Manual Host and Guest Installation under KVM for IBM z Systems

Holger Wolf - Holger.Wolf@de.ibm.com - KVM on z Test Lead

3rd February 2016

Agenda

- The environment
- Preparation Host Installation
- <Live Demo> Perform Host Installation
- Preparation Guest Installation
- <Live Demo> Perform Guest Installation

Considerations for Guest and Host Installation

Define disk type the guest should be installed
- Native SCSI or Image
- QCOW2, RAW → might need huge SCSI disk or LVM

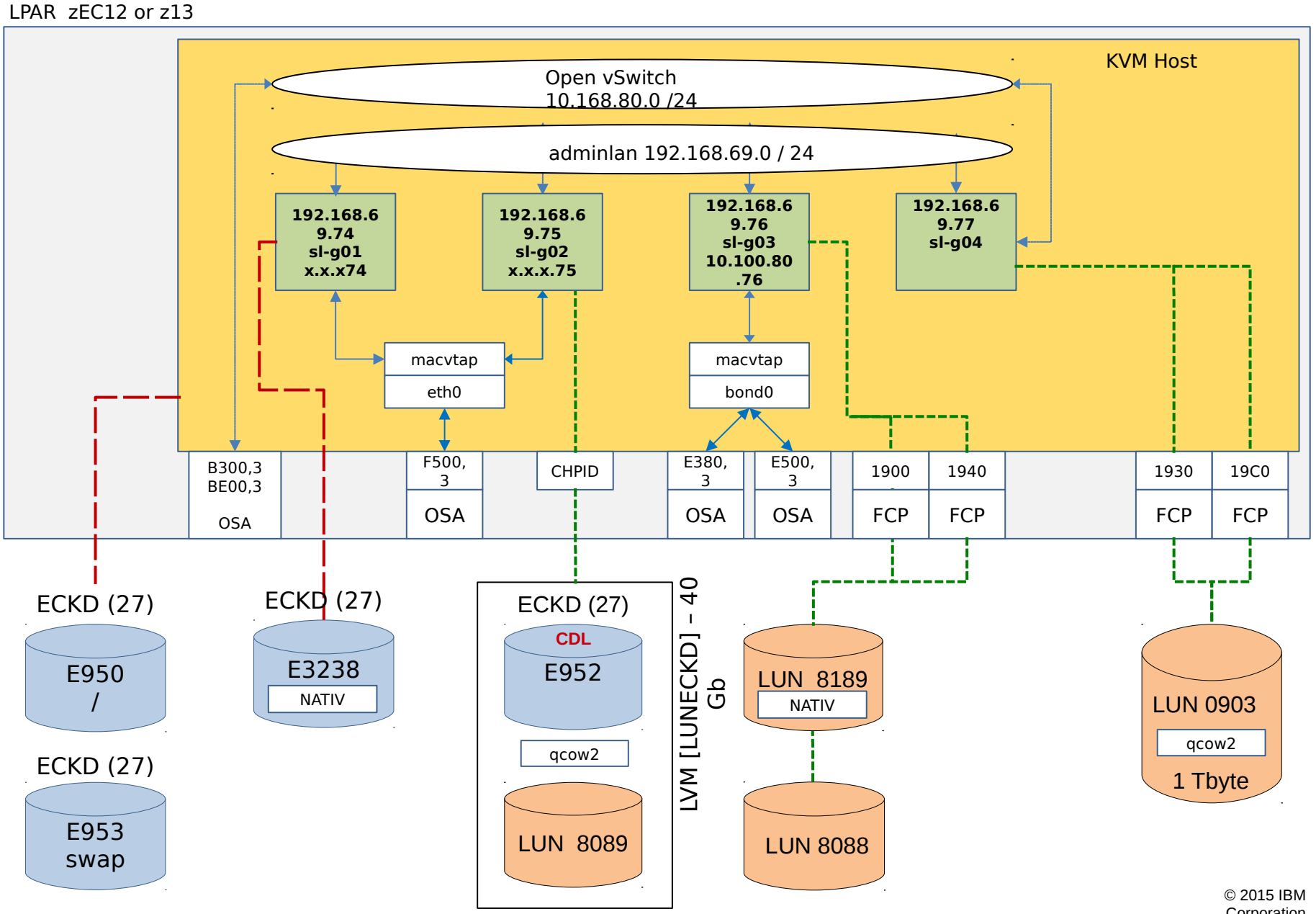
What network interface should be used connecting the guest to the world
- Which OSA card?
- Should they be bonded

Guest connected via OpenVSwitch networking or MacVTap
- You should not use any MAC-Address entries in xml-file
- Take care don't use a single interface, use BOND-Interface
- (for BOND configuration see “KVM Virtual Server Managment-Guide”)
- For OpenVSwitch consider to have latest firmware level installed

How many CPUs and Memory should your guest(s) have
- This drives usage of memory and CPU of your host

For Manual guest installation, a ftp-server setup is needed
- No difference to LPAR installation of distribution or KVM-host

KVM for z Systems – Sample Setup



Preparation for KVM host installation

- Carefully read the KVM for IBM z Systems: Planning and Installation Guide!
  - See: http://www-01.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/1/897/ENUS215-261/index.html&request_locale=en#epubx

- Setup a ftp server for installation which is reachable by your LPAR and/or Service Element
  - The requirement is the same as for existing Distributions like SLES / RHEL
  - If ftp access for Service Element is not allowed – DVD boot is a option

- Prepare space on the ftp server, upload and mount ISO
  - Mount KVM install DVD (iso file) appropriate:
    - mount -o ro,loop KVMIBM-1.1.0-20150806-s390x-dvd1.iso <KVM directory>
  - Copy content from DVD to ftp server

Preparation for KVM host installation

- Setup install infrastructure
  - Ftp server content should look like the following

```
.discinfo  generic.ins  images  Packages  repodata  s38lp78.ins  TRANS.TBL  .treeinfo
```

    - Generic.ins file available by default → created s38lp78.ins for installation Demo

  - Content of image directory looks like the following

```
generic.prm  initrd.addrsize  initrd.img  install.img  kernel.img  s38lp78.prm  TRANS.TBL
upgrade.img
```

- Test ftp setup with the following command:
  - curl ftp://<your ftp user>:<your ftp password>@<your ip>/<KVM dir>/.treeinfo

    curl ftp://ftpu:secure@192.168.0.76/ga/.treeinfo

  - If curl command fails → check selinux setup for ftp server

Update 1.1.0 Installer from Fixpack 1.1.0.3 or later

- There is a installer update available
  - Use the following Link to download the latest fixpack
    - http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=Virtualization%2Bsoftware&product=ibm/Other+software/KVM+for+IBM+z+Systems&release=1.1.0&platform=All&function=all

- Replace the installer as delivered from the GA ISO with the one on the Fixpack ISO
  - Follow the instruction procedure in the ISO README
    - See section “to copy installer.img from the ISO to your FTP server.”

Preparation for KVM host installation – INS AND PRM FILES

- s38lp78.ins → install file used by HMC/SE

```
* minimal lpar ins file
images/kernel.img 0x00000000
images/initrd.img 0x02000000
images/s38lp78.prm 0x00010480
images/initrd.addrsize 0x00010408
```

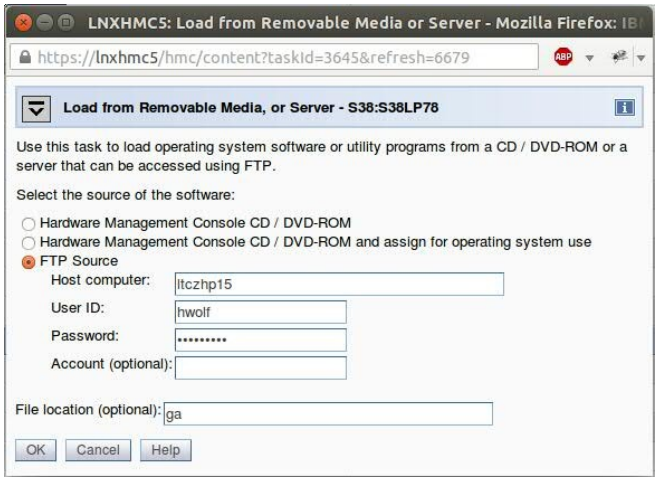
- S38lp78.prm defines kernel boot parameters for initial boot of installation system
  - ro ramdisk_size=20000 rd.zfcp=0.0.1900,0x50050763055341ae,0x4080409a00000000 rd.zfcp=0.0.1980,0x50050763055341ae,0x4080409a00000000 rd.zfcp=0.0.1900,0x50050763055041ae,0x4080409b00000000 rd.zfcp=0.0.1980,0x50050763055041ae,0x4080409b00000000 rd.dasd=0.0.3f8a-0.0.3f8d rd.znet=qeth,0.0.f500,0.0.f501,0.0.f502,layer2=1 ip=192.168.0.78::192.168.0.1:255.255.255.0:s38lp78:enccw0.0.f500:none inst.repo=ftp://ftpu:secure@192.168.0.76/ga

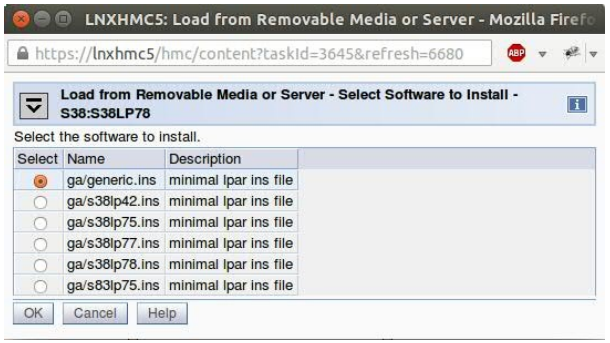
  - PRM file considerations:
    - No <CR>! Everything must be in one line!
    - Take care of character case – use lowercase

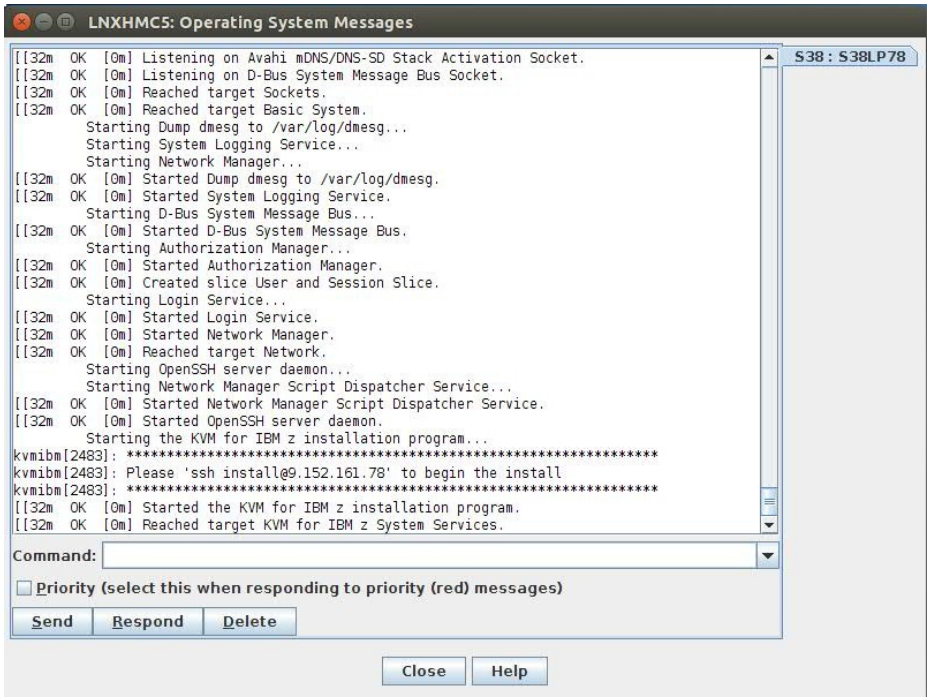
      0.0.3F8a is not the same as 0.0.3f8a

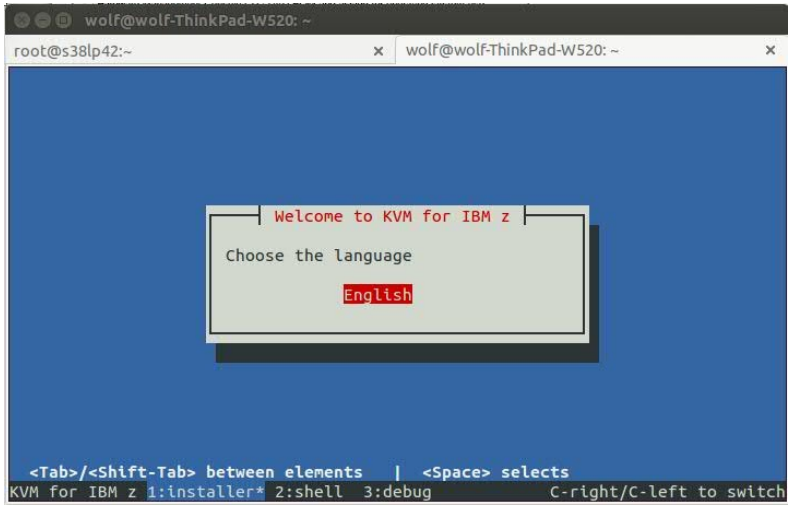
  - Dynamic change of DASD are not support – only zFCP
  - zFCP devices are only recognized in a multipath setup
  - In case of an error in the commandline → boot will fail → Check your PRM file again!

Host Installation DEMO









Steps to a successful guest installation on KVM for IBM z systems

1. Setup disk devices
2. Prepare one or more networks or using existing
   - Be aware that the KVM default network prevents connections to your guest
3. Create guest(s) definition (based on libvirt xml) using predefined kernel for installation
4. Install SLES 12 on the guest via ftp-server
5. Stop guest
6. Change guest(s) definition
7. Starting the guest

Prepare disk on Host system: Storage -qcow2 and raw files

- How to create a qcow2-, or raw-image

User libvirt command qemu-img

```
qemu-img <function> <type> <directory and name> <size in GB>

function: Create, info, resize
type: raw, qcow2
```

Example for a qcow2 image with minimum size of 10 GB:

```
kvmhost# qemu-img create -f qcow2 /home/images/suse120_qcow2.img 10G
```

Example for a raw image maximum size of 10 GB:

```
kvmhost# qemu-img create -f raw /home/images/suse120_raw.img 10G
```

Prepare disk on Host system: Storage DASD

- **Currently ECKD disks can not be used there is a bug which needs manual intervention – Therefore ECKD is not recommended as direct attached guest root disk.**

- How to prepare a ECKD-DASD

  - You have two possibilities to format a ECKD-DASD
    - `type=cdl (compatible disk layout)`
    - `Type=ldl (linux disk layout)`

  - There are just some problems with cdl on SLES 12 and cdl formatted ECKD Disks

  - We use for native BLOCK-Device cdl format
  - `dasdfmt -b 4096 -d cdl -y -P /dev/dasdc`

  - Create on partition on the whole ECKD-Device use fdasd
  - `fdasd /dev/dasdc`

  - `For ldl multiple partitions will not be supported`

Writing a domain XML file

- Write a domain XML file by a text editor

```
kvmhost# vim sl-g01.xml
```

- Define your guest (example file sl-g01.xml)

```
kvmhost# virsh define sl-g01.xml
Domain sl-g01 defined from sl-g01.xml
```

XML-File example – qcow2 disks on scsi

```
<domain type='kvm'>
  <name>sl-g01</name>
  <description>Guest-System SLES 12.0 qcow2 macvtap </description>
  <memory>524288</memory>
  <vcpu>2</vcpu>
  <iothreads>2</iothreads>
  <os>
    <type arch='s390x' machine='s390-ccw-virtio'>hvm</type>
    <kernel>/home/images/cd.ikr</kernel>
    <initrd>/home/images/initrd</initrd>
    <cmdline>linuxrcstderr=/dev/console</cmdline>
    <boot dev='hd'/>
   </os>
   <clock offset='utc'/>
   <on_poweroff>destroy</on_poweroff>
   <on_reboot>restart</on_reboot>
   <on_crash>preserve</on_crash>

   <devices>
    <emulator>/usr/bin/qemu-system-s390x</emulator>
    <disk type='file' device='disk'>
      <driver name='qemu' type='qcow2' cache='none' io='native' iothread='1'/>
      <source file='/home/images/suse120_qcow2.img'/>
      <target dev='vda' bus='virtio'/>
      <address type='ccw' cssid='0xfe' ssid='0x0' devno='0x0003'/>
    </disk>
    <disk type='file' device='disk'>
      <driver name='qemu' type='qcow2' cache='none' io='native' iothread='1'/>
      <source file='/home/images/suse120_2_qcow2.img'/>
      <target dev='vdb' bus='virtio'/>
      <address type='ccw' cssid='0xfe' ssid='0x0' devno='0x0004'/>
    </disk>
    <interface type='direct'>
        <mac address='02:BE:EF:83:22:69'/>
        <source dev='enccw0.0.f500' mode='bridge'/>
        <model type='virtio'/>
        <address type='ccw' cssid='0xfe' ssid='0x0' devno='0x0002'/>
    </interface>
    <console type='pty'>
      <target type='sclp' port='0'/>
    </console>
  </devices>
</domain>
```

XML-File example - scsi

```
<domain type='kvm'>
  <name>sl-g01</name>
  <description>Guest-System SLES 12.0 ftp to scsi macvtap </description>
  <memory>524288</memory>
  <vcpu>2</vcpu>
  <iothreads>1</iothreads>
  <os>
    <type arch='s390x' machine='s390-ccw-virtio'>hvm</type>
    <kernel>/home/images/cd.ikr</kernel>
    <initrd>/home/images/initrd</initrd>
    <cmdline>linuxrcstderr=/dev/console</cmdline>
    <boot dev='hd'/>
   </os>
   <clock offset='utc'/>
   <on_poweroff>destroy</on_poweroff>
   <on_reboot>restart</on_reboot>
   <on_crash>preserve</on_crash>

  <devices>
    <emulator>/usr/bin/qemu-system-s390x</emulator>
    <disk type='file' device='disk'>
      <driver name='qemu' type='raw' cache='none' iothread='1'/>
      <source file='/dev/mapper/36005076305ffc1ae0000000000008053'/>
      <target dev='vda' bus='virtio'/>
      <address type='ccw' cssid='0xfe' ssid='0x0' devno='0x8053'/>
    </disk>
    <interface type='direct'>
      <mac address='02:00:10:83:22:68'/>
      <source dev='enccw0.0.f500' mode='bridge'/>
      <model type='virtio'/>
      <address type='ccw' cssid='0xfe' ssid='0x0' devno='0x0001'/>
    </interface>
    <console type='pty'>
      <target type='sclp' port='0'/>
    </console>
  </devices>
</domain>
```

XML-File example – scsi - simplyfied

```
<domain type='kvm'>
  <name>sl-g04</name>
  <description>Guest-System SLES 12.0 ftp to scsi macvtap </description>
  <memory>524288</memory>
  <vcpu>2</vcpu>
  <iothreads>1</iothreads>
  <os>
    <type arch='s390x' machine='s390-ccw-virtio'>hvm</type>
    <kernel>/home/images/cd.ikr</kernel>
    <initrd>/home/images/initrd</initrd>
    <cmdline>linuxrcstderr=/dev/console</cmdline>
    <boot dev='hd'/>
   </os>
   <clock offset='utc'/>
   <on_poweroff>destroy</on_poweroff>
   <on_reboot>restart</on_reboot>
   <on_crash>preserve</on_crash>

  <devices>
    <emulator>/usr/bin/qemu-system-s390x</emulator>
    <disk type='file' device='disk'>
      <driver name='qemu' type='raw' cache='none' iothread='1'/>
      <source file='/dev/mapper/36005076305ffc1ae0000000000008052'/>
      <target dev='vda' us='virtio'/>
    </disk>
    <interface type='direct'>
      <source dev='enccw0.0.f500' mode='bridge'/>
      <model type='virtio'/>
    </interface>
    <console type='pty'>
      <target type='sclp' port='0'/>
    </console>
  </devices>
</domain>
```

install from ftp-server

- Setup a ftp server and place content of SLES12 DVD there reachable from KVM host
- Copy Installation Kernel and INITRD to the KVM host
  Files located on SLES12 DVD in Directory *boot*:
    cd.ikr
    initrd
- Change guest xml to boot from local installation kernel/initrd (cd.ikr/initrd)
  See next page

install from ftp-Server

- Kernel boot Option in xml

```
<domain type='kvm'>
     :
     :
 <os>
    <type arch='s390x' machine='s390-ccw-virtio'>hvm</type>
<!-- Kernel boot   -->
   <kernel>/var/lib/libvirt/images/s12-kernel.boot</kernel>
    <initrd>/var/lib/libvirt/images/s12-initrd.boot</initrd>
    <cmdline>linuxrcstderr=/dev/console</cmdline>
  </os>
```

For Installation follow SUSE Documenation as for LPAR or z/VM

- Installation sequence beginning ftp

```
Domain sl-g01 started
Connected to domain sl-g01
Escape character is ^]
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Initializing cgroup subsys cpuacct
…
no SUSE Linux Enterprise 12 SP1 repository found
Automatic setup not possible.

Please make sure your installation medium is available.

Choose the URL to retry.

0) <-- Back <--
1) cd:/
2) hd:/
3) Enter another URL

> 0
>>> linuxrc 5.0.61 (Kernel 3.12.49-11-default) <<<

Main Menu

0) <-- Back <--
1) Start Installation
2) Settings
3) Expert
4) Exit or Reboot

> 1
```

IMPORTANT after SUSE installation is finished

- cio_ignore is set
  for the available devices at installation time

- when adding new devices change cio_ignore configuration or switch off

After change of a Guest XML and make change active

- If guest started stop guest

```
virsh console sl-g01 (login and say halt)
or hard way
kvmhost# virsh destroy sl-g01
Domain sl-g01 destroyed
```

- Change guest with virsh edit

```
kvmhost# virsh edit sl-g01
Domain sl-g01 XML configuration edited.
```

- Save changed guest XML

```
kvmhost# virsh dumpxml sl-g01 > sl-g01.xml
```

- Start your guest after installation

```
kvmhost# virsh start sl-g01 --console
Domain sl-g01 started
```

Virsh undefine vs. edit

- Virsh undefine unregisters the guest uuid and id gets deleted
- With virsh define - new internal definitions are created
  like uuid/id and MAC address
- Recommendation is to create the guest and use virsh edit to modify.
- To hold the changes persistent use virsh dumpxml <guest> to save definition.

Available Documents

- Planning and Installation Guide (SC27-8236)
- System Administration Guide (SC27-8237)
- Installing SUSE Linux Enterprise Server 12 as a KVM Guest (SC34-2755)
- Virtual Server Management (SC34-2752)
- Virter Server Quick Start (SC34-2753)
- Device Drivers, Features, and Commands for LINUX as a KVM Guest (SC34-2754)

Any Questions ???

In case of questions - contact

- **Holger.Wolf@de.ibm.com**