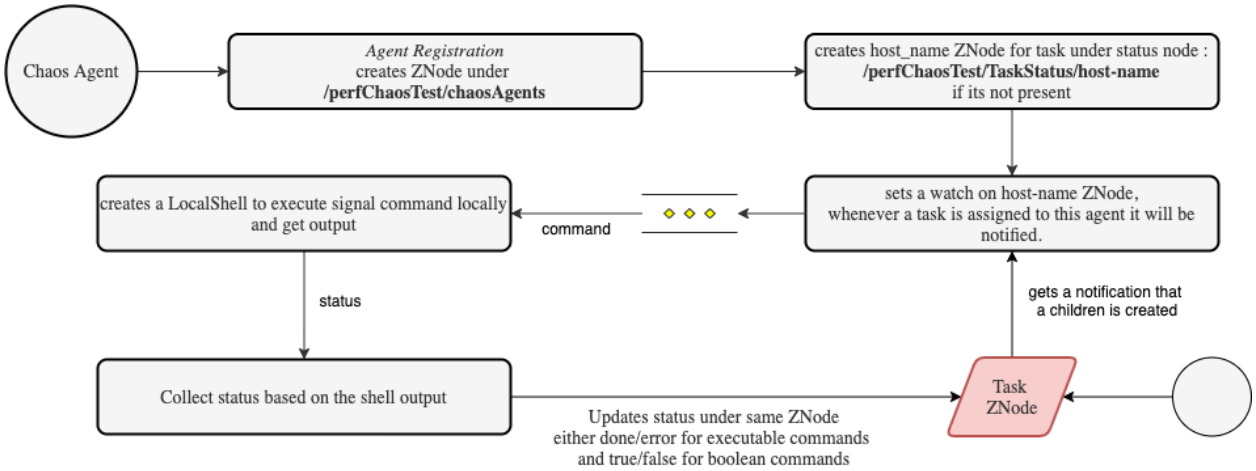
So to use our new ClusterManager a conf is already present we just have to change hbase.it.clustermanager.class and it'll redirect the destructive actions to ZNodeClusterManager, which will talk to ChaosZKClient and creates a ZNode under the hostname where command has to be executed and sets a watcher on same ZNode for status to be updated. Whenever data is changed from command to status Watcher will get notified and Logs the status.

NOTE: Here the same ZNode is being used for transmitting commands from ClusterManager to Agents and for transmitting status (error/done/true/false) from Agents to ClusterManager.



2) Chaos Agents which will be running on every node as a process waiting and watching for any task to be assigned.

WorkFlow of Chaos Agent:



Changes:

Not a single change in current code, just Introducing new Classes

- ZNodeClusterManager
- ChaosZKClient
- One small package for Chaos Agent