Foreman in Your **Data Center**

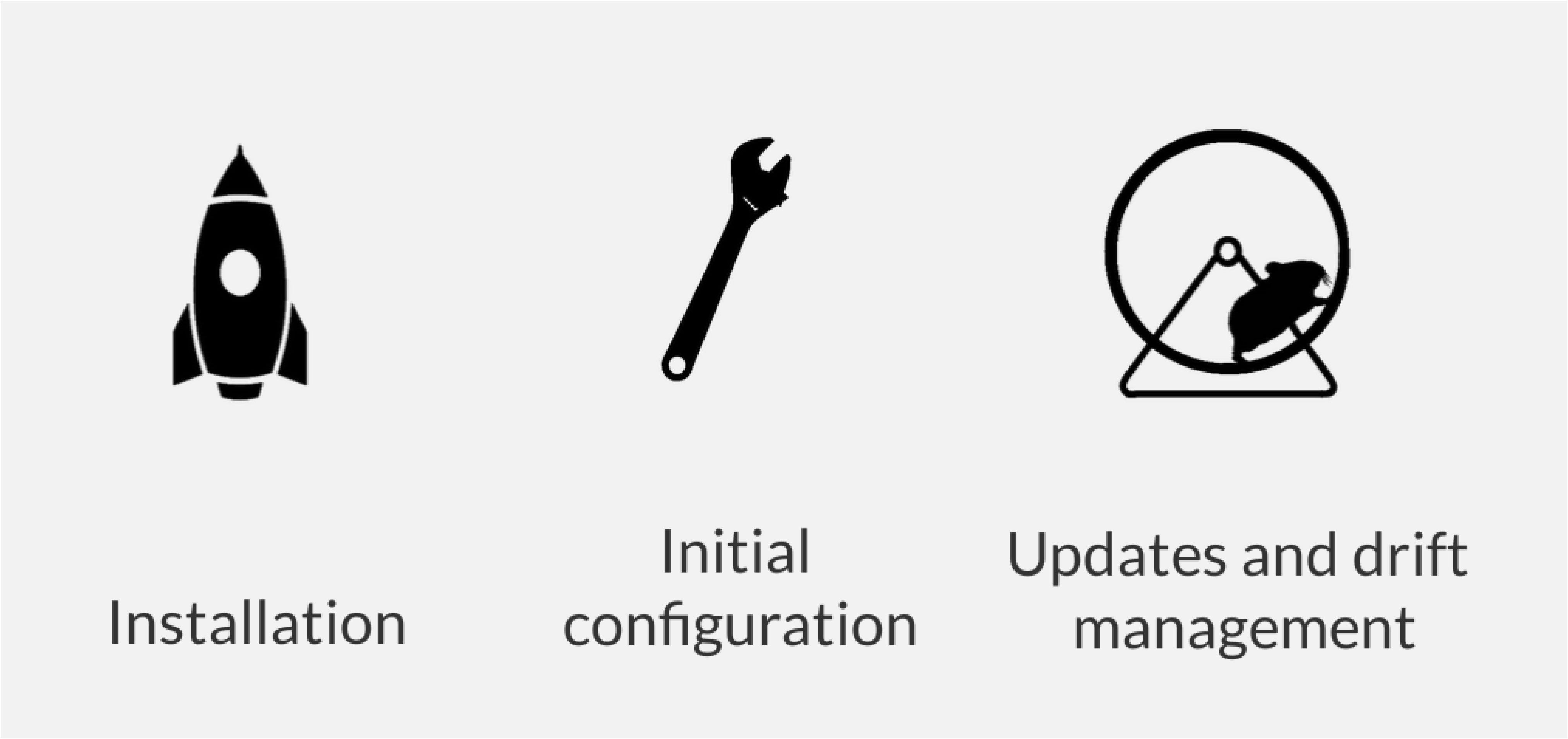
Lukáš Zapletal
@lzap

Agenda

- Introduction
  - Provisioning
  - Configuration
  - Monitoring
- History and Technology
- Architecture and Installation
- **Demo**: Foreman Basics
- Customizing foreman
  - Automating with CLI + API
  - Plugins (Bootdisk, Discovery, Docker, Katello, Chef, Salt, Hooks, Remote Execution)
- **Demo**: Foreman Plugins

Foreman's Realm

Managing the Lifecycle of your
Systems



Installation

Initial configuration

Updates and drift management

Foreman

Provisioning Configuration Monitoring

Provisioning

- Provision new machines or containers to (almost) anything
  - Bare metal, oVirt, libvirt, VMware, Docker, EC2, Rackspace, Digital Ocean, OpenStack, etc.
- If we don't support it today, we can via new plugins

Provisioning

- Provisioning types:
  - PXE - via PXELinux and kickstart, preseed, AutoYAST, etc
  - Image-based - cloning, configured over SSH or user data (cloudinit)
- For virtualization provider, we create the VM
- For everything we orchestrate related services through Smart Proxies
  - DNS - DHCP / TFTP
  - FreeIPA Realm - Configuration Management

Configuration

- Puppet
- Via plugins:
  - Chef
  - Salt
  - Ansible

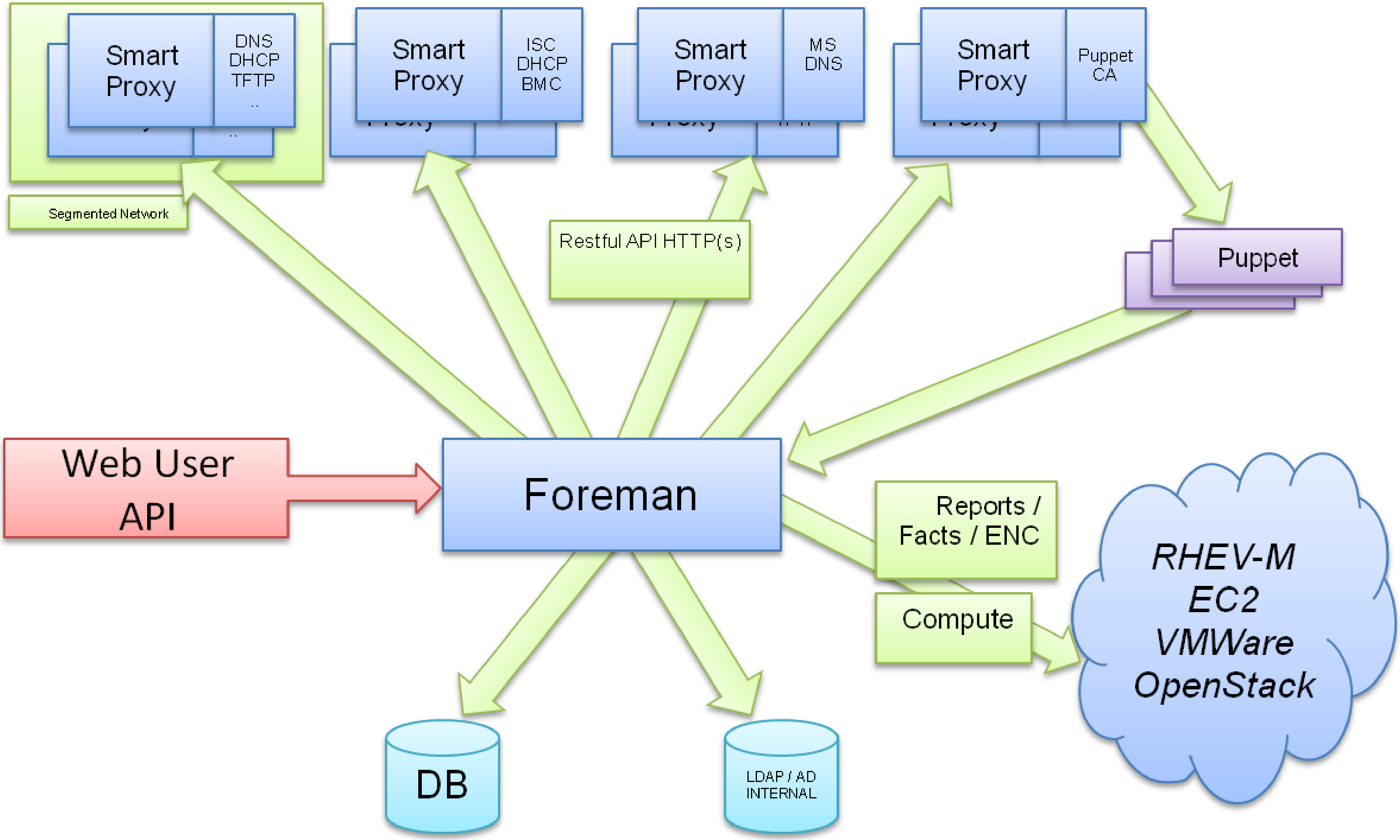
- Automatic registration & setup of clients, including autosigning certs/keys
- Defining:
  - Classes / states
  - Parameters / pillars
- Inventory data:
  - Facts / Grains
  - results of configuration runs

Monitoring

- Generic **Report API** with graphs/trends:
  - System Inventories
  - Reports from runs
  - Generic reports: ABRT, OpenSCAP
- Context sensitive search:
  - Not full-text (SQL level)
  - Keyword completion
  - Works across whole application

Distributed Architecture

- **Smart Proxies** located locally on Foreman itself or independent – (orchestration)
- Large organizations and/or multi-tenancy:
  - Organizations (Divisions)
  - Locations
- Strong **RBAC** model
  - Users / Groups
  - Permissions / **Filters**
- **LDAP** / MS-AD integration



Smart Proxy

DNS
DHCP
TFTP
..

Segmented Network

Smart Proxy

ISC
DHCP
BMC

Smart Proxy

MS
DNS

Smart Proxy

Puppet
CA

Restful API HTTP(s)

Puppet

Web User
API

Foreman

Reports /
Facts / ENC

Compute

RHEV-M
EC2
VMWare
OpenStack

DB

LDAP / AD
INTERNAL

History

- Project started in July 2009
- Initial set of features: Puppet + PXE provisioning
- 213 unique contributors (winter 2015)
- Core team sponsored by Red Hat (GMT +10 -8)
- Translated to 13 languages
- Healthy and friendly community
- Reported usage: Red Hat, CERN, EMC, Citrix, DHL, BBC, Digg, Good Data, Mozilla, eBay/Paypal (100,000 nodes)

Technology

Foreman itself

- Ruby on Rails application
- Targeted on UNIX platforms, Ruby 1.9+
- Steep learning curve (git clone, bundle install)

- Smart Proxy
  - Ruby / Sinatra application
  - Minimum dependencies
  - Quick start (git clone, bundle install)
  - On all Ruby 1.8+ platforms (incl. MS Windows)

Installation

- Repositories for RHEL/Fedora, Debian/Ubuntu
- Puppet-based installer
- Sane defaults for POC deployments
- Able to install, configure and manage:
  - Foreman app
  - Smart-proxies
  - Services: DNS, ~~DHCP~~, TFTP, Puppet
  - Selected plugins

```
# foreman-installer -h | wc -l
439
```

Demo: Foreman Basics

Customization

- Customize Foreman to support **your** workflows!
  - Configuration options in UI: Adminster → Settings
  - Smart proxy configuration values (features)
  - Automation with API + CLI
  - Foreman Plugins
  - Smart Proxy Plugins
  - Foreman Hooks Plugin

API & CLI

- Full UI coverage
- All of our API is documented
- Documentation DSL generates dynamic Ruby bindings
- Full RESTful API
  - Docs at /apidoc on your Foreman server
  - Also available at http://www.theforeman.org/api/1.9/index.html

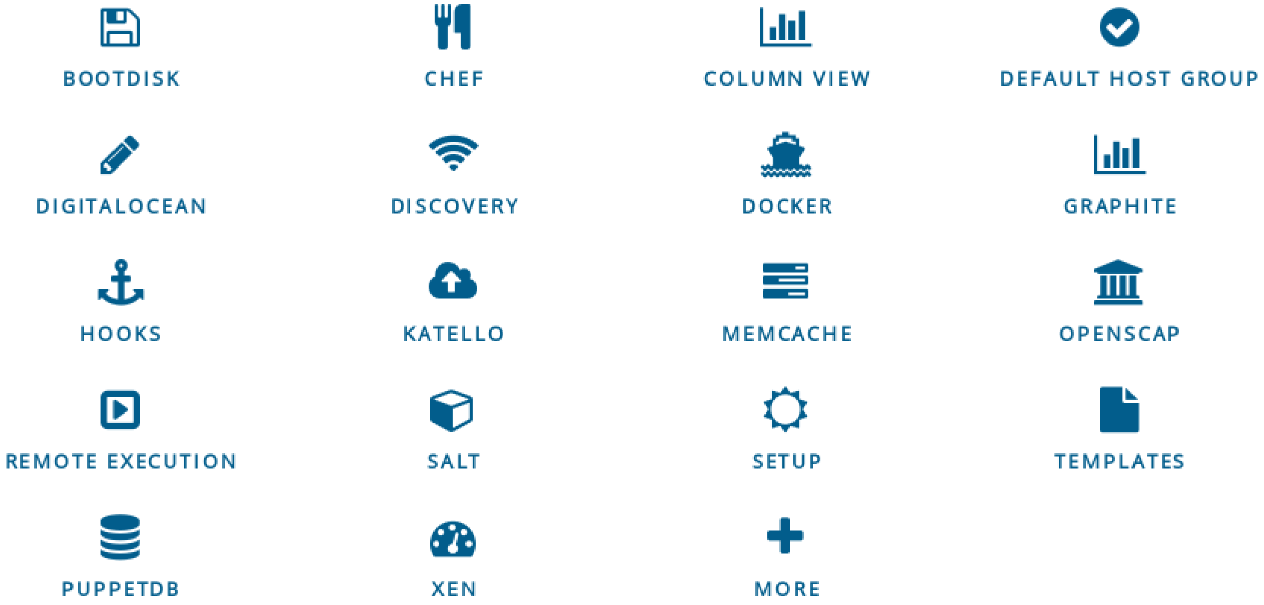
Hammer CLI

- On par with UI
- Username/Password authentication
- Easy to use, great for working in shell

```
hammer salt-key list --smart-proxy=smartproxy.example.com
```

Plugins

Rich ecosystem of existing plugins



BOOTDISK	CHEF	COLUMN VIEW	DEFAULT HOST GROUP
DIGITALOCEAN	DISCOVERY	DOCKER	GRAPHITE
HOOKS	KATELLO	MEMCACHE	OPENSCAP
REMOTE EXECUTION	SALT	SETUP	TEMPLATES
PUPPETDB	XEN	MORE	

More Info: http://projects.theforeman.org/projects/foreman/wiki/List_of_Plugins

Bootdisk plugin

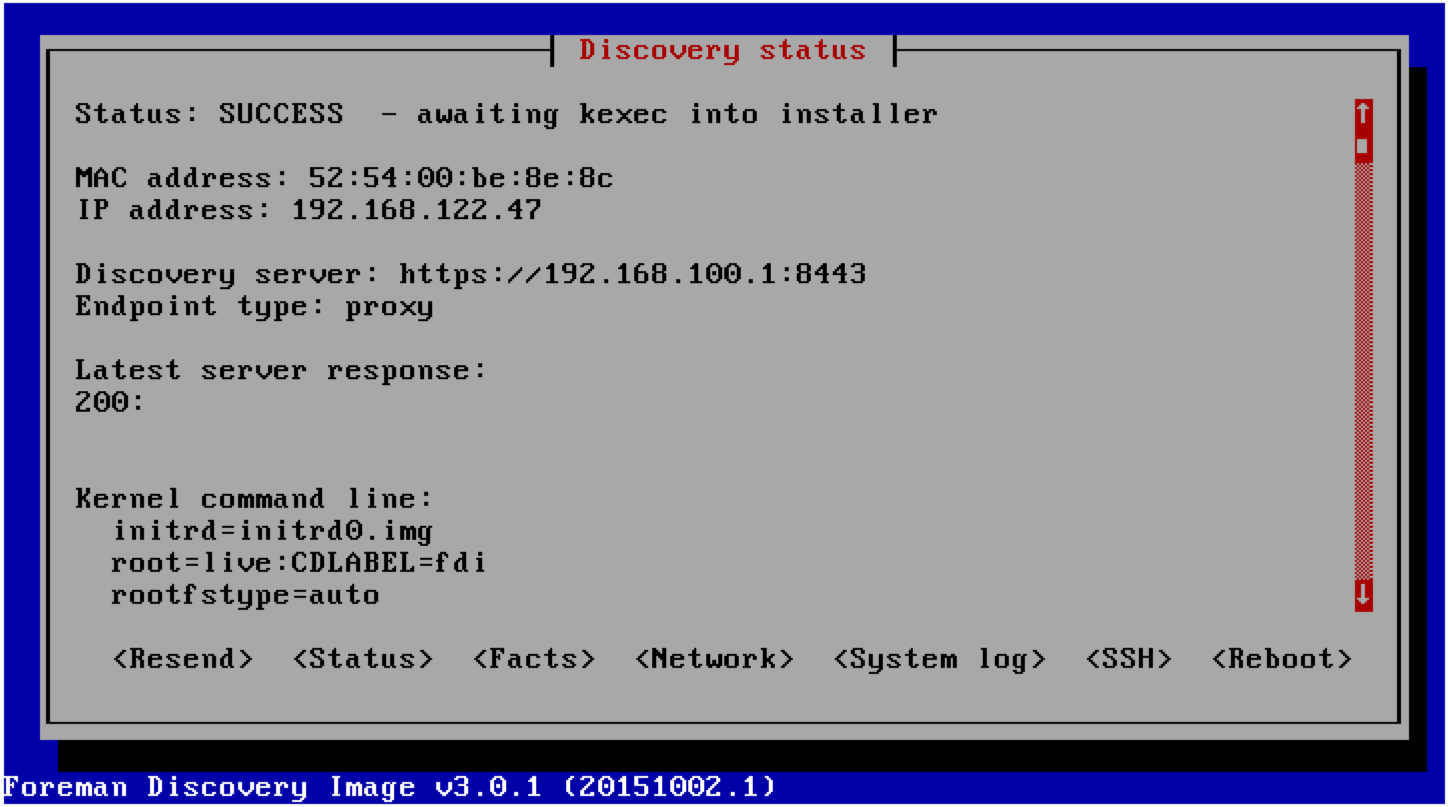
- Small hybrid ISO downloaded from Host UI page
- Unknown or pre-registred hosts boot chainloads from Foreman without PXE/TFTP
- Generic image
  - iPXE-based, DHCP required
- Host image
  - iPXE-based, DHCP not required
- Full host image
  - SYSLINUX-based, DHCP required, OS specific

Discovery plugin

- Unknown host boots via DHCP/PXE
  - Becomes available in Foreman as a “Discovered Host”
  - Workflow remains the same
  - Discovery image is RHEL7/CentOS7-based
- Provision with as few as **NO** clicks
  - Automatic provisioning via rules on arbitrary facts:
    - cpu_count < 8 → web server host group
    - cpu_count >= 8 → db box host group

Discovery plugin

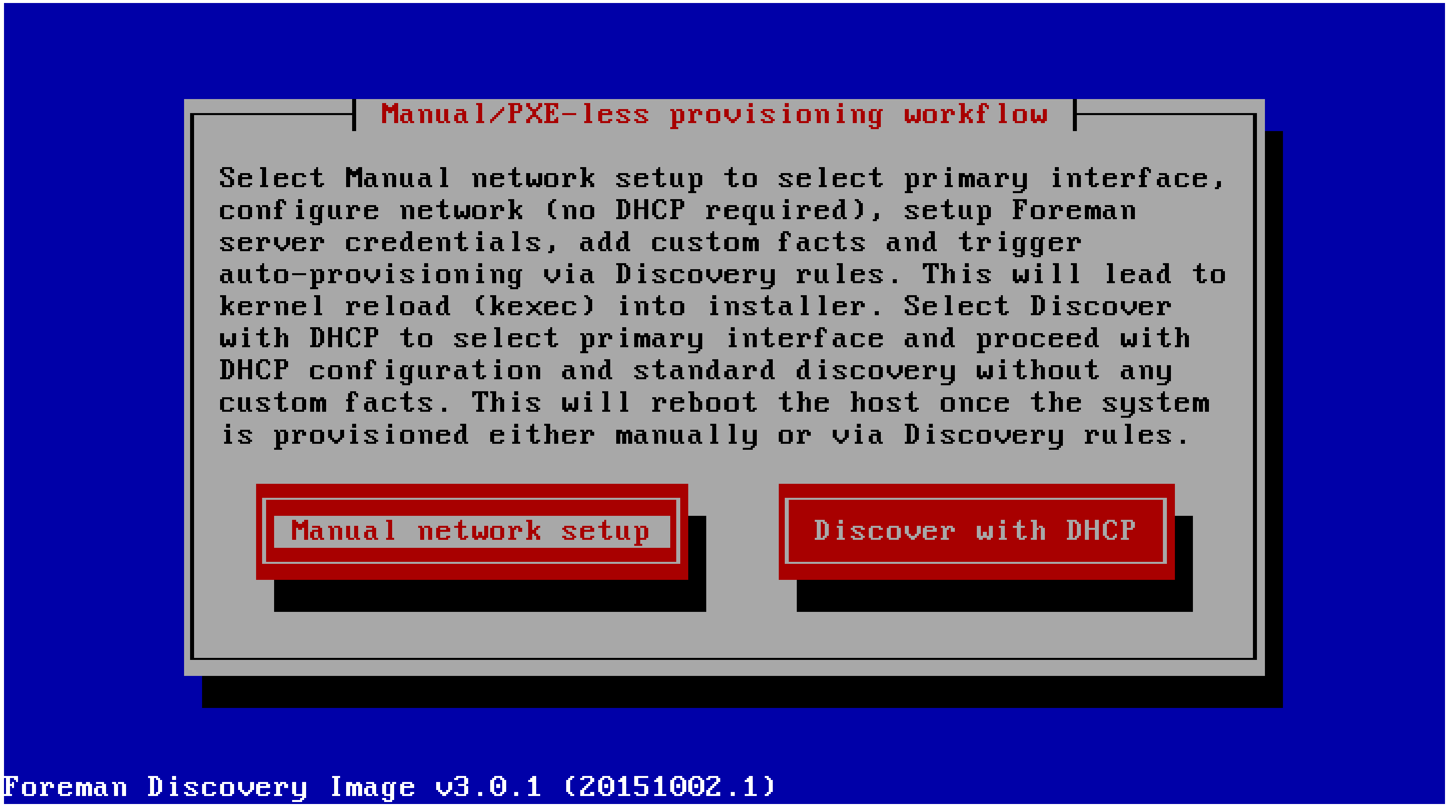
- Metal as a Service – PXE installation



```
Discovery status

Status: SUCCESS  - awaiting kexec into installer

MAC address: 52:54:00:be:8e:8c
IP address: 192.168.122.47

Discovery server: https://192.168.100.1:8443
Endpoint type: proxy

Latest server response:
200:


Kernel command line:
  initrd=initrd0.img
  root=live:CDLABEL=fdi
  rootfstype=auto

  <Resend>  <Status>  <Facts>  <Network>  <System log>  <SSH>  <Reboot>

Foreman Discovery Image v3.0.1 (20151002.1)
```

Discovery plugin

- PXE-less (un)attended workflow (supports EFI)



Manual/PXE-less provisioning workflow

Select Manual network setup to select primary interface, configure network (no DHCP required), setup Foreman server credentials, add custom facts and trigger auto-provisioning via Discovery rules. This will lead to kernel reload (kexec) into installer. Select Discover with DHCP to select primary interface and proceed with DHCP configuration and standard discovery without any custom facts. This will reboot the host once the system is provisioned either manually or via Discovery rules.

Foreman Discovery Image v3.0.1 (20151002.1)

Docker plugin

- Manage many docker hosts
- Deploy new containers easily & view their status, logs, etc
- Multiple registry support & integration with Katello
- https://github.com/theforeman/foreman-docker

Katello plugin

- Content Lifecycle Management
  - http://www.katello.org/
- Sync RPM, Docker, and Puppet content
- Spin repositories with filters using Content Views

Katello plugin

- Manage through a lifecycle
  - Dev → QA → Production (Environments)
- Patch Management
  - Emergency Patches
  - Errata Reports
- And much more!

Hooks plugin

- Hooks
  - Triggered on actions: on action, do X
    - host create/update/delete, build complete, etc.
    - X could be anything
      - add to nagios
      - send an email
    - Can be shell, python, ruby, etc.
  - More info: https://github.com/theforeman/foreman_hooks

Salt plugin

- Bootstrapping nodes
- Full interface to keys/autosign
- Define states, pillars via ext_node and ext_pillar
- Import reports (state.highstate results) and grains into Foreman
- API & CLI

Chef plugin

- Automatic bootstrapping of clients
- Import reports and attributes into Foreman
- Decomission nodes from Chef server when deleted in Foreman

Remote execution plugin

- Arbitrary commands on hosts
- Job Templates
  - Based on Foreman Templating engine
  - Input parameters
- Collected data available (Facts)
- Multiple providers architecture:
  - SSH (via Smart Proxies)

Writing Foreman Plugins

- Foreman:
  - Rails Engine
  - Extra Foreman API (plugin registration)
  - Distributed as a Ruby GEM
  - Template and HOWTO available
- Smart Proxy:
  - Sinatra app (REST API)
  - Small plugin registration API
  - Distributed as a Ruby GEM

What Next?

- Visit us http://theforeman.org/
- If you do something cool with Foreman, let us know!
- Find us:
  - IRC: irc.freenode.net
    - #theforeman
    - #theforeman-dev
  - Mailing Lists on Google groups
    - foreman-users
    - foreman-dev

Demo: Foreman Plugins