AIX 2020 Lessons Learned
By Jaqui Lynch

2020 has certainly been quite the year, but this year I have been working on a number of projects where I have had to do some new things. Below is a list of some of the things I have learned this year that I hope you find useful.

Nigel Griffith's AIXpert Blog

https://www.ibm.com/support/pages/aixpert-blog-nigel-griffiths-mrnmon
Nigel deserves his own paragraph here. I regularly check his AIXpert Blog to see what tips he has to share. While there are many useful articles on his blog Ihave picked a couple that I have found really useful in the past few weeks.

I'm frequently asked how to ensure data is removed from disks before the disks are removed from the system or when servers are being migrated. One answer is to smash the disks with a hammer or drill a hole through them. While that is very therapeutic it is also unnecessary if you scrub them properly. Nigel has provided a small C program (use at your own risk) that scrubs your disks.
https://www.ibm.com/support/pages/node/1136086

Recently I had to install Samba on AIX on a system that had been around for a long time. Many third-party rpms had been installed outside of the toolbox and it was a mess to clean up. I really wanted to use yum as it does a phenomenal job of taking care of prerequisites, something that is challenging using just rpm. I found the article below that provided the necessary steps to clean up my system and I now have a pristine system that uses only RPM packages from the toolbox that are managed by yum.
https://www.ibm.com/support/pages/aix-old-open-source-rpm-packages-clean-then-use-yum
This fix will require downtime as you are uninstalling and reinstalling third party packages. I highly recommend you take a backup and I usually use alt_disk_copy to take a clone to a spare disk so I have a fast fail back.

Rootvg clone

I have written on using alt_disk_copy before to take a backup. I use this often when doing maintenance as I can go back to the system before I made any chances by changing my bootlist and rebooting. This is much faster than restoring from a mksysb and I use it all the time. I do still take a mksysb though. If my rootvg is on hdisk0 and my free disk is hdisk1 then I would clone using:

```
alt_disk_copy -B -V -d hdisk1
```

The -V means verbose output and the -B means do not change the bootlist
I would still check the bootlist using:

```
bootlist -m normal -o
```

On my NIM server I see:

```
aix1nim:/> bootlist -m normal -o
hdisk1 blv=hd5 pathid=0
```

```
hdisk1 blv=hd5 pathid=1
hdisk1 blv=hd5 pathid=2
hdisk1 blv=hd5 pathid=3
hdisk1 blv=hd5 pathid=8
```

The bootlist command above only shows 5 entries—if you're mirrored or have lots of paths then you will only see the first 5. This is due to a limit on the bootlist size. As you can see below the limit show by the -v flag is 0x5 or 5 devices.

If you want more detailed information on the bootlist you can add a -v to use verbose mode

```
bootlist -m normal -o -v
```

On my NIM server this showed:

```
bootlist -m normal -o -v
'ibm,max-boot-devices' = 0x5
NVRAM variable: (boot-device=/vdevice/vfc-
client@30000046/disk@500507680d048ef6,1000000000000:2 /vdevice/vfc-
client@30000046/disk@500507680d048ef7,1000000000000:2 /vdevice/vfc-
client@30000047/disk@500507680d088ef6,1000000000000:2 /vdevice/vfc-
client@30000047/disk@500507680d088ef7,1000000000000:2 /vdevice/vfc-
client@3000005a/disk@500507680d108ef6,1000000000000:2)
Path name: (/vdevice/vfc-
client@30000046/disk@500507680d048ef6,1000000000000:2)
match_specific_info: ut=disk/fcp/mpioosdisk
hdisk1 blv=hd5 pathid=0
Path name: (/vdevice/vfc-
client@30000046/disk@500507680d048ef7,1000000000000:2)
match_specific_info: ut=disk/fcp/mpioosdisk
hdisk1 blv=hd5 pathid=1
……
```

If you need to have different disks in the bootlist but there are too many paths you can specify the specific disks and paths using bootlist as follows:

```
bootlist -m normal hdisk0 blv=hd5 pathid=0,1
bootlist -mode normal hdisk0 pathid=0,1 hdisk1 pathid=0,1
```

You can find the paths for an hdisk using:

```
lspath -t -l hdisk1
```

and also using:

```
lspath | grep “hdisk1 “
```

The following two links provide information on this:
https://www.ibm.com/support/pages/aix-boot-path-enablement

https://www.ibm.com/support/pages/why-vios-bootlist-does-not-list-all-path-ids-both-mirrored-san-deviceshdisks

Gareth Coates' Tips and Tricks

https://www.ibm.com/support/pages/node/1116939

I have attended many of Gareth's sessions on Tips and Tricks. He has kindly gathered them together on this webpage. Recently I was getting annoyed with my HMC as it was still showing the wrong release after I upgraded my VIO servers and NIM server. I have seen this before and it usually comes right after a while but not this time. The first link on Gareth's page told me how to solve this.

https://www.ibm.com/support/pages/node/6254331

On the HMC you use the –osrefresh flag as follows:

```
lssyscfg  -m  servername  -r lpar -F os_version  --osrefresh
```

Gareth also provides a script that refreshes the oslevel for all lpars on a server and that can be executed by using ssh to the HMC.

lsmpio

Another useful command is the lsmpio command. When I create storage luns in the storage subsystem I give them meaningful names. For example, my NIM boot disk is labelled NIM_100G_rtvg. I can use lsmpio to check disks before I did anything with them to make sure I am using the right disk and that it's the right size.

On my NIM server I am booted from hdisk1

```
#lsmpio -ql hdisk1
Device:  hdisk1
          Vendor Id:  IBM
         Product Id:  2145
           Revision:  0000
           Capacity:  100.00GiB
       Machine Type:  2078
       Model Number:  124
         Host Group:  P8NIM
        Volume Name:  NIM_100GB_rtvg
      Volume Serial:  60050763808100F7000000000000000A  (Page 83 NAA)
```

From this I can see it is 100GB and it is on a 2078-124 and it provides the volume name and the volume serial.

If you leave off the 1 it shows you the paths and their status

```
#lsmpio -l hdisk1
name    path_id  status   path_status  parent  connection

hdisk1  0        Enabled  Non          fscsi0  500507680d048ef6,1000000000000
hdisk1  1        Enabled  Sel,Opt      fscsi0  500507680d048ef7,1000000000000
```

```
hdisk1  2       Enabled  Non        fscsi1  500507680d088ef6,1000000000000
hdisk1  3       Enabled  Sel,Opt     fscsi1  500507680d088ef7,1000000000000
hdisk1  8       Enabled  Non        fscsi2  500507680d108ef6,1000000000000
hdisk1  9       Enabled  Sel,Opt     fscsi2  500507680d108ef7,1000000000000
hdisk1  10       Enabled  Non        fscsi3  500507680d0c8ef6,1000000000000
hdisk1  11       Enabled  Sel,Opt     fscsi3  500507680d0c8ef7,1000000000000
```

lsmpio with no flags shows you the status for all paths to all disks in the same format as just lsmpio -l above.

POWER9 and Adapter Firmware

For many years I have been encouraging people to keep their server and adapter firmware up to date. This has now become even more critical as we move into power9 at the 940 and higher firmware levels.

For FW940 (9009-22a, etc), there is a new firmware secure boot feature. If your boot adapters are not up to date on firmware the LPAR should still boot but you will get error messages with SRCs BA5400A5 or BA5400A6. This is documented in the server firmware readme.
For FW941 (new G models) it is more serious but there is no mention of the issue in the server firmware readme. I found this problem when I tried to boot the first VIO LPAR on a new 41G server. The LPAR would not boot and I saw fixed memory address alignment errors ending with the BA218003 error code.
I checked the adapter firmware readme and found the following:

"*Fixed Memory Address Not Aligned issue caused by FLOGI retry loop. Initial partition boot may fail with one of the following reference codes: BA210001, BA218001, BA210003, or BA218003."

This exactly describes what I saw. This applies to all of the 16Gb and higher HBAs. The LPAR would not boot until I removed the HBA from the profile. Fortunately, this was a vio that had internal disk so I was able to continue work. But be aware of this issue and make sure your microcode on your I/O adapters is up to date before bringing server firmware up to FW940 or higher.

Dealing With Missing or Failed Paths

When looking at lspath after migrating an LPAR or restoring an LPAR or after san problems you will sometimes see missing, defined or failed paths. I have seen this several times lately in my cloud systems. First you should check there are no outstanding issues that need to be resolved and cfgmgr should be run. If lspath shows the correct number of paths and running cfgmgr does not bring any paths back, then the only real option is to remove those failed paths. The following script identifies missing,failed and defined paths and builds remove scripts for them:

```
vi badpaths.sh
#!/bin/ksh
# badpaths - create cleanup scripts
```

```
echo “#!/bin/ksh“ >removeMpaths.sh
echo “#!/bin/ksh“ >removeFpaths.sh
echo “#!/bin/ksh“ >removeDpaths.sh
disks=$(lspv | awk '{print $1}')
for loop in $disks
do
lspath -l $loop -H -F "name:parent:connection:status" |grep Missing| awk -F: '{print
"rmpath -dl",$1,"-p", $2, "-w", $3}'>>removeMpaths.sh
lspath -l $loop -H -F "name:parent:connection:status" |grep Failed| awk -F: '{print
"rmpath -dl",$1,"-p", $2, "-w", $3}'>>removeFpaths.sh
lspath -l $loop -H -F "name:parent:connection:status" |grep Defined| awk -F: '{print
"rmpath -dl",$1,"-p", $2, "-w", $3}'>>removeDpaths.sh
done
exit 0
```

The script creates three new scripts: one each for missing, failed and defined. You should check the contents and, once you are happy with them, you can then make themexecutable and it will clean up your paths The original article where I found this can be found at:
https://sysaix.com/removing-failed-or-missing-paths

Creating bootable USB versions of the VIO to install

When you go to ESS (entitled software) to download your VIO server code there is now a flash image that you can download. As of today, the two DVDs are at 3.1.1.20 but the flash image is at 3.1.1.25. By using the flash image, you can save an upgrade step. I have been doing upgrades of my VIO servers from 2.2.6.32 to 3.1.1.25 using the mksysb from that flash image. For a new VIO you can also boot from the flash image on a USB stick.

In order to get the flash image on the USB stick from Windows, you need to be able to burn the ISO image to the USB. I have been most successful using an older version of rufus. I use rufus 2.7 as it lets me create a bootable image using dd from an ISO image. Under Format options about halfway down you will see an option to “create a bootable disk using.” Change ISO image to DD image and then click on the CD Icon to the right to select the ISO you want to use.
Make sure the correct USB key is chosen up the top then click on start to burn the bootable USB.

Another option is to download Fedora Media writer from Red Hat and follow their instructions on using it on Windows.
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/sect-making-usb-media?cm_mc_uid=44542485726616015575803&cm_mc_sid_50200000=79872101601557580333#sect-making-usb-media-windows

Migrating Users to a New System

Sometimes it is better to build a new system rather than migrate the old one over with all the things that have accumulated over time. One of the biggest complaints I see when people are not using LDAP or some other single signon relates to userids and passwords. In particular, users do not like being forced to change their password on the new system and it normally causes a lot of calls to the helpdesk during cutover.

I have a couple of scripts I user to gather the groups, users and passwords and create new scripts that I can then use on the new system to build their environments.

First, I tar up the home directories on the old system

Then, also on the old system I run the following:

```
#!/usr/bin/ksh
#
#  run using copyusers.sh
#  Creates a file that changes password for all accounts in /etc/passwd
#  Run on source system then bring setpwds.sh to new system
#  Create all the new users and groups PRIOR to running setpwds.sh and
untarring home dirs
#  Edit and remove accounts not to be touched
#
mkdir /holding
tardir="holding"
cd /home
tar -cvf $tardir/homedirs.tar *
cd $tardir
#
for user in `lsuser -a ALL`; do
        [ -n "$1" -a "$user" != "$1" ] && continue
        if grep -p ^${user}: /etc/security/passwd | grep -q "password = "; then
                hash=`grep -p ^${user}: /etc/security/passwd | grep "password = " | awk
-F " = " '{print $2}'`
                echo "echo '${user}:${hash}' | chpasswd -ec"
        fi
done >setpwds.sh
#
# Gather info on groups
#
lsgroup -c -a id ALL | grep -v ^# | awk -F: '{print "mkgroup id=" $2, $1}'
>creategrps.sh
#
# Gather users
#
lsuser -c -a id pgrp groups home shell gecos ALL | grep -v ^# | awk -F: '{print
"useradd -m -u", $2, "-g", $3, "-G", $4, "-c \"" $7 "\" -d ", $5, $1}' >createusers.sh
```

Copy the three scripts and the tarfile to the new system
I then edit each of the three scripts removing system users, etc and I also check to make sure there is no overlap in any of the group or userid numbers.
The order to run them is creategrps.sh, createusers.sh, setpwds.sh
I used relative backup for /home as I restore it to a holding directory on the new system as I don’t want to restore over the top of the ones I already have. Any that I don’t want to keep I move to a different directory.
I then use “cp -p -h -r * /home” to copy the home directories across to /home with their original permissions etc.
This works really well. I like to use a script to create the final script so that I can check the final version to make sure it is what I really want before I run it.

Summary

There is a lot of information out there to help administrators in their day-to-day jobs—but sometimes it’s hard to find. I’ve always found that a search of the blogs from Nigel Griffiths, Gareth Coates, Rob McNelly and Chet Mehta provide a ton of useful information. Finally, don’t forget to attend the virtual IBM Technical University (TechU) being held October 26-29, 2020 in your living room or home office. There is always plenty of great material at this conference.

References

1. IBM Virtual Techu – October 2020
    a. https://www.ibm.com/training/events/vtechu2020
2. Nigel Griffiths – AIXpert Blog
    a. https://www.ibm.com/support/pages/aixpert-blog-nigel-griffiths-mrnmon
3. Gareth Coates – Tips and Tricks
    a. https://www.ibm.com/support/pages/node/1116939
4. Rob McNelly AIXChange
    a. https://robmcnelly.com/
5. Rufus 2.7
    a. https://sourceforge.net/projects/rufus.mirror/files/v2.7/
    b. Portable download:
        i. https://sourceforge.net/projects/rufus.mirror/files/v2.7/rufus-2.7p.exe/download
    c.
6. IBM Power community
    a. https://community.ibm.com/community/user/power/communities/community-home?communitykey=71e6bb8a-5b34-44da-be8b-277834a183b0&tab=groupdetails
7. IBM Power Virtual User Group
    a. https://www.ibm.com/support/pages/node/1120377
8. IBM Powervm Virtual User Group

    a. https://www.ibm.com/support/pages/node/1110195

9. Chet Mehta Powervm
    a. https://community.ibm.com/community/user/power/network/members/profile/contributions/contributions-list?UserKey=58a0950f-9a24-49aa-85db-20d010c0cfd1