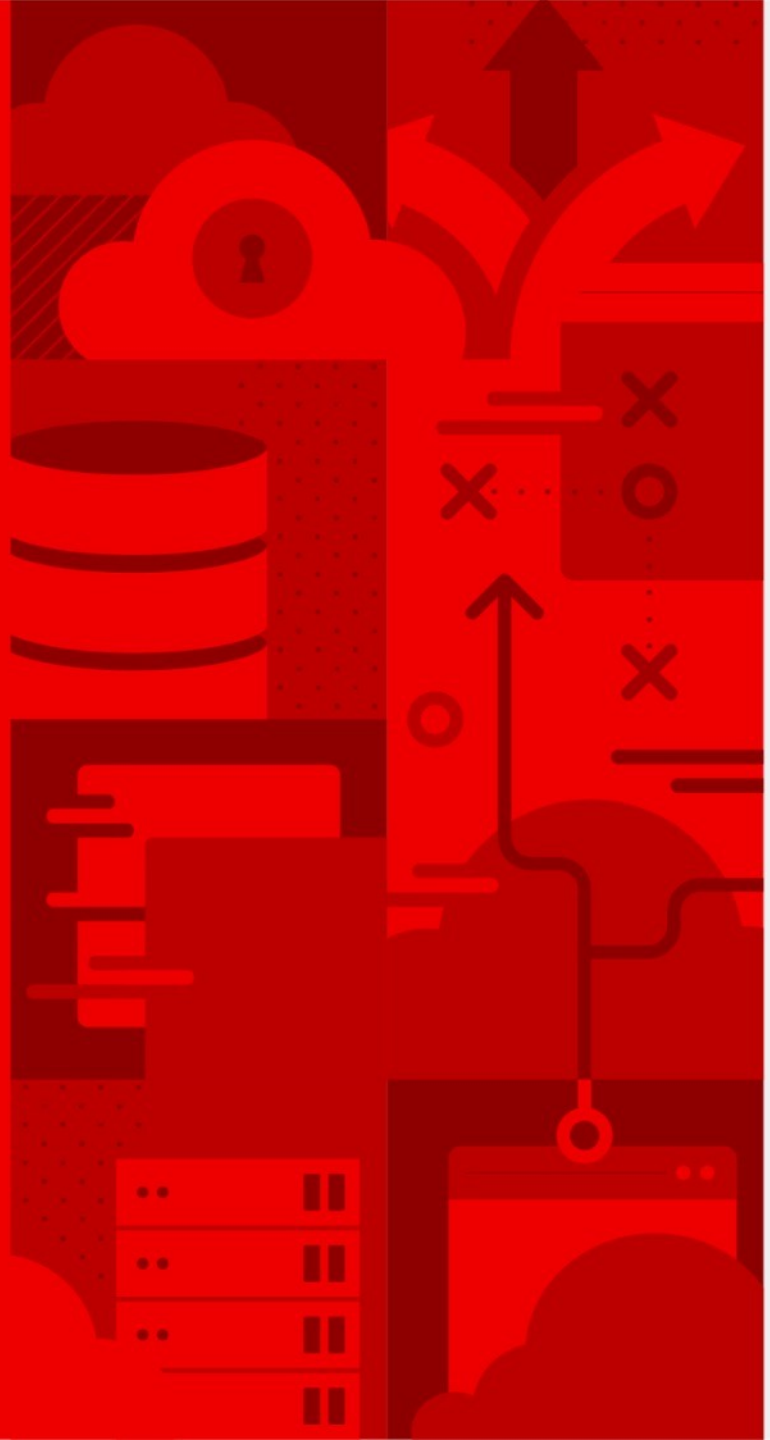


Foreman

Automating bare-metal provisioning of commodity hardware

Lukáš Zapletal
Principal Software Engineer
Satellite 6 Engineering, Red Hat Brno

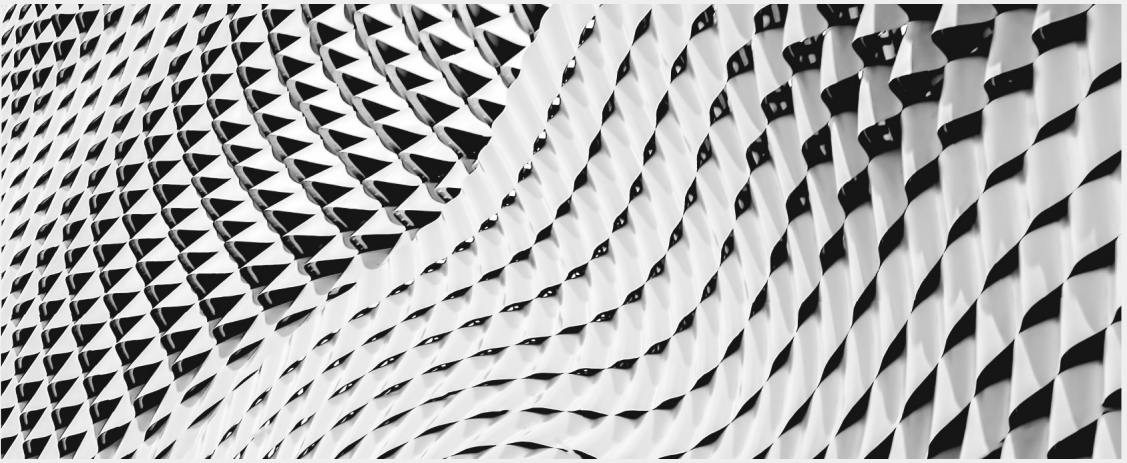
Red Hat

What we'll be discussing today

Foreman	Discovery	Post-discovery
Who?	Uh?	Meeeh.
Why?	Ah!	Wow?!
What?	Hmm.	Yeah!

Source:
http://www.theforeman.org

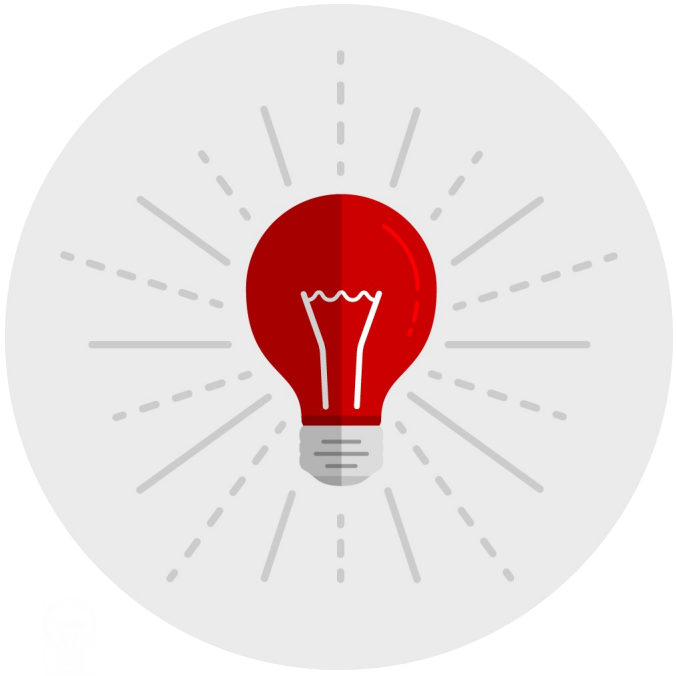
Foreman



Foreman is a complete lifecycle management tool for physical and virtual servers. We give system administrators the power to easily automate repetitive tasks, quickly deploy applications, and proactively manage servers, on-premise or in the cloud.

Red Hat

Foreman is good for:



Provisioning

Bare-metal, virtualization, cloud and managing of related resources (DHCP, DNS, TFTP, IdM)

Content management

Synchronization and managing of packages, puppet modules and images

Node bootstrapping

Getting Ansible/Puppet/xxx up and running

Inventory

Facts and reports from configuration management software or scanners (OpenSCAP) and external inventory system.

> Foreman is enterprise-grade application with powerful features like multi-tenancy, RBAC, LDAP/MSAD integration, IdM, REST API, CLI and localization (13 languages).

John Doe
CTO, Acme Unlimited

Foreman community

Foreman is 10 years old open-source project with about 250 unique contributors. Core team sponsored by several companies working 24 hours a day.

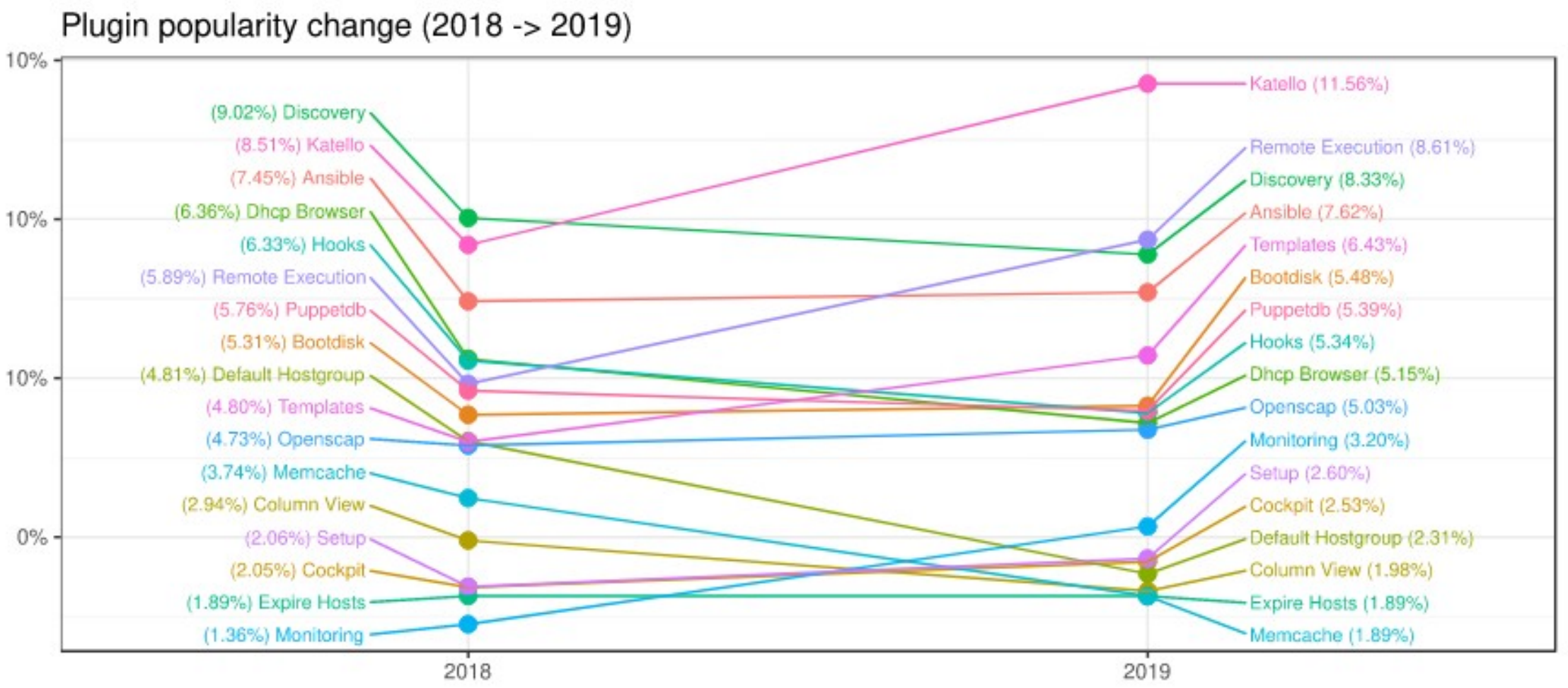
Foreman and Smart Proxy both have open core architecture. There are many plugins available.

Technology

Ruby on Rails application, Sinatra remote site application called “Smart Proxy” via HTTPS channel, runs on any UNIX, packages for CentOS and Debian.

Runs on PostgreSQL. Puppet-based installer.

Red Hat


Plugin popularity change (2018 -> 2019)
10%
10%
10%
0%
(9.02%) Discovery
(8.51%) Katello
(7.45%) Ansible
(6.36%) Dhcp Browser
(6.33%) Hooks
(5.89%) Remote Execution
(5.76%) Puppetdb
(5.31%) Bootdisk
(4.81%) Default Hostgroup
(4.80%) Templates
(4.73%) Openscap
(3.74%) Memcache
(2.94%) Column View
(2.06%) Setup
(2.05%) Cockpit
(1.89%) Expire Hosts
(1.36%) Monitoring
Katello (11.56%)
Remote Execution (8.61%)
Discovery (8.33%)
Ansible (7.62%)
Templates (6.43%)
Bootdisk (5.48%)
Puppetdb (5.39%)
Hooks (5.34%)
Dhcp Browser (5.15%)
Openscap (5.03%)
Monitoring (3.20%)
Setup (2.60%)
Cockpit (2.53%)
Default Hostgroup (2.31%)
Column View (1.98%)
Expire Hosts (1.89%)
Memcache (1.89%)
2018
2019

Provisioning workflows in Foreman

Bare-metal	Virtualization	Cloud
PXE / iPXE	Image-based	Image-based
Discovery	PXE / iPXE	
Bootdisk	Discovery	
Recipe disk	Bootdisk (AA)	

Source:
http://www.theforeman.org

Foreman Recipe Maker

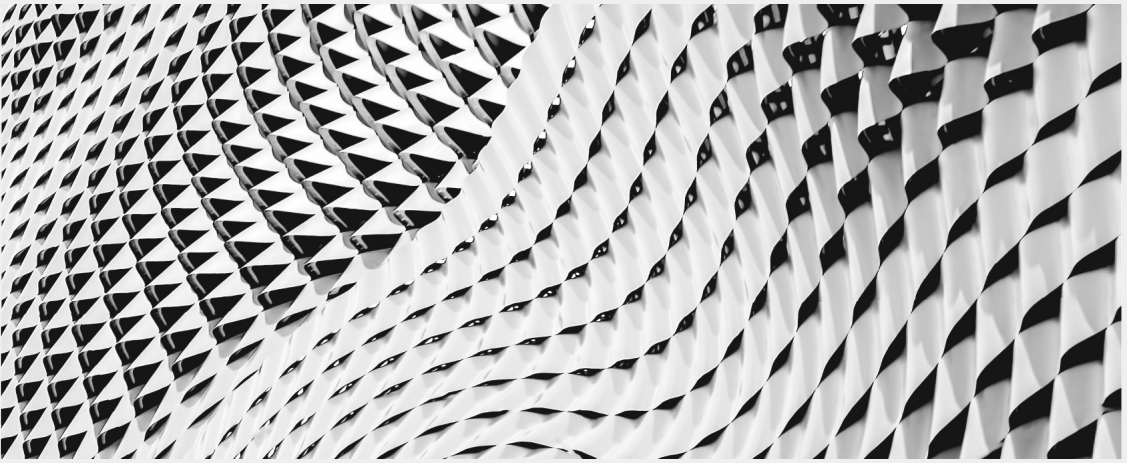
https://github.com/lzap/

foreman-recipe-maker

> Foreman is really good in commodity hardware provisioning, because our BMC integration is pretty poor.
>
> (We need help with this.)

Lukáš Zapletal
Foreman core team, bare-metal provisioning

Foreman Discovery



Automatic bare-metal discovery of unknown nodes on the provisioning network. This is often referenced as Metal-as-a-Service or simply MaaS. New nodes self-register into Foreman and upload facts collected by Facter. The registered nodes show up on Discovered Hosts page and provisioning can be initiated either manually (via UI/CLI or API) or automatically via predefined Discovery Rules.

Red Hat

Major limitations of Foreman Discovery

Network configuration

Not possible to change subnet or IP in semi or fully automated workflows. No IPAM integration (until 1.24) and no NIC reconfigurations.

Confusing and unreliable PXE-less

Interactive PXE-less workflow relies on unreliable kexec, also people are confused when to use bootdisk and when discovery.

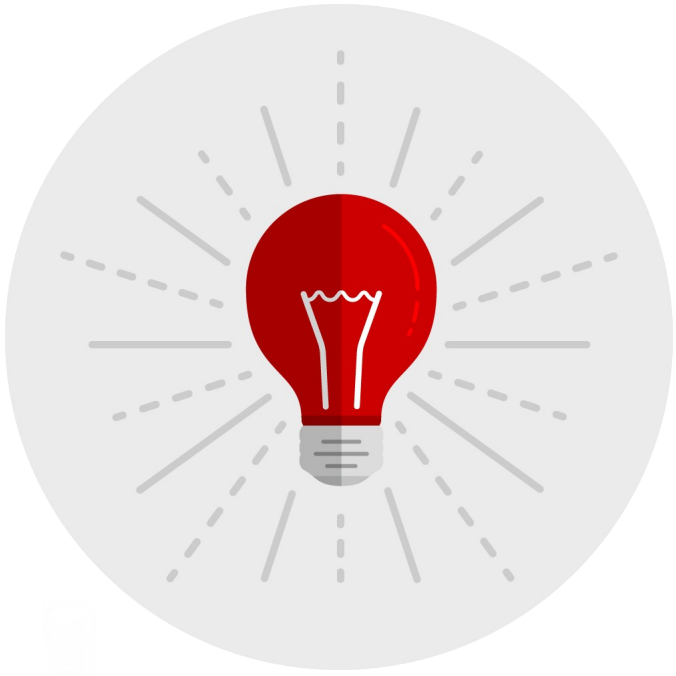
Not extensible

Fixed HTTPS API (“reboot” and “kexec”) is not extensible by users. Rebuilding from scratch is time consuming.

More info::
https://community.theforeman.org/t/improving-discovery-workflow/15315

Image: https://www.eglin.af.mil/News/Article/814477/40th-fts-expands-a-10-fuel-limitations-in-combat/

Proposal for Foreman Discovery NG



Initial discovery is the same

No major changes, discovery uses facter to report facts via Foreman HTTPS API to create a discovered host. New mechanism to download and verify server (CA) certificate – fingerprint check.

Remote Execution for flexibility

Discovered nodes are remote execution nodes – all other communication is performed via ReX: post discovery hook, provisionining hook. Smart-proxy on the discovered image is dropped completely,

Foreman templating engine leveraged

Templates to reboot node, kexec node, image-based or reboot-into-os-installer ships with Foreman. Users can customize them or write ERB to reconfigure hosts during provisioning (NICs, Domains, Subnets, IPAM).

More info::
https://community.theforeman.org/t/improving-discovery-workflow/15315

Secondary goals

Upgrade facter to facter 3.x or higher.

Upgrade to CentOS 8 LiveCD build tool (lorax).

Custom fact to report master-slave NIC relationships.

Red Hat

theforeman / **community-templates**

Unwatch 30 | Unstar 126 | Fork 174

Code | Issues 12 | Pull requests 25 | Actions | Security | Insights

A place to share templates for various OSes for Foreman's provisioning

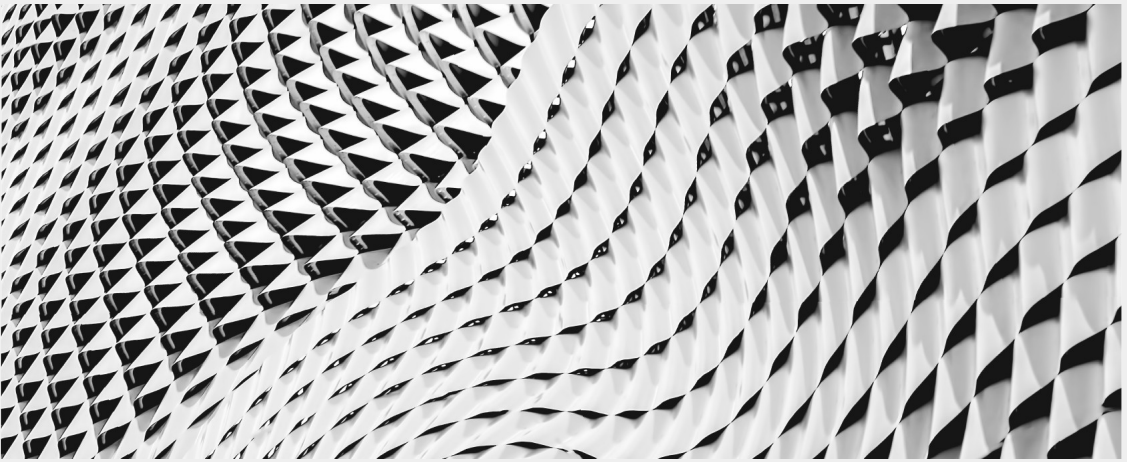
535 commits | **19** branches | **0** releases | **98** contributors

Branch: **develop** | **New pull request** | **Create new file** | **Upload files** | **Find file** | **Clone or download**

uvNikita and **mmoll** Fixes #28023 - fix rancheros PXE template ··· Latest commit `e1d3bd1` 3 days ago

job_templates	Place the check_update template to the right place (#629)	17 days ago
partition_tables_templates	Apply preseed grub-installer/bootdev in partition template	6 months ago
provisioning_templates	Fixes #28023 - fix rancheros PXE template	3 days ago
report_templates	Fixes #28020 - Ansible Inventory report template (#635)	3 days ago
test	fix minitest deprecations	8 days ago
.gitignore	Update pykickstart to the latest version (#485)	last year
.travis.yml	Update pykickstart to the latest version (#485)	last year
Gemfile	update and relax nokogiri version constraint	last year
README.md	Updated README with template render trick	10 days ago
Rakefile	Refs #2310,#3873 - test preseed configs	3 years ago

Post Foreman Discovery Era



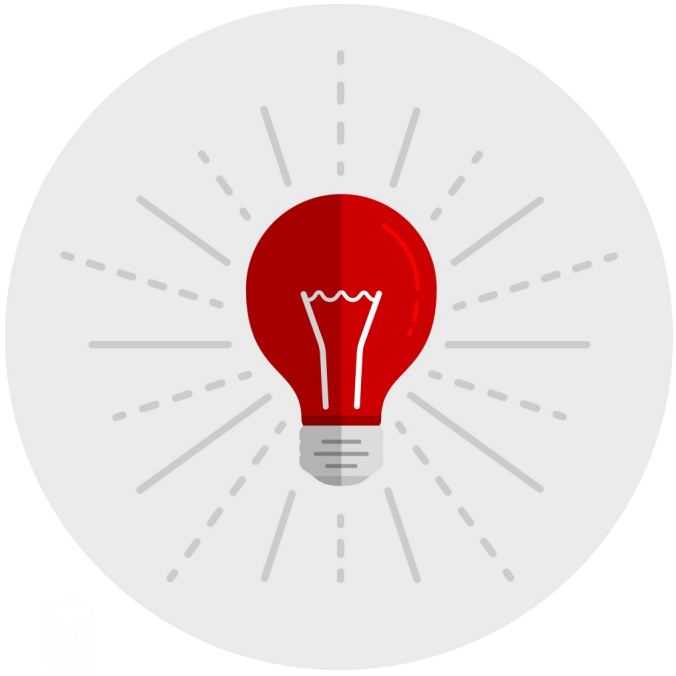
What if there was a way to make OS installers like Anaconda or Debian Installer to register systems in Foreman.

No discovery needed.

EOD (end-of-dream)

Red Hat

Proposal for OS Installation Wait Protocol



Send basic hardware facts in HTTP request

MAC addresses, CPUs, memory, serial number and vendor and optionally total storage space

Wait if HTTP response is 408 Request Timeout

Indefinitely (?) wait in a loop and keep trying every 1-10 minutes until the response is HTTP 200 OK. Code 404 is probably easier to implement in busybox curl/wget.

Transparent with minimum changes

Small patches into Anaconda, Debian Installer and Foreman, opt-in via kernel command line options. No regressions or changes in behavior for regular PXE workflows.

More info::
https://community.theforeman.org/t/improving-discovery-workflow/15315

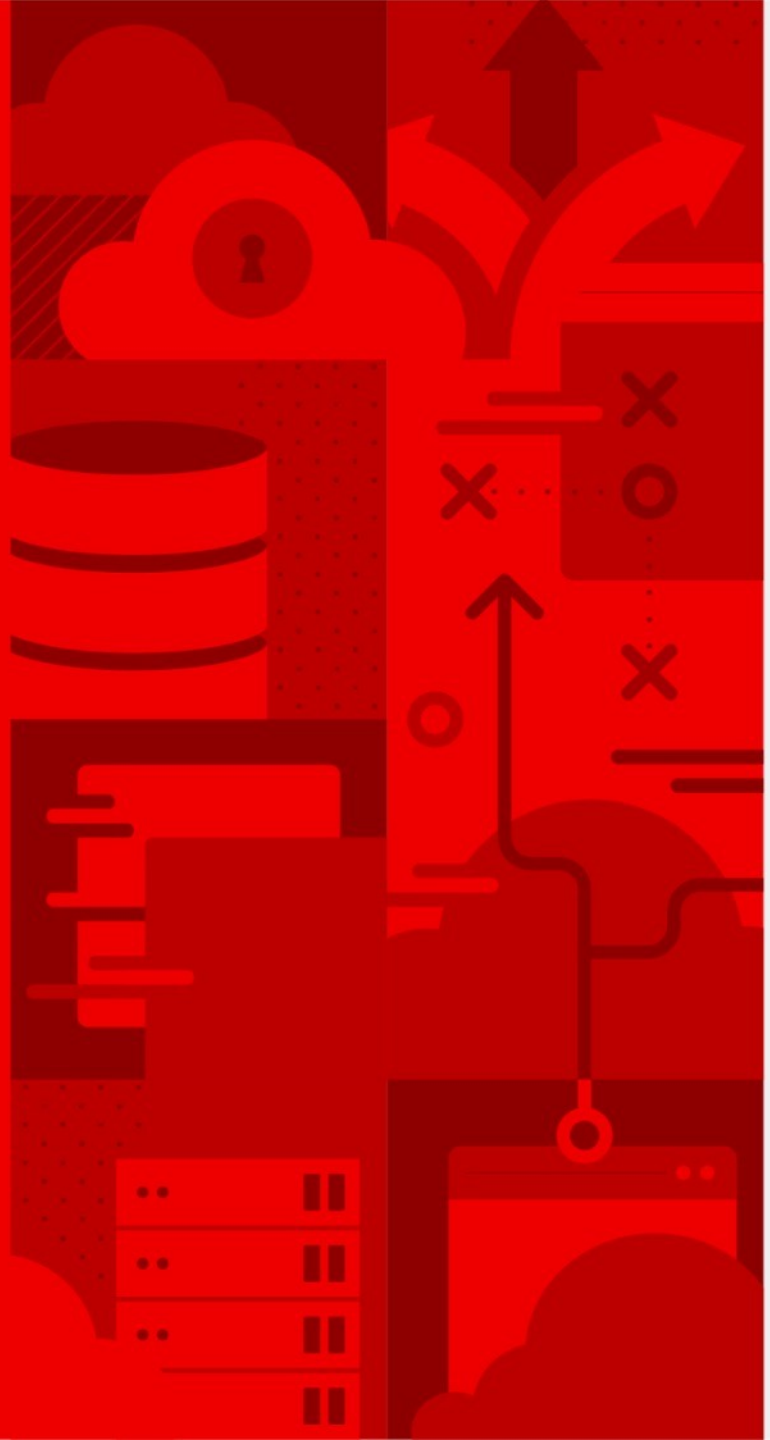
Q: Is it a good idea overall?

Q: Would you use this workflow?

Q: Wait forever? Reboot after an hour?

Q: Which HTTP code to use for wait?

Q: Someone here to help with DI?



Danke schön

Red Hat is the world’s leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat