POWER up your Linux!

Georgia IBM **P**OWER **U**ser **G**roup
April 21, 2016

Mehboob Mithaiwala, IBM Lab Services

Agenda

- IBM Power Systems - Linux Portfolio
- Why Linux on Power?
- Which Linux distro?
- Power Virtualization Options
- Linux Installation (PowerVM)
- Installing PowerKVM
- Linux Installation (PowerKVM)

IBM Power Systems - Linux Portfolio

Industry-leading performance, resilience and virtualization flexibility for business-critical computing

Power E880

Power E870

Power E850

Power S824
Power S814

Power S824L

Power S822L
Power S812L

Power S822

PowerKVM PowerVM

PowerVM PowerHA

Linux on POWER8 Portfolio – Spans Workloads of all Sizes

redhat SUSE ubuntu Supported by Canonical fedora openSUSE debian

Linux

Designed for big data

Power S812LC

New

- 8, 10 Cores
- 2.92 – 3.32 GHz
- 14 LFF drives
- KVM, Bare metal

Ideal for data in the cloud

Power S822LC

New

- 16, 20 Cores
- 2.92 – 3.32 GHz
- 0 -2 NVidia GPUs
- KVM, Bare metal

Power S812L

- 10, 12 Cores
- 3.02 – 3.42 GHz
- KVM, PowerVM, Bare metal

Power S822L

- 16, 20, 24 Cores
- 3.02 – 4.15 GHz
- KVM, PowerVM, Bare metal

Power S824L

- 16, 20, 24 Cores
- 3.02 – 4.15 GHz
- 0 -2 NVidia GPUs
- KVM, PowerVM, Bare metal

- S822/S822L nodes
- Storwize V7000
- OpenStack Mgmt
- PowerVM, KVM

Power E850

- 16 to 48 Cores
- 3.02 – 3.72 GHz
- PowerVM

Power E870

- 8 to 80 Cores
- 4.02 - 4.19 GHz
- PowerVM

Power E880

- 8 - 192 Cores
- 4 - 4.35 GHz
- PowerVM

'LC line' Linux only Scale-out 'L line' PurePower Power Enterprise IFLs

New

Clients value initial cost of acquisition

Clients value performance and price/performance

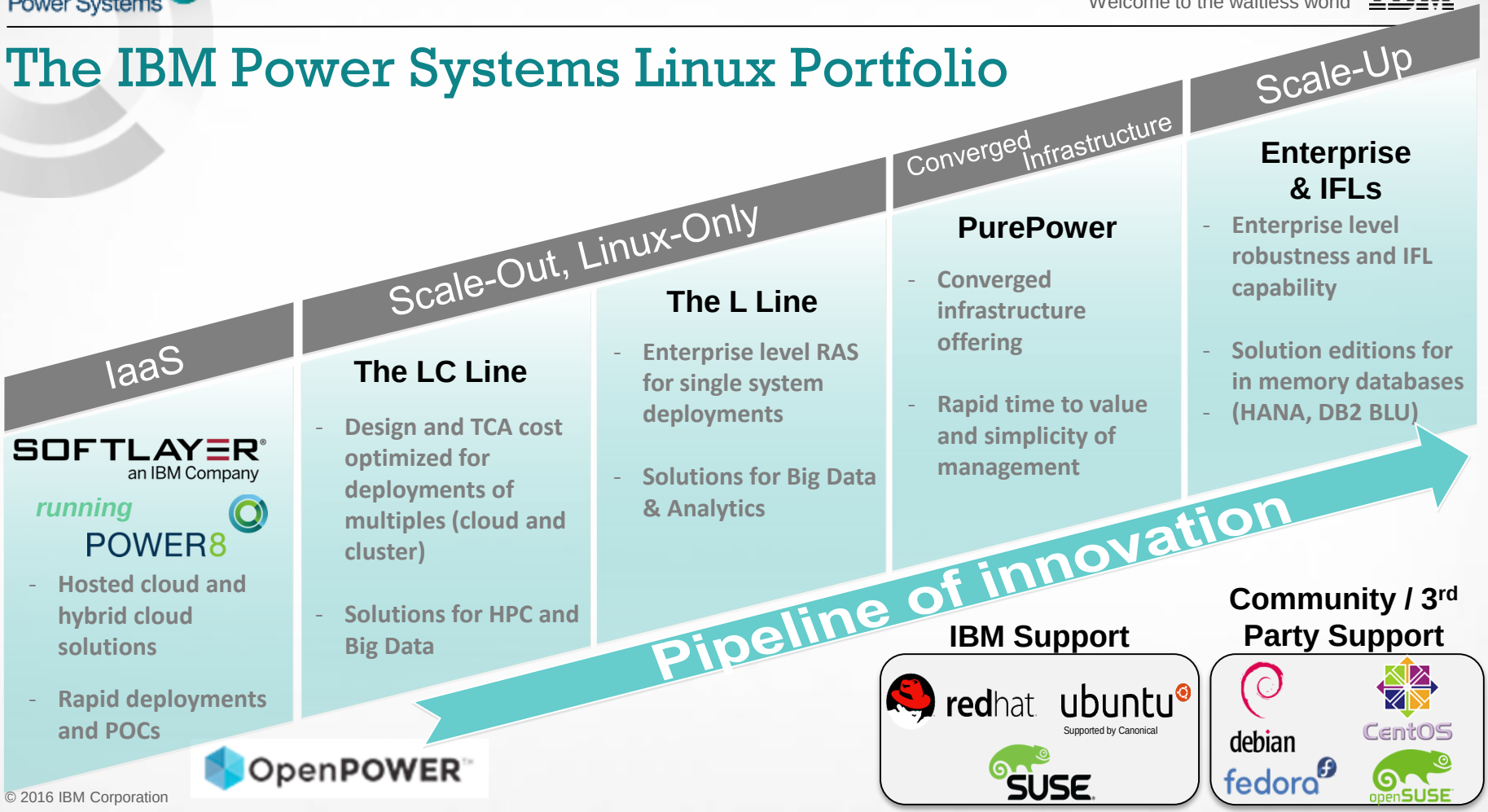
Clients value converged infrastructure

Clients value enterprise class features, robustness

70 PVUs for IBM SW on any Core running Linux

docker openstack PowerVC PowerKVM PowerVM

The IBM Power Systems Linux Portfolio



Why Linux on Power?

POWER8:

Architecture Matters - features that delivers better performance

4X

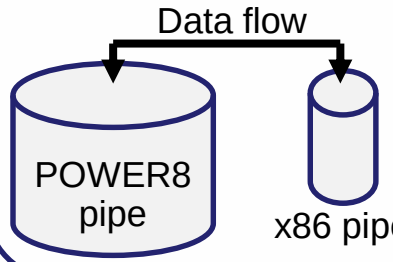
Threads per core*

POWER8 SMT8 | x86 Hyperthread

Parallel Processing

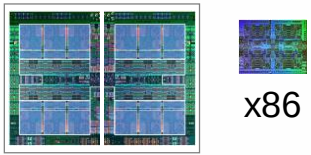
4X

Mem. Bandwidth*



5X

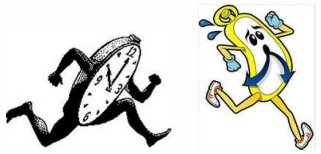
More cache*



POWER8 | x86

1.4 - 2.3X

Clock Frequency*



POWER8 | x86

These design decisions result in best performance for all types of workloads such as:

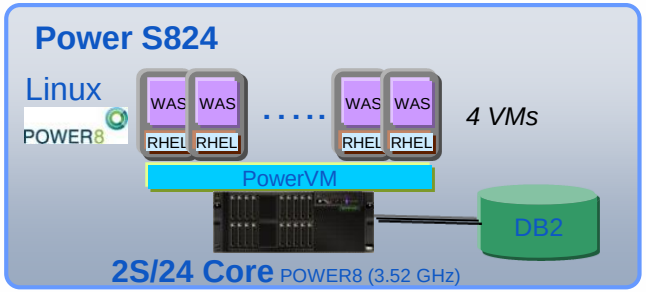
Analytics, Big Data, Java, OLTP, HPC

SMT=Simultaneous Multi-Threading
OLTP = On-Line Transaction Processing
HPC=High Performance Computing

*** POWER8 compared to Haswell EX**

Sources: Haswell EX: http://ark.intel.com/products/84685/Intel-Xeon-Processor-E7-8890-v3-45M-Cache-2_50-GHz
POWER8: http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?subtype=BR&infotype=PM&appname=STGE_PO_PO_USEN&htmlfid=POB03046USEN

POWER8 with Linux:
Delivers Better Performance at LESS cost than x86 Linux

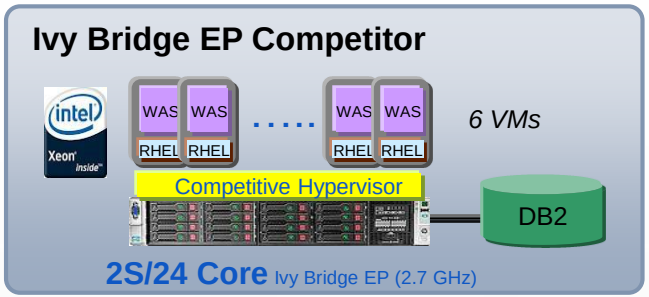


188,184 User Interactions per second

$2.97 per UI per sec

WebSphere on platform
Database off platform

Web Application Online Banking Workload v3.6



84,262 User Interactions per second

$6.37 per UI per sec

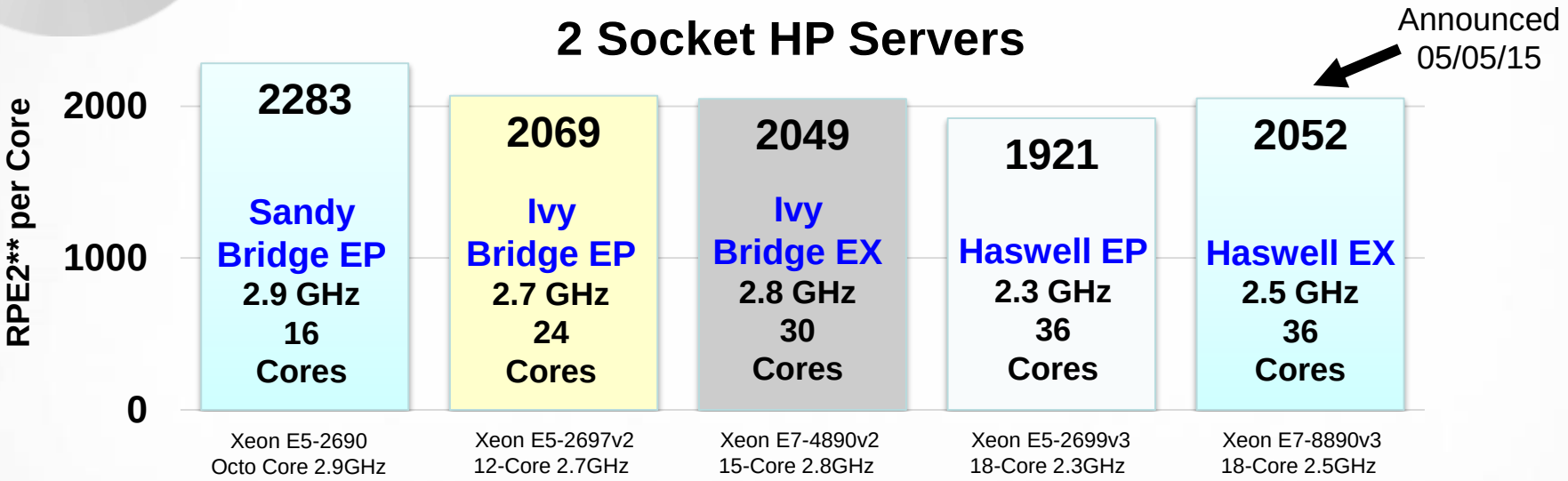
WebSphere on platform
Database off platform

2.2x Faster

53% Lower cost

This is an IBM internal study designed to replicate a typical IBM customer workload usage in the marketplace. It consists of a POWER8 S824 with 24 cores, 256GB Memory, 3.52 GHz, RHEL 7.0, WAS 8.5.5.2, DB2 9.7, JDK 7.0 FP1 compared to an Ivy Bridge EP 24 cores, 256GB Memory, 2.7 GHz, RHEL 6.5, WAS 8.5.5.1, DB2 9.7, JDK 7.0 FP1. The results were obtained under laboratory conditions, and not in an actual customer environment. IBM's internal workload studies are not benchmark applications, nor are they based on any benchmark standard. As such, customer applications, differences in the stack deployed, and other systems variations or testing conditions may produce different results and may vary based on actual configuration, applications, specific queries and other variables in a production environment. Prices, where applicable, are based on published US list prices for both IBM and competitor, and the Total Cost of Acquisition (TCA) includes the list HW and SW prices and 3 years of service & support which is then divided by the number of transactions to get $ per user interaction per second.

The last three generations of x86 architecture:
Performance per core has remained flat



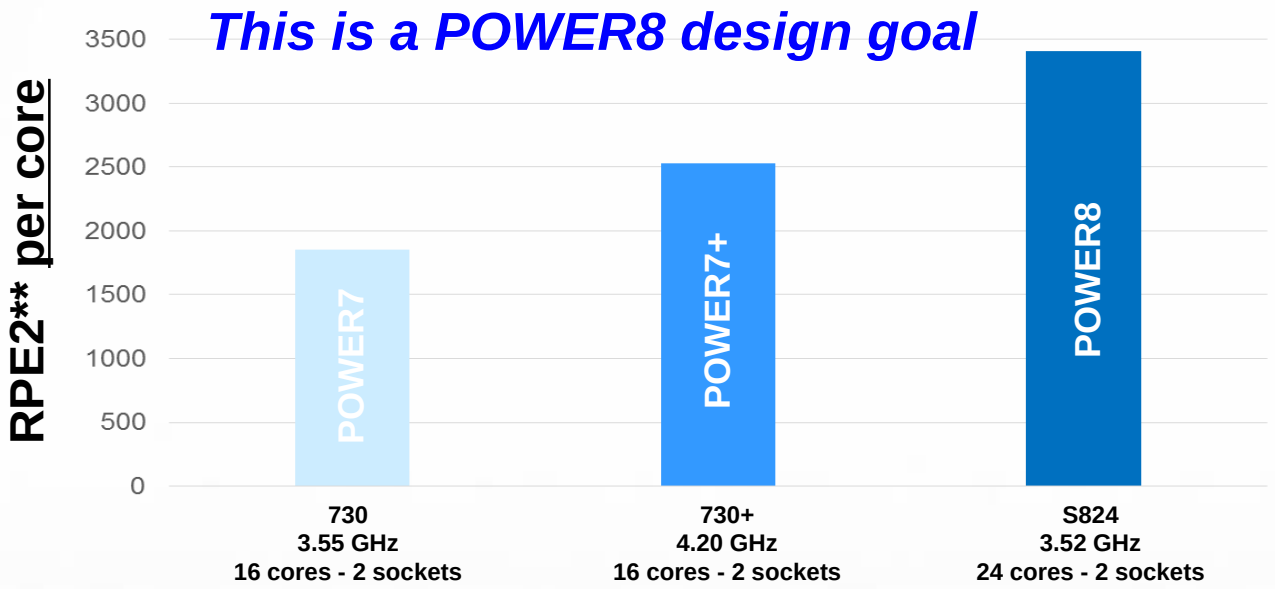
The number shown is best in each category (sockets and number of cores)

**Gartner RPE2 Details:
http://www.gartner.com/technology/research/RPE2-methodology-details.jsp

RPE2** numbers are derived from the following six benchmark inputs:

SAP SD Two-Tier, TPC-C, TPC-H, SPECjbb2006 and two SPEC CPU2006 components

POWER architecture:
Performance per core is increasing!



The number shown is best in each category (sockets and number of cores)

**Gartner RPE2 Details:
http://www.gartner.com/technology/research/RPE2-methodology-details.jsp

RPE2** numbers are derived from the following six benchmark inputs:

SAP SD Two-Tier, TPC-C, TPC-H, SPECjbb2006 and two SPEC CPU2006 components

Why is this important?

POWER8 design focus and results

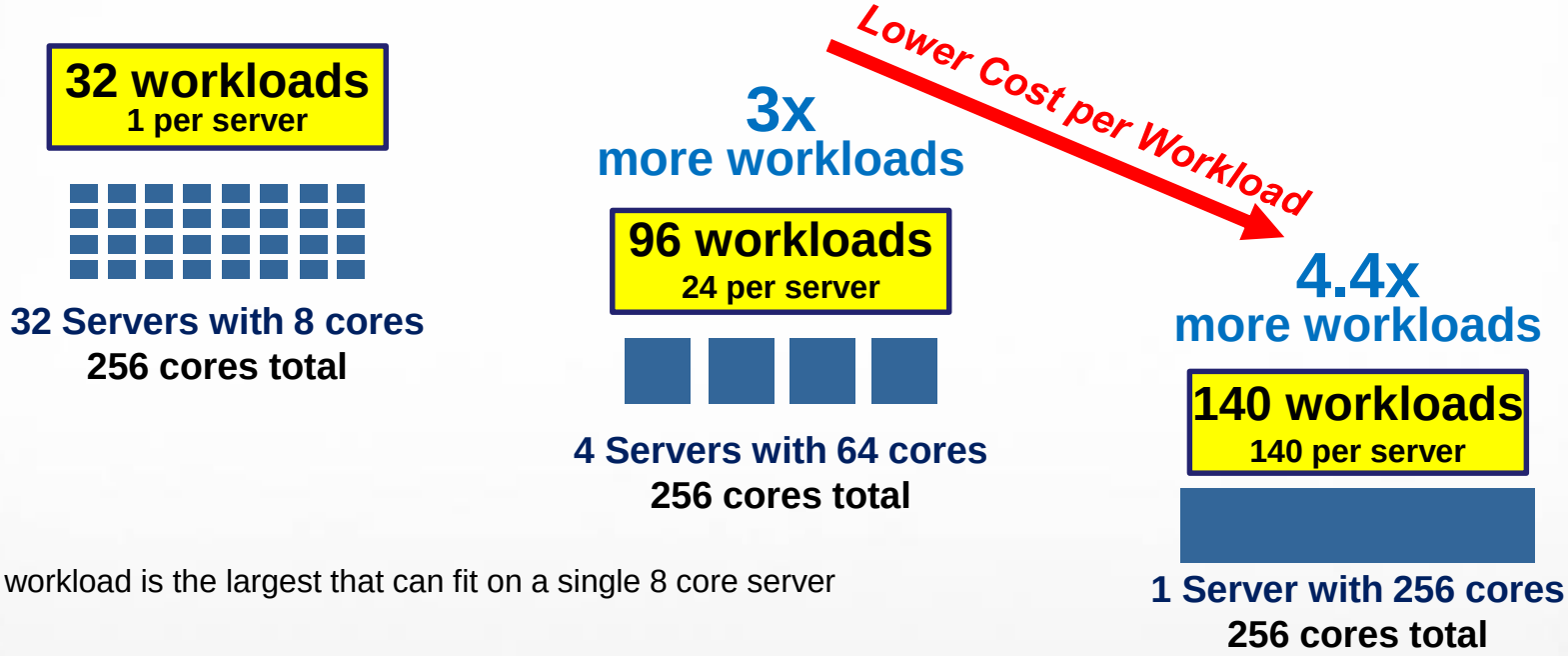
- More threads - More memory bandwidth - More cache - Higher clock frequencies	**- Fastest performance for all types of workloads** **- Ready to address larger workloads** (Analytics, Big Data)
Higher Performance per core	- Lower software costs and Total Cost of Acquisition (TCA) - Fewer servers, lower support costs

x86 Design Focus and Results

- More sockets and cores per server	- Faster overall **SERVER** performance - Higher cost per server
Performance per core remains the same	- Overall solution cost is higher

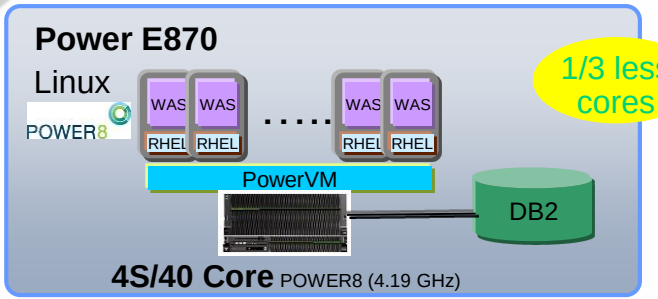
Statistical Multiplexing drives higher utilization, more workloads and lower cost per workload on large servers

A single virtualized server with a large pool of shared processors can run more workloads than several smaller servers with the same total number of processors



Each workload is the largest that can fit on a single 8 core server

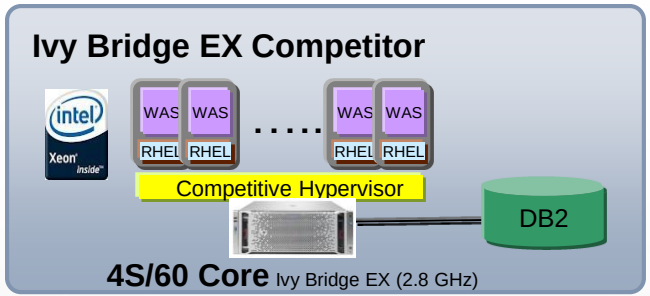
E870 with PowerVM supports 2.3x more workloads with 1/3rd fewer cores than competitive hypervisor



1/3 less cores

72 Workloads
$21,632 per Workload

Web Application
Online Banking Workloads each running
1960 User Interactions/sec



31 Workloads
$61,061 per Workload

WebSphere on platform, Database off platform for both tests

2.3x More Workloads

64% Lower cost per Workload

This is an IBM internal study designed to replicate a typical IBM customer workload usage in the marketplace. It consists of a POWER8 E870 with 40 cores, 4.19 GHz, RHEL 7, WAS 8.5.5.2, DB2 10.5, JDK 7.0 FP1. compared to an Ivy Bridge EX 60 cores 2.8 GHz, RHEL 7, WAS 8.5.5.2, DB2 10.5, JDK 7.0 FP1 and a Competitive hypervisor. The results were obtained under laboratory conditions, and not in an actual customer environment. IBM's internal workload studies are not benchmark applications, nor are they based on any benchmark standard. As such, customer applications, differences in the stack deployed, and other systems variations or testing conditions may produce different results and may vary based on actual configuration, applications, specific queries and other variables in a production environment.

IBM has a Strong History of Collaboration on Open Technology

+ IBM

15+ Years of Collaboration on Open Source

1999 2000 2001 2007 2007 2011 2011 2012 2013 2015

200+
IBM software products

500+
patents donated

600+
developers

Fueling the OpenPOWER Community

OpenPOWER™

Accelerates software innovation

- Over **1,900 Linux ISVs** developing on Power
- 50 IBM Innovation Centers
- Compelling PoCs
- Support for little endian applications

HPC

CHARMM
GROMACS
NAMD
AMBER
RTM
GAMESS
WRF
HYCOM
HOMME
LES
MiniGhost
AMG2013
OpenFOAM
miniDFT
CTH
BLAST
Bowtie
BWA
FASTA
HMMER
GATK
SOAP3
STAC-A2
SHOC
Graph500
Ilog

Cloud

openstack
TOMCAT
Apache
Chef
STORIX
REFLEXIS
DIASOFT
OneView Commerce
POWERSENSE
IFS
KANA.
Fis
YUCHENG™ Technologies Limited
sky solutions
betasystems
EXIGEN.
ProjectWare
ALPHINAT

Big Data & Machine Learning

hadoop
PostgreSQL
MariaDB
redislabs
SUGARCRM
gpudb
Datameer
Spark
neo4j
EDB ENTERPRISEDB
AsiaINFO
InterSystems
CROSSVIEW.
POLARIS
ZAO
Valogix

Mobile Enterprise

zend
亿阳信通 BOCO Inter-Telecom
php
CSC
RabbitMQ Open Source Enterprise Messaging
Kingdee 金蝶，企业管理专家
SYBASE
Fis
NARI 南瑞集团公司 NARI GROUP CORPORATION
FINEOS
ALPHINAT
comptel
Corent
TEMENOS The Banking Software Company
KUTIR

Major Linux Distros

ubuntu
suse
redhat
FreeBSD
Linux
Power
PowerVM

Which Linux distro?

Linux support for IBM Power Systems

- Same source and distribution release schedules as x86. No special IBM code.
- Simplified x86 application migration with little endian distributions.
- Enterprise support for all three from IBM or distributors.

redhat fedora

- RHEL 7 (BE/LE)
  - POWER8 (native mode) and POWER 7/7+ at GA (BE)
  - Available June 2014
  - LE distribution (7.1)
  - Bare metal on L/LC models (7.2)

New

- RHEL 6
  - POWER8 supported with U5 (P7-compatibility mode)
  - Full support of POWER6 and POWER7 (native mode)
- Fedora
  - Fedora 16 was first release to re-launch POWER
  - Fedora 20 has POWER8 support
  - Docker in Rawhide, target Fedora 23

New

- Supported add-ons
  - JBoss
  - High Performance Network Add-on

SUSE openSUSE

- SLES 12 (LE)
  - POWER8 (native mode)
  - Technology preview of KVM host support
  - Bare metal support target of SP1 ('L' models only)

New

- SLES 11 (BE)
  - POWER8 with SP3 (P7-compatibility mode)
  - POWER7+ encryption, RNG accelerators with SP3
  - Full support of POWER7 (native mode)
- openSUSE
  - openSUSE 12.2 re-launched for IBM POWER
  - POWER in openSUSE 13.2
  - Docker enabled in Tumbleweed
  - Power enablement in Leap 42.1 underway

New

- Supported add-ons
  - SUSE Linux Enterprise HA

ubuntu debian

- Ubuntu 15.10 (LE)
  - S822LC enablement
  - CAPI enablement

New

- Ubuntu 15.04 (LE)
  - Docker enablement
  - FPGA support
- Ubuntu 14.10 (LE)
  - S824L GPU enablement
- Ubuntu 14.04 (LE)
  - POWER8 enabled (native mode)
  - No official support for POWER7+ and older systems
  - 64-bit only
  - KVM hosting enabled (.2)
  - S812LC enabled (.3)

New

- Supported add-ons
  - JuJu Charms
  - MaaS (Metal as a Service)
  - Landscape
- Debian
  - Stable supports POWER8 LE in Jessie (8) release

Power Virtualization Options

Power Virtualization Options

PowerKVM

Q2 2014
Initial Offering

PowerKVM provides an open source choice for Power Virtualization for Linux workloads. Best for clients that aren't familiar with Power and Linux centric admins.

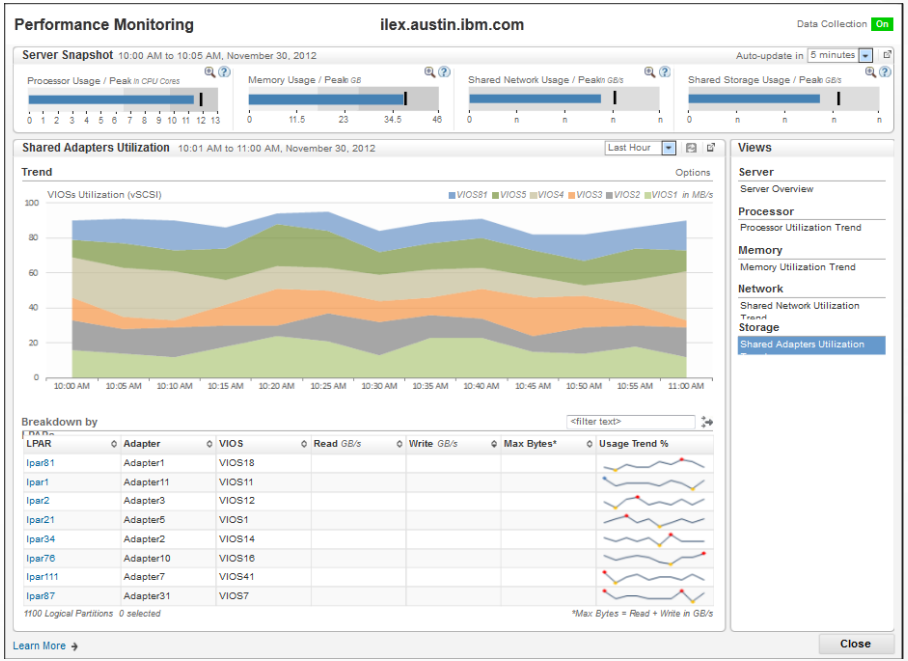
PowerVM is Power Virtualization that will continue to be enhanced to support AIX, IBM i, as well as Linux workloads

PowerVM

2004 Initial Offering

PowerVM: Virtualization without limits

- New monitoring views help simplify performance and capacity management
- Support for latest POWER8 servers and capabilities
- SR-IOV NIC support for enhanced network virtualization

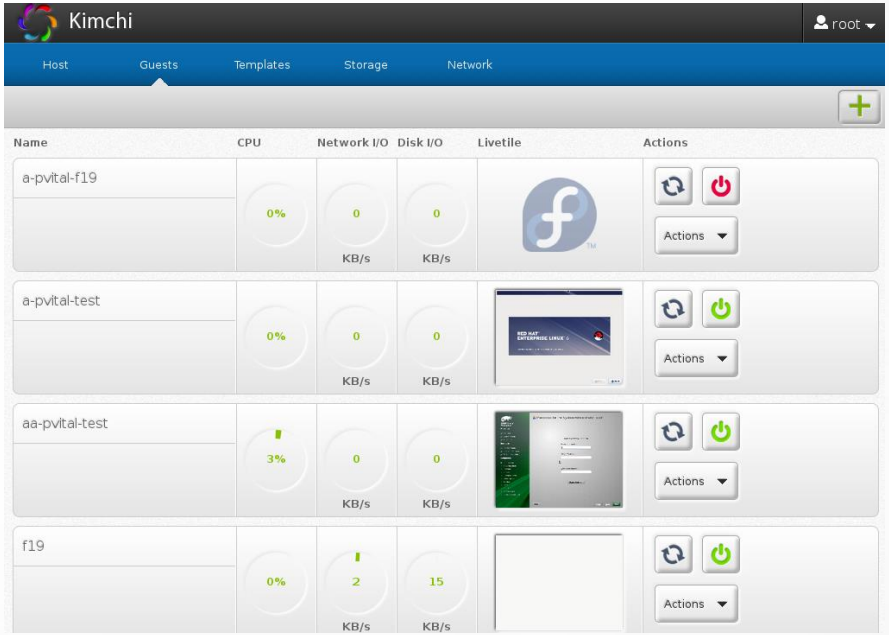


PowerVM

AIX for Business Linux

PowerKVM: open virtualization for scale-out servers

- Kernel-Based Virtual Machine (KVM) virtualization targets new Linux workloads
- Provides simplicity and familiarity for VMware and KVM x86 Linux admins
- Enables cloud providers to integrate Linux on Power into OpenStack environments
- Managed by PowerVC or open source tools such as Kimchi
- Exploits POWER8 micro-threading for performance efficiency



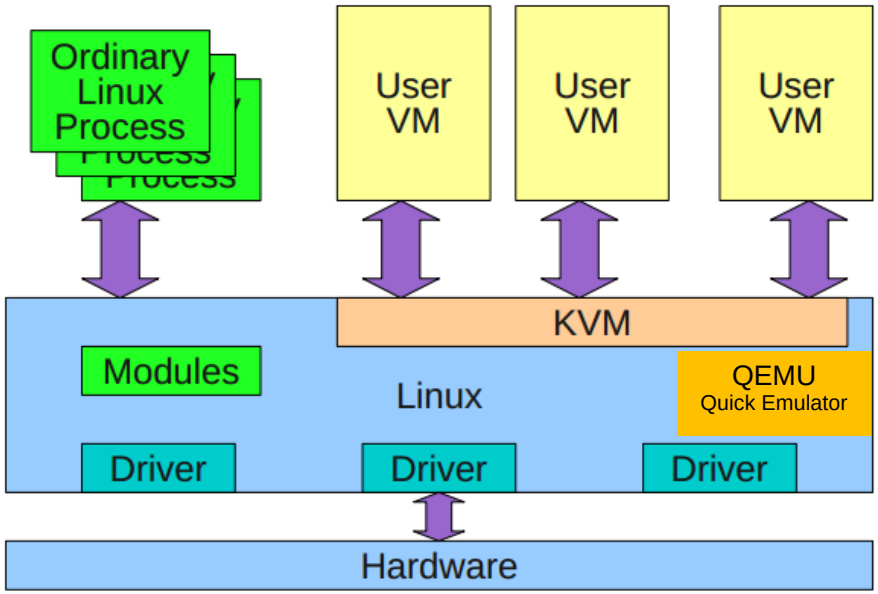
PowerKVM

redhat. SUSE ubuntu Supported by Canonical

Power Firmware – OPAL (Open Power Abstraction Layer)

- PowerKVM runs over a different firmware stack than PowerVM
  - New system firmware layer developed (OPAL)
- Hardware Management Console (HMC) does not play a role
  - Console access using open source tool called **ipmitool**
- System Management Services (SMS) is replaced by Petitboot
  - Petitboot is a platform independent bootloader based on the Linux **kexec** warm reboot mechanism

KVM Architeture Overview

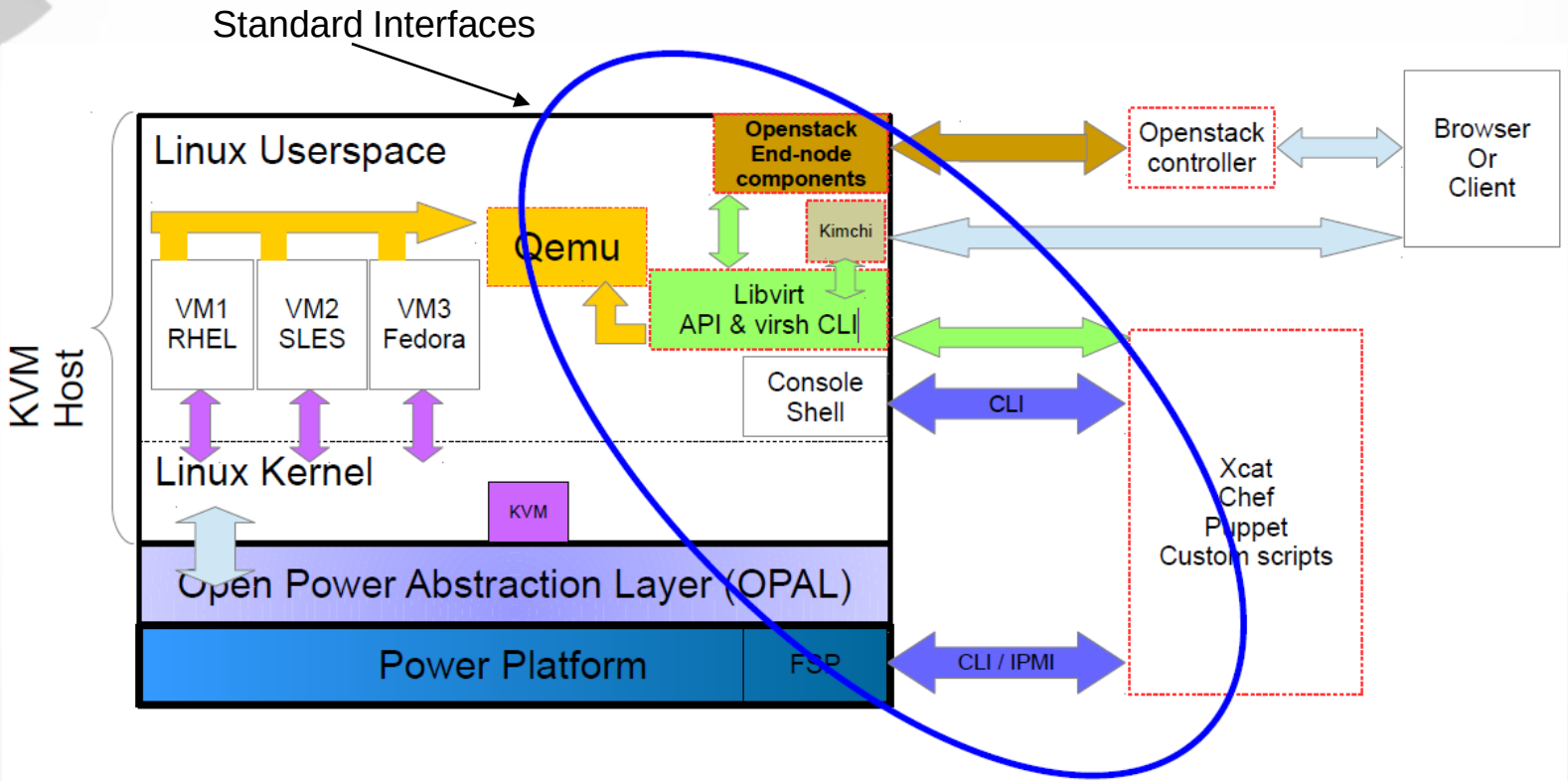


KVM - Kernel-Based Virtual Machine Loadable Kernel Module that provides server virtualization for Memory and CPU
QEMU – Quick Emulator virtualizes I/O

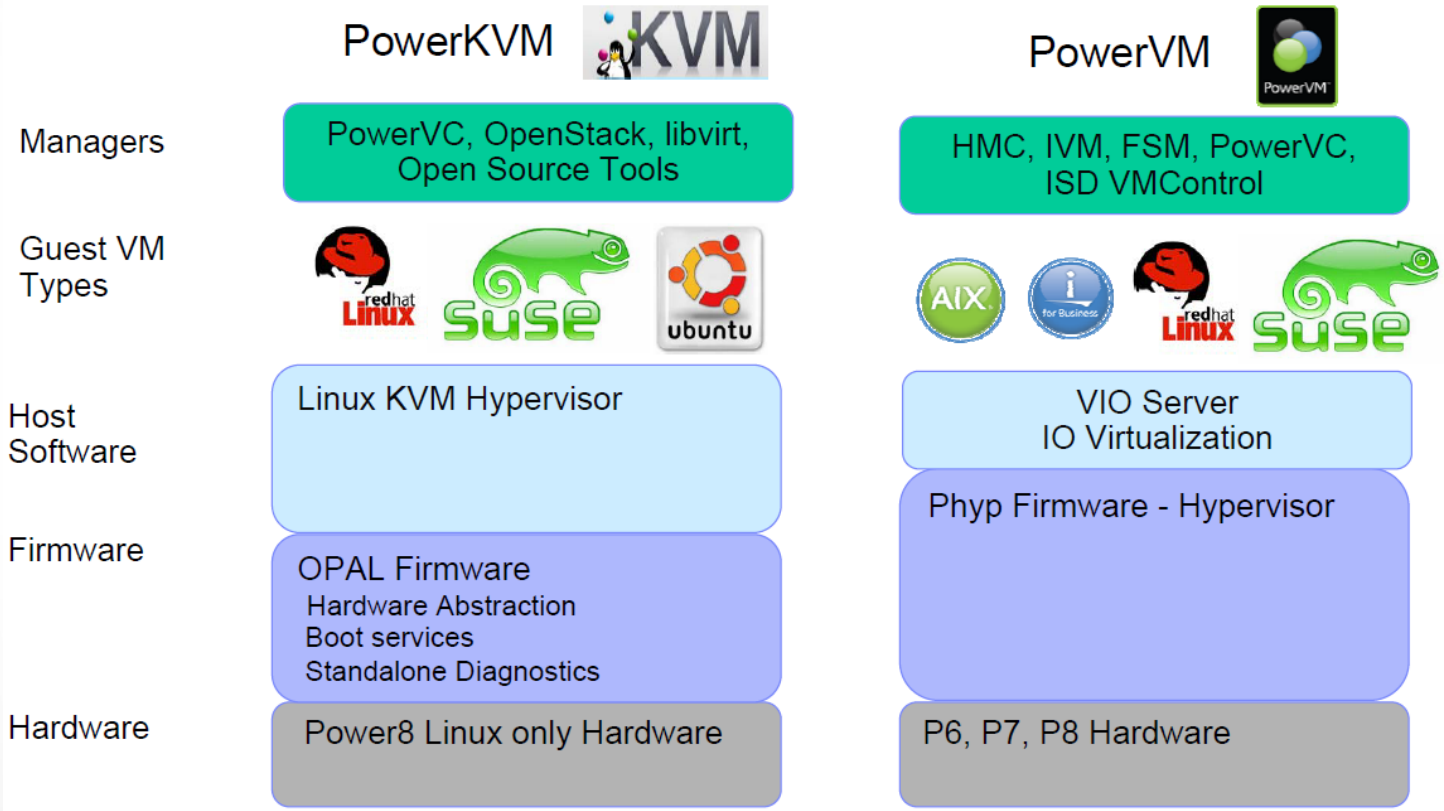
Guests run a normal Linux Process scheduled by the Linux Scheduler

Originally designed for x86 and uses hardware assists. Intel VT, AMD-V

KVM on Power - PowerKVM



Component Overview: PowerKVM vs. PowerVM



Terminology comparison

IBM PowerKVM	KVM on x86	IBM PowerVM
Guest, virtual machine	Guest, virtual machine	LPAR
Hypervisor, Host	Hypervisor, Host	Hypervisor
Flexible Service Processor (FSP)	Integrated Management Module (IMM)	Flexible Service Processor (FSP)
Open Power Abstraction Layer (OPAL)	Unified Extensible Firmware Interface (UEFI) and BIOS	PowerVM hypervisor driver (pHyp)
KVM Host User space (QEMU)	KVM Host User space (QEMU)	Virtual IO Server (VIOS)
Kimchi and virsh	Kimchi and virsh	HMC
Intelligent Platform Management Interface	Intelligent Platform Management Interface	Hardware Management Console
Kernel Same-page Merge (KSM)	Kernel Same-page Merge (KSM)	Active Memory Deduplication
zswap	zswap	Active Memory Expansion (AME)
SLOF	SeaBIOS	Open Firmware, SMS
Preboot eXecution Environment (PXE)	Preboot eXecution Environment (PXE)	bootp and tftp, NIM
Virtio drivers, ibmvscsi, ibmveth	Virtio drivers	ibmvscsi, ibmveth
Hot plug	Hot plug	DLPAR

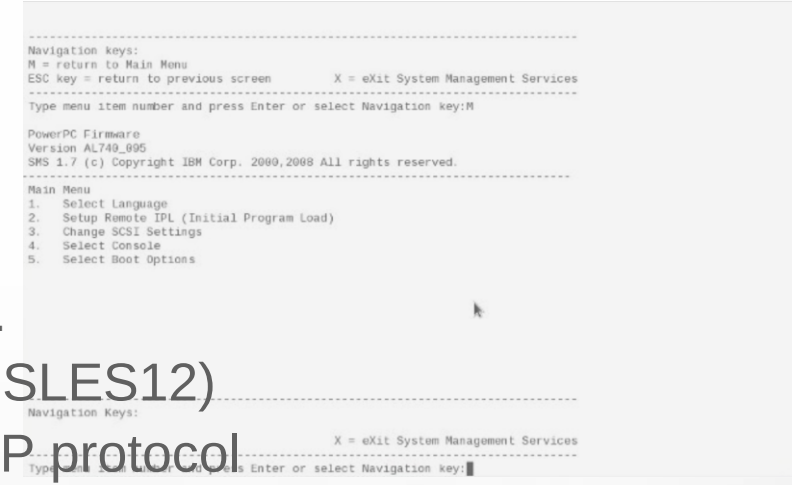
Linux Installation (PowerVM)

Installation of a Linux OS
Either natively or into a partition / virtual machine

- DVD (physical or virtual optical)
- To make life easier:
  - Use VNC installation mode for headless nodes
    - **boot: linux vnc**
    - Needs network connection and assumes DHCP available, otherwise specify network settings also
  - Automated installation using configuration file available on network
    - Kickstart - RHEL or Ubuntu
    - Preconfiguration file - Ubuntu Installer
    - AutoYAST (rules and profiles) - SUSE
- Network

Linux is Linux
(but there are some differences related to the architecture)

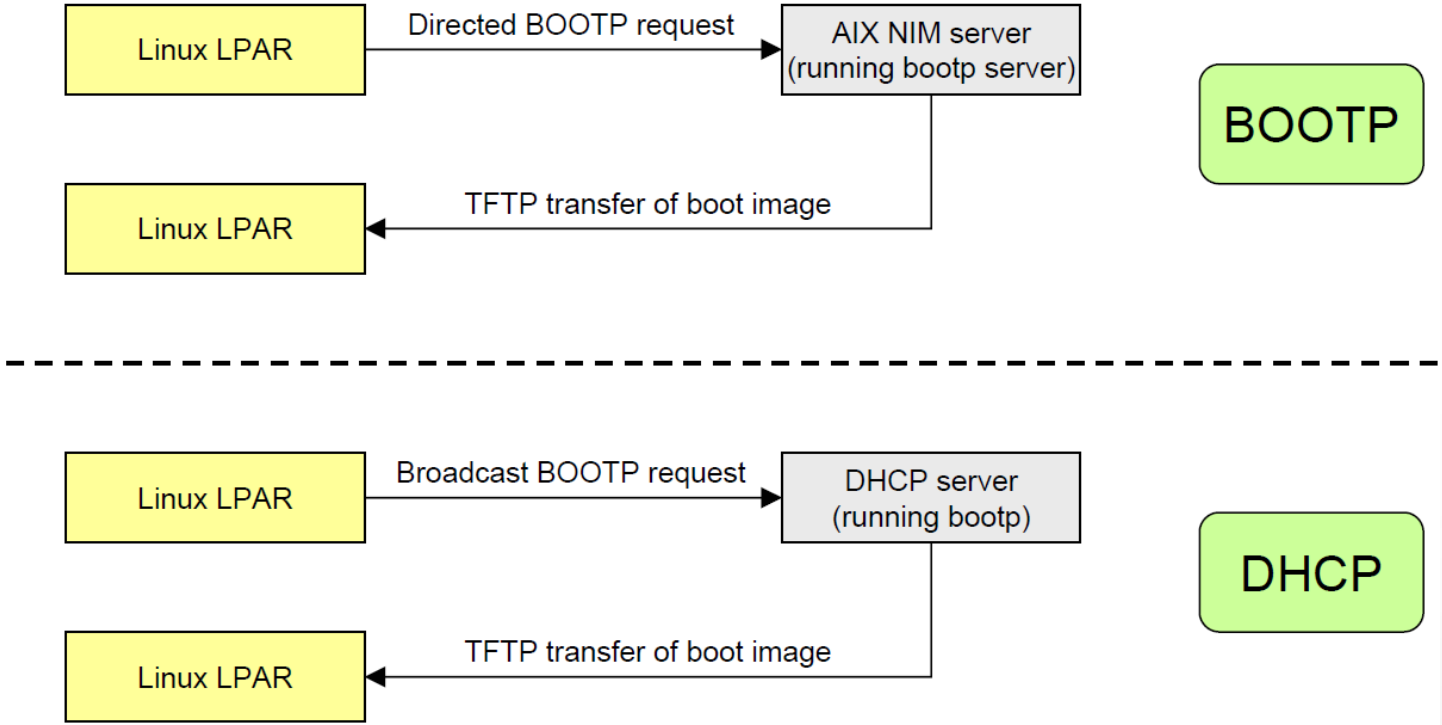
- Partitioning:
  - additional PPC PReP Boot partition which holds bootloader
  - MBR is present, but only holds partition table
- System firmware
  - System Management Services (SMS)
  - Press 1 at boot time to enter, x to exit
  - Setup Remote IPL
  - Select Boot Order, bootp network boot, ...
- Bootloader: yaboot (grub2 for RHEL71, SLES12)
- No PXE boot, network boot uses BOOTP protocol
- Additional value-add packages: IBM Service and Productivity Tools



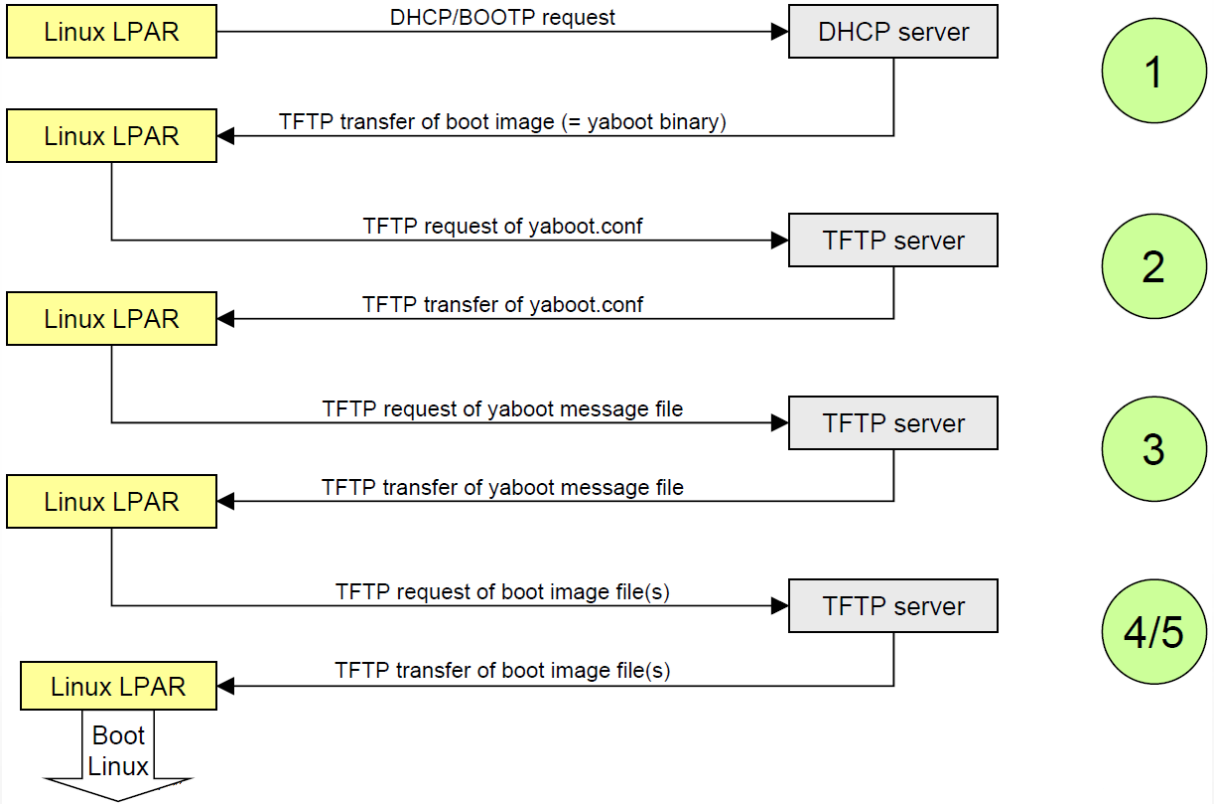
Netbooting Linux on Power

- Netbooting Linux on Power uses – same protocols as AIX:
  - BOOTP is an IP protocol that informs a computer of its IP address and where on the network to obtain a boot image
  - TFTP (Trivial File Transfer Protocol) is used to serve the boot image to the client
- Two basic approaches:
  - AIX NIM server setup as Linux install server
    - Use a directed bootp request
    - Does not require you to know the MAC address of the network boot adapter
  - (Linux) DHCP server
    - Use a broadcast bootp request
    - DHCP server must support BOOTP protocol
    - Requires you to know the MAC address of the network boot adapter

Directed BOOTP vs. broadcast BOOTP request



Netbooting yaboot (1/2)



Netboot yaboot (2/2)

- Linux LPAR sends out directed BOOTP or broadcast DHCP request
- DHCP server answers BOOTP/DHCP request and transfers boot image back to Linux LPAR via TFTP
- Linux LPAR executes transferred boot image (= yaboot binary)
- Yaboot now requests via TFTP its config file yaboot.conf:
  - Depending on Linux distribution client-specific yaboot.conf can be located in different locations (i.e., directories)
  - TFTP server transfers yaboot.conf back to Linux LPAR
- If “yaboot.conf“ contains a statement “message=yaboot.txt“ then the yaboot message file is requested via TFTP and transferred back from the TFTP server
- Linux LPAR now requests the boot image file(s)
- TFTP server transfers boot image files(s) back to Linux LPAR

Netbooting Linux on Power – AIX NIM bootp setup

```
/etc/bootptab:
# Legend:
#   first field -- hostname (may be full domain name and probably should be)
#   hd -- home directory
#   bf -- bootfile
#   sa -- server IP address to tftp bootfile from
#   gw -- gateways
#   ha -- hardware address
#   ht -- hardware type
#   ip -- host IP address
#   sm -- subnet mask
#   tc -- template host (points to similar host entry)
#   hn -- name switch
#   bs -- boot image size
#   dt -- old style boot switch
js21-5-rhel5:bf=/tftpboot/js21-5-rhel5:ip=10.0.21.52:ht=ethernet:sa=10.0.0.8:sm=255.255.0.0:
js21-6-sles10:bf=/tftpboot/js21-6-sles10:ip=10.0.21.62:ht=ethernet:sa=10.0.0.8:sm=255.255.0.0:
```

```
root@nim:/tftpboot> ls -la js*
lrwxrwxrwx    1 root     system           19 Mar 03 2013  js21-5-rhel5 -> rhel5u8-netboot.img
lrwxrwxrwx    1 root     system           14 Mar 02 2013  js21-6-sles11 -> sles11-sp2-inst64
```

Netbooting Linux on Power – Linux DHCP server

```
[root@ppclinux:/tftpboot]# ls -al
drwxr-xr-x  14 root root      640 Apr 19 00:29 .
drwxr-xr-x  26 root root      888 Apr 16 21:32 ..
lrwxrwxrwx   1 root root       10 Apr  7 13:32 LPAR00001 -> yaboot.RHEL65
drwxr-xr-x   2 root root      104 Apr  7 14:09 RHEL6U5
-rw-r--r--   1 root root     2916 Apr  7 14:40 yaboot.conf
lrwxrwxrwx   1 root root       11 Apr  7 13:51 yaboot.conf-36-9a-c0-00-40-0b -> yaboot.conf
-rw-r--r--   1 root root   263760 Apr  7 13:27 yaboot.RHEL65
-rw-r--r--   1 root root      213 Apr  7 13:58 yaboot.txt

[root@ppclinux:/tftpboot]# ls –al RHEL6U5
./RHEL6-U5:
-rw-r--r--   1 root root 28205108 Apr  7 14:09 initrd.img
-rwxr-xr-x   1 root root 17020184 Apr  7 14:09 vmlinuz
```

Netbooting Linux on Power – Linux DHCP server

/etc/dhcpd.conf:

```
option domain-name "[redacted].ibm.com";
option domain-name-servers 10.0.0.8;
option routers 10.0.0.1;
option ntp-servers 10.0.0.8;
ddns-update-style none;
ignore unknown-clients;
allow bootp;
subnet 10.0.0.0 netmask 255.255.0.0 {
  range 10.0.56.230 10.0.56.250;
  default-lease-time 86400;
  max-lease-time 604800;
}
host LPAR00001 {
  hardware ethernet 36:9A:C0:00:40:0B;
  filename "LPAR00001";
  fixed-address 10.0.56.4;
  next-server 10.0.0.51;
}
```

Netbooting Linux on Power – Linux DHCP server

```
[root@dhcp-server:/tftpboot]# cat yaboot.conf
## This yaboot.conf is for netbooting different Linux distributions

message=yaboot.txt

timeout=100

default=sles11-sp3

image[64bit]=sles11-sp3-inst64
    label=sles11-sp3
    append="quiet sysrq=1 insmod=sym53c8xx insmod=ipr install=nfs://10.0.0.51/export/PPC/SLES11-SP3"

image[64bit]=sles11-sp2-inst64
    label=sles11
    append="quiet sysrq=1 insmod=sym53c8xx insmod=ipr install=nfs://10.0.0.51/export/PPC/SLES11-SP2"

image=/RHEL6-U5/vmlinuz
    label=rhel6-u5
    initrd=/RHEL6-U5/initrd.img
    append="ks=nfs:10.0.0.51:/export/RHEL_Kickstart/kickstart.cfg ksdevice=eth0"
    read-only

. . . .
```

Installing PowerKVM

OPAL Model: IPMI on the FSP

- IPMI (which stands for Intelligent Platform Management Interface) is a standardized interface and protocol set for managing systems out of band, or in-band and monitoring operation
  - Common protocol on x86 server BMCs, although HP, DELL, Lenovo have customized BMC software
- Open source command line tool on Linux: ipmitool
- There is no explicit IPMI user on the FSP
  - No user to specify when using ipmitool
- IPMI password is set within the ASMI interface
- Firmware updates are not done with IPMI on FSP-based systems

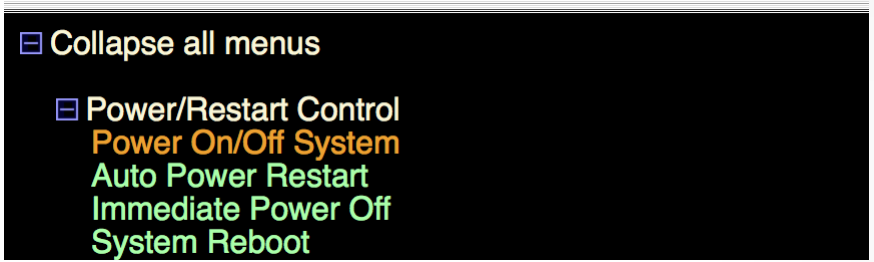
Determining IP address – Systems with FSPs

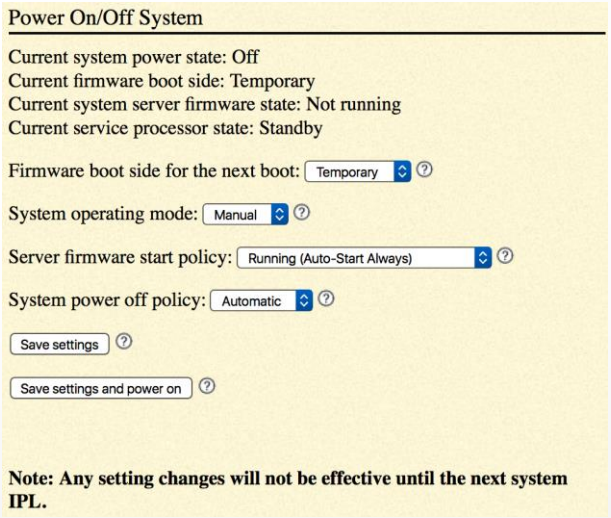
- The FSP ports (usually labeled HMC1 and HMC2) are initially configured to obtain network address(es) via DHCP request
- The op console can be used to determine the IP address of the FSP
- The following steps will determine the IP Address
  - STEP 1: Use the op-console to set the system into manual mode
  - STEP 2: Use the up/down arrow keys to select option 30
  - STEP 3: Press Enter
  - STEP 4: Use the up arrow keys to select 3000
  - STEP 5: Press Enter
  - STEP 6: Observe IP Address

Configuring the system for Opal Mode

Power off the system

- POWER8 Scale-Out Linux only Models can be configured to support either the PowerVM hypervisor or OPAL (bare metal)
- Configuring the system to support OPAL mode requires the system to be in a powered-off state and not managed by an HMC
- **Step 1**: Check the Power state of the system via ASM and power off if necessary

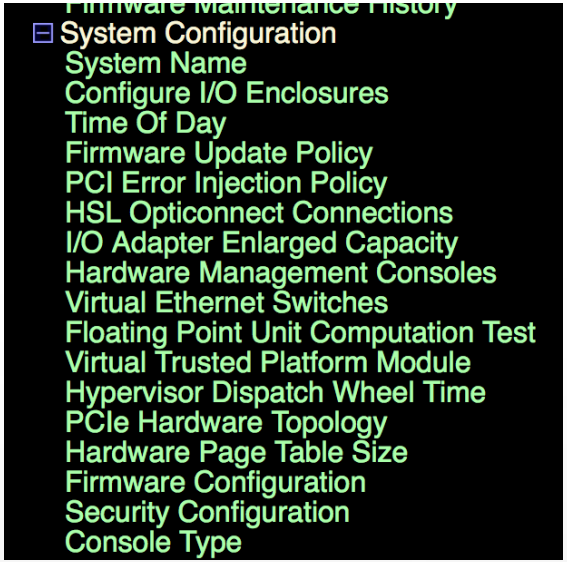


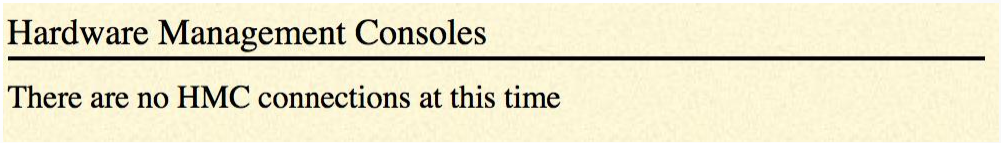


Configuring the system for Opal mode

Verify that there is no HMC connection to the system

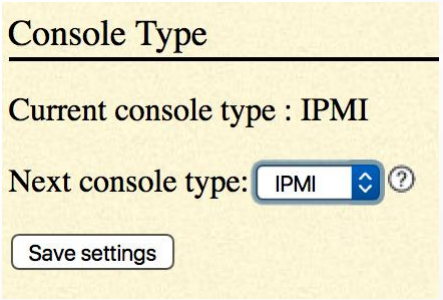
- To change a system to OPAL mode all HMC connections need to be removed
- **Step 2**: Remove any configured HMC connections





Configuring the system for OPAL mode
Check Console Type

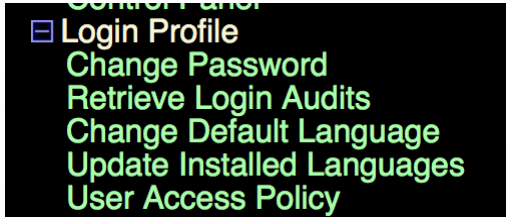
- In OPAL mode there are two console types, serial and IPMI
  - Serial requires a connection to the serial port on the back of the system
  - IPMI requires a network connection to the Service Processor
- **Step 3**: Ensure that the console type is set to IPMI

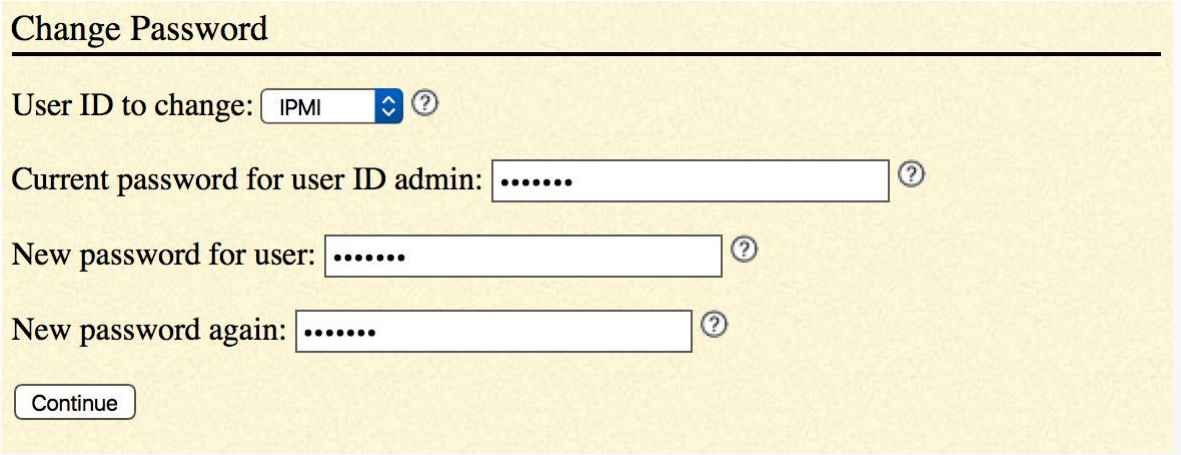


Configuring the system for OPAL mode

Setting the IPMI password

- The password defined for the ‘IPMI’ user is used to access the Flexible Service Processor via ipmi tools
- **Step 4:** Set the IPMI password





Useful IPMI commands

- Power on server
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com chassis power on`
- Power off server
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com chassis power off`
- Check server status
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com chassis status`
- Gracefully shut down the server
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com chassis power soft`
- Server hard reset
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com chassis power reset`

Useful IPMI commands

- Activate Serial Over LAN (SOL) console
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com sol activate`
- Deactive SOL system console
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com sol deactivate`
- Get error log
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com sel list`
- List status of all sensors
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com sdr list`

Useful IPMI commands

- Get firmware version
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com mc info`
- Get a specific sensor
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com sdr get "OS Boot"`
- Get a list of sensor types
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com sdr type list`
- Set boot device
  - `ipmitool -I lanplus -H myserver.ibmpowerlinux.com chassis bootdev cdrom`

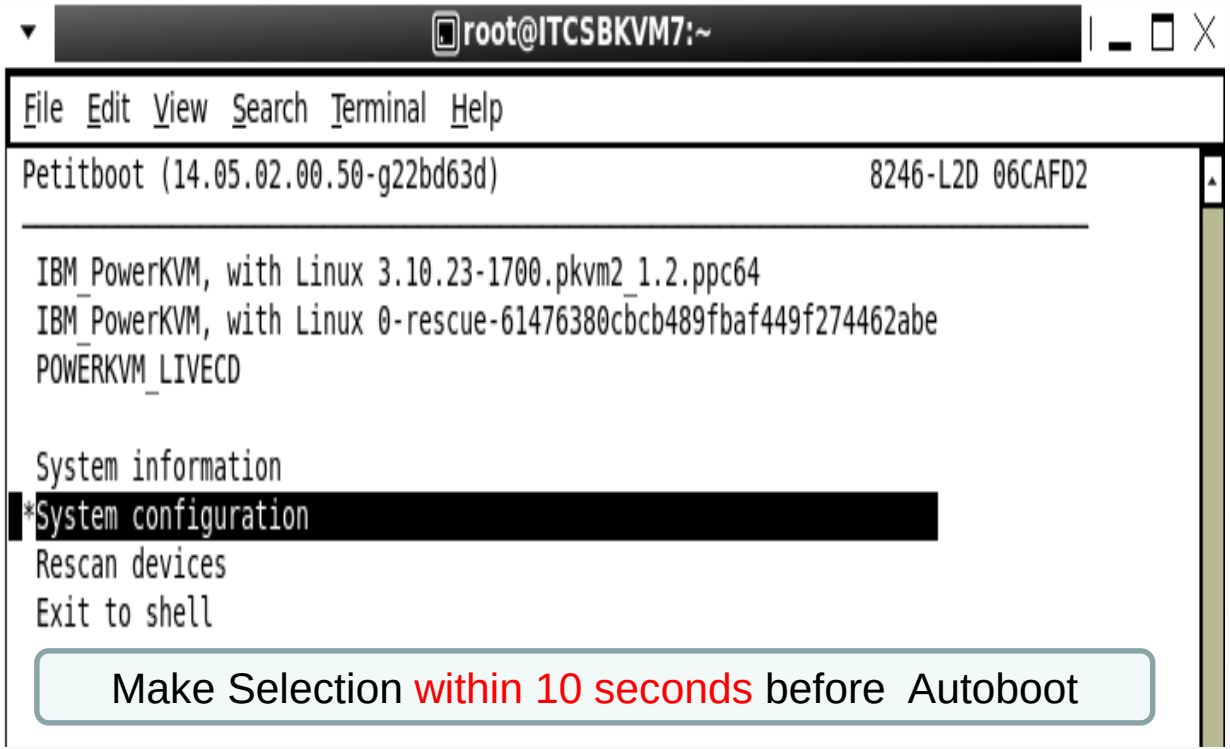
Petitboot Utility

- Petitboot is a platform independent boot loader based on the Linux **kexec** warm reboot mechanism

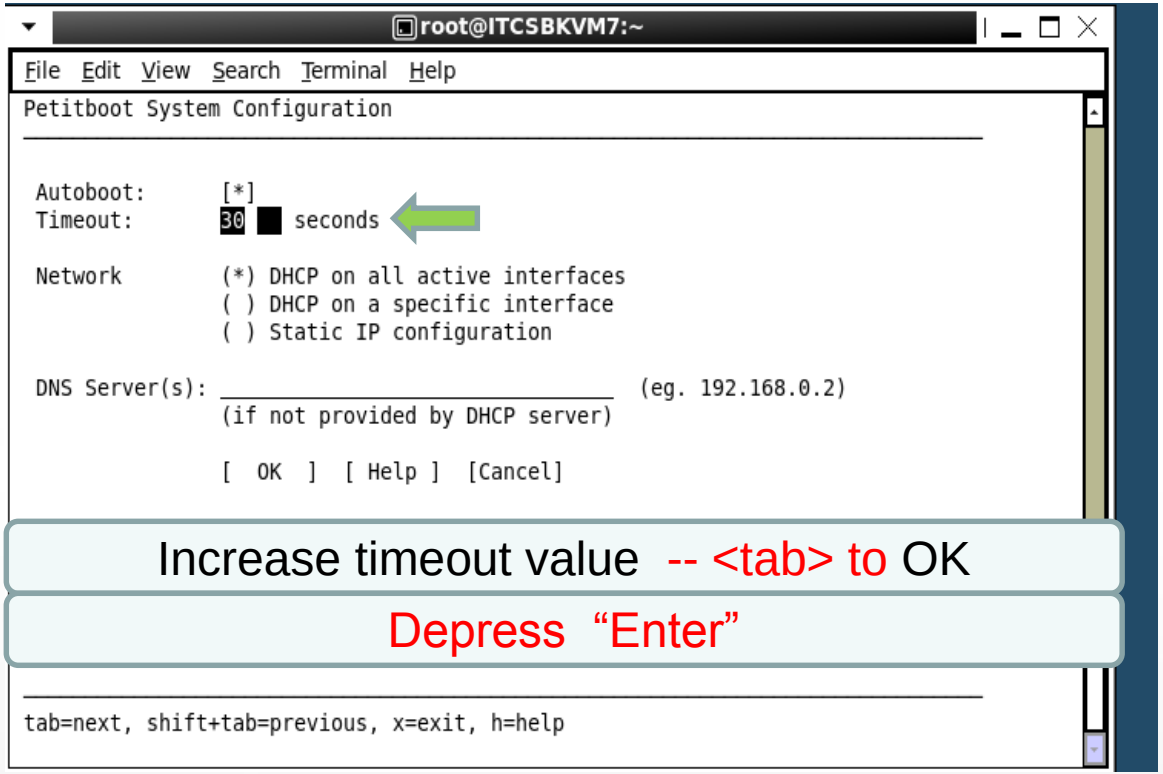
- Petitboot is part of the OPAL firmware that provides the following functions
  - Automatically detects installation media as well as installed Linux instances
  - Displays all physical adapters / ports attached to the Power Server
  - Configures storage on attached storage adapters
  - Sets Petitboot auto timeout and other settings
  - Interfaces with boot device (DVD, Linux Kernel, etc)

- Power systems configured for OPAL mode will disable PHYP and boot into Petitboot

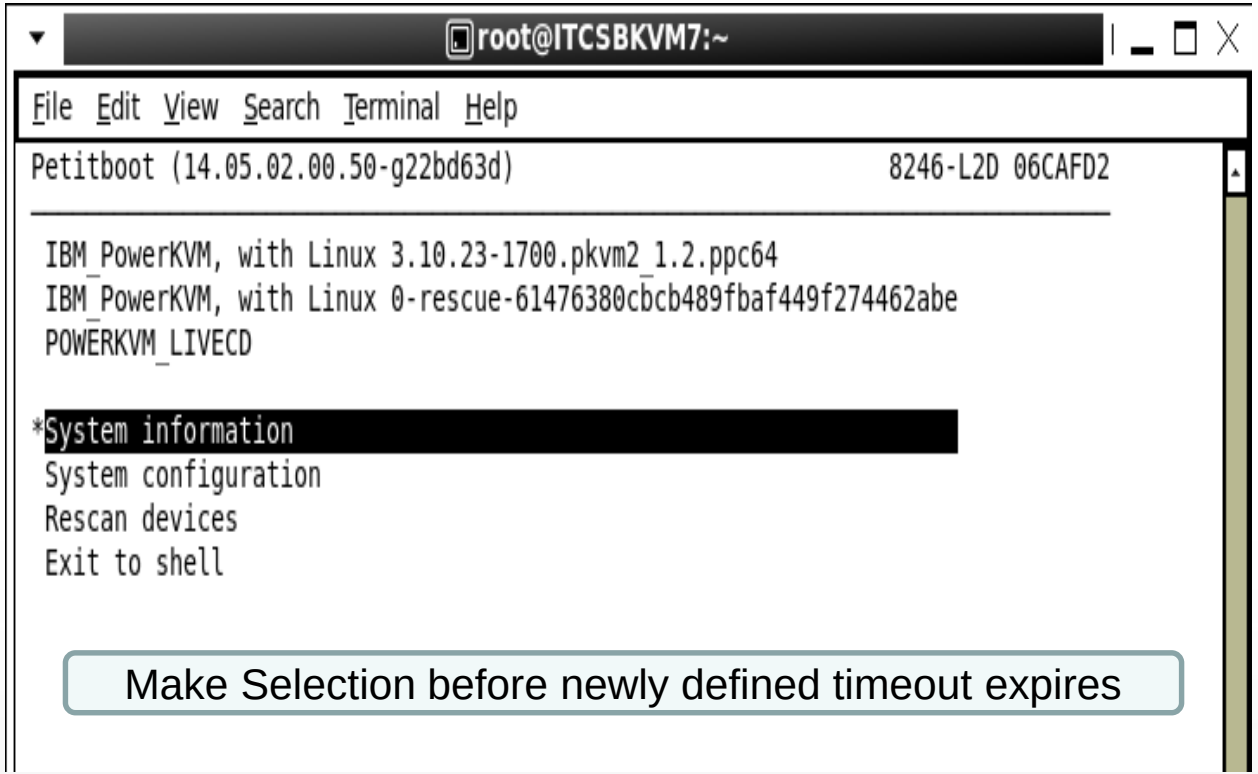
Petitboot Main Menu



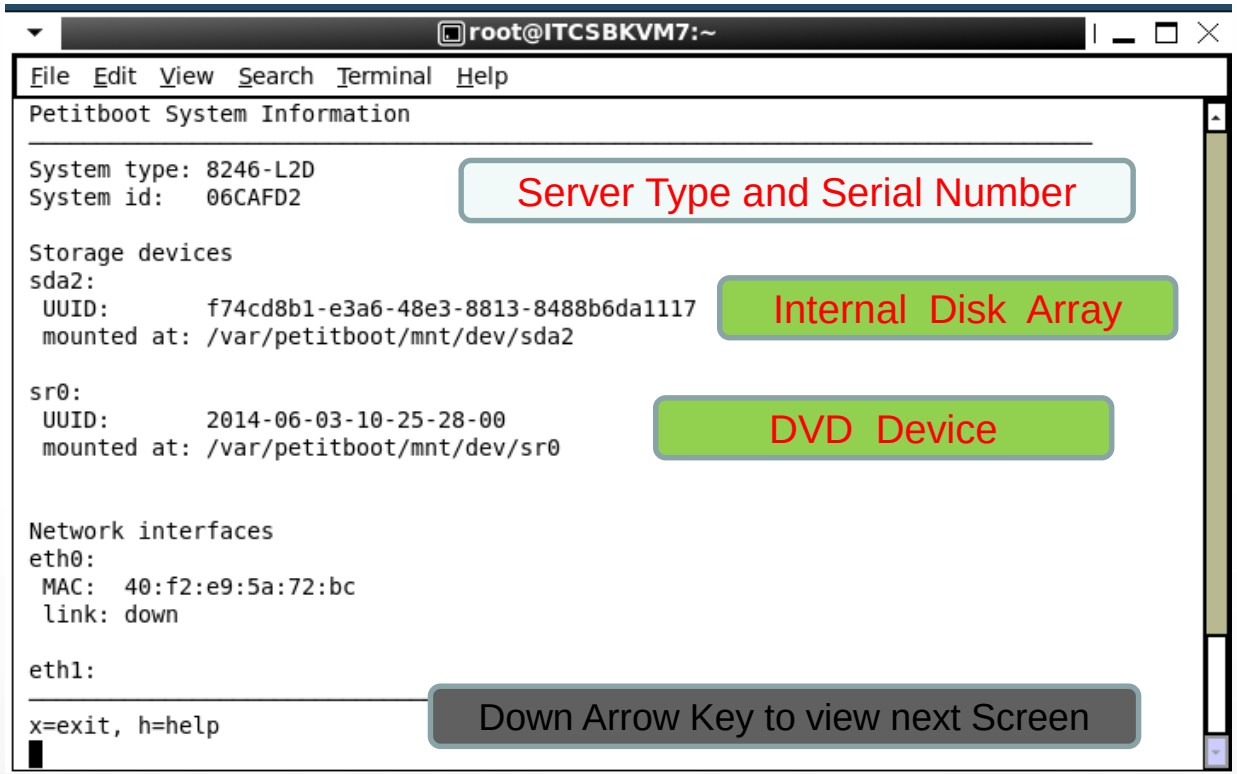
Petitboot – System Configuration
Increasing the Timeout



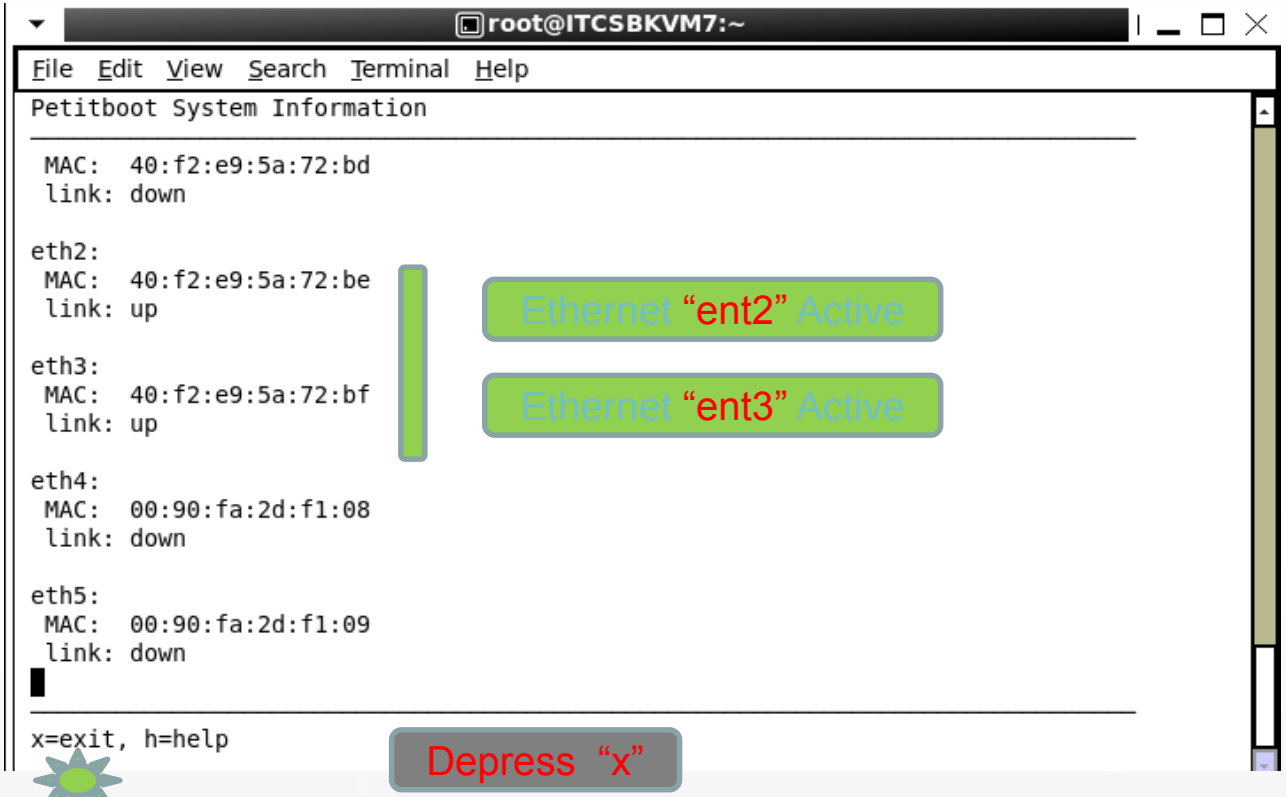
Petitboot – Working with System Information



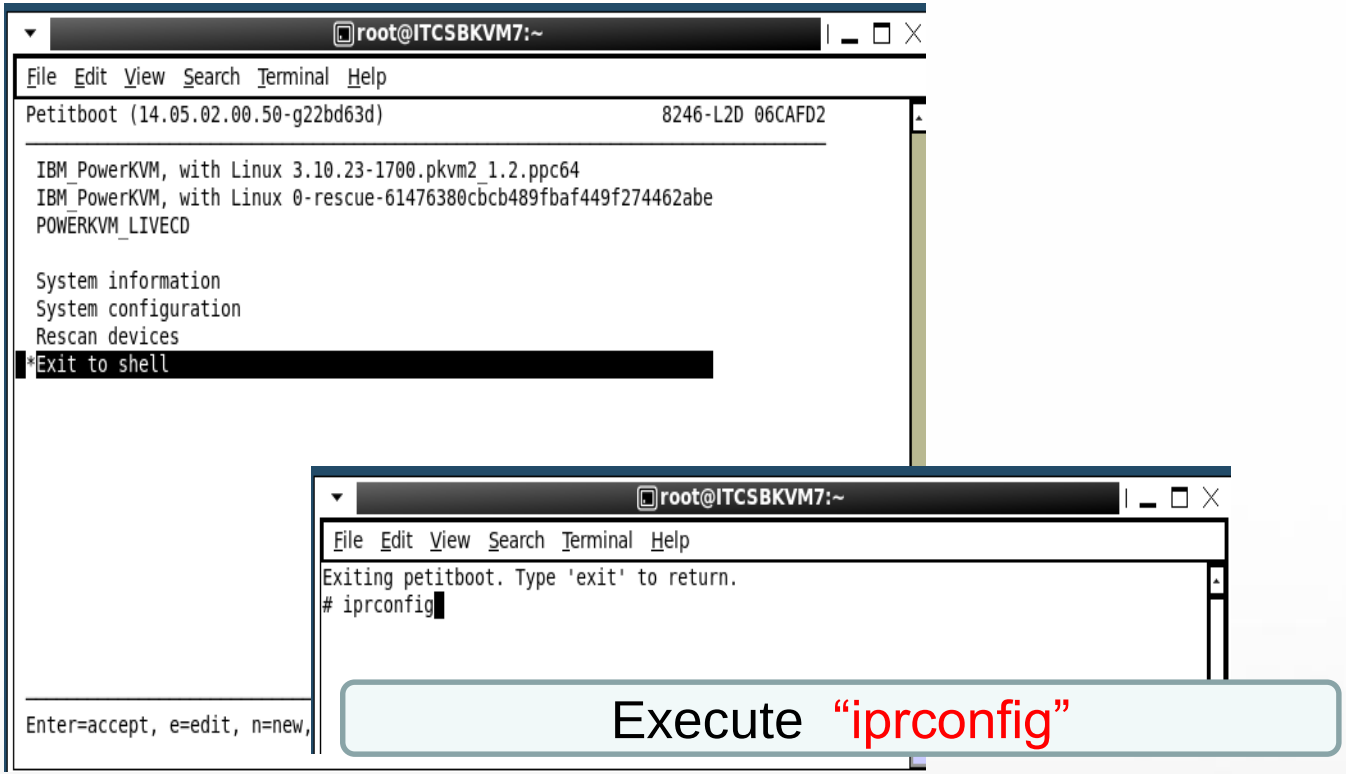
Petitboot – Working with System Information



Petitboot – Working with System Information

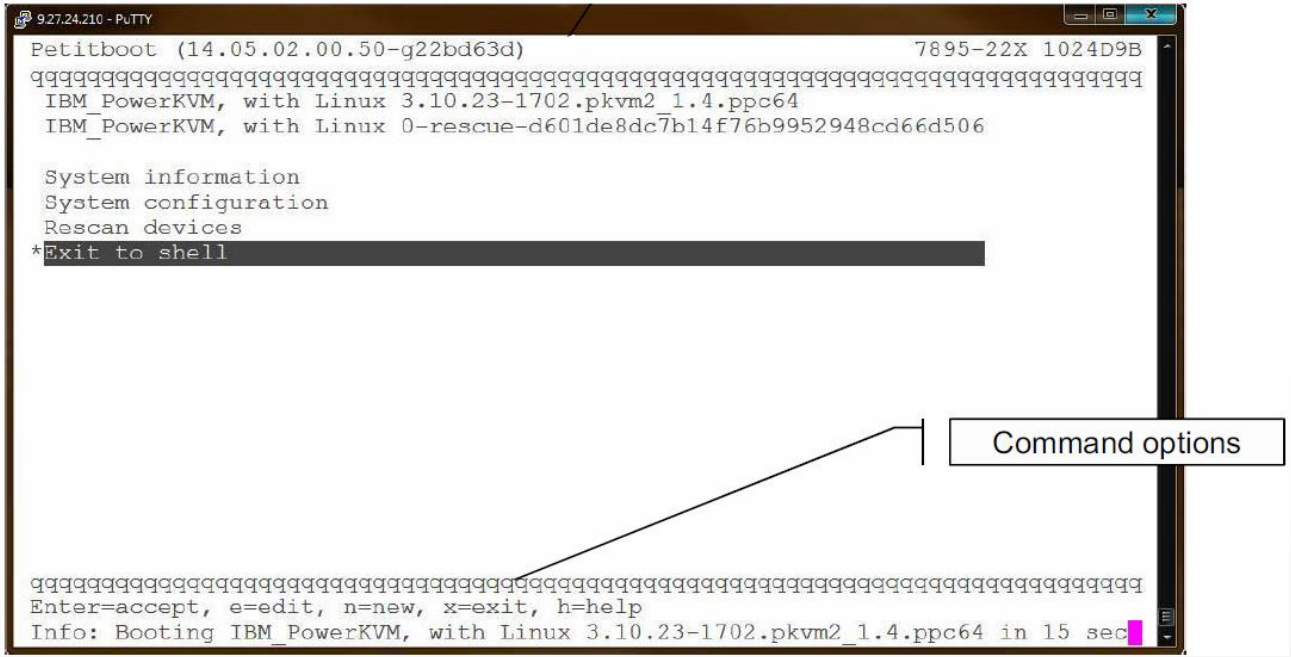


Petitboot – Working with Hardware RAID Adapters



Petitboot – Network boot: Option Editor (1/3)

- From the main **Petitboot** menu, use the **N** key to create a new entry

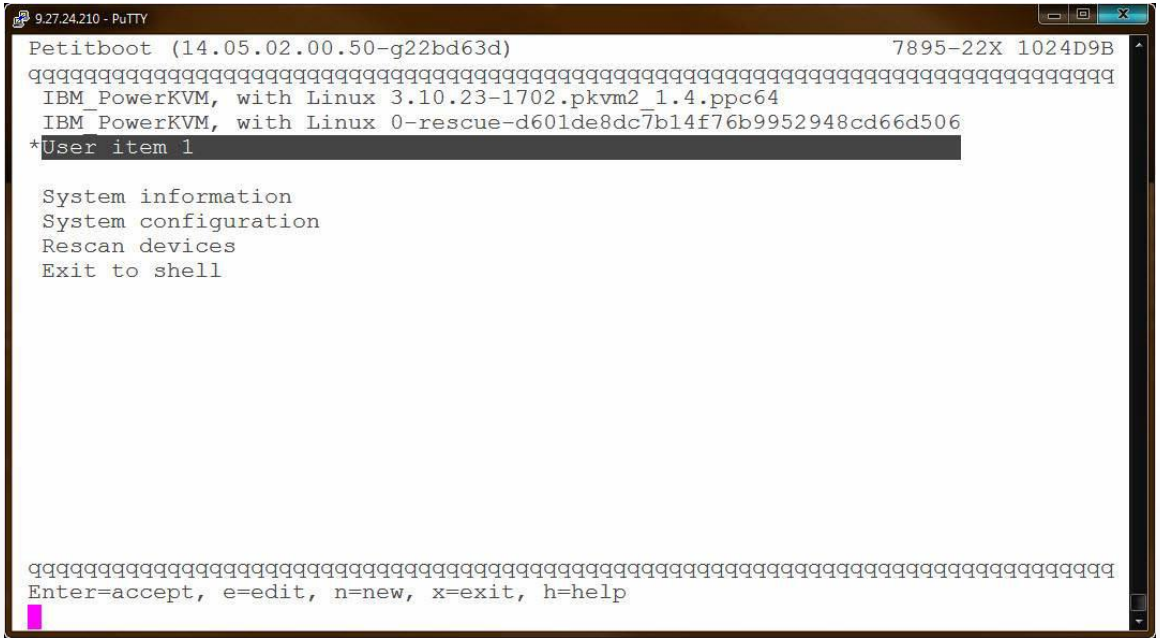


Petitboot – Network boot: Option Editor (2/3)

- Provide required information
  - lines in the example extend beyond the screen capture

Petitboot – Network boot: Option Editor (3/3)

- Select the boot source:
  - example shows new user item just created



PowerKVM Connections

- Direct connection to system via serial port
- Network connection to KVM host once PowerKVM has been installed
- Serial Over LAN (SOL) console from workstation with network connection/access to KVM Host
  - SOL is supported by ipmitool on Linux and MAC workstations and ipmiutil on Windows workstations

PowerKVM Installation

- Power Cycle the Power Server via ipmitool
- Open SOL console to the server via ipmitool
- Select “DVD” Media” from Petitboot Menu
- Follow “PowerKVM Installation Menus”

"ipmitool" Syntax

- Power Control Command Options

```
$ ipmitool -I lanplus -H <fsp – IP> -P <fsp password> power <option>
```

status
on
off
cycle

- Serial Over LAN – Console Connection

```
$ ipmitool -I lanplus -H <fsp – IP> -P <fsp password> sol activate
```

activate
deactivate

System Power-On and petitboot access

FSP IP Address and IPMITOOL Password

```
File  Edit  View  Search  Terminal  Help
[root@ITCSBKVM7 ~]# ipmitool -I lanplus -H 192.168.100.10 -P lnxteam power status
Chassis Power is off
[root@ITCSBKVM7 ~]#
[root@ITCSBKVM7 ~]# ipmitool -I lanplus -H 192.168.100.10 -P lnxteam power on
Chassis Power Control: Up/On
[root@ITCSBKVM7 ~]#
[root@ITCSBKVM7 ~]# ipmitool -I lanplus -H 192.168.100.10 -P lnxteam sol activate
[SOL Session operational.  Use ~? for help]
```

verify Chassis power is "OFF"

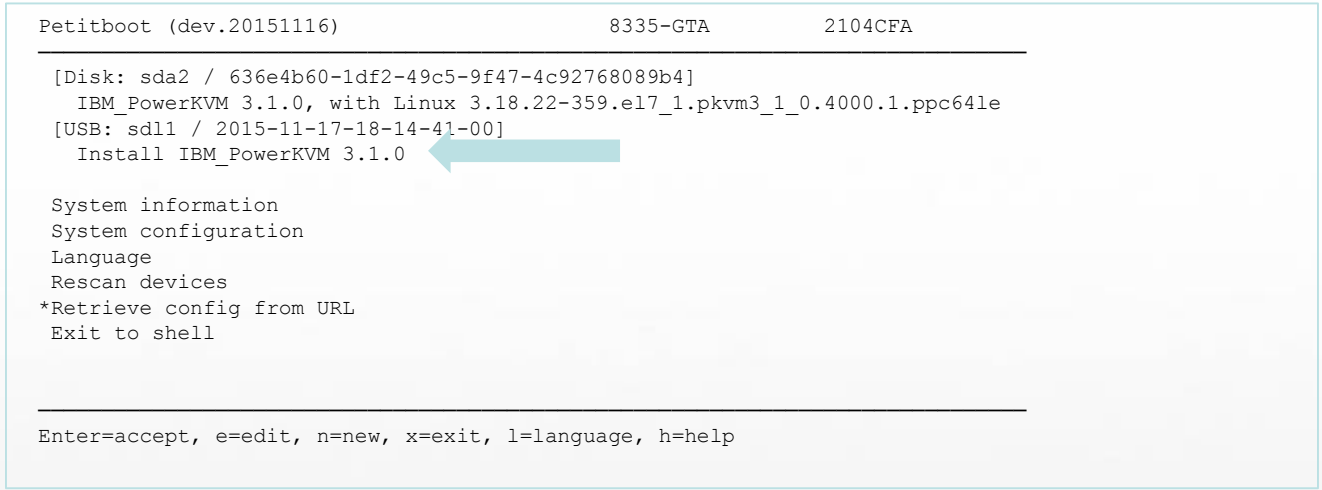
Activate Chassis power "ON"

Activate Serial Over Lan - Console

Be Patient Can take up to 5 minutes before "Pettiboot" Menu Appears

Booting the Installer

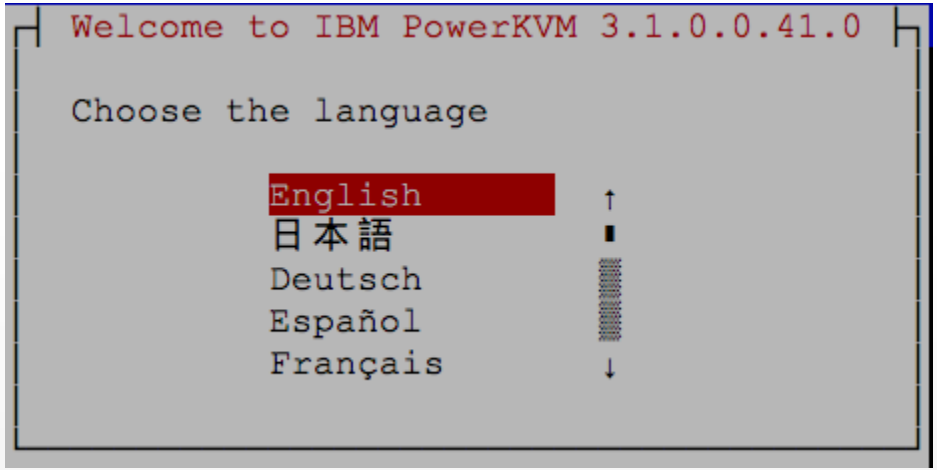
- When the system is started petitboot scans the system looking for any bootable devices (such as HDDs/SSDs with Linux installed or USB keys/Media Drives with installation media)
- To install PowerKVM use the arrow keys to highlight ‘Install IBM_PowerKVM” and press <ENTER>



Installing PowerKVM

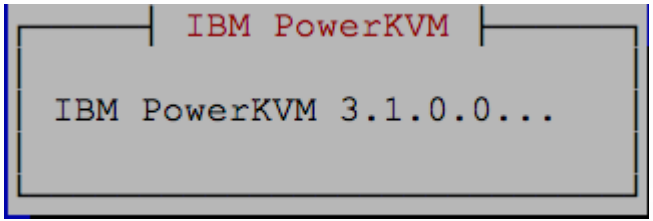
- When the installer is selected, petitboot uses a mechanisms called ‘kswap’ to swap itself out of memory in favor of the bootable device selected (in this case the installer)

- The first step in the installation is to select the language to use for the installation.
  - Highlight the desired selection and press <ENTER>



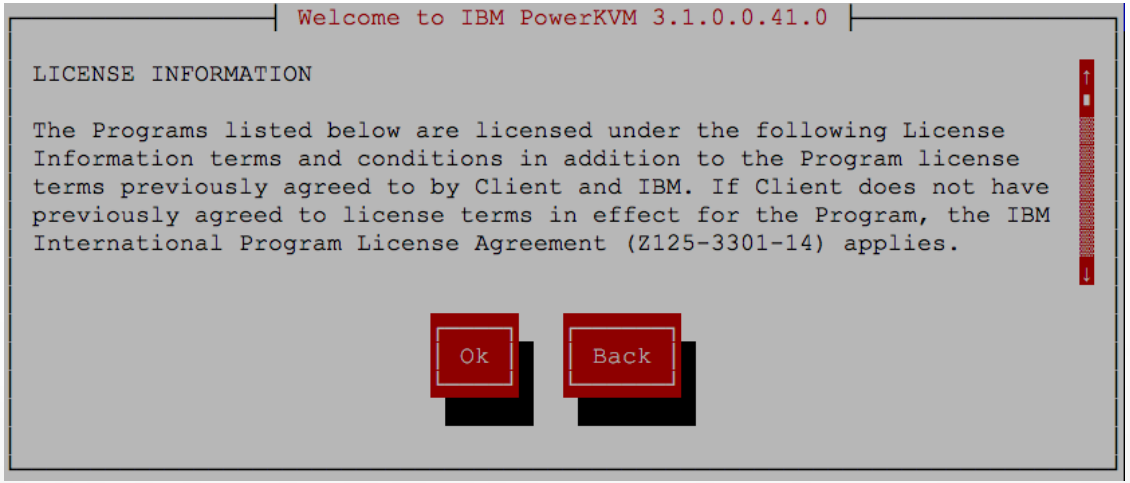
Installer Loading

- After the language selection additional installation files are loaded
- There is no action to take until the next dialog is displayed



PowerKVM License Agreement #1

- Select <OK> to accept the displayed License Agreement
  - NOTE: In the installer buttons are highlighted by pressing the <TAB> key and selected via the <ENTER> key

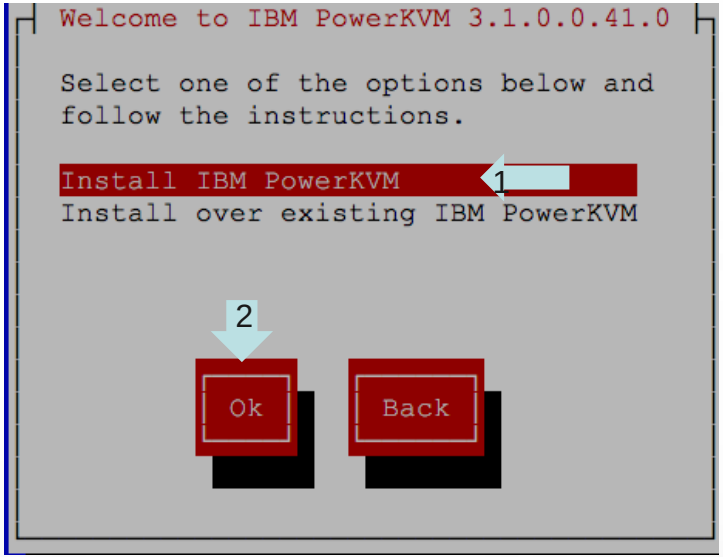


PowerKVM License Agreement #2

- PowerKVM has a second set of License Agreements that need to be accepted
- Select <I Accept both IBM and non-IBM terms>

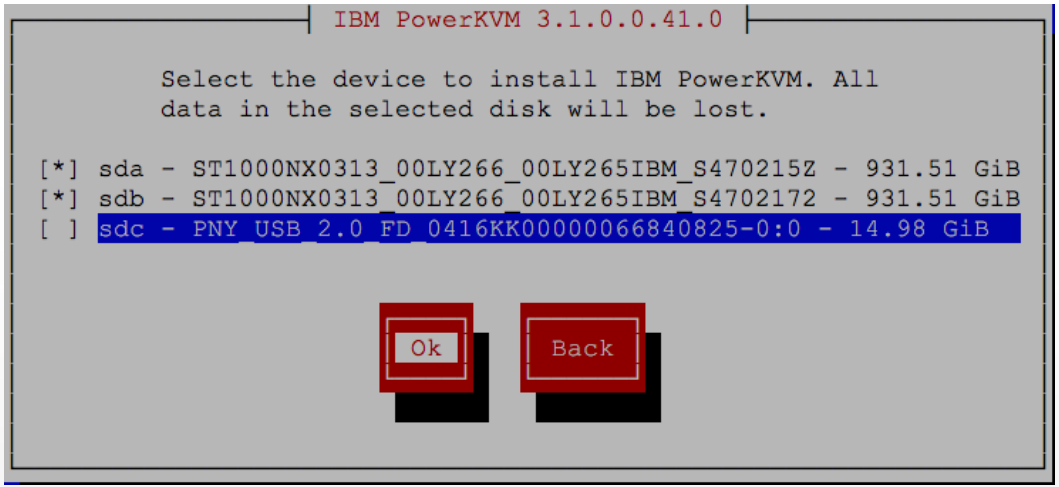
Install New vs. Over Existing

- If PowerKVM is being installed on a system that has a previous installation of PowerKVM then the option is provided to overwrite the existing installation
  - What this means is that the storage volumes for any installed guests are not overwritten
  - Keep in mind that all configurations are overwritten so any guests previously defined will need to be redefined – but the virtual disks for the guests will still be available



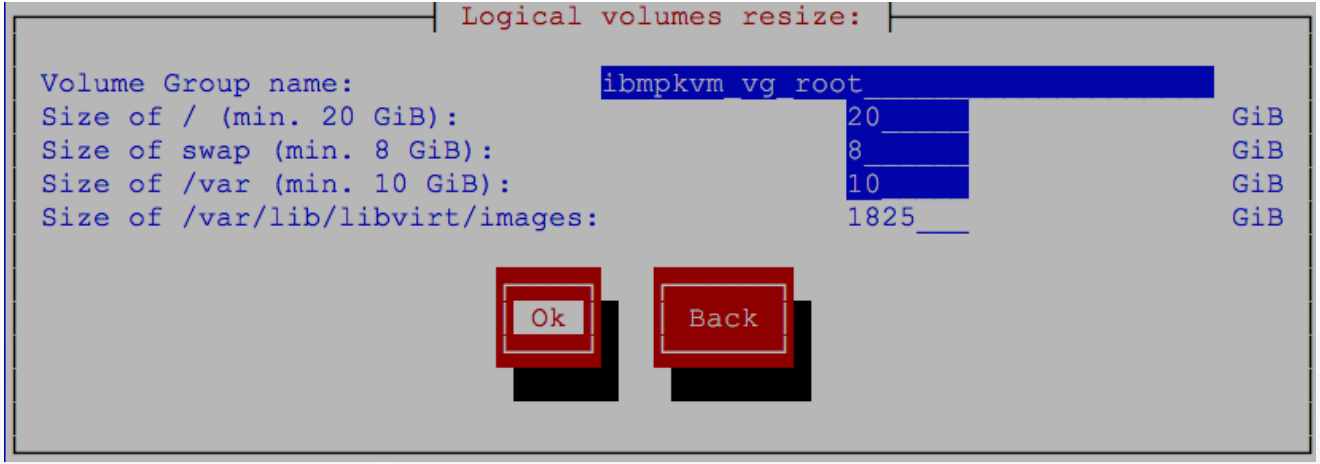
Disk Selection

- The disk selected will be the disk that the installer will partition, format, and install the operating system onto
- Use the arrow keys to highlight the desired disk and press the <SPACE BAR> to select it
  - NOTE: Multiple disks can be selected. PowerKVM uses Logical Volume Manager (LVM) and will spread the volumes across all of the disk devices selected
- Once the disk(s) are selected press <TAB> to OK and then press <ENTER>



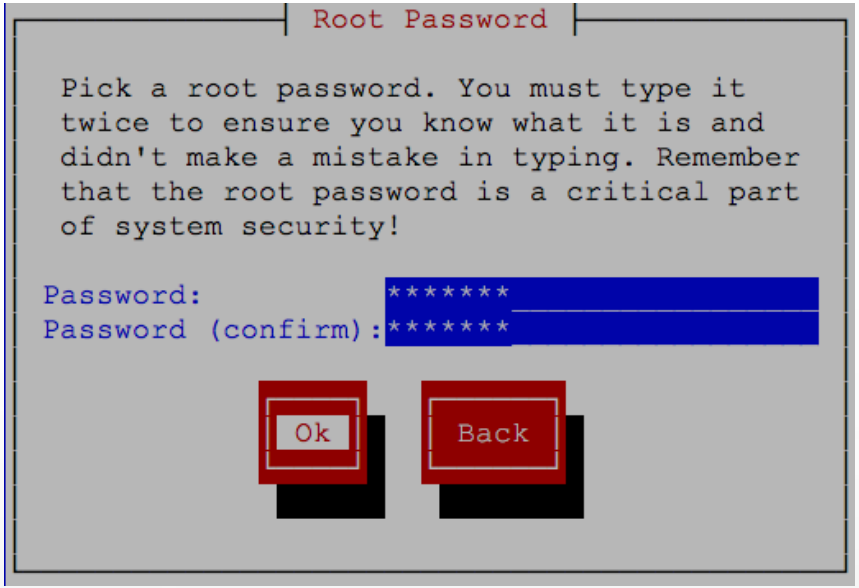
Logical Volume Resize

- The installer will establish the size of the Logical Volumes based on default (minimum) sizes for the root, swap and var file systems. The remaining space is put in the /var/lib/libvirt/images volume
  - /var/lib/libvirt/images is the default storage directory for the KVM guest image files (virtual disks)
- Tab to <OK> and press <ENTER>



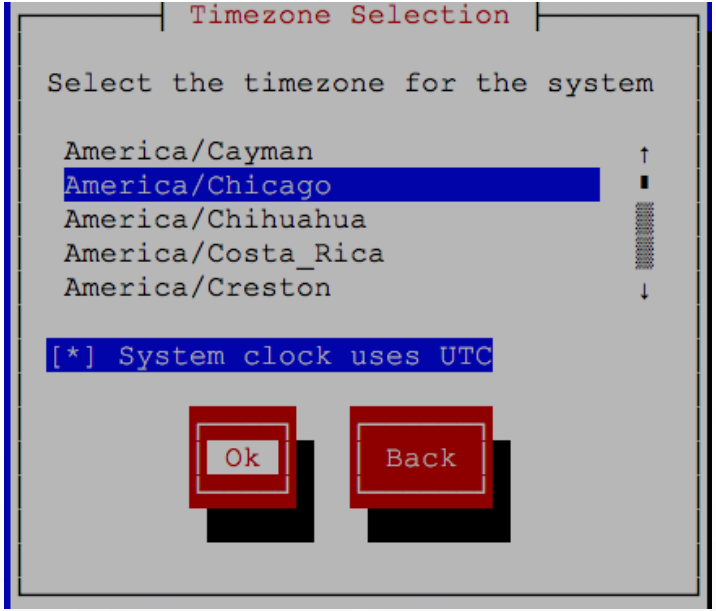
root password definition

- Define a password for the root user
- Tab to OK and press <ENTER>



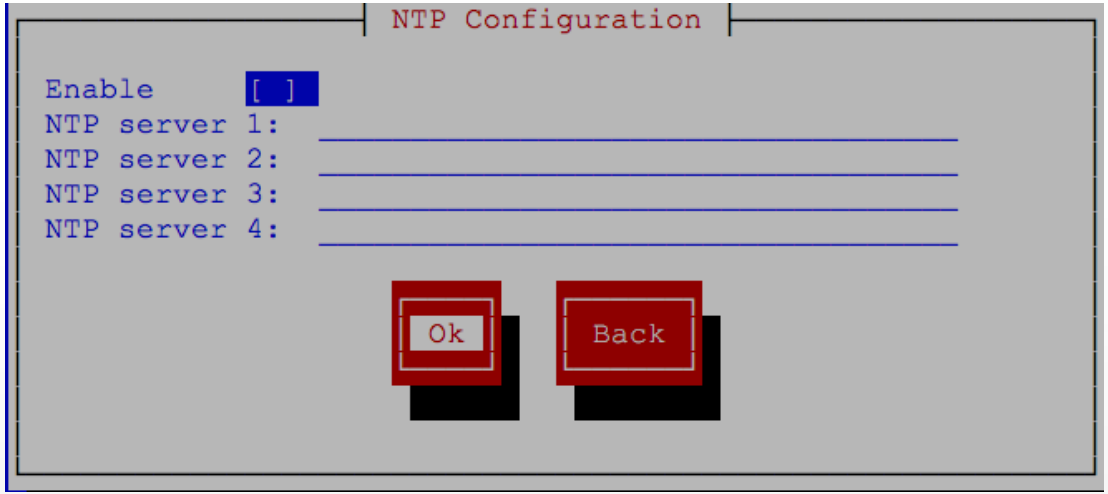
Timezone Selection

- Use the arrow keys to highlight the appropriate time-zone
- Press <TAB> to highlight OK and then press <ENTER>



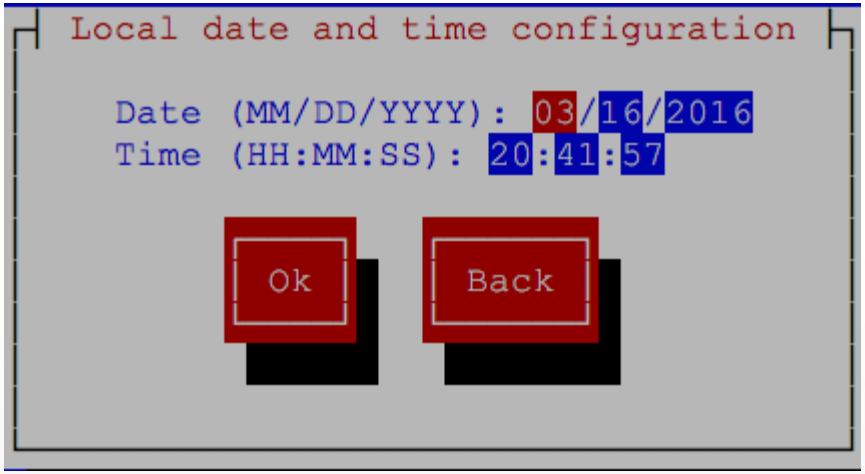
NTP Configuration

- If an NTP server is available on the network then press the <SPACE BAR> to enable NTP and then provide the network address for each NTP server
- Press <TAB> to highlight OK and then press <ENTER>



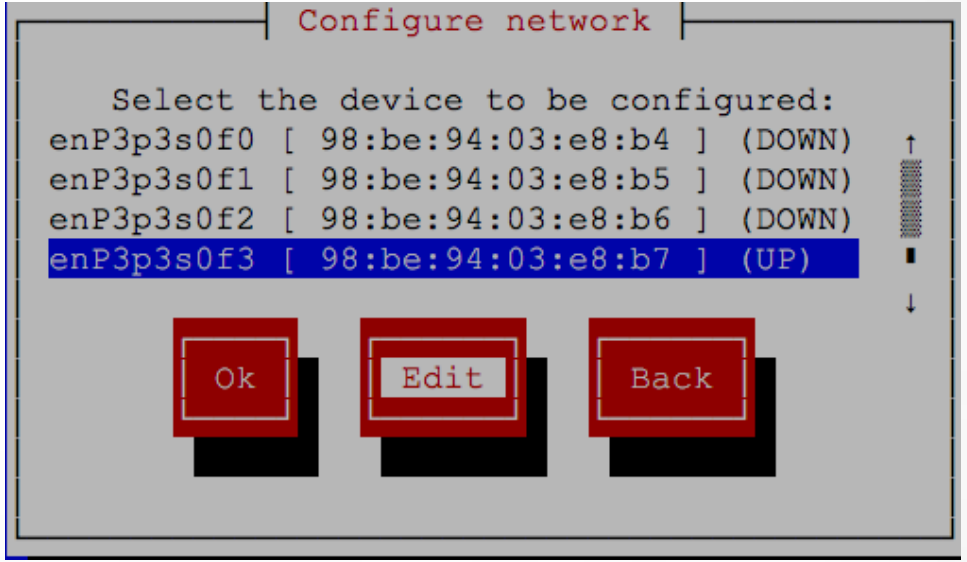
Date / Time Configuration

- Update the Date and Time as appropriate
- Press <TAB> to highlight OK and then press <ENTER>



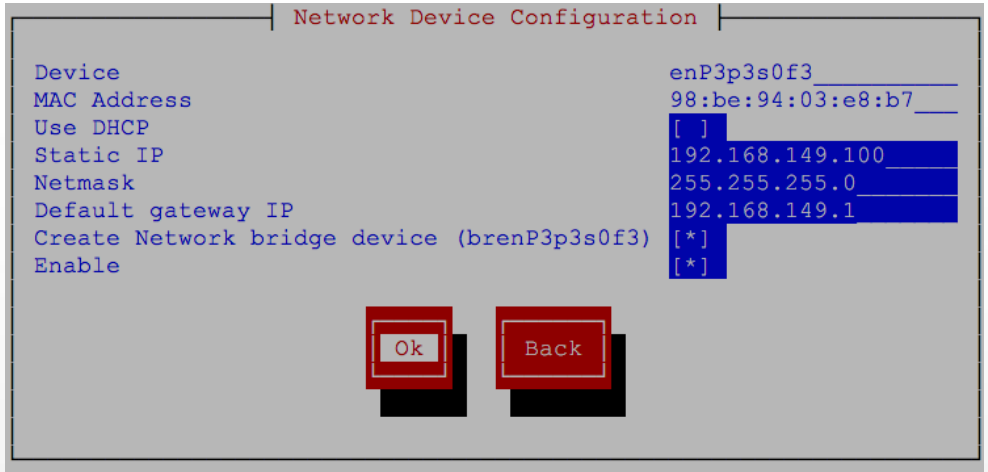
Network Configuration

- The first page of the network configuration shows the link status of each network interface discovered on the system
- Use the arrow keys to highlight an interface with a link status of ‘UP’
- Tab to Edit and then press <ENTER> to advance to the second page of Network Configuration



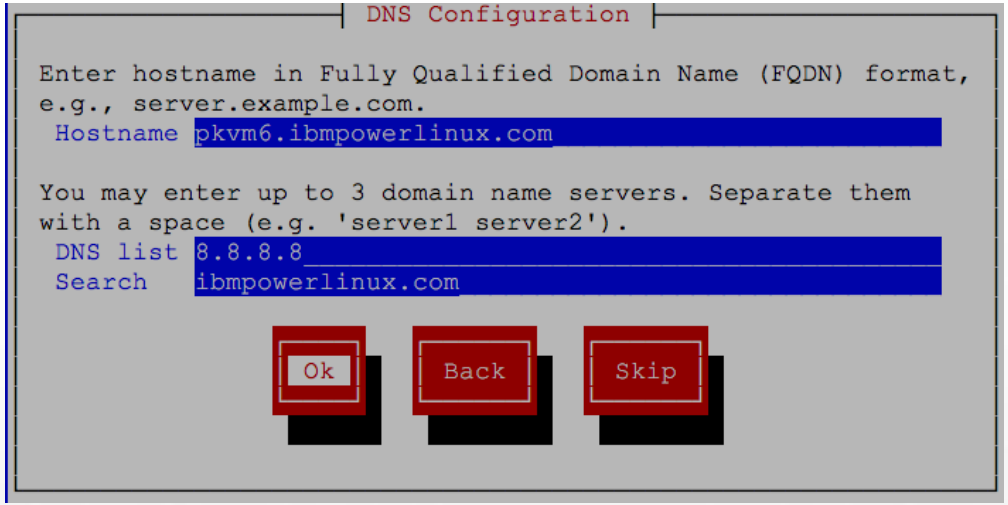
Network Device Configuration

- Most of the fields on the Network Device Configuration are self explanatory; however, there are a couple that need further explanation
  - '**Create Network bridge device**' – this will create a bridge device in Linux that can be used by KVM to route network traffic from a KVM guest via this network adapter to the external network.
  - '**Enable**' – this will cause the adapter to be enabled on start of KVM.
- <TAB> to OK and press <ENTER>
- Back on the 'Configure Network' screen <TAB> to OK and press <ENTER>



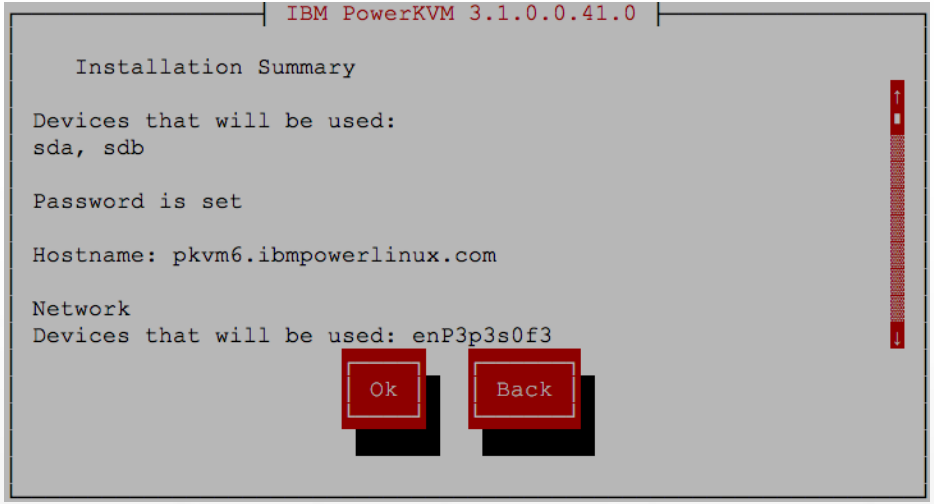
Network Configuration – DNS Configuration

- Provide a network name for the system
- Specify the DNS server(s) as well as any domains to have in the search list
  - NOTE: 8.8.8.8 is a public DNS server provided by google
  - NOTE: the 'ibmpowerlinux.com' domain is a domain that can be used for POCs – it does not resolve to anything on the public network
- Press <TAB> to highlight OK and press <ENTER>



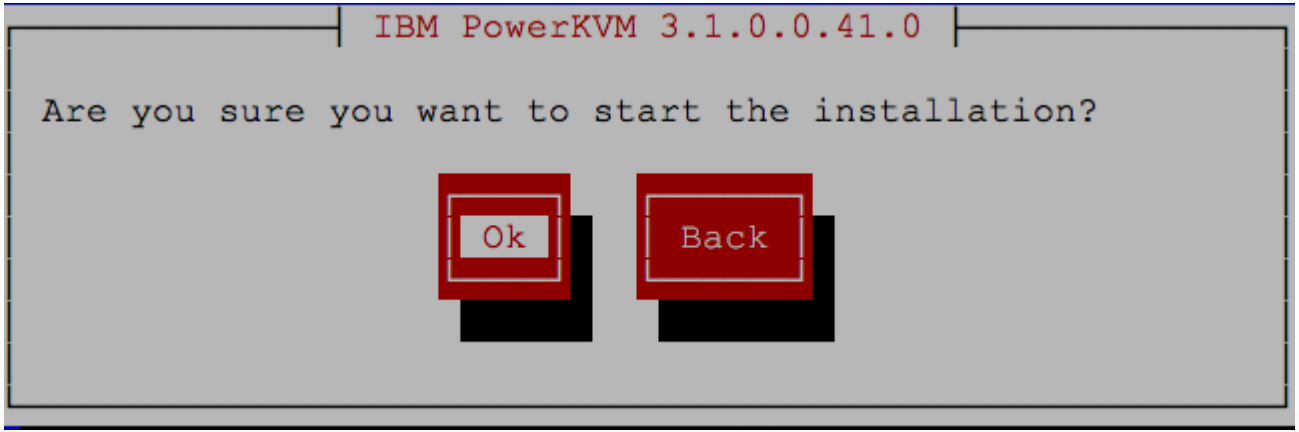
Installation Summary

- Once all of the installation settings have been specified a summary of the installation is provided
- After reviewing the settings if they are correct press <TAB> to highlight OK and press <ENTER>

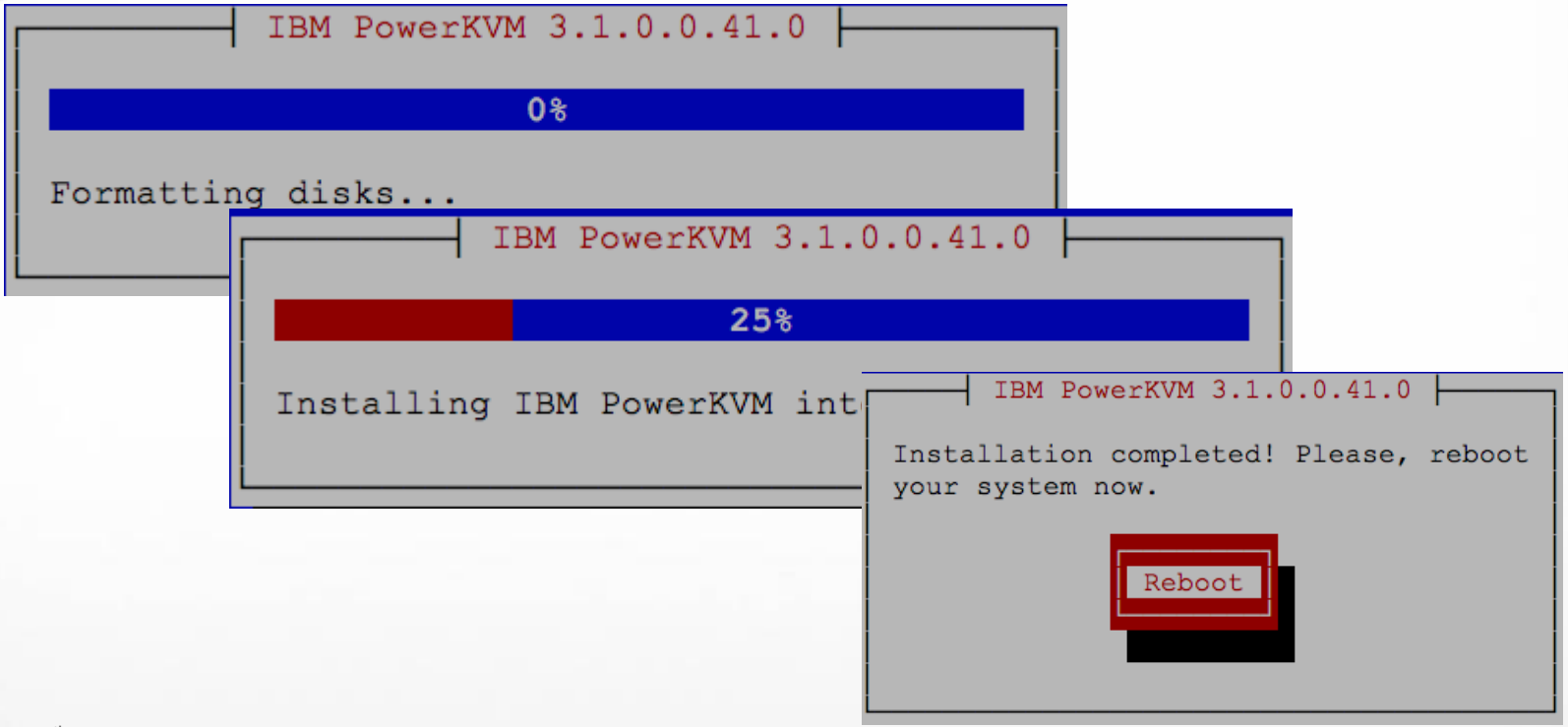


Final Confirmation

- A final confirmation for the installation is displayed.
- Press <TAB> to highlight OK and press <ENTER>

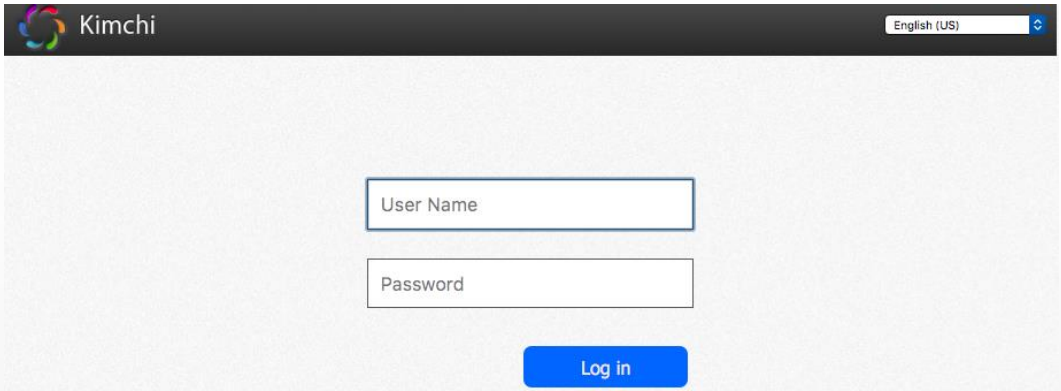


Installation Progress



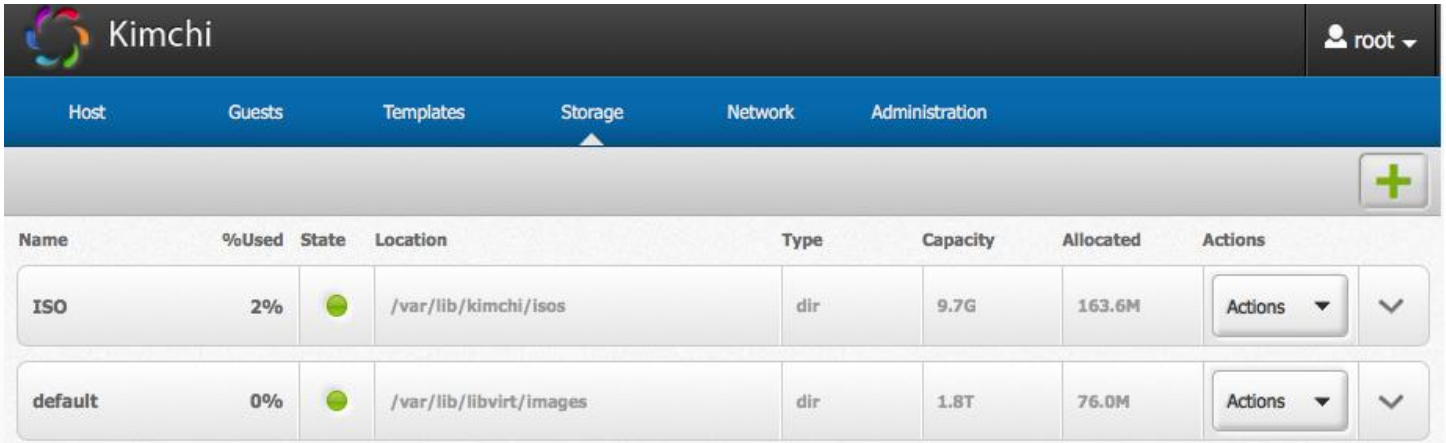
Post Installation Setup

- After the installation the system reboots into ‘petitboot’
- The KVM host is flagged as the first O/S to boot so after the ‘petitboot’ timeout the just installed KVM host will boot
- We will use the KIMCHI interface to configure networking and validate storage for the KVM host
- Point a web browser at the IP Address of the KVM host and specify port 8001 (i.e., https://<ip address of KVM host>:8001)
- Login with the root user and password



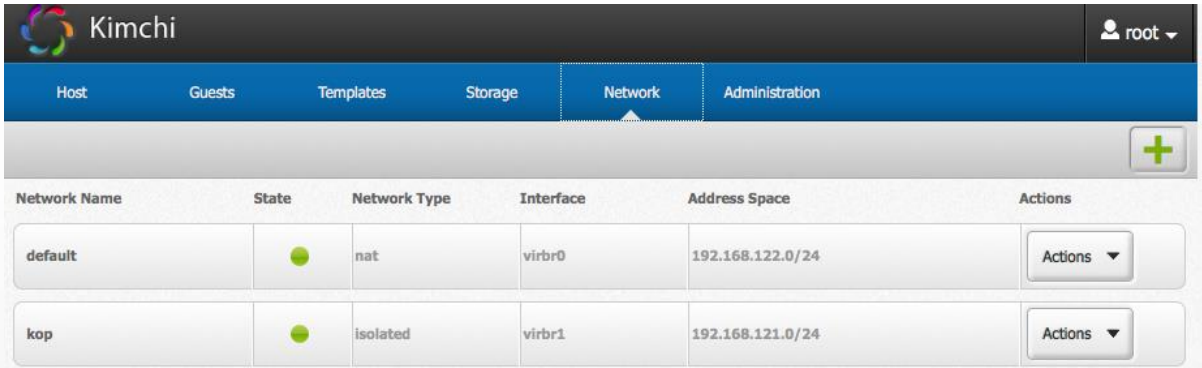
Post Installation Setup
Validate Default Storage Setup

- The Installer sets up two storage pools by default.
  - The ISO storage pool is used to store ISO images used for installations
  - The default storage pool is where storage volumes used by the guests are stored by default



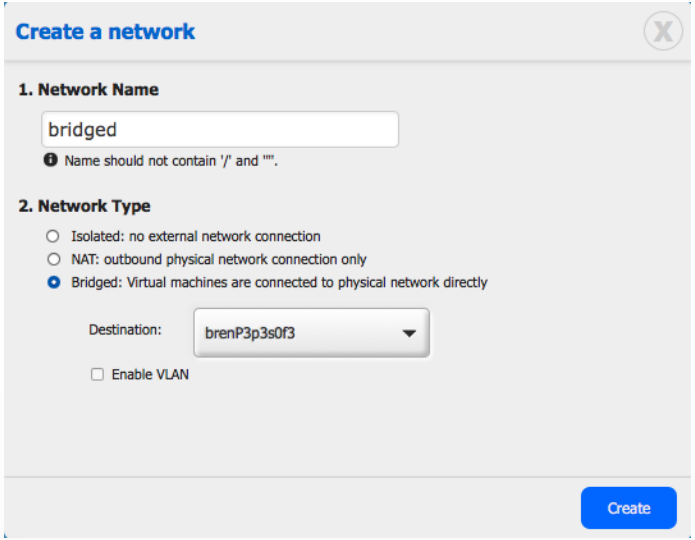
Post Installation Setup
Bridged Network Definition

- The KVM installer defines networks by default
  - The ‘default’ network is a network that uses Network Address Translation (NAT) to route guest traffic to the external network by masquerading through the IP address of the KVM host.
  - The kop network is an isolated network that only allows communication between the guests and host and does not allow external network communication
- On the next slide we will create a new bridged network definition.



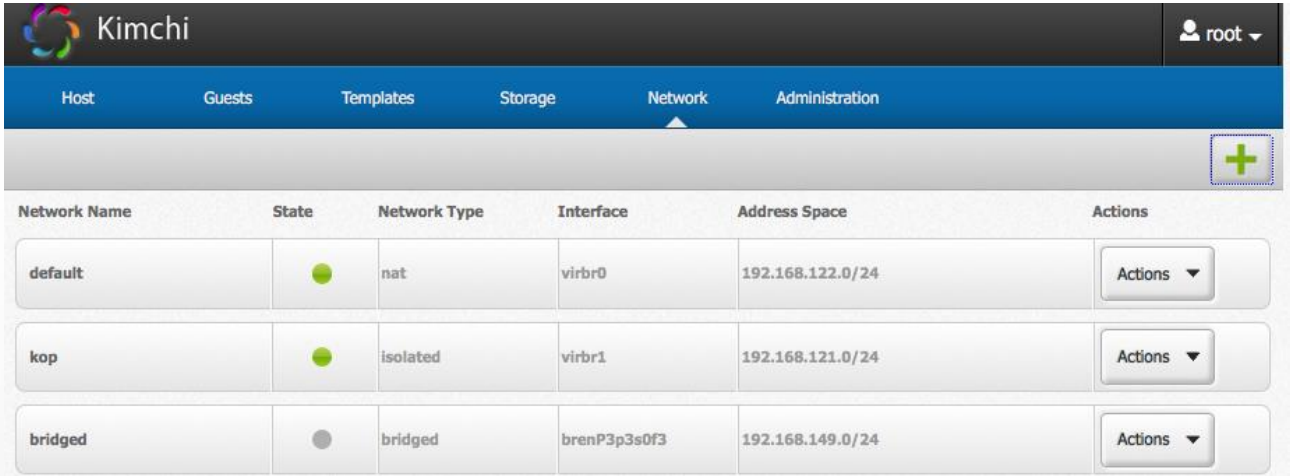
Post Installation Setup
Bridged Network Definition

- To create a new network select the '+' on the Network page
- Complete the 'Create a network' dialog as follows:
  - Network Name – this is a user defined name for network
  - Network type – bridged
  - Destination – the bridged network interface defined during the KVM host installation
    - This interface name will begin with 'br'



Post Installation Setup
Bridged Network Definition

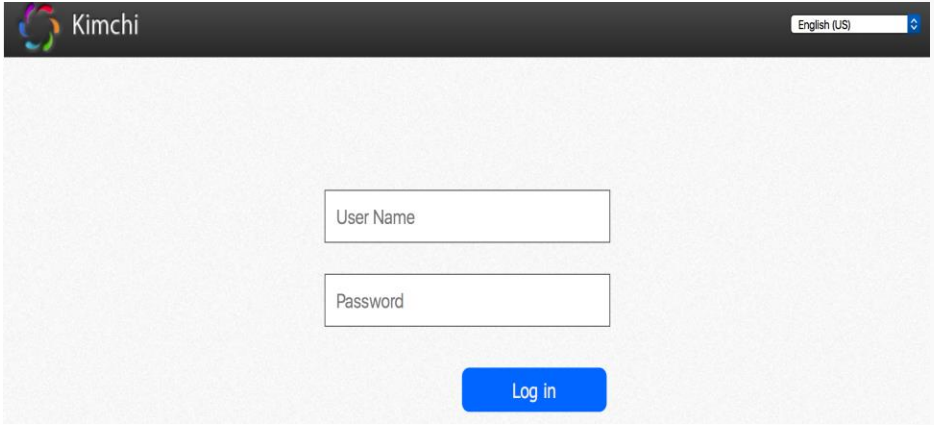
- Notice that the displayed networks now include the newly created network
- Also notice that the network is currently not active
- The network can be activated by selecting ‘Start’ from the networks ‘Actions’ drop-down list



Linux Installation (PowerKVM)

Using KIMCHI to create RHEL 7.1 Guest

- KIMCHI will be used to setup the Linux guest that will be used as the baseline for PowerVC
- KIMCHI can be accessed by pointing a web browser to port 8001 of the KVM Host:
  - https://<KVM host>:8001
- When the login screen is displayed use the root account and password



Linux Installation ISO Images

- Following table shows where Linux ISO images can be obtained from:

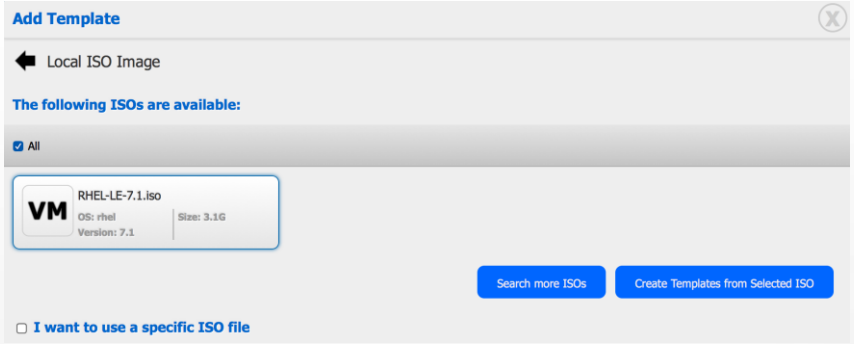
Disribution	Location
RedHat	https://access.redhat.com/products/red-hat-enterprise-linux/evaluation
SuSE	https://www.suse.com/products/server/download/
Ubuntu	http://www.ubuntu.com/download/server/power8

Regardless of where obtained, the installation ISO images should be placed in the `/var/lib/kimchi/isos` directory so they are available for template creation.

Using KIMCHI to create RHEL 7.1 Guest

Template Creation

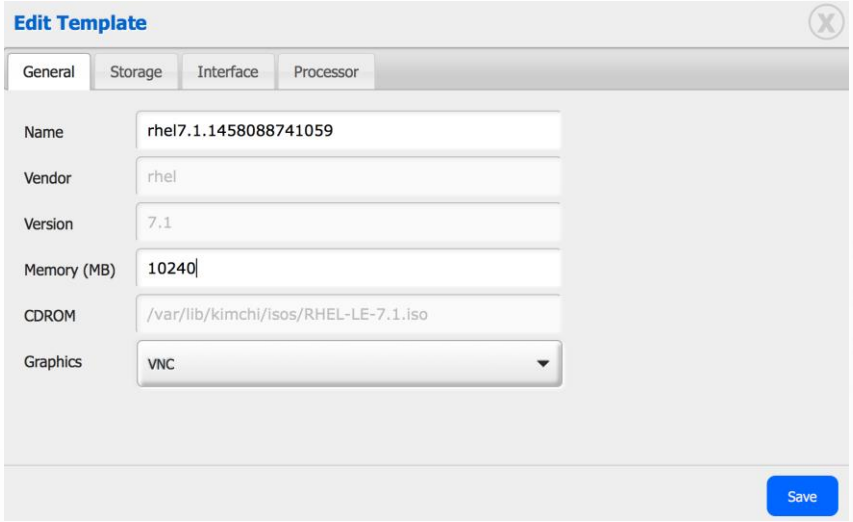
- The first step to creating a KVM guest via KIMCHI is to create a template
- The template is used to define items such as the installation source, number of processors, amount of memory, size of disk, and networks to attach to
- The first step is to select the '+' on the Guests page
- Then Select 'Local ISO Image'
- Finally select the ISO and the <Create Templates from Selected ISO>



Using KIMCHI to create RHEL 7.1 Guest

Template Creation – General Settings

- After the template is created use the ’Edit’ function under ‘Actions’ to make changes as detailed on the following pages
- Change the value of memory to ‘10240’



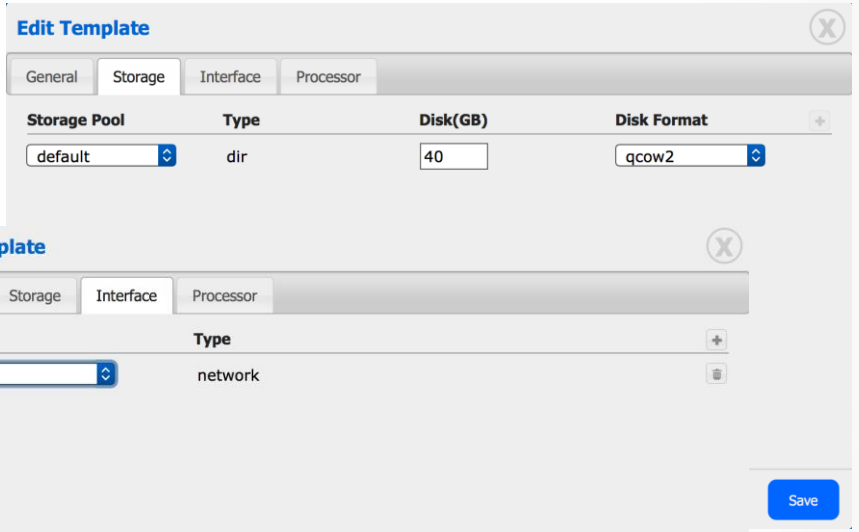
NOTE: The modified settings meet the resource requirements for PowerVC

Using KIMCHI to create RHEL 7.1 Guest

Template Creation – Storage & Interface Settings

- Change the size of the disk to 40GB

- Change the Network to ‘bridged’
  - The network that was previously defined as the bridged network

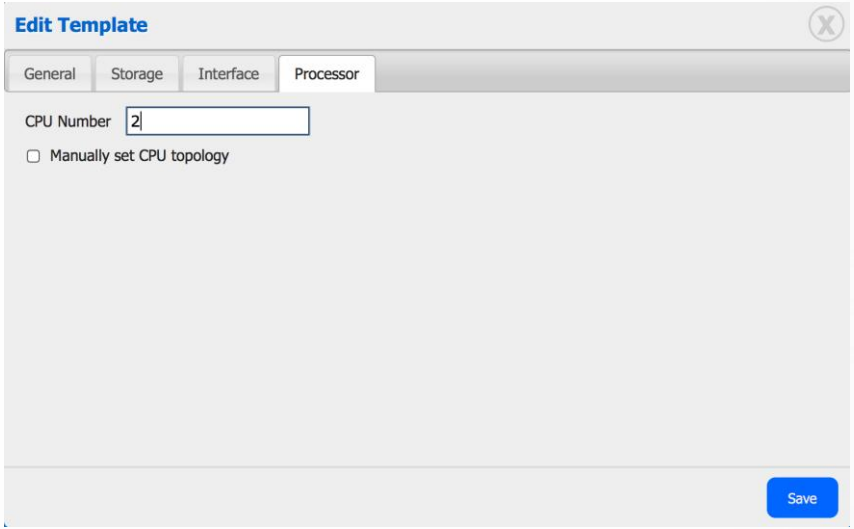


Using KIMCHI to create RHEL 7.1 Guest

Template Creation – Processor Settings

- Change the CPU Number to 2

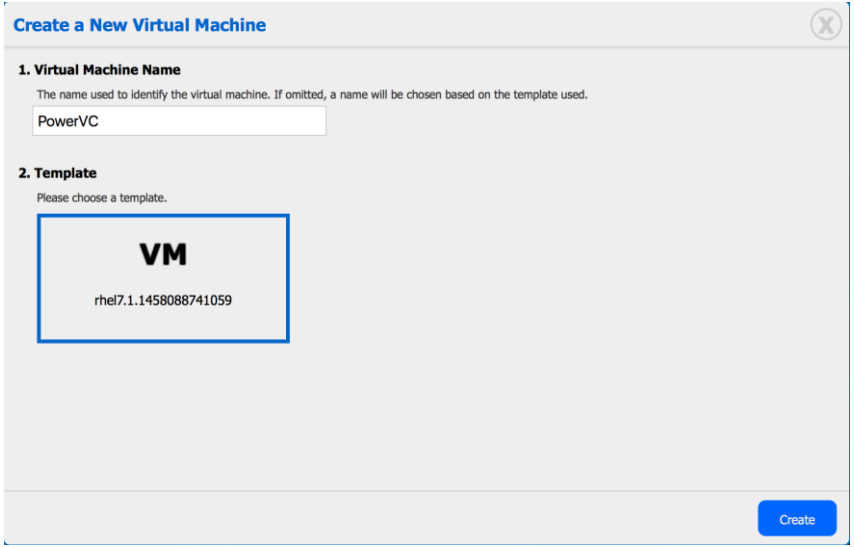
- Click on the <Save> button to save the template settings



Using KIMCHI to create RHEL 7.1 Guest

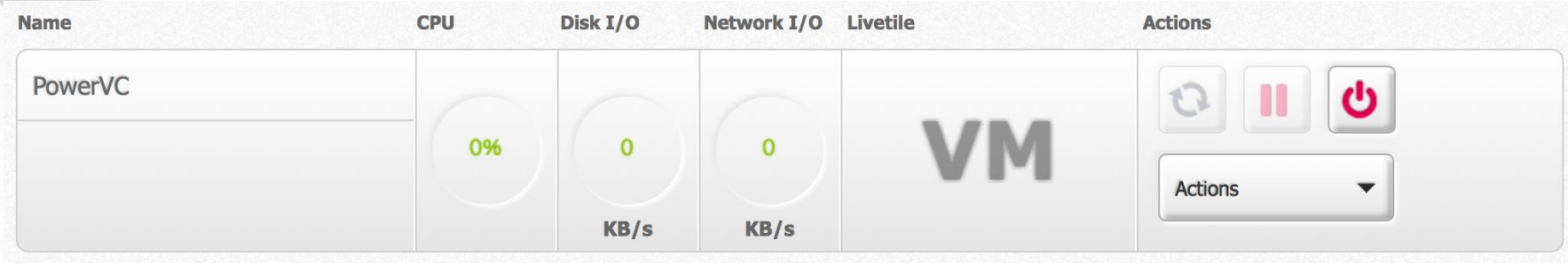
Guest Creation

- Select the '+' on the 'Guests' page
- Provide a name for the Guest
- Select the previously created template
- Click on <Create> to create the guest.

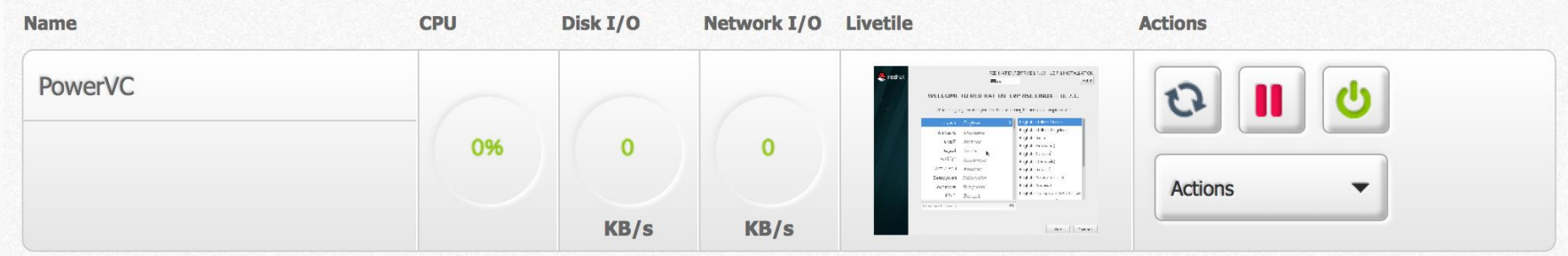


Using KIMCHI to create RHEL 7.1 Guest

- Guests, including the newly created guest, are displayed on the guests page



- Newly created guest can be started by selecting the power button or 'Start' from the 'Actions' drop-down

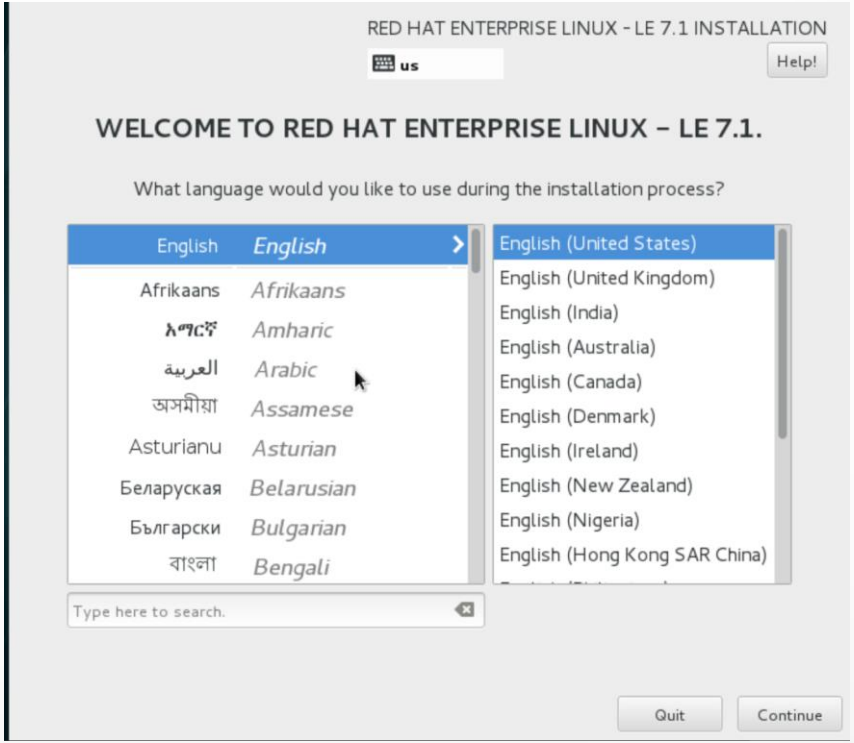


- Console can be accessed by clicking on the Livetile (NOTE: popups have to be enabled in the browser)

Installing RHEL 7.1

Language Selection (Installation)

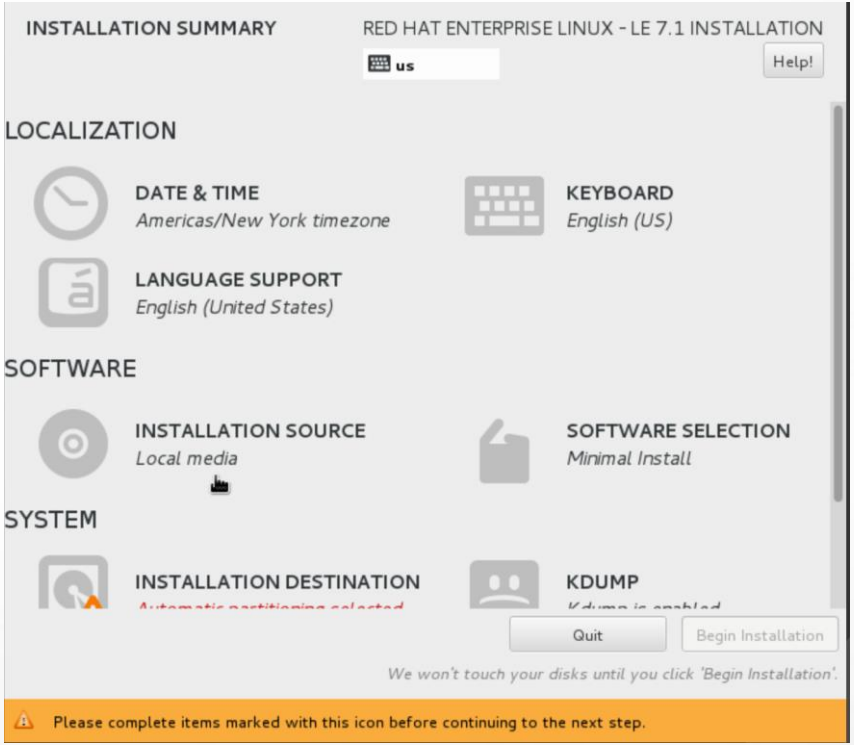
- Select the language to use for the installation
- Select <Continue>



Installing RHEL 7.1
Installation Summary

- For PowerVC the size of the swap partition needs to be changed

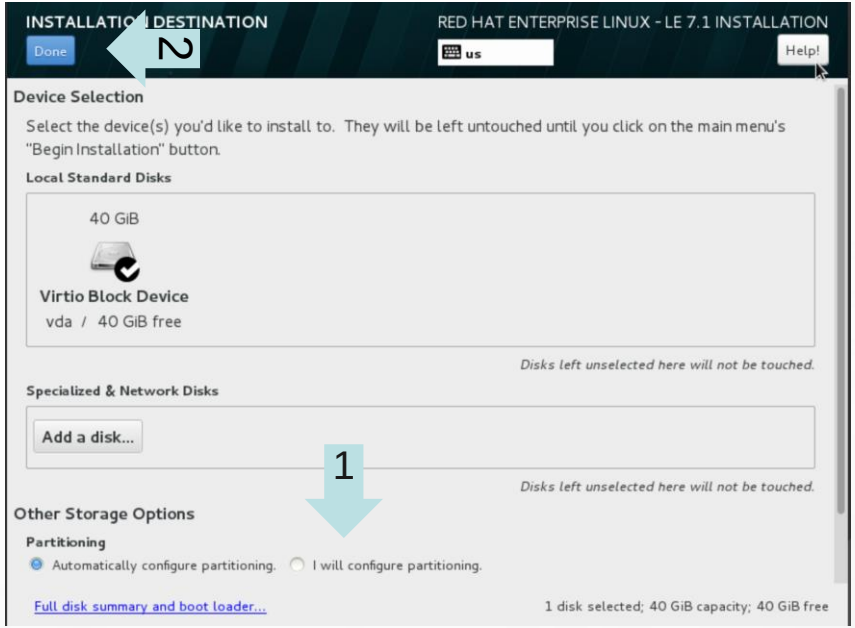
- This is accomplished by selecting ’Installation Destination’ from the ‘Installation Summary’ screen and using the steps provided on the following slides



Installing RHEL 7.1
Installation Destination

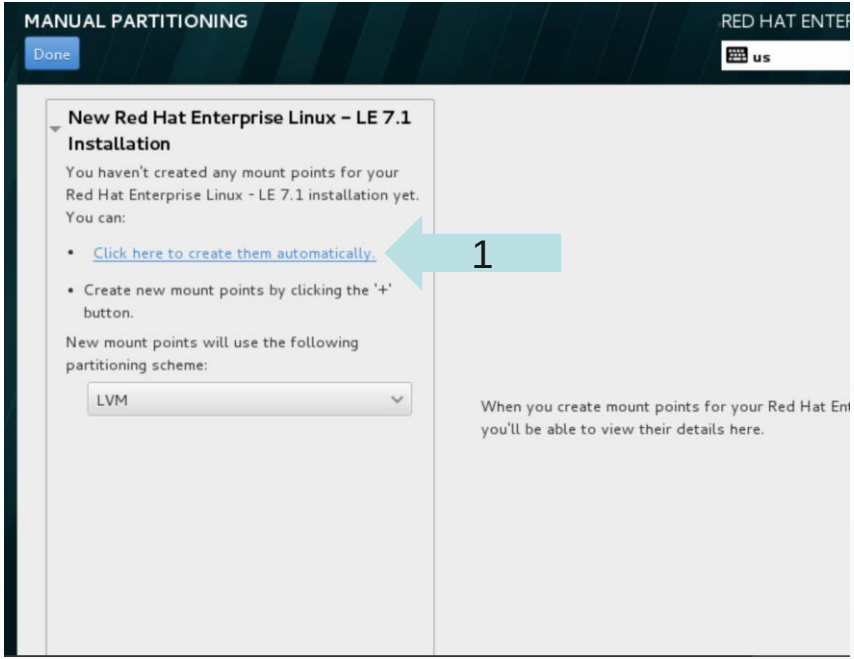
- To change the size of the ’swap’ partition select ‘I will configure partitioning’ under ‘Other Storage Options’

- Select <Done> to display the manual partitioning page



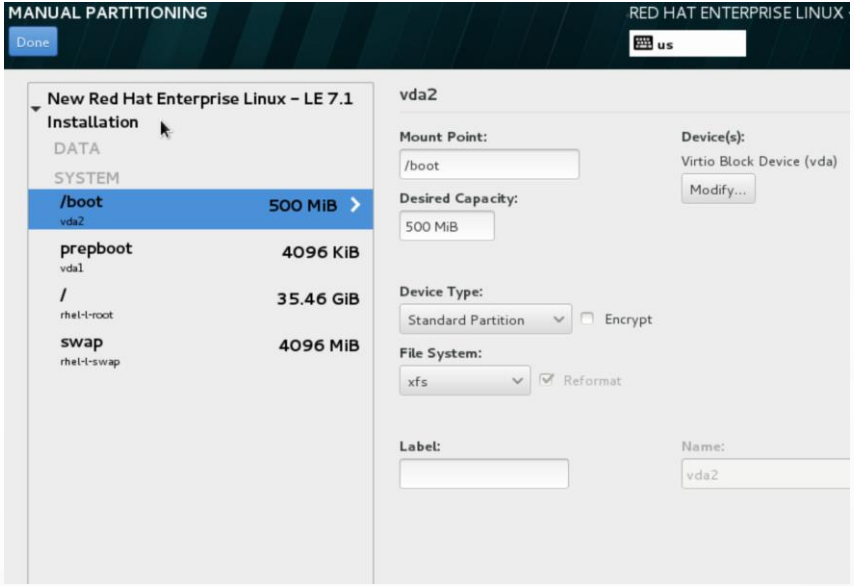
Installing RHEL 7.1
Manual Partitioning

- On the Manual Partitioning page select ‘Click here to create them automatically’
  - This will cause a default partition scheme to be created
  - The default partition scheme to be displayed and allow for changes to be made



Installing RHEL 7.1
Manual Partitioning

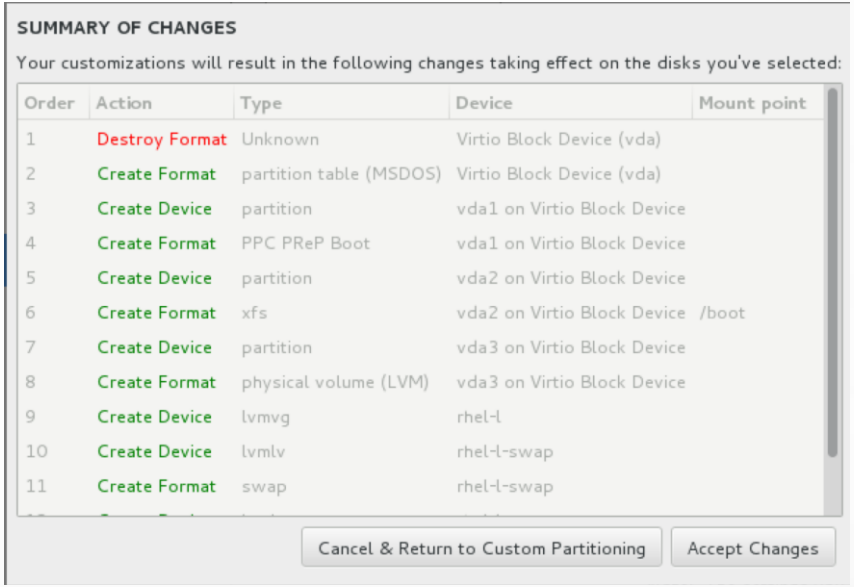
- The swap partition, which is used for virtual paging when physical memory is full, needs to be at least 10 GiB in size for the PowerVC installation
- To increase the size of swap, space needs to be freed up from another partition
- Select the / partition and reduce it's size to 28 GiB
  - NOTE; The change will be reflected when the next partition is selected
- Select the swap partition and change it's size to 11 GiB
- When the changes are completed select <Done>



Installing RHEL 7.1

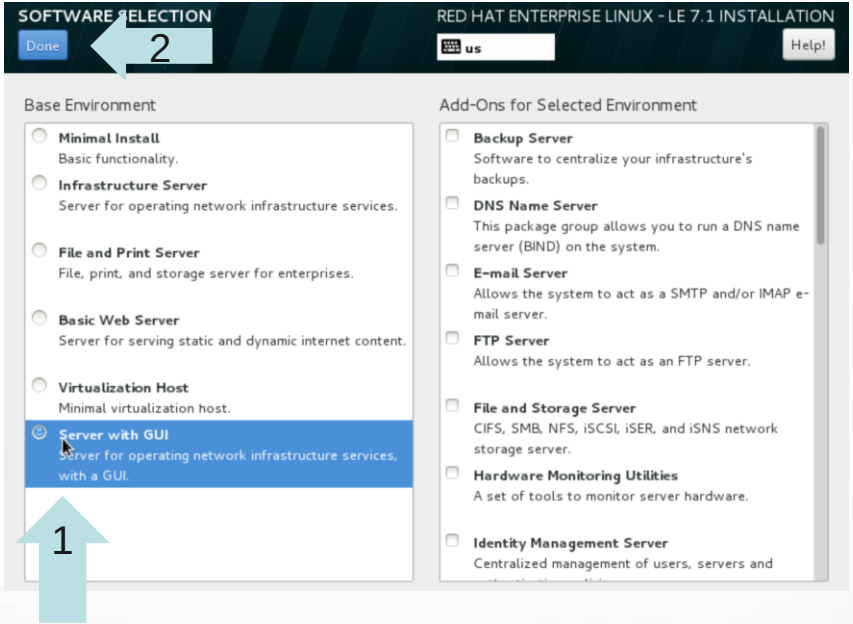
Summary of Changes (Disk Partitioning)

- The Summary of Changes dialog shows the partitioning steps that will take place as a result of the partitioning selections made.
- Select <Accept Changes> to confirm the partitioning for the RHEL installation



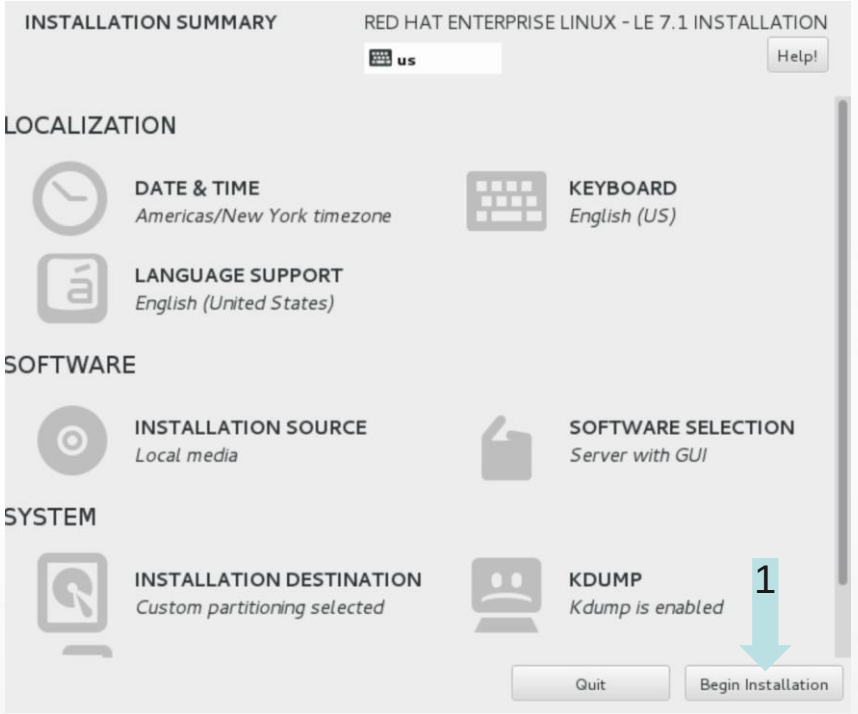
Installing RHEL 7.1
Software Selection

- On the Installation Summary page select the ‘Software Selection’ item
- From the ‘Software Selection’ screen select ‘Server with GUI’
- Select <Done>



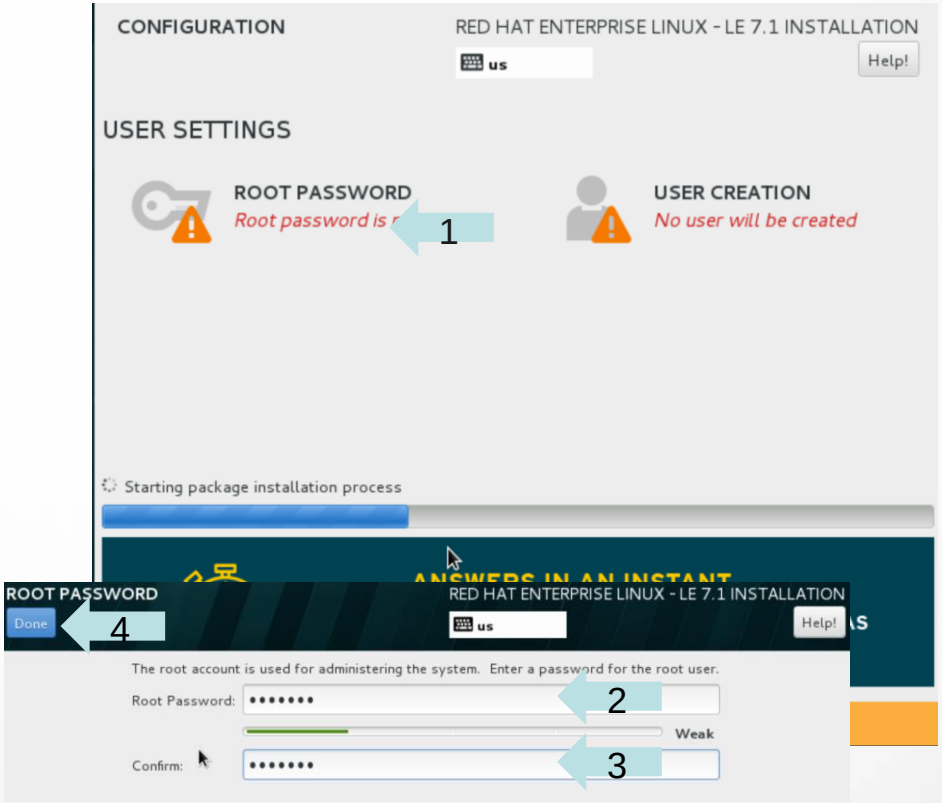
Installing RHEL 7.1
Installation Summary

- Once the previous changes have been completed select <Begin Installation> on the Installation Summary page



Installing RHEL 7.1
Configuration

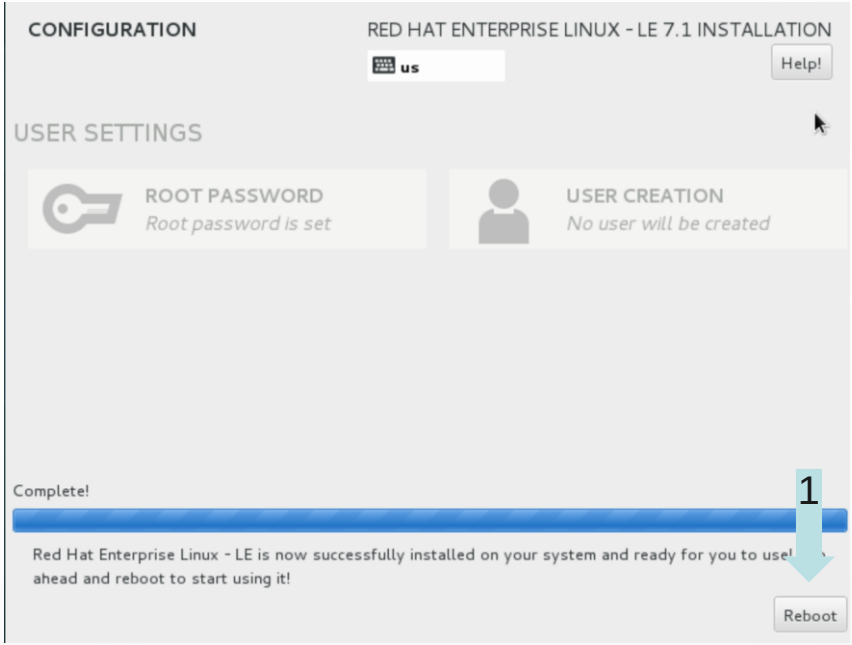
- During the installation of packages, the password for the root user needs to be set
- Select 'Root Password'
- Type (and confirm) the root password
- Select <Done>
  - NOTE: If a simple (weak) password is used then <Done> will need to be selected twice



Installing RHEL 7.1
Configuration

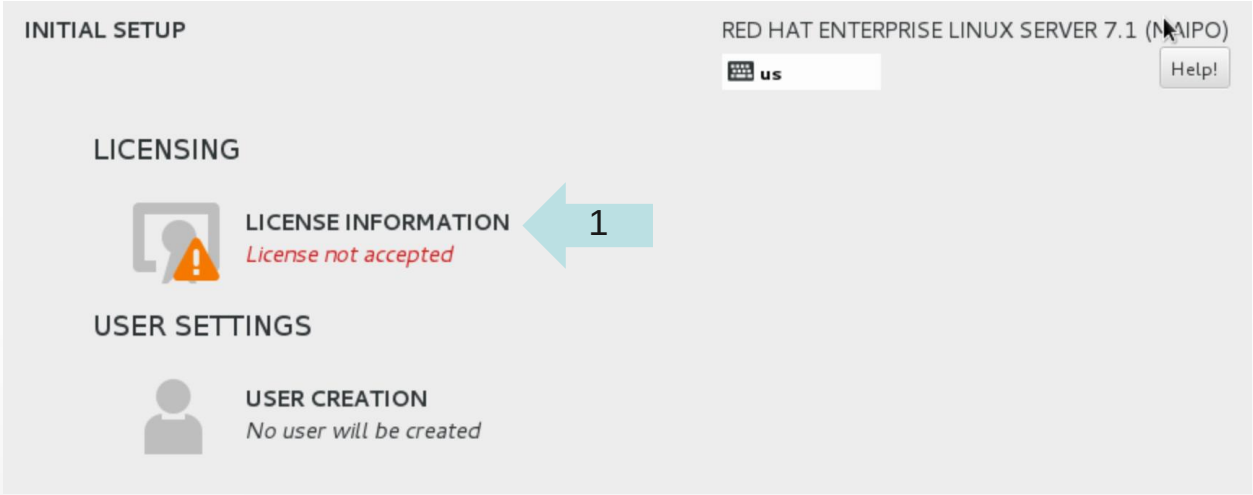
- Once the installation of packages is completed additional activities take place including
  - Post installation setup
  - Bootloader installation
  - initramfs generation

- Select the <Reboot> button



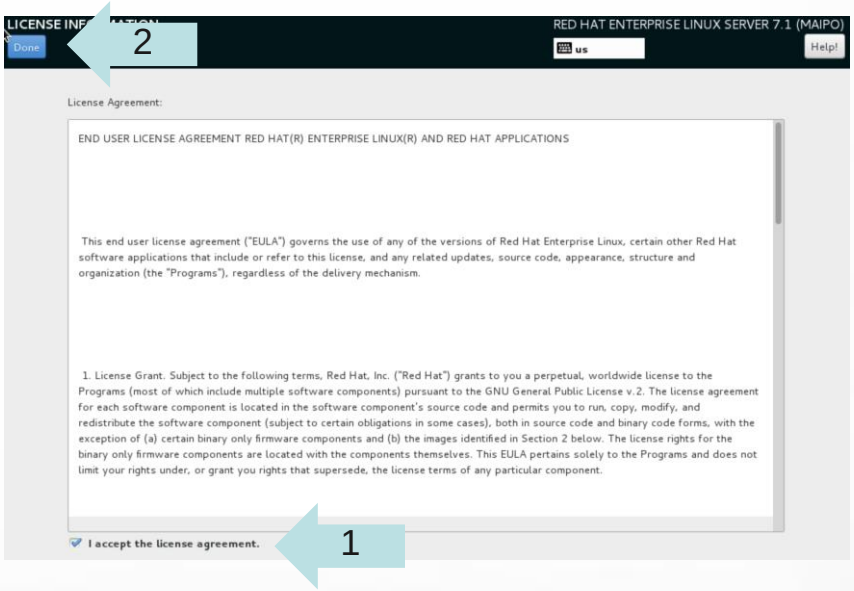
Installing RHEL 7.1
Initial Setup

- A number of initial setup steps need to be completed prior to using the newly installed Linux operating system
- Red Hat's License Agreement needs to be accepted, click on 'License Information'



Installing RHEL 7.1
License Information

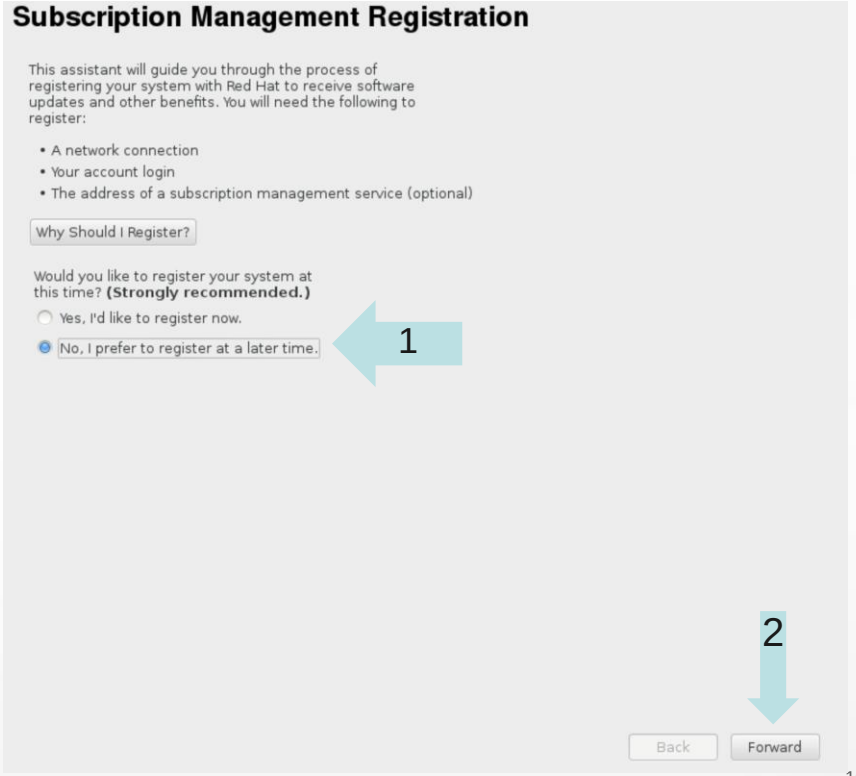
- Select ‘Accept the license agreement’
- Select <Done>
- Back on the ‘Initial Setup’ window select the <FINISH CONFIGURATION> button



Installing RHEL 7.1

Subscription Management Registration

- Since we don't have an active Red Hat Network subscript, select 'No, I prefer to register at a later time'
- Select <Forward>



Installing RHEL 7.1

- Select the desired Language
- Select <Next>

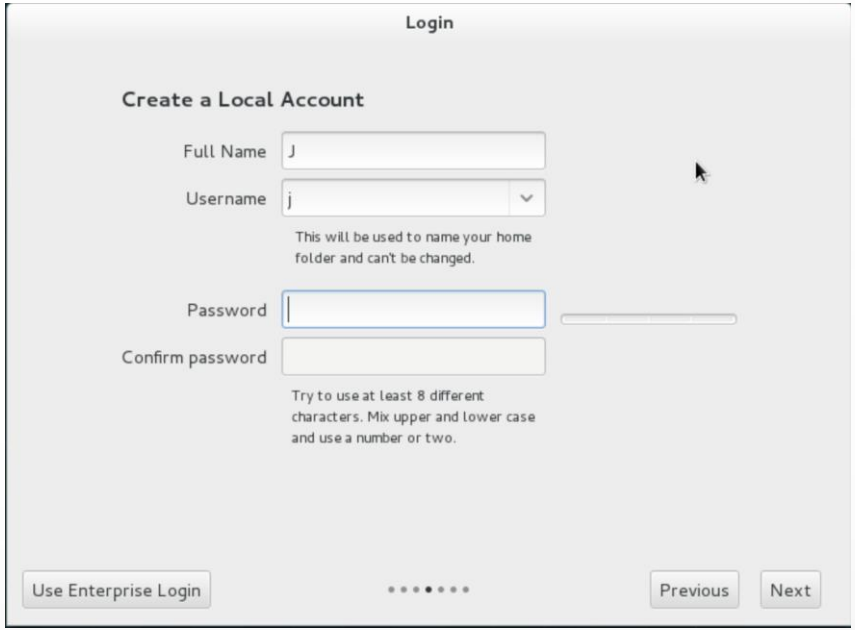
Installing RHEL 7.1
Input Sources

- Select <Next> to validate the detected input sources

Installing RHEL 7.1

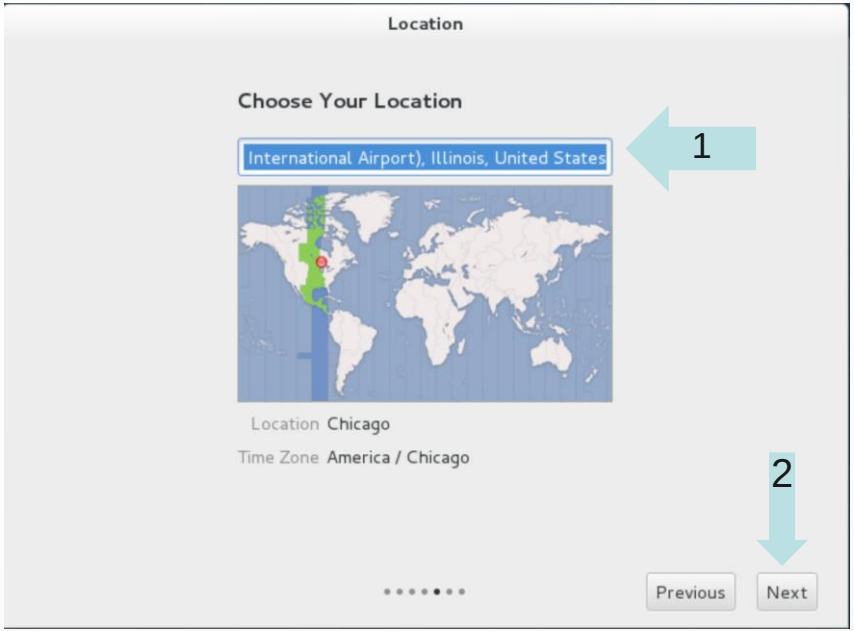
Create a Local Account

- The RedHat installer forces the creation of a new user
- Enter the required information
- Select <Next>



Installing RHEL 7.1
Location Selection

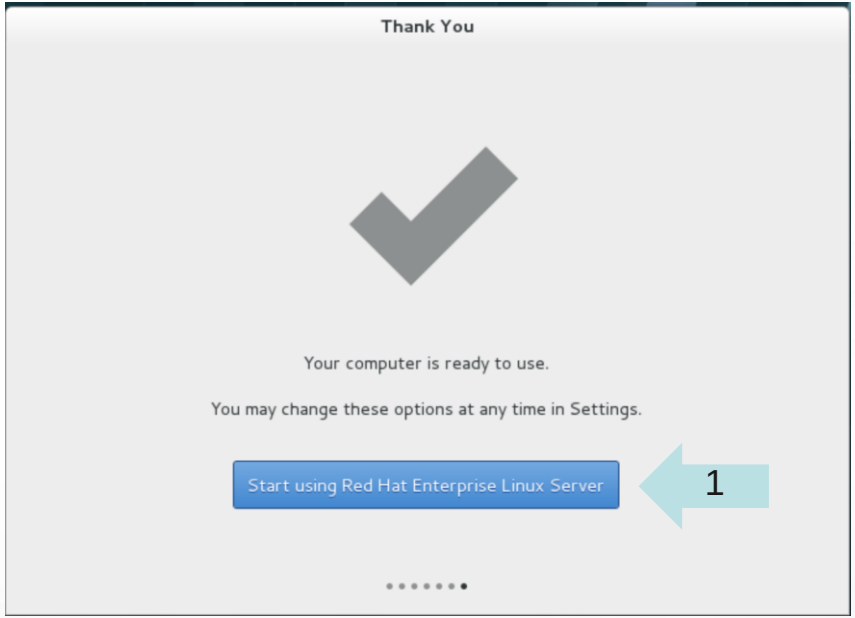
- Select the Location
- Select <Next>



Installing RHEL 7.1

Setup Complete

- Initial setup is complete and you can start using Linux by selecting the <Start using Red Hat Enterprise Linux Server> button



Learn more about Linux on Power Systems

- Power System Linux web site:
  http://www-03.ibm.com/systems/power/software/linux/
- The Linux on Power Community:
  www.ibm.com/developerworks/group/tpl/
- AIXpert blog:
  http://tinyurl.com/AIXpert
- Google+ Linux on Power Community:
  https://plus.google.com/communities/100156952249293416679

In Summary….

- There are many benefits to running Linux on Power whether you are coming from a Power background or an x86 background
  - For scale out workloads the default answer should not be Linux on x86. Linux on Power has comparable TCA, better TCO, better performance and reliability
  - In an existing Power environment Linux can be better suited for emerging / open source workloads

- Have a go at it!
  - Download a version of Linux and try it out on your existing hardware
  - Customers - don't have hardware but have a focus workload? Get us involved in a proof of concept
  - Unsure about your Linux strategy? Get us involved to look more closely at your existing workloads and their suitability for moving

Reference Charts

The IBM Power Systems L Line

	Key Client Value Proposition	Specifications
S812L 	• Entry price offering for Linux on POWER8 • Delivers equal performance to a 2 socket x86 server • Ideal for clients who do not want to virtualize	• 1-socket, 2U • Up to 12 cores (2.9-3.3Ghz) • 512 GB Memory (8 DIMMs) • 192 GB/sec sustained memory bandwidth • 12 x 2.5” SAS drives, 21.6TB Storage • 7 PCIe slots, 2 CAPI enabled • Default 3 yr 9x5 Warranty
S822L 	• Strong performance that can be maximized with virtualization • PowerVM Support allows easier datacenter management of heterogeneous OS environments • 65% Utilization Guarantee • 8 core fast offering, delivering 16 cores, SMT4 at 4.15 GHz for workloads requiring highest single thread performance • NEBS compliant option for TELCO	• 2-socket, 2U • Up to 24 cores (2.9-3.3Ghz) • 1 TB Memory (8 DIMMs) • 384 GB/sec sustained memory bandwidth • 12 x 2.5” SAS drives, 21.6TB Storage • 9 PCIe slots, 4 CAPI enabled • Default 3 yr 9x5 Warranty
S824L 	• Integrated GPU acceleration (2 K40s or K80s) targeting HPC with support for Ubuntu BareMetal • Non GPU acceleration with PowerVM support version available • 2 TB of memory available on system with Linux pricing model	• 2-socket, 4U • Up to 24 cores (2.9-3.3Ghz) • 2 TB Memory • 384 GB/sec sustained memory bandwidth • 18 x 2.5” SAS drives, 32.4TB Storage • 11 PCIe slots, 4 CAPI enabled • Default 3 yr 9x5 Warranty

The IBM Power Systems LC Line

KVM

	Key Client Value Proposition	Specifications
S812LC 	• Optimized for Hadoop and Spark • Largest internal storage in portfolio • Lowest priced offering delivering 1TB memory • Complete the same Spark workloads for less than ½ the cost of Intel Xeon E5-2690 v3 systems	• 1-socket, 2U • Up to 10 cores (2.9-3.3Ghz) • 1 TB Memory (32 DIMMs) • 115 GB/sec sustained memory bandwidth • 14x 3.5” drives, 84TB Storage • 4 PCIe slots, 2 CAPI enabled • Default 3 yr 9x5 Warranty 100% CRU
S822LC *Commercial* 	• Ideal for compute intensive data workloads in the cloud • 40% better price performance vs Xeon E5-2600 v3 based servers • Over 2X memory bandwidth than Xeon E5-2600 v3 with fully configured memory capacity	• 2-socket, 2U • Up to 20 cores (2.9-3.3Ghz) • 1 TB Memory (32 DIMMs) • 230 GB/sec sustained memory bandwidth • 2x SFF (HDD/SSD), 2 TB storage • 5 PCIe slots, 4 CAPI enabled,IB Add-in • Default 3 yr 9x5 Warranty 100% CRU
S822LC *High Performance Computing*	• 2 GPU accelerators integrated for HPC workloads • Over 2X memory bandwidth than Xeon E5-2600 v3 with fully configured memory capacity	• Deltas from S822LC - 2 integrated K80 NVIDIA GPUs - 3 PCIe slots, 2 CAPI enabled, IB Add-in

Linux Distro Support on POWER8 Models

Linux Distribution (Big Endian / Little Endian)	S812 LC S822 LC	S812 L S822 L	S824 L	S814 / S822 S824	S850	S870 S880
RED HAT LINUX						
Big Endian - Red Hat Linux 6.5 and updates		X	X (opt)	X		X
Big Endian - Red Hat Linux 6.6 and updates		X	X (opt)	X	X	X
Big Endian - Red Hat Linux 7 and updates		X (opt)	X (opt)	X (opt)		X (opt)
Big Endian - Red Hat Linux 7.1 and updates		X (opt)	X (opt)	X (opt)	X (opt)	X (opt)
Little Endian - Red Hat Linux 7.1 and updates		X (opt)	X (opt)	X (opt)	X (opt)	X (opt)
Little Endian - Red Hat Linux 7.2 and updates	X (opt)	X (opt)	X (opt)	X (opt)	X (opt)	X (opt)
SUSE LINUX						
Big Endian – SUSE Linux Enterprise 11 SP3 and updates		X		X		X
Big Endian – SUSE Linux Enterprise 11 SP4 and updates		X		X	X (opt)	X
Little Endian – SUSE Linux Enterprise 12 and updates		X (opt)	X (opt)		X (opt)	X (opt)
UBUNTU LINUX						
Little Endian – Ubuntu 14.04		X (opt)				
Little Endian – Ubuntu 14.04.2		X (opt)	X (opt)		X (opt)	X (opt)
Little Endian – Ubuntu 14.04.3 and updates	X (opt)	X (opt)	X (opt)		X (opt)	X (opt)

X (opt) - This version is optimized for POWER8 technology

Big Endian byte order support

Little Endian byte order support

Hypervisor Support on POWER8 Models

Linux / Hypervisor Distribution	S812 LC S822 LC	S812 L S822 L	S824 L	S814 S822 S824	S850	S870 S880
RED HAT LINUX						
PowerVM		X	X	X	X	X
PowerKVM	X	X	X			
SUSE LINUX						
PowerVM		X	X	X	X	X
PowerKVM	X	X	X			
UBUNTU LINUX						
PowerVM		X	X	X	X	X
PowerKVM	X	X	X			

POWER8 Endian Support

	Big-Endian	Little-Endian
AIX	AIX 6.1, AIX 7.1	N/A
IBM i	IBM i 7.1, IBM i 7.2	N/A
SUSE	SLES11	SLES12
Red Hat	RHEL 6.5 - RHEL 7.0, RHEL 7.1, RHEL 7.2	RHEL 7.1, RHEL 7.2
Canonical	N/A	Ubuntu 14.04 / 15.04
PowerVM	Mixed endian LPAR support as of FW8.30	
PowerKVM	Mixed endian guest support as of 10/2014	

Note: See IBM documentation for specific OS/Hypervisor release support information