Installer - Feature #29820

Add ability to have a "hook" that is dedicated to defining new parameters to the installer

05/14/2020 03:52 PM - Eric Helms

Status:	New		
Priority:	Normal		
Assignee:			
Category:			
Target version:			
Difficulty:		**Fixed in Releases:**	
Triaged:	No	**Found in Releases:**	
Bugzilla link:		**Red Hat JIRA:**	
Pull request:			

Description

Today, we use the boot hook to create new parameters to the installer. Given that creating a new parameter and then executing some chunk of code based on that parameters value is a first class citizen of the installer, could we build support into Kafo for this?

For example:

https://github.com/theforeman/foreman-installer/blob/develop/katello/hooks/boot/11-clear_pulp_data_hook.rb defines the option for clearing pulp data
https://github.com/theforeman/foreman-installer/blob/develop/katello/hooks/pre/11-clear_pulp_data_feature.rb defines the action to take if that option is set

One benefit of this could be finding a way to build in :noop support directly instead of having to remember to check for it.

Related issues:

Related to Installer - Tracker #29746: Merge hooks/ and katello/hooks into a ...	**Closed**

History

#1 - 05/14/2020 03:54 PM - Eric Helms

- Related to Tracker #29746: Merge hooks/ and katello/hooks into a single directory structure added

#2 - 05/14/2020 04:40 PM - Ewoud Kohl van Wijngaarden

I don't see how. A parameter can be used in any stage, but clamp must be configured before execution starts. You can't parse the CLI and later on decide an additional parameter should be added.

In theory you can add options like "do this in the pre hook phase", but that just kills all the visibility of the pre phase. The order will also become complicated.

https://github.com/theforeman/kafo#hooks defines the stages and this just works well for me.

As for noop, we can implement a noop? helper as a shortcut for app_value(:noop).