Give Your PXE Wings!

It’s not magic! How booting actually works.

Presentation for virtual SREcon 2020
By Rob Hirschfeld, RackN

Rob Hirschfeld

@zehicle
Co-Founder of RackN

We created Digital Rebar
Bare Metal Provisioning ++

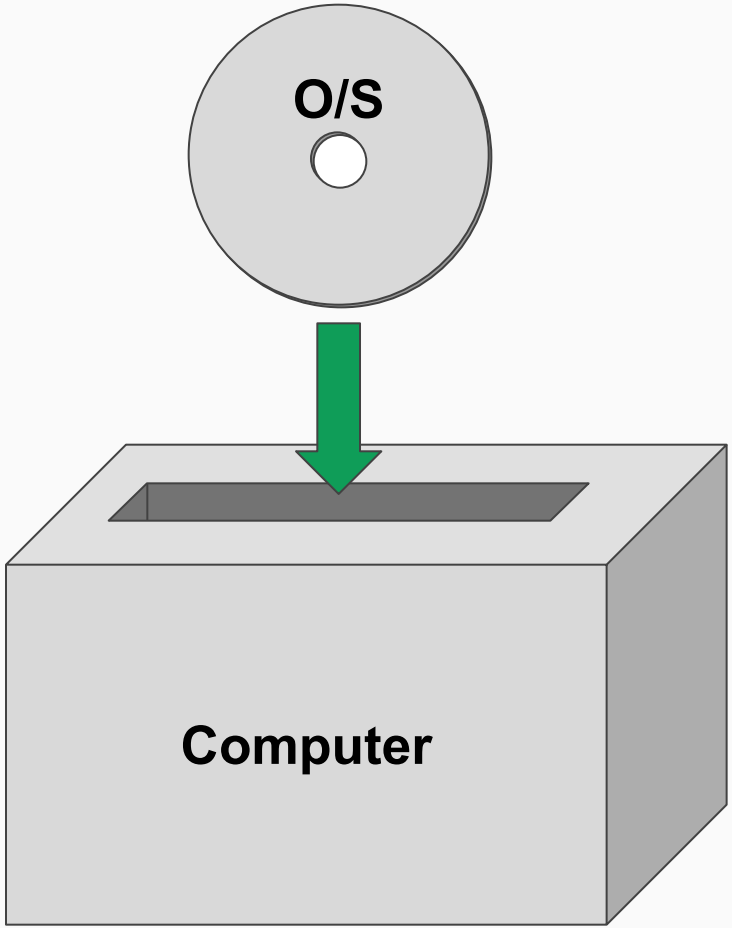
@L8istSh9y Podcast on PXE:
*http://bit.ly/**pxewings***

In concept, Provisioning is Easy!

We're just installing an operating system on a server or switch!

Why is that so hard?!

- Bootstrapping
- Firmware Limitations
- Variation
- Networking
- Security
- Performance
- Post configuration

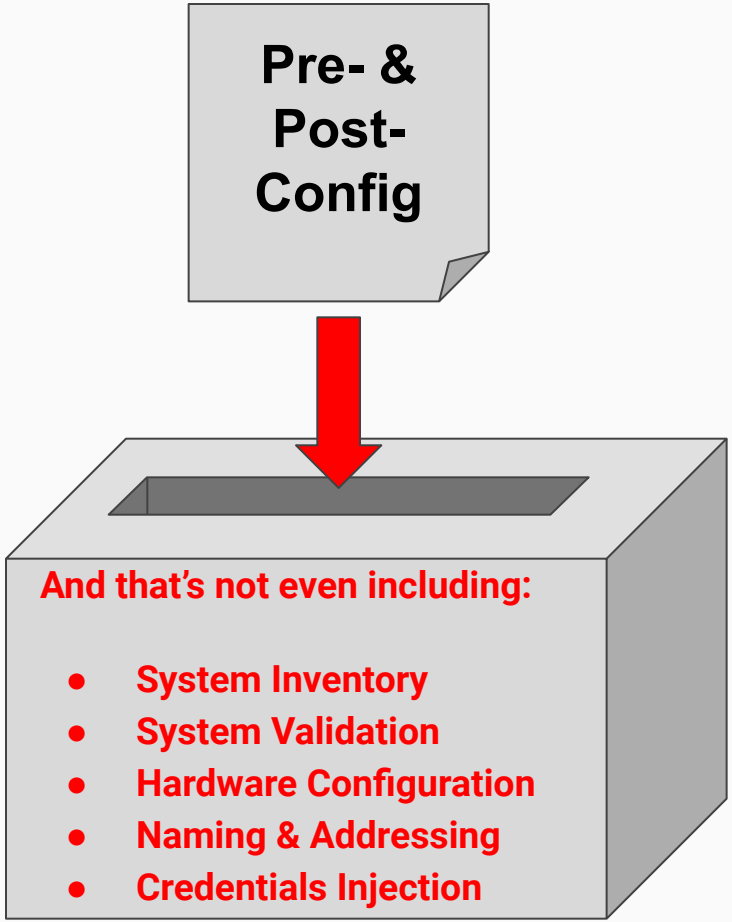


In concept, Provisioning is Easy!

We're just installing an operating system on a server or switch!

Why is that so hard?!

- Bootstrapping
- Firmware Limitations
- Variation
- Networking
- Security
- Performance
- Post configuration



Exploring Provisioning Approaches

Netboot (25 min)

- PXE
- iPXE
- ONIE
- Kickstart
- Preseed

Image Deploy (10 min)

- Packer
- Write Boot Part
- Cloud Init

Esoteric Flavors (5 min)

- kexec
- Secure Boot
- BMC Boot

Exploring Provisioning Approaches

Netboot (25 min)

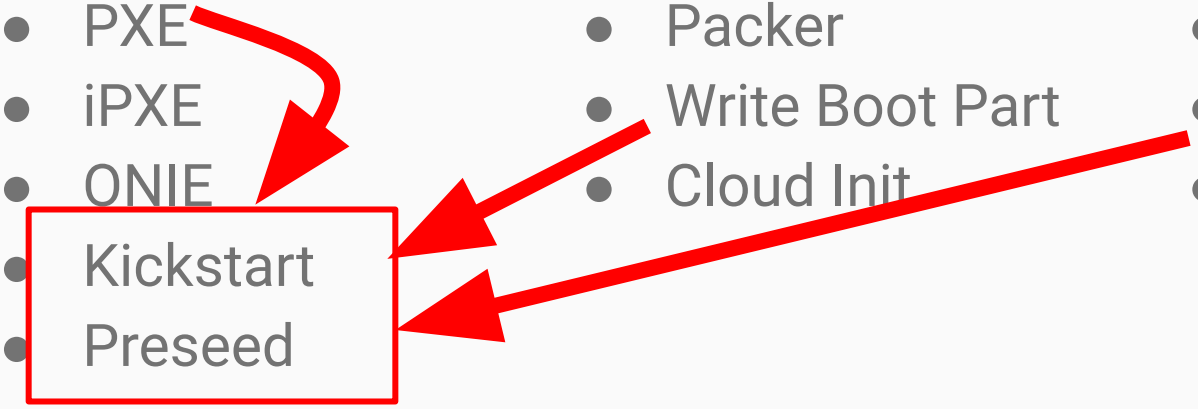
- PXE
- iPXE
- ONIE
- Kickstart
- Preseed

Image Deploy (10 min)

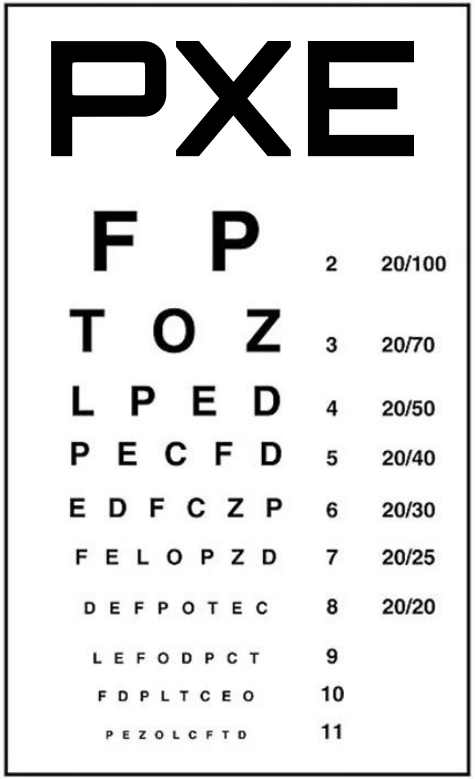
- Packer
- Write Boot Part
- Cloud Init

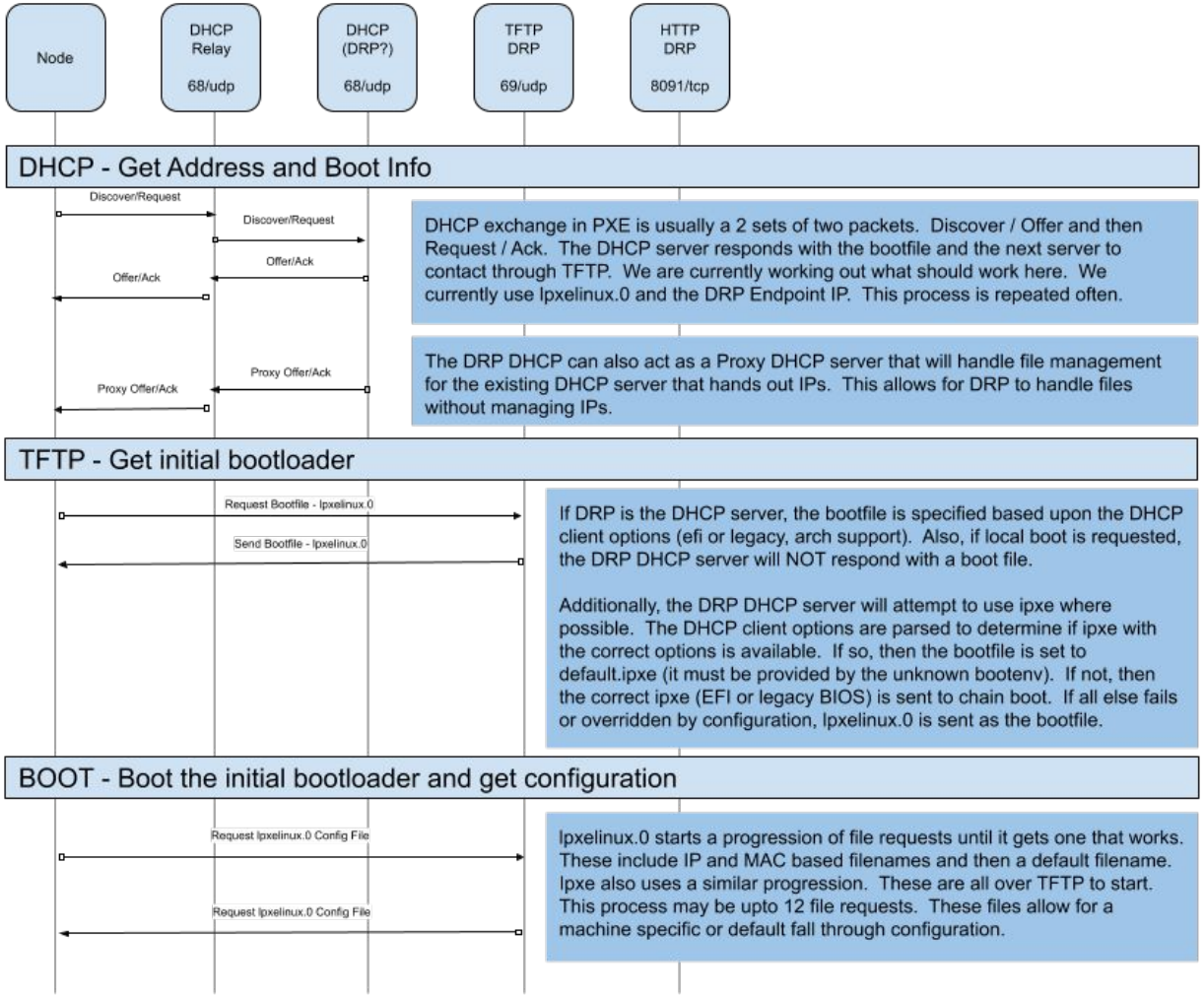
Esoteric Flavors (5 min)

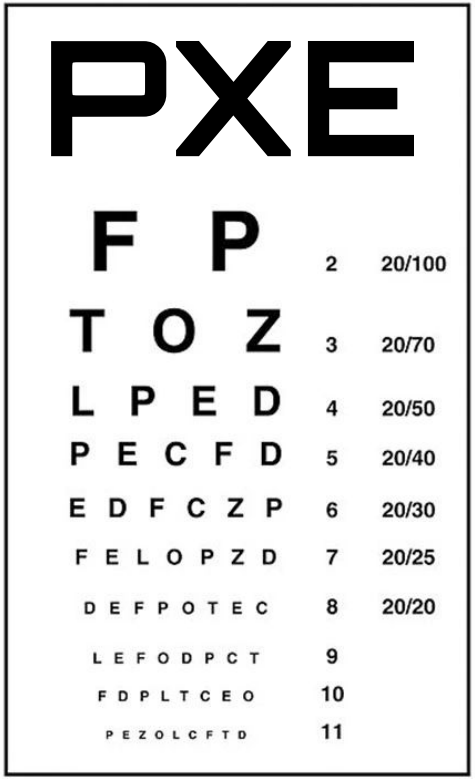
- kexec
- Secure Boot
- BMC Boot

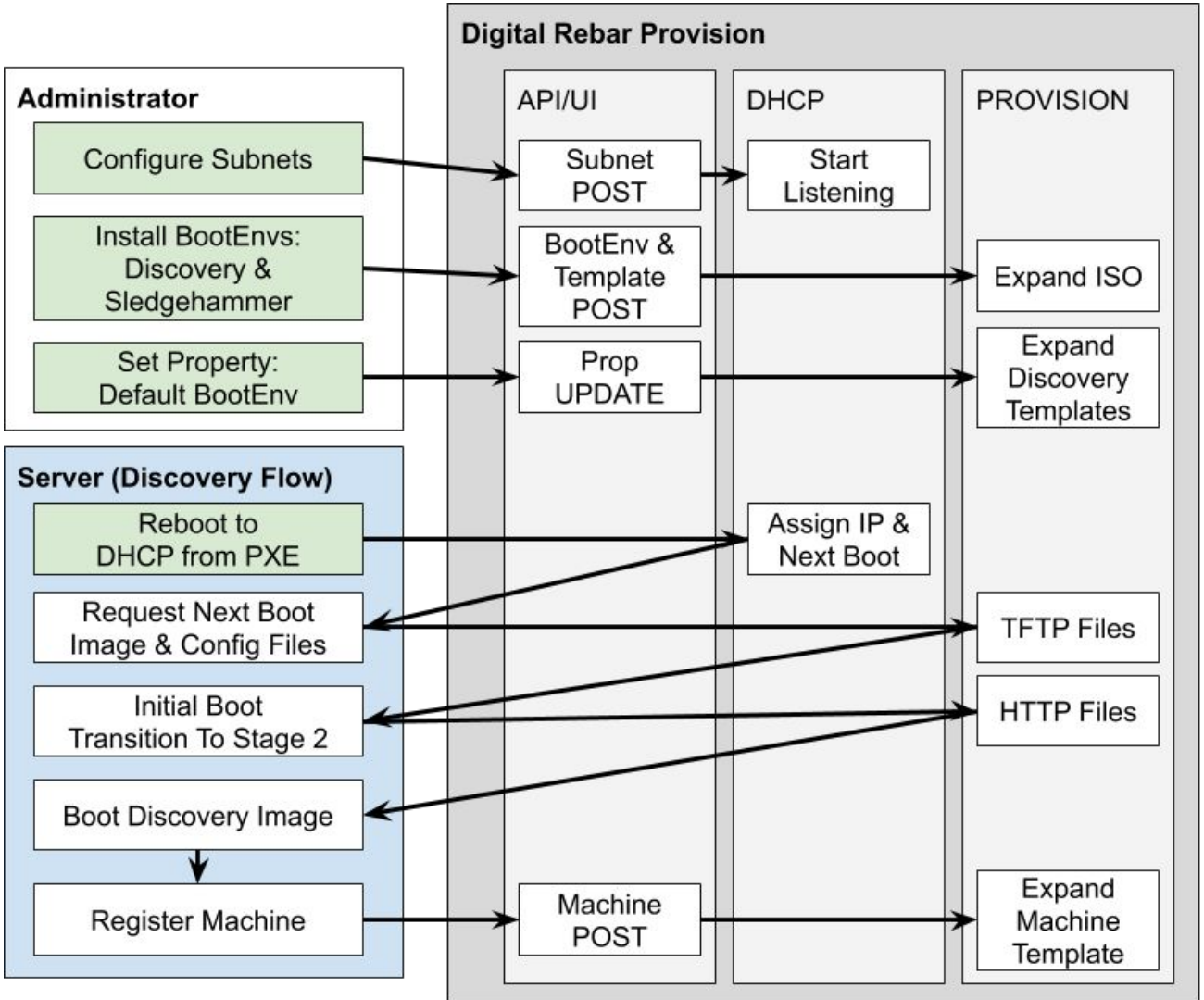


All roads lead to a kernel init process


PXE
F P
2 20/100
T O Z
3 20/70
L P E D
4 20/50
P E C F D
5 20/40
E D F C Z P
6 20/30
F E L O P Z D
7 20/25
D E F P O T E C
8 20/20
L E F O D P C T
9
F D P L T C E O
10
P E Z O L C F T D
11

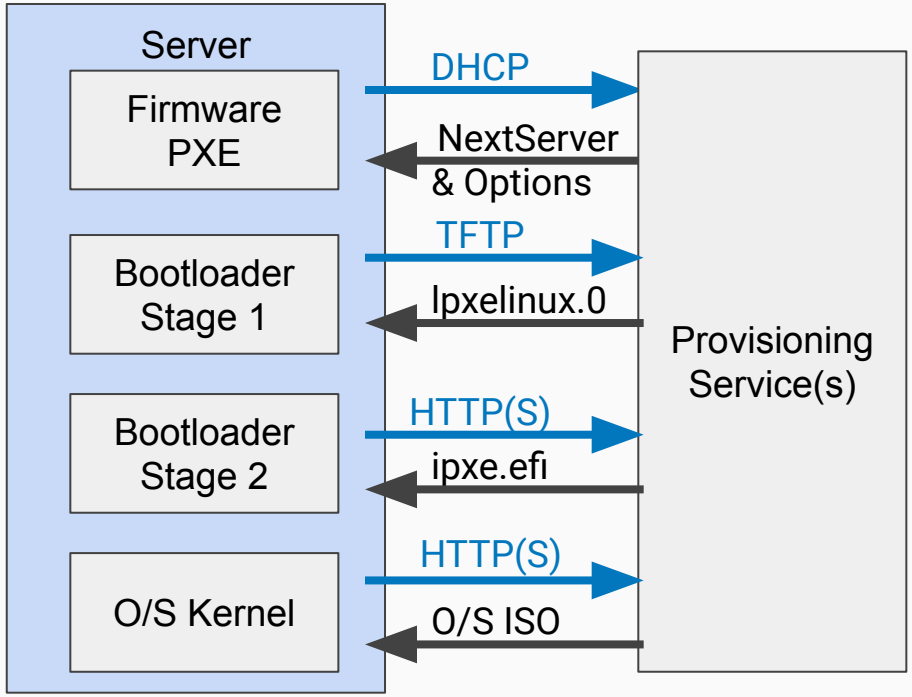

Node
DHCP Relay 68/udp
DHCP (DRP?) 68/udp
TFTP DRP 69/udp
HTTP DRP 8091/tcp
DHCP - Get Address and Boot Info
Discover/Request
Discover/Request
Offer/Ack
Offer/Ack
DHCP exchange in PXE is usually a 2 sets of two packets. Discover / Offer and then Request / Ack. The DHCP server responds with the bootfile and the next server to contact through TFTP. We are currently working out what should work here. We currently use lpxelinux.0 and the DRP Endpoint IP. This process is repeated often.
Proxy Offer/Ack
Proxy Offer/Ack
The DRP DHCP can also act as a Proxy DHCP server that will handle file management for the existing DHCP server that hands out IPs. This allows for DRP to handle files without managing IPs.
TFTP - Get initial bootloader
Request Bootfile - lpxelinux.0
Send Bootfile - lpxelinux.0
If DRP is the DHCP server, the bootfile is specified based upon the DHCP client options (efi or legacy, arch support). Also, if local boot is requested, the DRP DHCP server will NOT respond with a boot file.
Additionally, the DRP DHCP server will attempt to use ipxe where possible. The DHCP client options are parsed to determine if ipxe with the correct options is available. If so, then the bootfile is set to default.ipxe (it must be provided by the unknown bootenv). If not, then the correct ipxe (EFI or legacy BIOS) is sent to chain boot. If all else fails or overridden by configuration, lpxelinux.0 is sent as the bootfile.
BOOT - Boot the initial bootloader and get configuration
Request lpxelinux.0 Config File
Request lpxelinux.0 Config File
lpxelinux.0 starts a progression of file requests until it gets one that works. These include IP and MAC based filenames and then a default filename. Ipxe also uses a similar progression. These are all over TFTP to start. This process may be upto 12 file requests. These files allow for a machine specific or default fall through configuration.


PXE
F P
T O Z
L P E D
P E C F D
E D F C Z P
F E L O P Z D
D E F P O T E C
L E F O D P C T
F D P L T C E O
P E Z O L C F T D
2 20/100
3 20/70
4 20/50
5 20/40
6 20/30
7 20/25
8 20/20
9
10
11

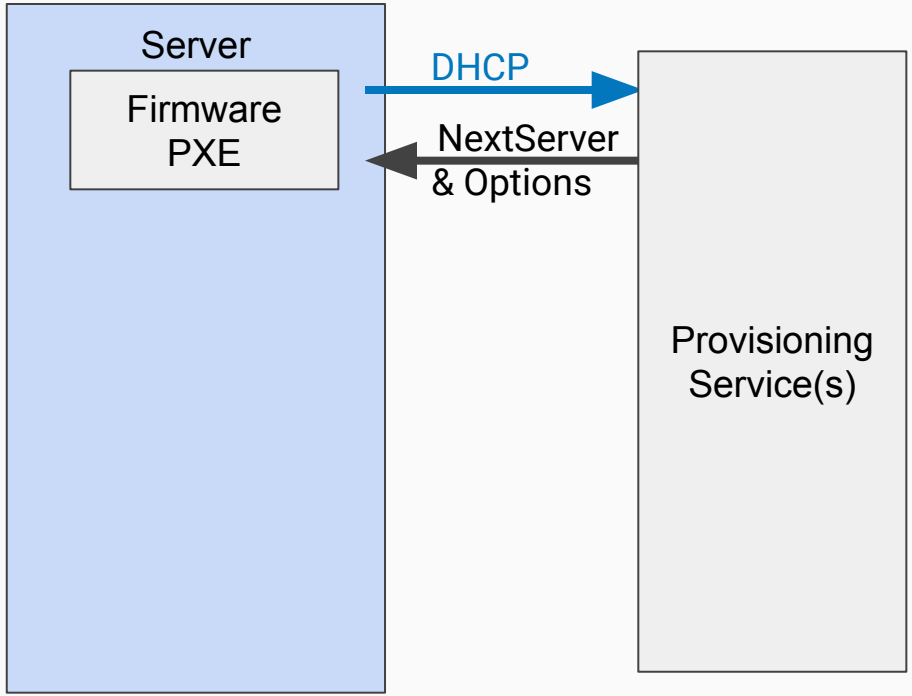

Digital Rebar Provision
Administrator
Configure Subnets
Install BootEnvs: Discovery & Sledgehammer
Set Property: Default BootEnv
Server (Discovery Flow)
Reboot to DHCP from PXE
Request Next Boot Image & Config Files
Initial Boot Transition To Stage 2
Boot Discovery Image
Register Machine
API/UI
Subnet POST
BootEnv & Template POST
Prop UPDATE
Machine POST
DHCP
Start Listening
Assign IP & Next Boot
PROVISION
Expand ISO
Expand Discovery Templates
TFTP Files
HTTP Files
Expand Machine Template

Let’s PXE!

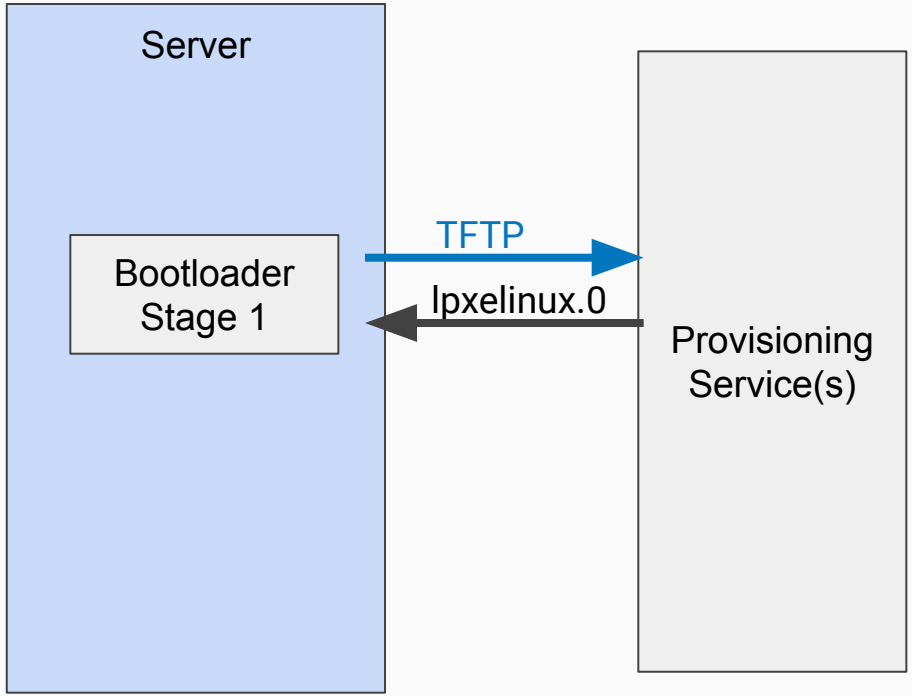
Bootstrapping is a multi-stage process



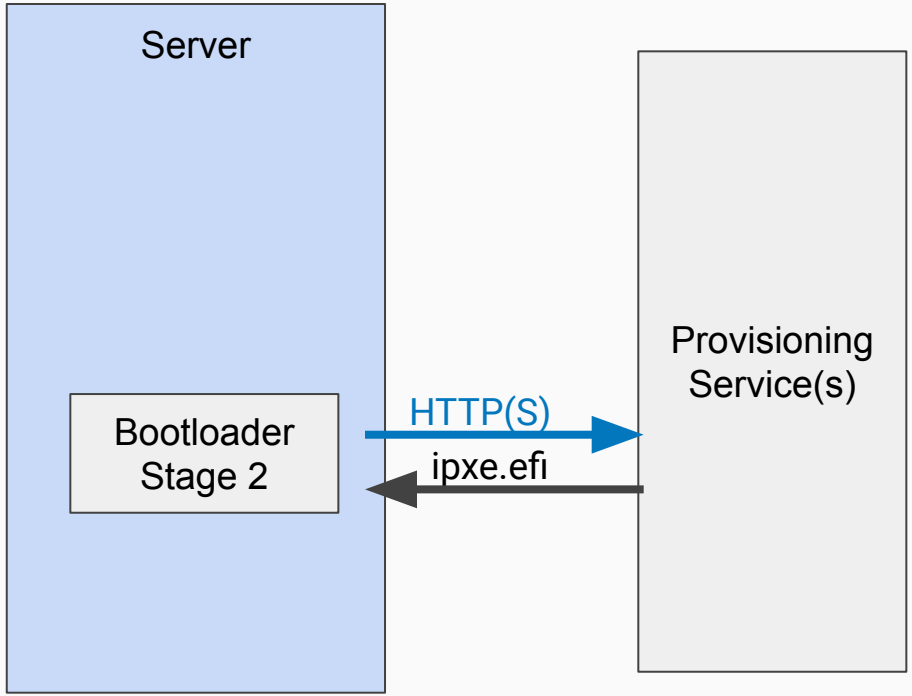
First: Get on the network



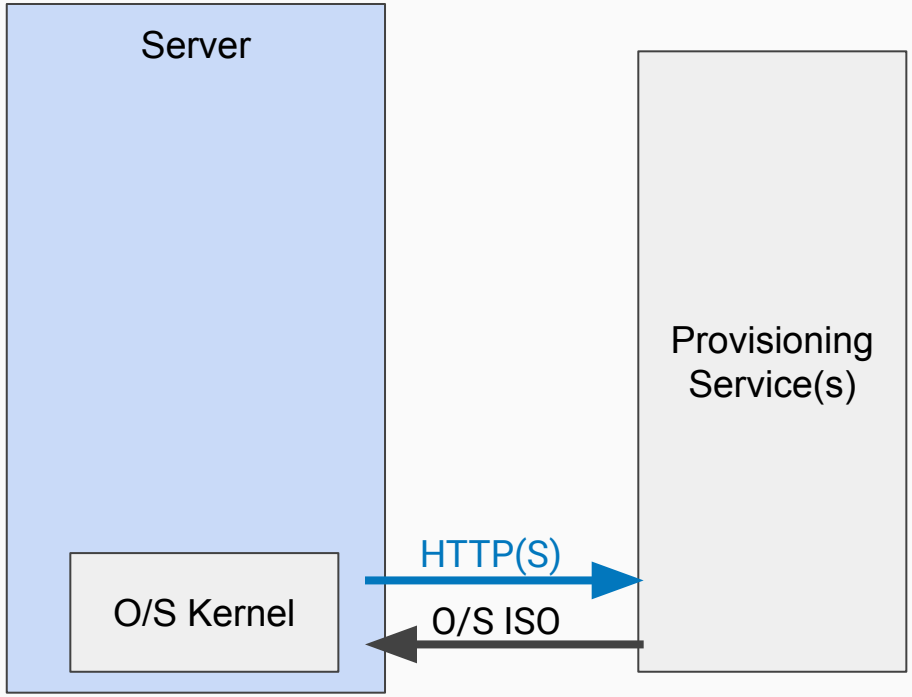
Then: Download a Bootloader



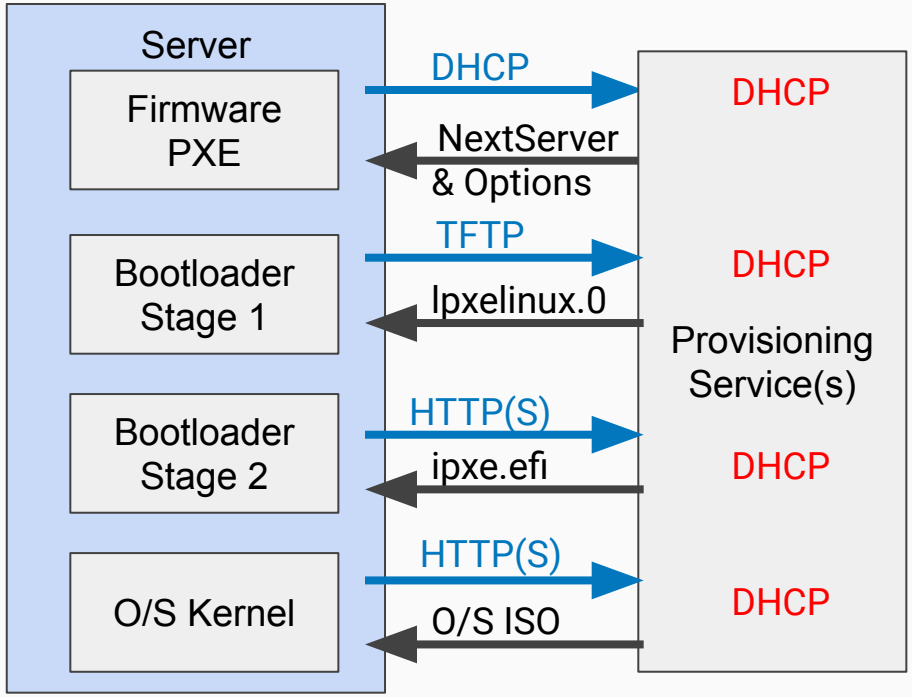
Then get a BETTER Bootloader



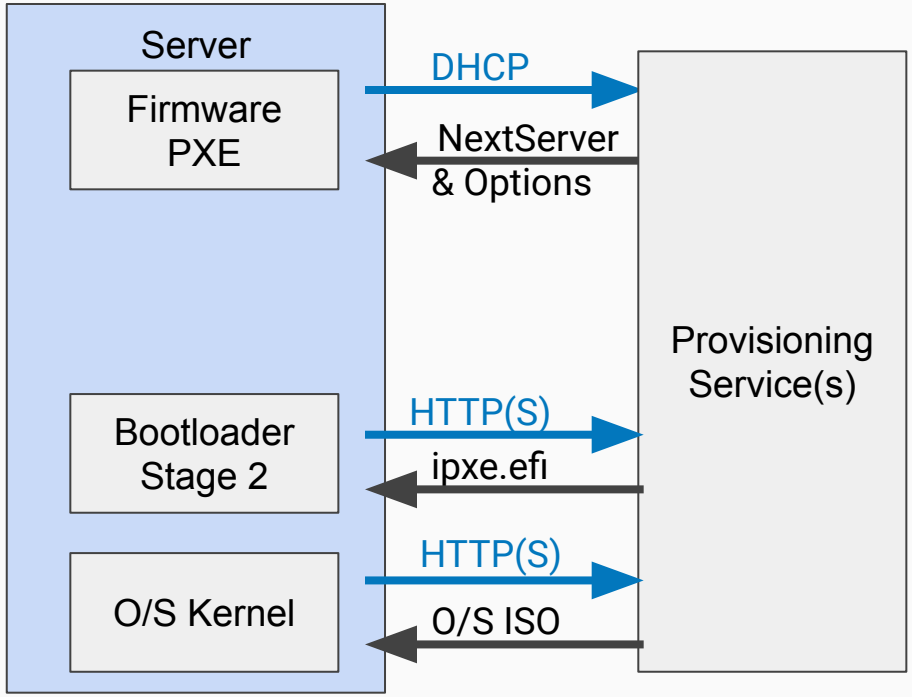
Finally load a “real” operating system



Each stage is actually a NEW O/S Load

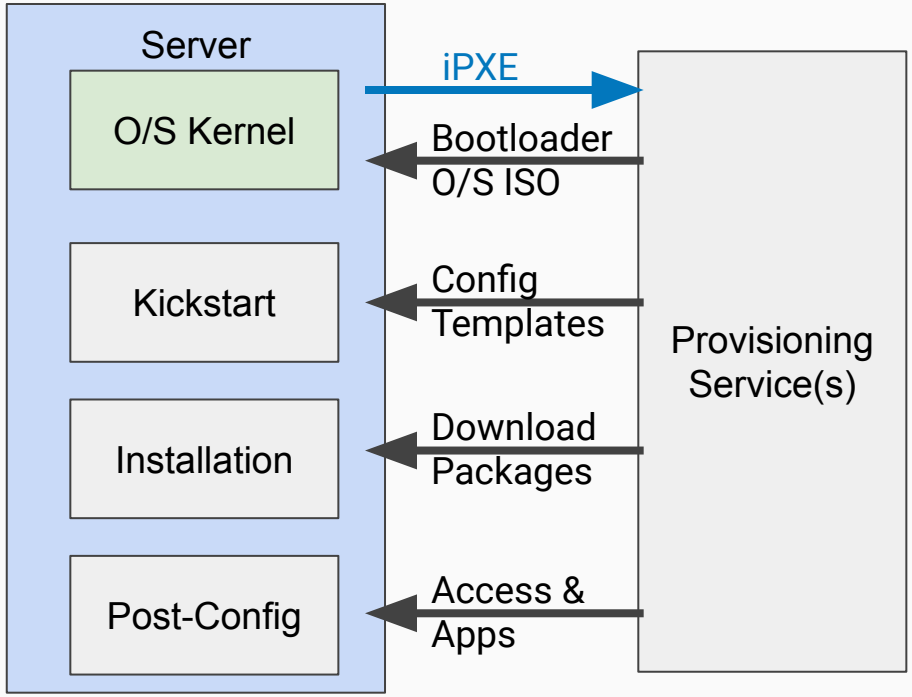


And modern servers can skip TFTP!
So… technically, no longer PXE

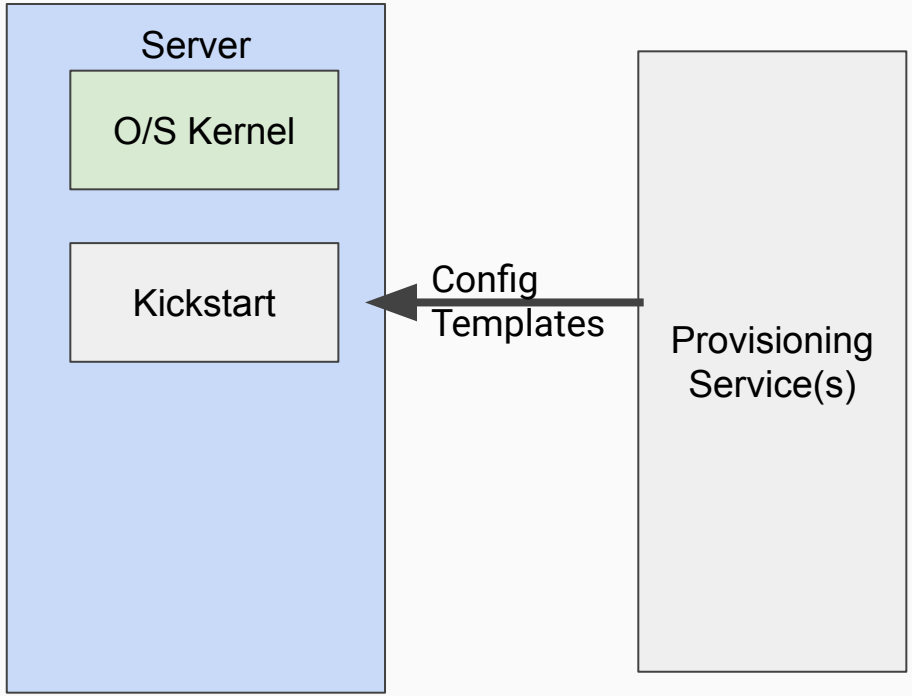


Yay! We’re done, right?

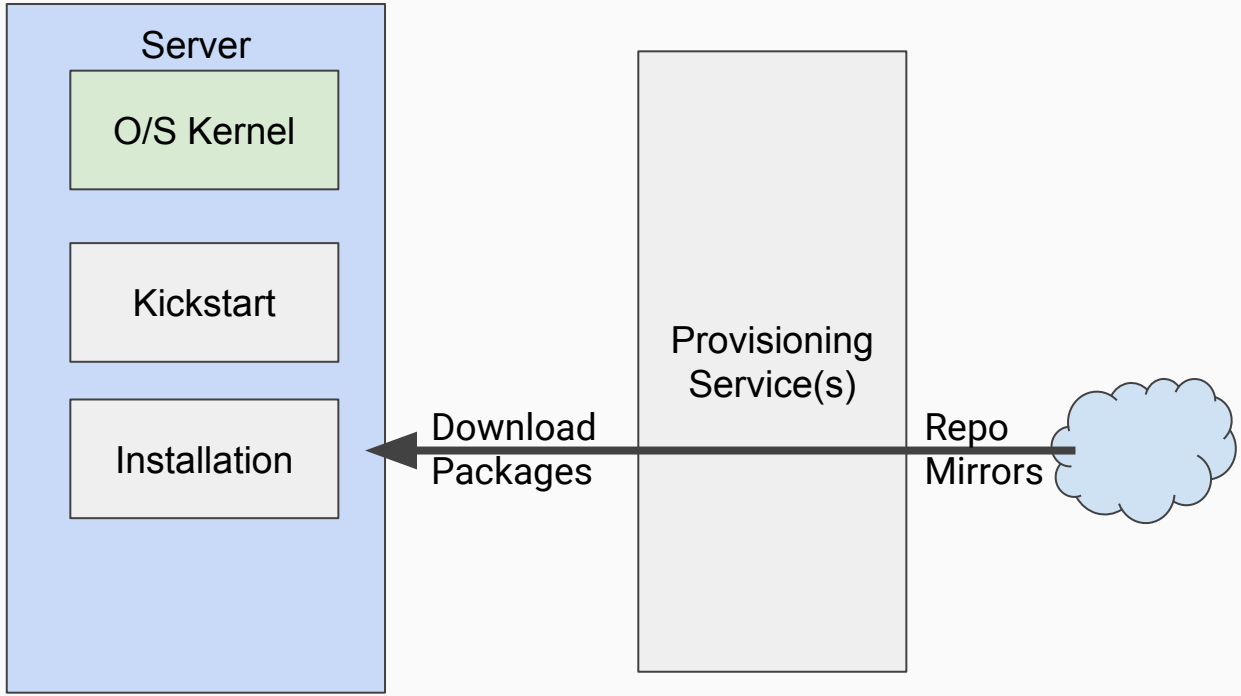
Provisioning is more than PXE



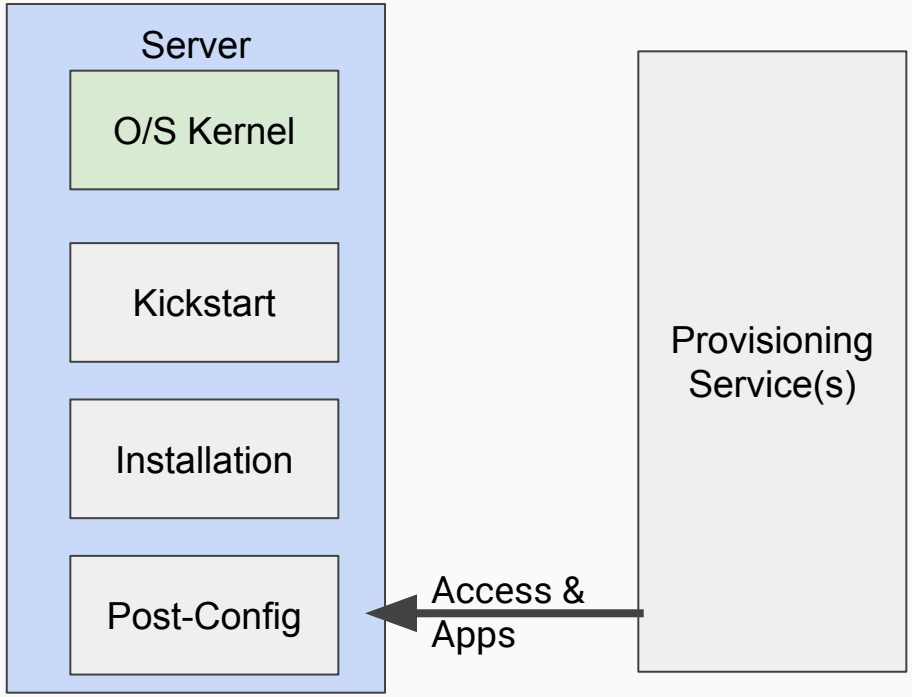
Hardware varies, so
Install must be guided by templates



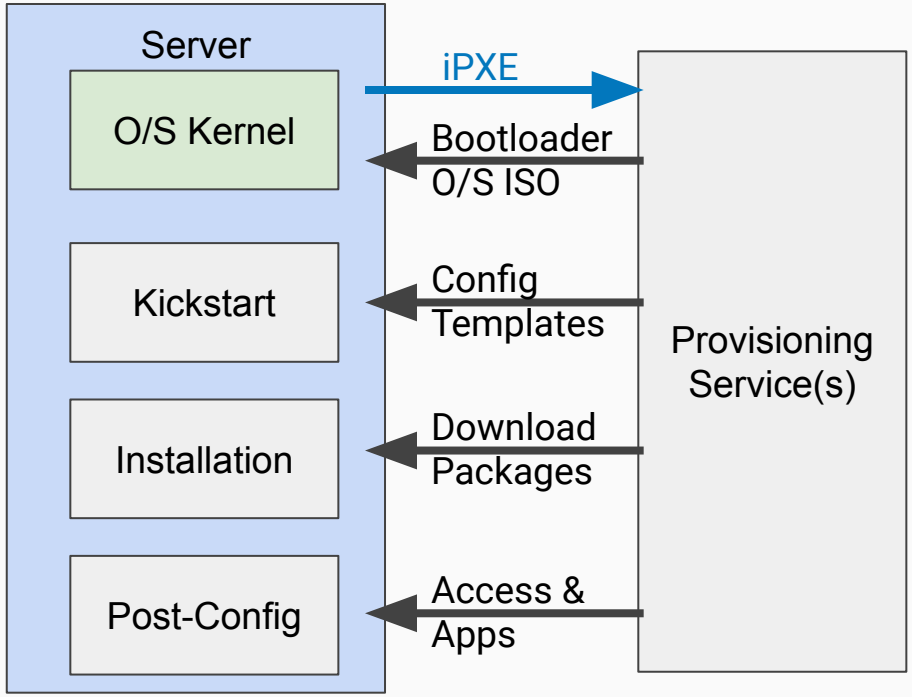
ISOs are minimal and stale
So they must be updated



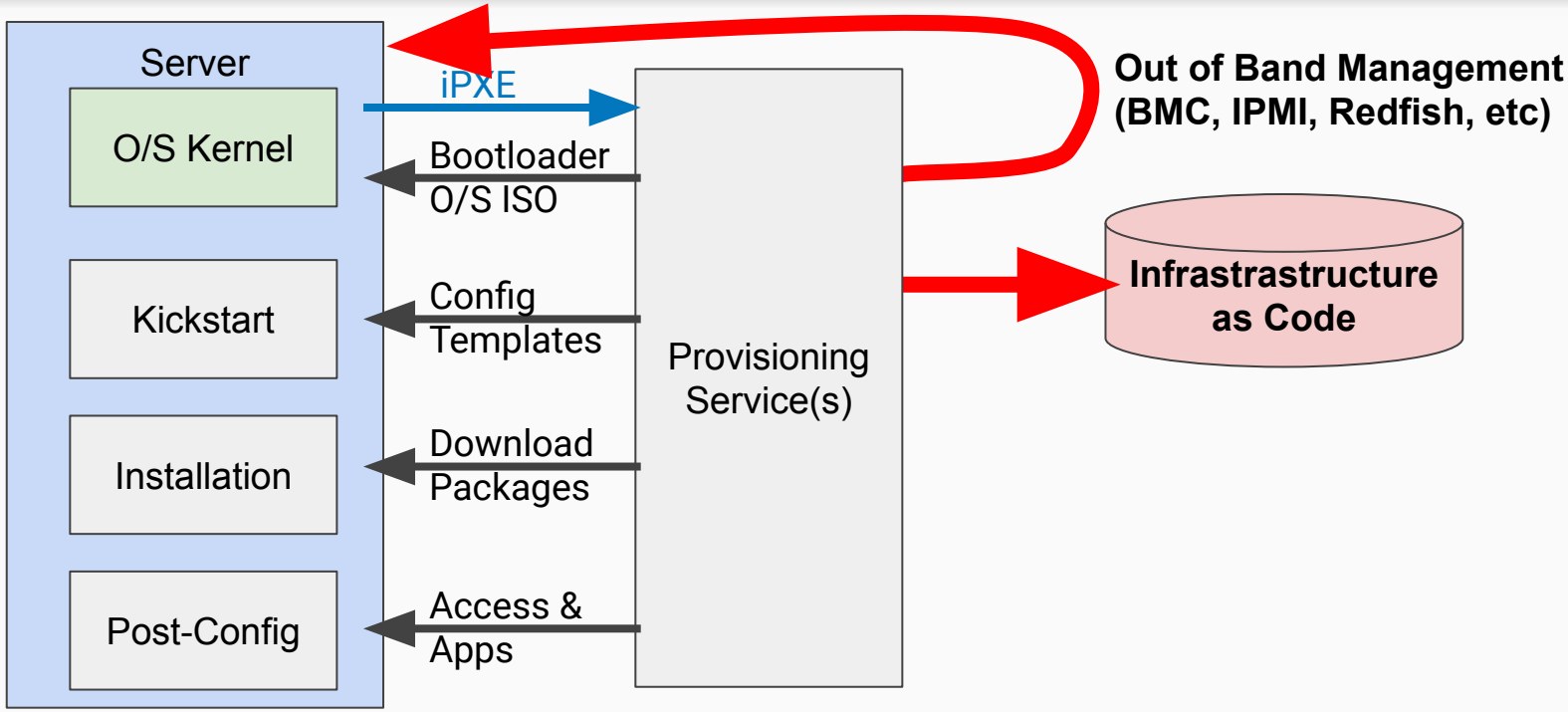
And then you can actually connect to start configuring the system!



Automating Provisioning means Connecting all these steps together



But wait….
There's more to consider!



IaC?
Show us some templates!

Typical PXE Questions

Why is this so fragile?

What about PXE over Wifi?

What about using a VLAN?

Can I dockerize this?

What about setting BIOS & RAID?

How can I make this faster?

How can we simplify that?!!

At RackN, we've been using an in memory operating system, "sledgehammer," based on CentOS.

It's highly optimized to

- Run on nearly any hardware
- Load very quickly
- Collect deep inventory
- Have built-in tools for system tasks like hardware config

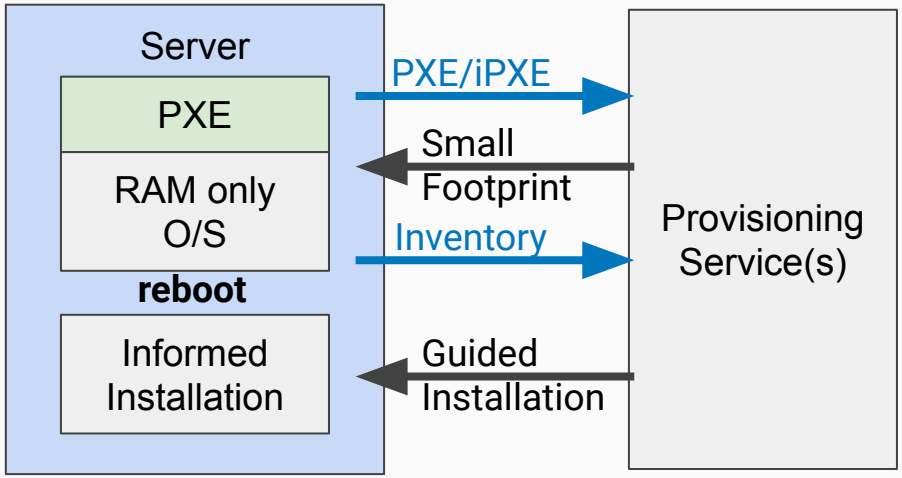
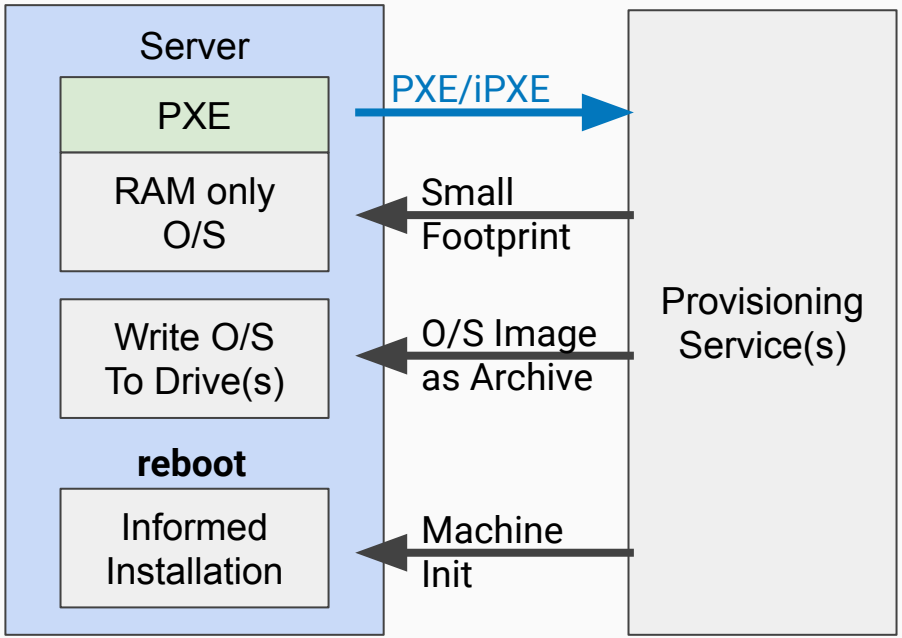
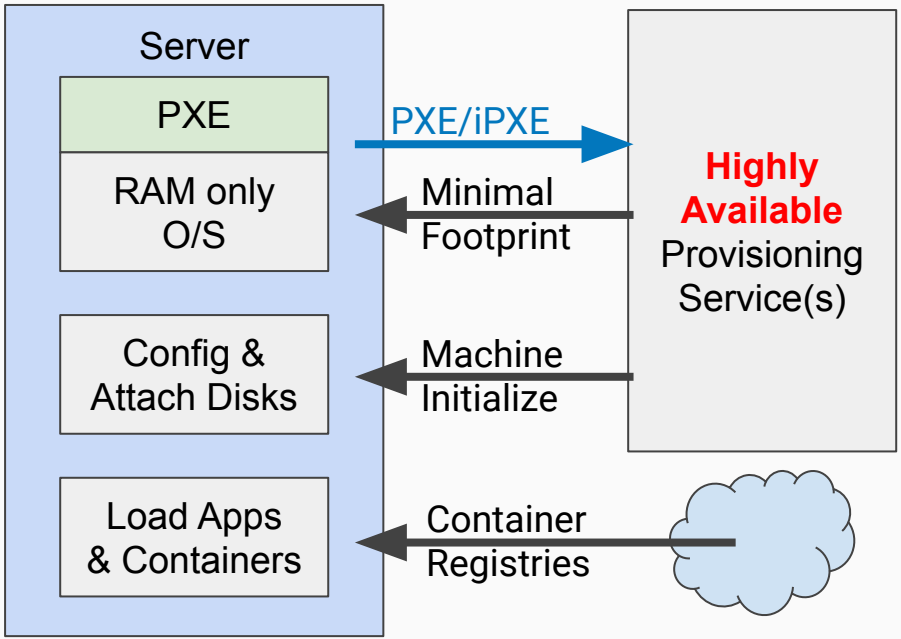


Image Based Deployment *(10x faster!)*

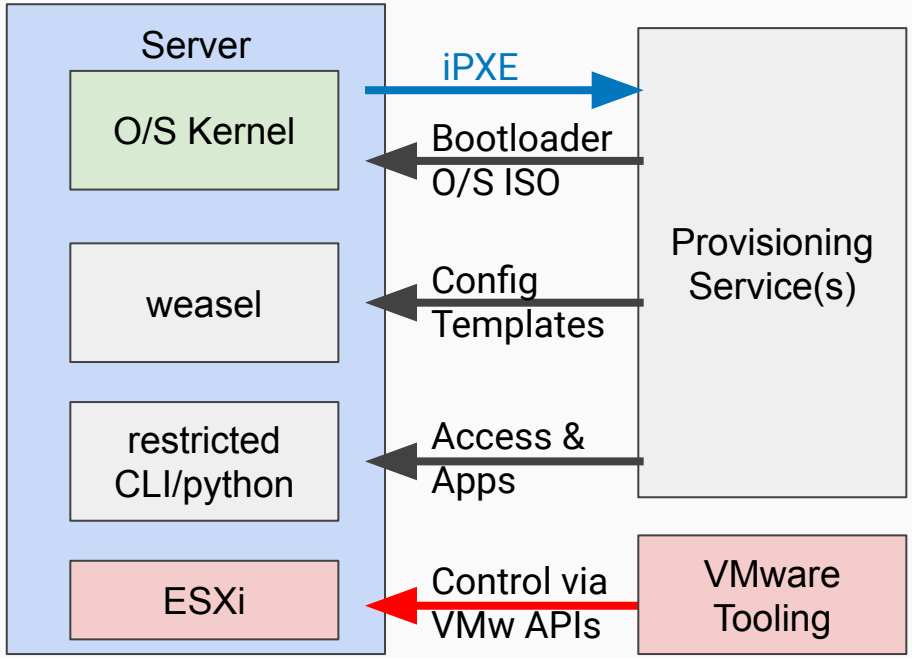


Immutable Provisioning

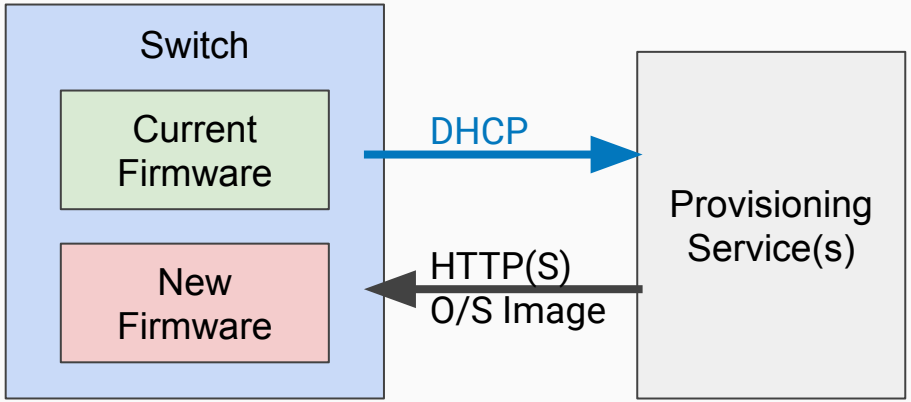


And now…
Advanced Provisioning!

ESXi Provisioning



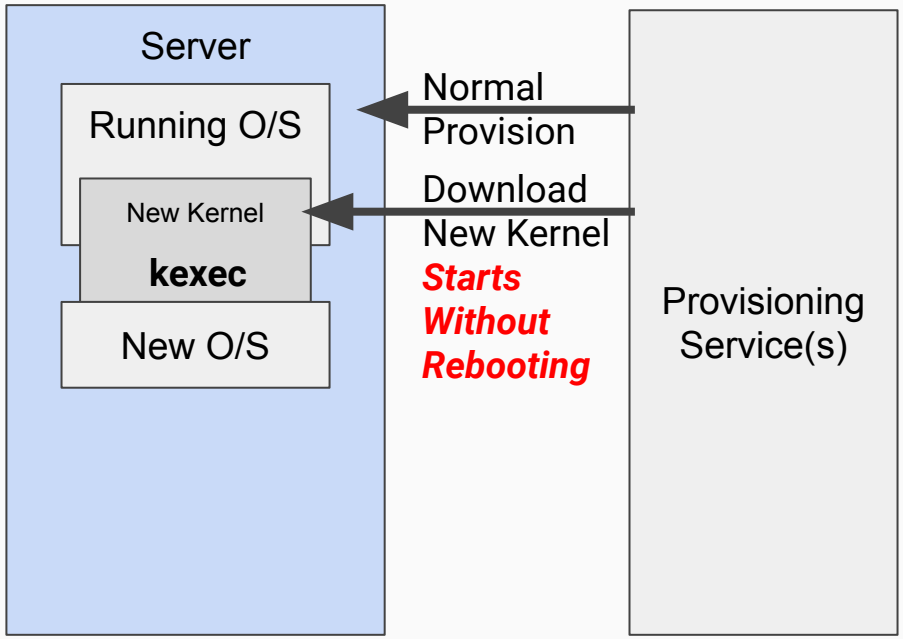
ONIE: Open Network Install Environment



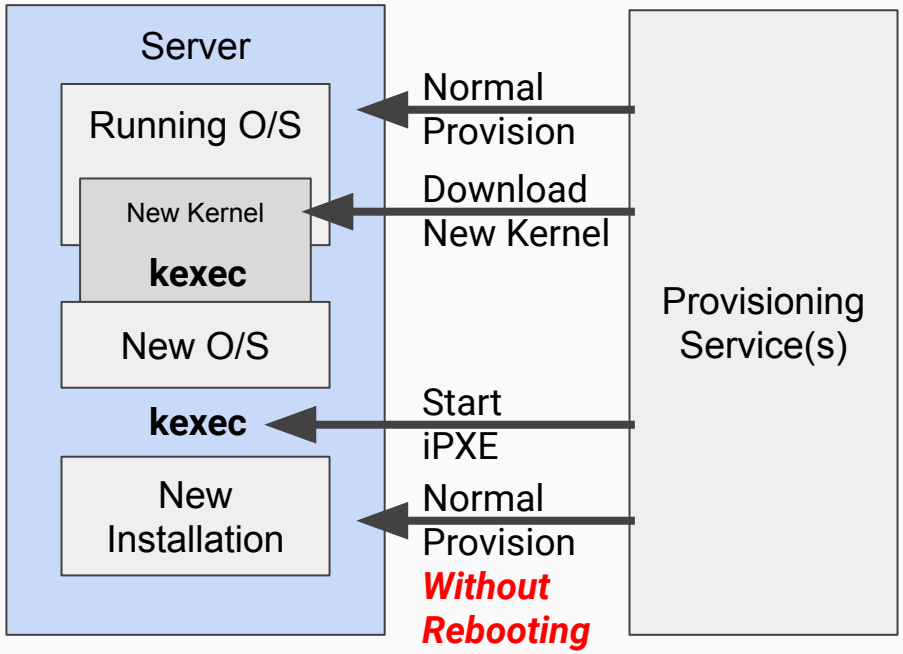
Designed for Embedded Systems where we're replacing the O/S as a complete image.

Does have DHCP options for a startup script.

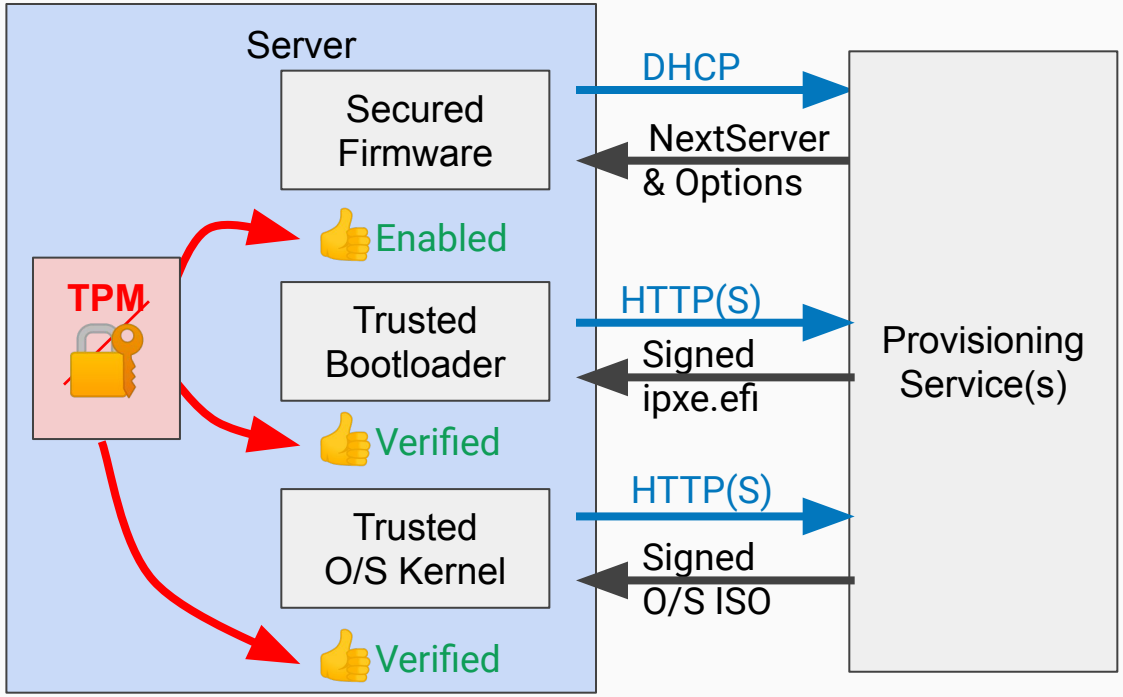
kexec (kernel execute)



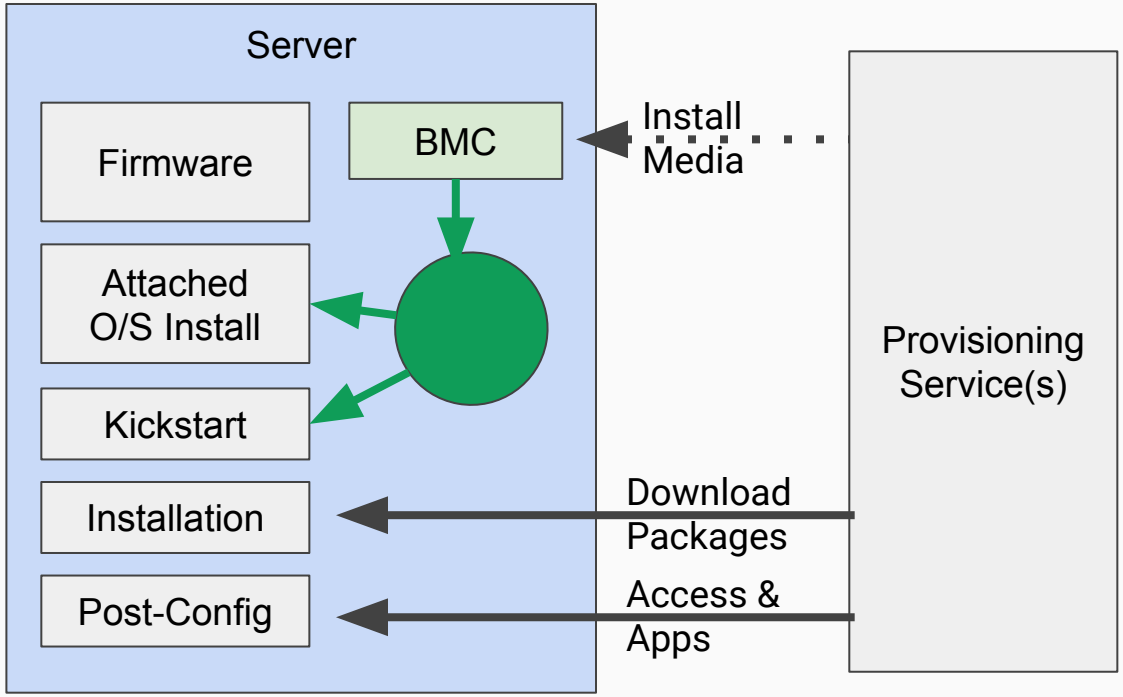
kexec (kernel execute)



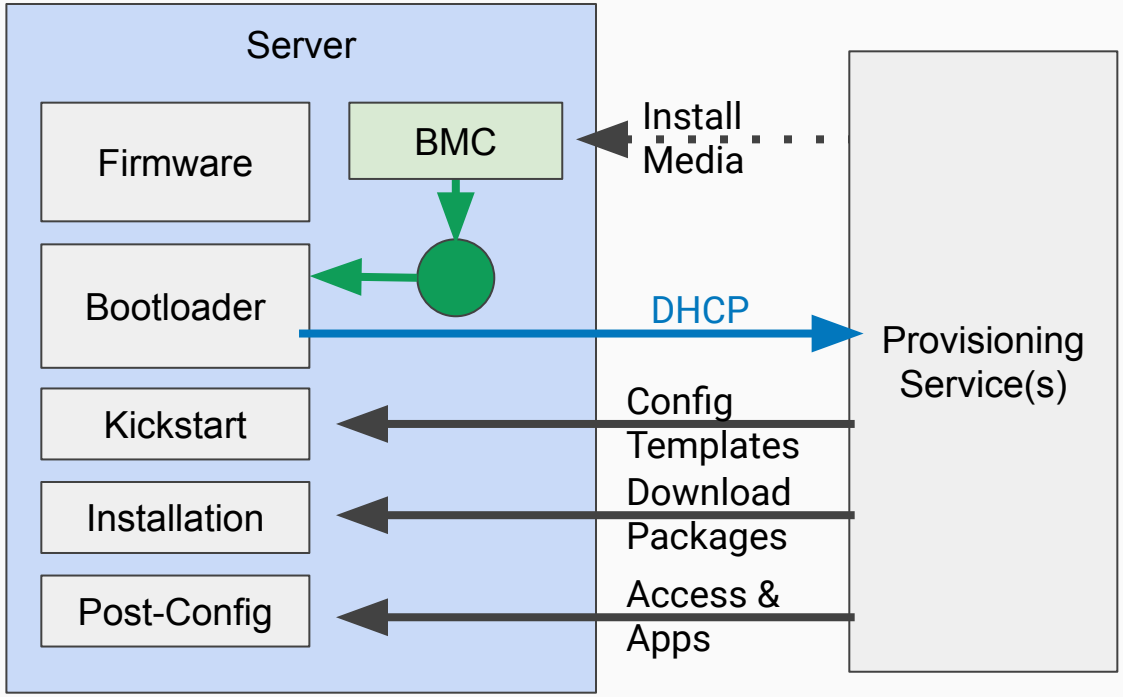
Secure Boot Required SIGNED Bootloaders



BMC Boot option 1



BMC Boot option 2



Thanks!

Contact us:

Rob Hirschfeld, RackN.com

Digital Rebar
Behind the Firewall, Self-Service
Infrastructure as Code

Self-Trials: rebar.digital