

## Hammer CLI - Feature #945

### foreman CLI

05/26/2011 03:37 AM - Ohad Levy

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b> Brian Gupta	
<b>Category:</b>	
<b>Target version:</b>	
<b>Difficulty:</b>	<b>Fixed in Releases:</b>
<b>Triaged:</b>	<b>Found in Releases:</b>
<b>Bugzilla link:</b>	<b>In Kanboard:</b>
<b>Pull request:</b>	<b>Red Hat JIRA:</b>
<b>Team Backlog:</b>	
<b>Description</b>	
All of the operations over the UI could be done via the API as well.	
Foreman should have an command line interface, where basic functionaily could be performed:	
list hosts, inventory(facts) and their report status. create / update / delete hosts ...	
an example for api usage could be found at the <a href="#">API</a> page, or an example scripts :	
<ul style="list-style-type: none"><li>• Using curl - <a href="https://gist.github.com/937501">https://gist.github.com/937501</a></li><li>• Using search - <a href="https://gist.github.com/919578">https://gist.github.com/919578</a></li><li>• Creating a new host - <a href="https://gist.github.com/913305">https://gist.github.com/913305</a></li></ul>	
<b>Related issues:</b>	
Related to Foreman - Refactor #1173: remove legacy query interface	<b>Closed</b>
Related to Foreman - Feature #1222: API enhancement	<b>Closed</b> <b>10/12/2011</b>
Related to Foreman - Bug #1245: API bug with search a dot in front of domain ...	<b>Feedback</b> <b>10/21/2011</b>

### Associated revisions

#### Revision 71ce3e9b - 10/30/2011 04:13 AM - Brian Gupta

fixes #945 - Added basic Command Line Interface for foreman

Signed-off-by: Brian Gupta <[brian.gupta@brandor.com](mailto:brian.gupta@brandor.com)>

### History

#### #1 - 05/26/2011 03:39 AM - Ohad Levy

Foreman has a legacy query script<sup>1</sup> which would probably needs to be part of the CLI as well, once exists, we could remove the old API.

[1] - [source:extras/query](#)

#### #2 - 05/26/2011 03:43 AM - Brian Gupta

- Assignee set to Brian Gupta

#### #3 - 05/27/2011 04:01 AM - Brian Gupta

Ok so I have a script here: <https://gist.github.com/994819>

Note the "collections" array isn't used yet, but I was thinking to turn it into a hash with {"collectionname == path", "description"}, which would really enable me to remove a lot of the repetition in the get.opts and the get\_list calls.

I started thinking a bit further, and was wondering if it would make sense to add another "collection" to the API, that is called from "/API/collections" that returns the hash I am describing?

Thoughts?

P.S. - When I say collections, I am talking about a REST collection that when you use a GET method, it returns a list of "elements" e.g. GET /hostgroups returns a list of hostgroups

#### #4 - 05/27/2011 04:30 AM - Brian Gupta

I maybe overthinking things, but I could see the /API paradygm extended further:

GET /API returning a list of what you can get from it. e.g.

/API/collections - returns a list of REST collections the API supports

/API/versions - returns a list of API versions

you could even do nested stuff like:

/API/collections/hostlist - return the URLs and methods that hostlist supports

with API versioning we could do GET /v1.0/API/collections

#### #5 - 05/27/2011 05:06 AM - Brian Gupta

Ohad suggested we build a "lib of objects" like has been down for smart-proxy: [https://github.com/ohadlevy/foreman/blob/develop/lib/proxy\\_api.rb](https://github.com/ohadlevy/foreman/blob/develop/lib/proxy_api.rb)

#### #6 - 05/28/2011 03:53 AM - Brian Gupta

Updated Ohad's optimization, by converting the Hashes to arrays, and updating code to address arrays vs hashes. <https://gist.github.com/996702>

#### #7 - 09/20/2011 10:19 AM - Ohad Levy

and now with hostgroup as defaults attributes its as simple as <https://gist.github.com/1229194>

#### #8 - 10/07/2011 09:53 PM - Brian Gupta

- % Done changed from 0 to 10

I have not yet refactored to accomodate "lib of objects", but at this point I have a basic script that can query most collections and filter on a few. (It's now useful to us, and I'm hopin it might be useful to others.)

It has nothing hardcoded that is site specific, and takes arguments for URL, username and password. (I have tested with SSL and basicauth enabled. It should work without auth and http, but I don't have a test server setup anymore without them).

Error handling for options parsing is nonexistent. I need to figure out how to handle whether or not to print in JSON or not.

Here is the script, comments welcome: <https://gist.github.com/1271734>

#### #9 - 10/08/2011 10:04 PM - Brian Gupta

Updated again. Fixed some issues working with basic auth, and refactored the get/searchcollections, to be more elegant, in addition, I now delete the "options hash" elements as the instance variables are instantiated (and all class variables were converted to instance variables). In addition the output now defaults to plain text and one needs to use --json to get the results in "pretty" JSON.

Please check it out, and give me your thoughts (in this issue): <https://gist.github.com/1271734>

Also as an aside, I still think we need a fully functionally command line tool that can do everything including provisioning. (Similar to Chef's "knife" utility). I think a good name would be "hammer".

One open question, is a name for this read-only script. I am using foremancli as a working name, but that implies a bit more functionality than I am providing. As it stands this script only lists collections and searches collections. ("hammer" will be the fully functional tool that can read and set any data, as well as trigger provisioning.)

#### #10 - 10/08/2011 10:10 PM - Brian Gupta

Ohad Levy wrote:

and now with hostgroup as defaults attributes its as simple as <https://gist.github.com/1229194>

Could you please elaborate? We don't use hostgroups, so I am not sure, what this does, or how to incorporate this functionality into the script.

#### #11 - 10/09/2011 04:28 AM - Ohad Levy

Brian Gupta wrote:

Ohad Levy wrote:

and now with hostgroup as defaults attributes its as simple as <https://gist.github.com/1229194>

Could you please elaborate? We don't use hostgroups, so I am not sure, what this does, or how to incorporate this functionality into the script.

This simply creates a new host and provision it.  
assuming that the host group contain all important attributes (e.g. vm properties(mem, vcpu etc) network, os attributes etc).

#### #12 - 10/09/2011 05:03 PM - Brian Gupta

Ohad starting coding a framework for high level objects (based on active resource) here:  
<https://github.com/ohadlevy/stacker/blob/master/lib/foreman.rb>

#### #13 - 10/09/2011 08:38 PM - Brian Gupta

Ok, now has another option to support any custom nested requests like /hosts.fqdn/facts

```
--custom COLLECTION          Custom COLLECTION string, see: http://theforeman.org/projects/foreman/wiki/API  
https://gist.github.com/1271734
```

#### #14 - 10/22/2011 05:25 PM - Brian Gupta

Added support for three optional ENVIRONMENT VARIABLES:

FOREMAN_SERVER	Foreman Server URL
FOREMAN_USER	Foreman user
FOREMAN_PASSWORD	Foreman password

CLI options take precedence over ENV\_VARS.

<https://gist.github.com/1271734>

#### #15 - 10/23/2011 11:16 AM - Brian Gupta

Basic --status (Foreman server status) has been added: <https://gist.github.com/1271734> Currently spits out JSON status report e.g.:

```
{  
  "result": "ok",  
  "version": "0.3",  
  "db_duration_ms": "1",  
  "status": 200  
}
```

Not sure what people would like to see here.

#### #16 - 10/30/2011 04:14 AM - Ohad Levy

- Category set to 44
- Target version set to 0.4

#### #17 - 10/30/2011 04:16 AM - Brian Gupta

- Status changed from New to Closed
- % Done changed from 10 to 100

Applied in changeset commit:"71ce3e9bbefebff5dc18a26f9d6b64fd50c2df0".

#### #18 - 10/16/2013 04:11 PM - Dominic Cleal

- Project changed from Foreman to Hammer CLI
- Category deleted (44)
- Target version deleted (0.4)