

CLI tools for Foreman

Martin Bačovský

mbacovsk@redhat.com

Agenda

- overview of the tools
- live examples for individual options
- troubleshooting
- new features

Foreman API

- general purpose API
- versioned - v2
- extensible from Foreman plugins
- same features as in UI

Foreman API

- general purpose API
- versioned - v2
- extensible from Foreman plugins
- same features as in UI

- + fast
- + well documented
- + wide range of tools/libs

Foreman API

- general purpose API
- versioned - v2
- extensible from Foreman plugins
- same features as in UI

+ fast

+ well documented

+ wide range of tools/libs

— usability from shell (curl options, parsing JSON output)

— general complexity (routes, inconsistencies across plugins)

Documentation

- documented with Apipie
- <https://theforeman.org/api/1.23/index.html>
- <https://<your foreman instance>/apidoc/>
- structured docs in JSON
- localized

Example

```
In [203]: ! curl -k -u admin:changeme -X GET \
          -H "Accept:application/json" \
          https://centos7-luna-devel.pichi.example.com/api/v2/architectures \
          | python -m json.tool
```

```
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                               Dload  Upload   Total   Spent    Left   Speed
100    354      0    354      0      0   1616      0  --:--:--  --:--:--  --:--:--   1616
{
  "total": 2,
  "subtotal": 2,
  "page": 1,
  "per_page": 20,
  "search": null,
  "sort": {
    "by": null,
    "order": null
  },
  "results": [
    {
      "created_at": "2018-10-08 17:16:11 UTC",
      "updated_at": "2018-10-08 17:16:11 UTC",
      "name": "x86_64",
      "id": 1
    }
  ]
}
```

Apipie-bindings

- ruby wrapper around Foreman API
- simple static library
- the API description is loaded from the server
- API introspection

Apipie-bindings

- ruby wrapper around Foreman API
 - simple static library
 - the API description is loaded from the server
 - API introspection
-
- + easy to use from Ruby
 - + well tested
 - + it finds the best route based on params
 - + works with any ApiPie documented API
 - + logging

Introspection demo

```
In [204]: %%writefile foreman_api.rb
require 'apipie-bindings'
require 'awesome_print'

def api
  ApipieBindings::API.new(
    {
      :uri => 'https://centos7-luna-devel.pichi.example.com/',
      :api_version => 2,
      :username => 'admin',
      :password => 'changeme'
    },
    :verify_ssl => false
  )
end
```

Overwriting foreman_api.rb

```
In [205]: %%ruby
require './foreman_api.rb'

ap api
```

```
#<ApiPieBindings::API:0x000000000261d5d0 @uri="https://centos7-luna-devel.pichi.example.com/", @api_version=2, @language=nil, @apidoc_cache_dir="/home/mbacovsk/.cache/apipie_bindings/https__centos7-luna-devel.pichi.example.com_/v2", @apidoc_cache_name="13c937025f2570a497cca8a51a31a5cc33d0a7a5", @apidoc_authenticated=true, @follow_redirects=:default, @dry_run=false, @aggressive_cache_checking=false, @fake_responses={}, @logger=#<Logger:0x0000000002eabdc8 @level=3, @progname=nil, @default_formatter=#<Logger::Formatter:0x0000000002eabd50 @datetime_format=nil>, @formatter=nil, @logdev=#<Logger::LogDevice:0x0000000002eabd00 @shift_period_suffix=nil, @shift_size=nil, @shift_age=nil, @filename=nil, @dev=#<IO:<STDERR>>, @mon_owner=nil, @mon_count=0, @mon_mutex=#<Thread::Mutex:0x0000000002eabc88>>>, @authenticator=#<ApiPieBindings::Authenticators::BasicAuth:0x0000000002eaace8 @user="admin", @password="changeme">, @resource_config={:timeout=>nil, :headers=>{:content_type=>"application/json", :accept=>"application/json;version=2"}, :verify_ssl=>false}, @config={:uri=>"https://centos7-luna-devel.pichi.example.com/", :api_version=>2, :username=>"admin", :password=>"changeme"}>
```

API has resources

```
In [206]: %%ruby
require './foreman_api.rb'

ap api.resources[0,10]
```

```
[
  [0] <Resource :hosts>,
  [1] <Resource :interfaces>,
  [2] <Resource :host_subscriptions>,
  [3] <Resource :fact_values>,
  [4] <Resource :hostgroups>,
  [5] <Resource :smart_proxies>,
  [6] <Resource :architectures>,
  [7] <Resource :audits>,
  [8] <Resource :auth_sourceexternals>,
  [9] <Resource :auth_sourceinternals>
]
```

Resources have actions

```
In [207]: %%ruby
require './foreman_api.rb'

ap api.resource(:architectures).actions

[
  [0] <Action architectures:index>,
  [1] <Action architectures:show>,
  [2] <Action architectures:create>,
  [3] <Action architectures:update>,
  [4] <Action architectures:destroy>
]
```

Actions can be called

```
In [208]: %%ruby
require './foreman_api.rb'

ap api.resource(:architectures).action(:index).call

{
  "total" => 2,
  "subtotal" => 2,
  "page" => 1,
  "per_page" => 20,
  "search" => nil,
  "sort" => {
    "by" => nil,
    "order" => nil
  },
  "results" => [
    [0] {
      "created_at" => "2018-10-08 17:16:11 UTC",
      "updated_at" => "2018-10-08 17:16:11 UTC",
      "name" => "x86_64",
      "id" => 1
    },
    [1] {
      "created_at" => "2018-10-08 17:16:11 UTC",
      "updated_at" => "2019-11-04 11:40:41 UTC",
      "name" => "i486",
      "id" => 2
    }
  ]
}
```

The following notations are equivalent

- `api.resource(:architectures).action(:index).call`
- `api.resource(:architectures).call(:index)`
- `api.call(:architectures, :index)`

Actions also have params

```
In [209]: %%ruby
require './foreman_api.rb'

ap api.resource(:architectures).action(:update).params

[
  [0] <Param location_id (Numeric)>,
  [1] <Param organization_id (Numeric)>,
  [2] <Param *id (String)>,
  [3] <Param *architecture (Hash)>
]
```


Actions also have params

```
In [209]: %%ruby
require './foreman_api.rb'

ap api.resource(:architectures).action(:update).params

[
  [0] <Param location_id (Numeric)>,
  [1] <Param organization_id (Numeric)>,
  [2] <Param *id (String)>,
  [3] <Param *architecture (Hash)>
]
```

Params can be nested

```
In [210]: %%ruby
require './foreman_api.rb'

ap api.resource(:architectures).action(:update).params[3].params

[
  [0] <Param name (String)>,
  [1] <Param operating_system_ids (Array)>
]
```

```
In [212]: %%ruby
require './foreman_api.rb'

params = {
  :id => 2,
  :architecture => {
    :name => 'i586'
  }
}
ap api.call(:architectures, :update, params)

{
  "created_at" => "2018-10-08 17:16:11 UTC",
  "updated_at" => "2019-11-07 13:38:19 UTC",
  "name" => "i586",
  "id" => 2,
  "operatingsystems" => [
    [0] {
      "id" => 3,
      "name" => "TestOS1",
      "title" => "TestOS 1"
    },
    [1] {
      "id" => 2,
      "name" => "Windows7EM2019-01",
      "title" => "Windows 7 EN (2019-01)"
    }
  ],
  "images" => []
}
```

Alternatives for Python

- Aypie (<https://github.com/Apipie/aypie/tree/master/aypie>)
- Nailgun (<https://nailgun.readthedocs.io/en/latest/#>)

Hammer CLI

- command-line client for Foreman
- same functionality as UI

Hammer CLI

- command-line client for Foreman
 - same functionality as UI
-
- ✚ can combine multiple API calls in a single command
 - ✚ extensible with plugins
 - ✚ processing of input and output
 - ✚ has help
 - ✚ configurable (auth methods, different servers, ...)

Hammer CLI

- command-line client for Foreman
- same functionality as UI
- + can combine multiple API calls in a single command
- + extensible with plugins
- + processing of input and output
- + has help
- + configurable (auth methods, different servers, ...)
- slow to load

Structure of commands

hammer [global opts] <resource>* <command> [command opts]

- global opts (-d, --username, --output, ...)
 - hammer --help
- commands (list, info, create, update, delete, ...)
 - hammer <resource> --help
- command opts (--id, ...)
 - hammer <resource> <command> --help

In [213]: `! hammer architecture --help`

Usage:

hammer architecture [OPTIONS] SUBCOMMAND [ARG] ...

Parameters:

SUBCOMMAND	Subcommand
[ARG] ...	Subcommand arguments

Subcommands:

add-operatingsystem	Associate an operating system
create	Create an architecture
delete	Delete an architecture
info	Show an architecture
list	List all architectures
remove-operatingsystem	Disassociate an operating system
update	Update an architecture

Options:

-h, --help	Print help
------------	------------

Configuration

```
tree ~/.hammer
├── certs
├── cli_config.yml
├── cli.modules.d
│   ├── foreman_remote_execution.yml
│   ├── foreman.yml
│   └── katello.yml
├── defaults.yml
├── log
│   └── hammer.log
└── sessions
```

Auth methods

- basic auth (username and password)
- sessions/ no sessions
- auth with user access token
- OpenID Connect (Keycloak)
- client cert

Example access token usage

```
$ hammer user access-token create --user admin --name demo
```

Personal access token [demo] created:
jmz0hwHZy9P2fSmRms3i6A

```
In [214]: ! hammer --username admin --password jmz0hwHZy9P2fSmRms3i6A \
          user access-token info --name demo --user admin
```

```
Id:          12
Name:        demo
Active:      yes
Expires at:
Created at:  2019/11/06 11:52:45
Last used at:
```

Output tuning

Global options

- `--output [base, table, csv, yaml, json, silent]`
- `--no-headers`
- `--show-ids`

Command options

- `--fields`
- `--page`
- `--per-page`

Predefined field sets for host list

Fields	ALL	DEFAULT	THIN
Id	x	x	x
Name	x	x	x
Operating System	x	x	
IP	x	x	
MAC	x	x	
Global Status	x	x	
Organization	x		
Location	x		
Content View	x	x	
Bugfix	x		
.....			
Enhancement	x		

```
In [215]: ! hammer --output yaml host list --fields THIN,organization,ip --per-page 2
```

```
---  
- Id: 4  
  Name: centos7.pichi.example.com  
  Ip: 192.168.121.172  
  Organization:  
    Name: Default Organization  
    Id: 1  
- Id: 6  
  Name: drew-whyte.pichi.example.com  
  Ip: 192.168.121.10  
  Organization:  
    Name: Default Organization  
    Id: 1
```

Search and order lists

- `--search` - scoped search syntax as in the UI
- `--order` - 'id DESC' (may be inconsistent in plugins)

```
In [216]: ! hammer host list --search "ip=192.168.121.10" --fields name,ip
```

-----	-----
NAME	IP
-----	-----
drew-whyte.pichi.example.com	192.168.121.10
-----	-----

Taxonomies

- each API endpoint accepts organization_id and location_id
- it sets the scope in which the request is evaluated
- it is similar to Org/Loc selector in UI

Troubleshooting

- `--debug` global option
- check if error messages comes from hammer or API
- check for lost API error messages
- check server logs (debug verbosity)
- see what UI sends if it works there

```
In [ ]: ! hammer -d os list
```

```
In [219]: %%capture request_id
! hammer -d os list 2>&1 | grep x_request_id \
  | sed 's/\([^"]*"([a-z0-9]*\)-.*\)/\2/' | tr -d '[:cntrl:]'
```

```
In [219]: %%capture request_id
! hammer -d os list 2>&1 | grep x_request_id \
| sed 's/\([^"]*"([a-z0-9]*\)-.*\)/\2/' | tr -d '[:cntrl:]'
```

```
In [220]: print(request_id)
```

5e456046

```
In [219]: %%capture request_id
! hammer -d os list 2>&1 | grep x_request_id \
| sed 's/\([^"]*"\"([a-z0-9]*\)-.*\)/\2/' | tr -d '[:cntrl:]'
```

```
In [220]: print(request_id)
```

5e456046

```
In [221]: ! ssh vagrant@centos7-luna-devel.pichi.example.com \
"grep {request_id} foreman/log/development.log"
```

```
2019-11-07T13:49:55 [I|app|5e456046] Started GET "/api/operatingsystems?page=1&per_page=40" for 192.168.121.1 at 2019
-11-07 13:49:55 +0000
2019-11-07T13:49:55 [I|app|5e456046] Processing by Api::V2::OperatingsystemsController#index as JSON
2019-11-07T13:49:55 [I|app|5e456046] Parameters: {"page"=>"1", "per_page"=>"40", "apiv"=>"v2", "operatingsystem"=>
{}}
2019-11-07T13:49:55 [D|app|5e456046] Authenticated user admin against INTERNAL authentication source
2019-11-07T13:49:55 [I|app|5e456046] Authorized user admin(Admin User)
2019-11-07T13:49:55 [D|tax|5e456046] Current location set to none
2019-11-07T13:49:55 [D|tax|5e456046] Current organization set to none
2019-11-07T13:49:55 [D|tax|5e456046] Current location set to none
2019-11-07T13:49:55 [D|tax|5e456046] Current organization set to none
2019-11-07T13:49:55 [I|app|5e456046] Rendering api/v2/operatingsystems/index.json.rabl within api/v2/layouts/index_
layout
2019-11-07T13:49:55 [I|app|5e456046] Rendered api/v2/operatingsystems/index.json.rabl within api/v2/layouts/index_l
ayout (39.7ms)
2019-11-07T13:49:55 [D|app|5e456046] Body: {
2019-11-07T13:49:55 [I|app|5e456046] Completed 200 OK in 605ms (Views: 42.5ms | ActiveRecord: 337.8ms)
```

Troubleshooting

- ask on community.theforeman.org
- report issue in projects.theforeman.org
- attach debug output

GraphQL

- powerfull query language
- fresh addition to Foreman
- lots of potential

GraphQL

- powerfull query language
 - fresh addition to Foreman
 - lots of potential
-
- + comes with handy console in UI
 - + introspection
 - + flexibility in queries

GraphQL

- powerfull query language
 - fresh addition to Foreman
 - lots of potential
-
- + comes with handy console in UI
 - + introspection
 - + flexibility in queries
-
- no mutations yet
 - not much feedback on performance

GraphiQL - interactive UI console

- <https://centos7-luna-devel.pichi.example.com> > Toolbox > GraphiQL

```
In [222]: %%writefile graphql_params.json
{
  "query": "query {
    users(first:2) {
      totalCount
      nodes {
        login
        mail
      }
    }
  }"
```

Overwriting graphql_params.json

```
In [223]: ! curl -k -u admin:changeme -X POST -H "Content-Type:application/json" \
-H "Accept:application/json" --data @graphql_params.json \
https://centos7-luna-devel.pichi.example.com/api/graphql | python -m json.tool
```

```
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
100    335      0   126    100     209     118     197   0:00:01   0:00:01   -:--:--    316
{
  "data": {
    "users": {
      "totalCount": 6,
      "nodes": [
        {
          "login": "techuser",
          "mail": ""
        },
        {
          "login": "admin",
          "mail": "root@pichi.example.com"
        }
      ]
    }
  }
}
```

Report templates

- ERB templates
- direct access to the data model
- macros for easies report formating
- see the Foreman Demo #68 for details on usage
- predefined reports are easy to clone and extend

Report templates

- ERB templates
 - direct access to the data model
 - macros for easies report formating
 - see the Foreman Demo #68 for details on usage
 - predefined reports are easy to clone and extend
-
- + complex reports possible
 - + output format according to your needs
 - + instant and scheduled execution

Report templates

- ERB templates
- direct access to the data model
- macros for easies report formating
- see the Foreman Demo #68 for details on usage
- predefined reports are easy to clone and extend

- + complex reports possible
- + output format according to your needs
- + instant and scheduled execution

- runs in protected environemnt
- have to create the template first

Available report templates

In [224]: `! hammer report-template list`

```
-----|-----  
ID      | NAME  
-----|-----  
143     | Applicable errata  
157     | Applied Errata  
105     | Host statuses  
142     | Registered hosts  
144     | Subscriptions  
-----|-----
```

Report generation

```
In [225]: ! hammer report-template generate --name "Applicable errata" \
          --report-format yaml
```

```
---
- Host: centos7.pichi.example.com
  Operating System: Centos 7.5
  Environment: Development
  Erratum: RHEA-2012:0055
  Type: security
  Published: '2012-01-27'
  Applicable since: 2018-10-08 18:03:07 UTC
  Severity: ''
  Packages:
  - penguin
  - shark
  - walrus
  CVEs: []
  Reboot suggested: false
```

Foreman ansible modules

- perfect for Foreman setup with Ansible
- Foreman clones for testing
- fresh addition
- multiple alternatives
- <https://github.com/theforeman/foreman-ansible-modules>

Foreman maintain

- tool to make maintenance tasks easier
 - health checks
 - backup / restore
 - services
 - upgrades
 - ...
- current support for Foreman is not complete
- upgrades of Foreman and Debian support is WIP
- plans to wrap various maintenance scripts

Service status with foreman-maintain

```
In [226]: ! ssh 'root@centos7-luna-devel.pichi.example.com' \  
          "foreman-maintain service status -b"
```

Running Status Services

=====

Get status of applicable services:

Displaying the following service(s):

rh-mongodb34-mongod, postgresql, qdrouterd, qpidd, squid, pulp_celerybeat, pulp_resource_manager, pulp_streamer, pulp_workers, tomcat, httpd, puppetserver, foreman-proxy

\ displaying rh-mongodb34-mongod

[OK]

\ displaying postgresql

[OK]

\ displaying qdrouterd

[OK]

\ displaying qpidd

[OK]

\ displaying squid

[OK]

\ displaying pulp_celerybeat

[OK]

\ displaying pulp_resource_manager

[OK]

\ displaying pulp_streamer

[OK]

\ displaying pulp_workers

[OK]

\ displaying tomcat

[OK]

\ displaying httpd

[OK]

\ displaying puppetserver

[OK]

| displaying foreman-proxy

[OK]

| All services are running

[OK]

Questions?