



Pulp Containers and the Katello Registry

Ian Ballou
GitHub: [ianballou](#)
IRC: [iballou](#)



Foreman



Provisioning



Configuration



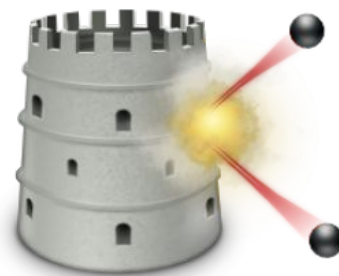
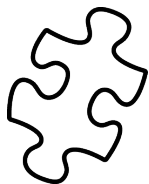
Monitoring



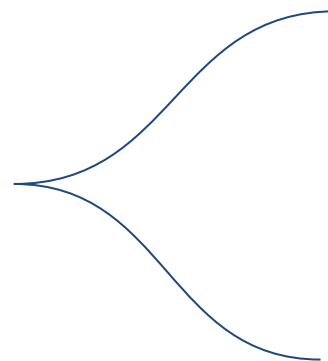
The Stack



Foreman



Katello



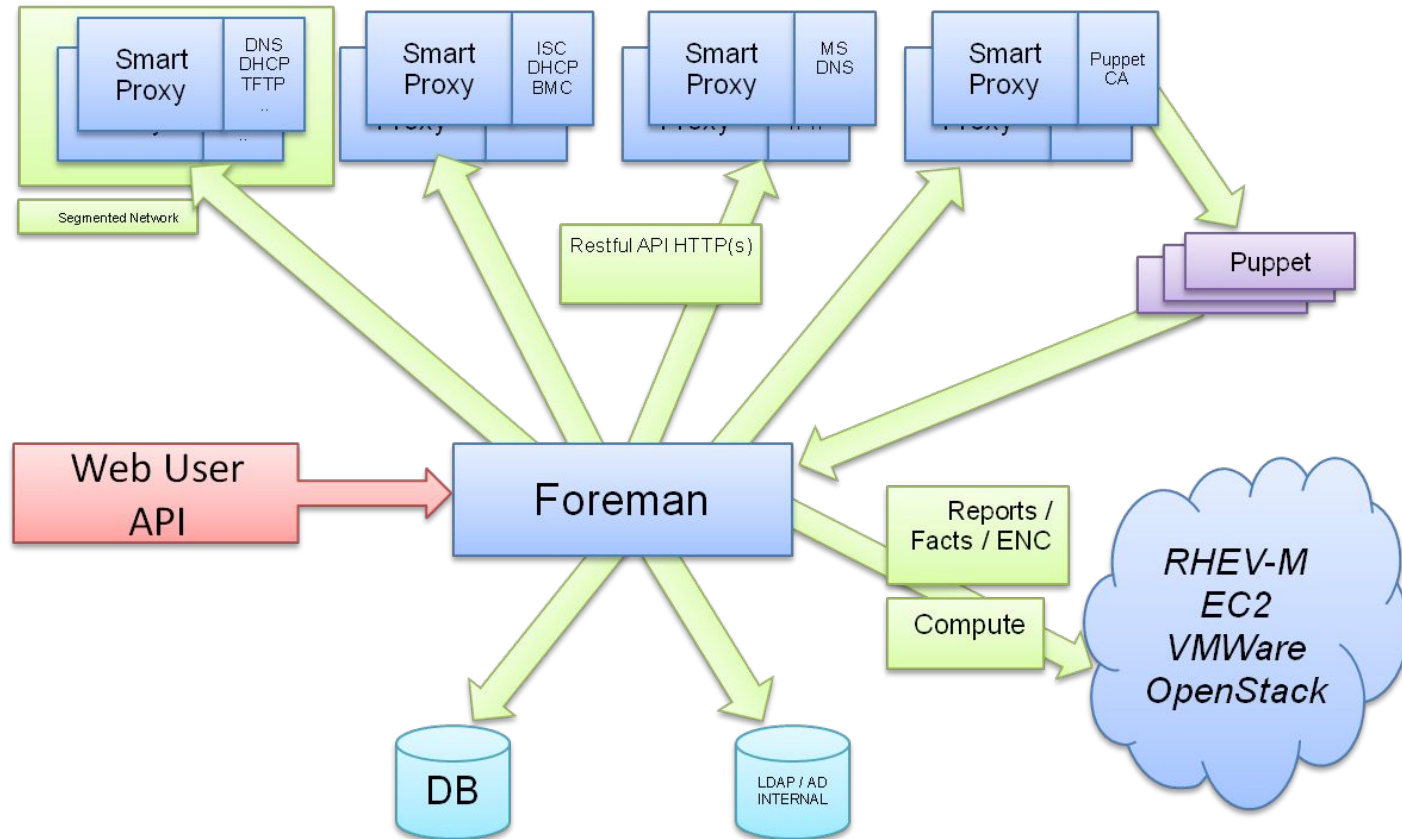
Candlepin



Pulp

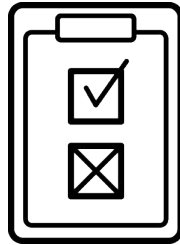


Foreman

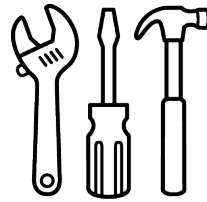


Katello

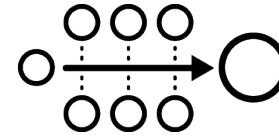
Content Management



Subscription
Management



Patch Management



Lifecycle
Management



Katello Container Registry



Registry: Katello vs Pulp functions

Katello

- Token authentication
- Catalog
- Search

Pulp

- Content distribution



Registry: Apache Setup

```
<Location "/pulpcore_registry/v2/">  
  RequestHeader unset REMOTE_USER  
  RequestHeader set REMOTE_USER "%{SSL_CLIENT_S_DN_CN}s"  
  env=SSL_CLIENT_S_DN_CN  
  ProxyPass unix:///run/pulpcore-api.sock|http://pulpcore-api/v2/  
  ProxyPassReverse unix:///run/pulpcore-api.sock|http://pulpcore-api/v2/  
</Location>
```

Pulp-Container's /v2 endpoint clashes with Katello's /v2 smart proxy endpoint



Example Interaction

```
podman login foreman.example.com
```

...

Started GET

```
"/v2/token?account=admin&service=centos8-katello-devel.cannolo.example.com"
```

Completed 200 OK With body:

```
{"token": "$2a$10$1b6453892473a467d0737uawQSD8Z  
6HKUQo.p3cKq3H5eWXFf2GUS", "expires_in": 594827,  
"issued_at": "2023-10-27T18:35:11+00:00", "expires_at": "2023-11-03T15:48:59+00:00"}
```



Example Interaction

```
podman search `hostname`/
```

```
...
```

```
<authentication token dance>
```

```
Started GET "/v2/_catalog"
```

```
Completed 200 OK
```

```
With body: {"repositories":["org-podman_hello"]}
```



Example Interaction

```
podman pull `hostname`/org-podman_hello
```

...

Started GET

`"/v2/org-podman_hello/manifests/latest"`

Parameters: `{"repository"=>"org-podman_hello",
"tag"=>"latest"}`

Sending GET request to Registry:

`/v2/org-podman_hello/manifests/latest`

Then the manifests by SHA...

Then the blobs by SHA...



The Future – Container Push

```
podman push 66e562cf5405 `hostname`/org/hello:v9
```

...

To Be Implemented !



Container Push Steps - Katello

1. podman push is run
2. Katello handles auth and redirects to Pulp
3. Pulp creates “push repository”
4. Katello copies contents to normal pulp repository
5. Katello creates a Katello product if necessary
6. Katello creates a Katello repository if necessary
7. Katello tells Pulp how to distribute the contents
8. User uses container repo in Katello like any other



Container Push Notes - Katello

- Product will be detected in the container name
 - If no product, create/use a “container push product”
- Katello repositories will be reused
 - Detected by name
- Smart proxies won't have push support to begin
 - Use smart proxy syncing instead
 - Auto-sync after container push perhaps...
 - Request routing from smart proxy to Katello?



Status

- Current status:
 - Upstream RFC out and without objections
 - <https://community.theforeman.org/t/rfc-container-registry-push/34478/>
 - Proof-of-concept branch for basic pushing
 - <https://github.com/ianballou/katello/tree/container-push-testing>
 - Designs made for implementation
- Ideal plan: container push developed for Katello 4.12
 - Katello 4.11 is currently wrapping up development





FOREMAN

Thanks!

<https://theforeman.org/>

<https://github.com/Katello/katello>

https://github.com/Katello/smart_proxy_container_gateway