

Wednesday, April 30, 2014

Solaris 11.2 announcement - a collection of links and hints (Part 3)

In this blog entry i want to write about some small features you wouldn't read about on page 1, but which are really really helpful. Sean Wilcox wrote a nice article about SMF stencils. Stencils are a nice and build-in way to fill-in configuration information from SMF properties into legacy configuration files.

Another small but very nice feature is an extension of netstat. It now shows the pid, the user and the command of the process owning a socket. Casper Dik wrote about it in User, Pid and Commands in netstat(1m)
root@sol112:~# netstat -aun | more

UDP: IPv4

Local Address	Remote Address	User	Pid	Command	State
.	root	123	in.mpathd	Unbound	
.	root	123	in.mpathd	Unbound	
.	netadm	659	nwamd	Unbound	
.	netadm	659	nwamd	Unbound	
*.111	daemon	851	rpcbind	Idle	
.	daemon	851	rpcbind	Unbound	
*.48105	daemon	851	rpcbind	Idle	
*.111	daemon	851	rpcbind	Idle	
.	daemon	851	rpcbind	Unbound	
*.40150	daemon	851	rpcbind	Idle	
.	root	870	in.ndpd	Unbound	
*.631	root	948	cupsd	Idle	
*.68	root	985	dhcpageant	Idle	
*.546	root	985	dhcpageant	Idle	

With svcs -Lv a really helpful option found it's way to Solaris 11.2 . It's shows the logs for an SMF server. So you don't have to search for the log location anymore. root@sol112:~# svcs -Lv svc:/network/smtp:sendmail
svc:/network/smtp:sendmail (sendmail SMTP mail transfer agent)
[Apr 30 20:48:10 Enabled.]
[Apr 30 20:48:10 Rereading configuration.]
[Apr 30 20:48:46 Executing start method ("/lib/svc/method/smtp-sendmail start").]
WARNING: local host name (sol112) is not qualified; see cf/README: WHO AM I?
/etc/mail/aliases: 13 aliases, longest 10 bytes, 145 bytes total
[Apr 30 20:49:47 Method "start" exited with status 0.]

With tcpstat and ipstat two commands from the *stat family were integrated into 11.2. Useful to find out what's happening on your network interfaces.

root@sol112:~# tcpstat -c 1

Please wait...

ZONE	PID	PROTO	SADDR	SPORT	DADDR	DPORT	BYTES
global	979	TCP	sol112	22	MBP-of-c0t0d0s0.	51418	64,0

Total: bytes in: 0,0 bytes out: 64,0

ZONE	PID	PROTO	SADDR	SPORT	DADDR	DPORT	BYTES
global	979	TCP	sol112	22	MBP-of-c0t0d0s0.	51418	336,0

root@sol112:~# ipstat -c 1

Please wait...

SOURCE	DEST	PROTO	INT	BYTES
sol112	MBP-of-c0t0d0s0.fritz.box	TCP	net0	96,0
MBP-of-c0t0d0s0.fritz.box	sol112	TCP	net0	32,0

Total: bytes in: 32,0 bytes out: 96,0

SOURCE	DEST	PROTO	INT	BYTES
sol112	MBP-of-c0t0d0s0.fritz.box	TCP	net0	448,0

Blog Export: c0t0d0s0.org, <http://www.c0t0d0s0.org/>

MBP-of-c0t0d0s0.fritz.box sol112 TCP net0 64,0
Total: bytes in: 64,0 bytes out: 448,0

Posted by Joerg Moellenkamp in English, Oracle, Solaris at 21:18

Tuesday, April 29. 2014

Openstack in Solaris 11.2

There is a nice "Getting started" guide for Openstack support in Solaris 11.2 written by David Comay: "Getting Started with OpenStack on Oracle Solaris 11.2". It's part of a large heap of information available about this topic at OTN.

Posted by Joerg Moellenkamp in English, Solaris at 23:22

Solaris 11.2 announcement - a collection of links (Part 2)

Glenn Faden is writing about Authenticated Rights Profiles. Now you can assign administrative rights to a user, so she or he doesn't need to assume a role with the role password but still require him to authenticate.

A second really interesting new feature of Solaris 11.2 Glenn is writing about is the "Qualified User Attributes" feature, this is answering the challenge to limit for example an expanded set of privileges to a host or a group of hosts when you are putting the user information into LDAP. Qualified User Attributes solves this nicely.

By the way, such nice feature as limiting access or privileges to a certain timespan is available as well is in Solaris 11.2 as well. So you can just allow your junior administrator to do certain tasks for example just on weekdays between 09:00 and 17:00 .

Mike Gerds points in this blog entry to the various pieces of information regarding Kernel Zones and shows how to install a Kernel zones in three easy steps. Just a tip: Please keep in mind, that you can't do Kernel Zones inside a Virtualbox environment out of obvious reasons. Another blog entry is about the new dynamic reconfiguration of zones. In "Need another disk in your zone? No problem!" he is talking how you can add for example a disk to a zone without rebooting the zone.

Posted by Joerg Moellenkamp in English, Solaris at 22:12

Solaris 11.2 announcement - a collection of links (Part 1)

You may ask why Oracle made such a large fuzz around Solaris 11.2 ... however a large load of new features found their way into this release. I just want to point you to some sources of information about all this new things. There is a nice summary of all the new features: ORACLE SOLARIS 11.2 BETA – WHAT'S NEW.

The most boring feature at first: Binary compatibility ... all the new features ... still you are able to run your application you've compiled last millennium. It's a different question if you should do something like that, but the idea behind it is something different. Safe effort when moving forward to newer OS releases and newer application releases. When you have your own application, you don't have to think if something has changed inside the OS.

The second most boring feature, but extremely helpful: A Oracle DB prerequisite package ... install it and you can directly install Oracle DB 12c on it with the graphical installer.

Kernel Zones is a really great new features and now i can answer the question "Can i run different patch sets or different Solaris releases with zones with a resounding "Yes, when you use kernel-zones". You find the documentation for this feature here.

Unified Archive is extremely helpful to. Think of it as Flash Archives on steroids. Another frequently customer question i have a great answer now. You find the documentation of this feature at docs.oracle.com: Unified Archives are a new native archive type for Oracle Solaris. Unified Archives enable multiple system instances to be archived in a single unified file format. Unified Archives may contain one or more archived instances of Oracle Solaris from a single host. You can select individual installed zones to include during archive creation, and the host itself is optional.

You can deploy Unified Archives to recover a system that needs to be replaced due to failure, duplicate or clone a system configuration that you want installed on multiple machines, or migrate an existing system to new hardware or to a virtual system..

However, there are much more features available and some of my colleagues have already written about it: Darren Moffat has two articles in his blog. On one side he is reporting about the new compliance reporting framework. One command to assess the compliance of a system to a set of compliance rules, one command to make a nice report out of it. Another small, but important new feature solved a problem that many admin had for quite some time: Herding changed to the `/etc/system` file. There is now a directory called `/etc/system.d` and the `/etc/system` is assembled out of the contents of the directory and the file. Content in the file wins if you define something twice.

Bart Smaalders is writing about a nice addition in Solaris 11.1 called `pkg exact-install`: It functions exactly as the `install` command does except that it behaves as if you're installing onto a blank system with respect to package tenancy - any installed packages not a direct or indirect dependent of the packages specified on the command line are removed. Note that configuration files, etc. on remaining packages are left alone just as if you used `pkg install`.

David Comay is giving a walkthrough for the admin panel and project panel of the Openstack integrated into Solaris 11.2.

Nicolas Droux is reporting about "Application-Driven SDN and Beyond" talking about features like an build-in Elastic Virtual Switch (EVS), OpenStack Neutron Networking, VXLAN, the integration of probing in DLMP (YAY, one point of criticism away), Reflective Relay, Precision Time Protocol and (really, really interesting) SR-IOV VNICs in Solaris itself.

Dan Anderson writes about `verify boot`

Verified Boot here refers to verification of object modules before execution using digital signatures. If enabled, Solaris Verified Boot checks the factory-signed signature in a kernel module before loading and executing the module.

Krishna Yenduri reports in his blog about `SO_FLOW_SLA`, a clever way set resource controls from within an application. In a second blog entry he is reporting about extensions to the configuration of network flows that were introduced with 11.0 but are now even more useful (i use it quite often to ensure the priority of ssh or other important interfaces to the system).

More hints and links will follow soon.

Posted by Joerg Moellenkamp in English, Solaris at 21:22

Monday, April 28. 2014

We can provide you used days wholesale

Had a lot of used days in the last few days:

Day n-8: Eastern ... a few days off ... bank holiday. What a nice opportunity to get a sinusitis. Okay ... bad mood and headache galore. Will be a nice weekend. Yay!

Day n-6: Oh, friendly people at the counter in the airline lounge in HAM. I just want to scan my boarding card to get into the lounge and sit down. Get a coffee and forget the guy who drove with 130 on the fast lane almost from Lüneburg to Hamburg. Waited for a moment. Normaly it's not a problem just to hold your boarding card on the reader even while people are talking with other customers. Otherwise the usual high-traffic hours in that lounge would lead to a queue from A40 to the security in HAM. I definitely don't want to wait until you have stop talking with another guest. And i definitely don't want to wait after such a unfriendly comment telling me that i should wait. Sorry.

Day n-6: A long queue at rental car company. What a nice time to to a lunch break. At least in the opinion of the people there at Frankfurt airport station of the rental car company. And the "manager on duty" just looked and did nothing, nor stopping the employee to go into lunch break nor doing some work on her own. And then the car was at the end of nowhere. Seemingly the last possible parking bay in the last possible rental car parking lot. Is this the way of the rental car people telling you "F*ck you" because you was just minimally friendly after waiting between 10 and 15 minutes to get to the counter.

Day n-6: Oh nice ... a hotel room at the top of the hotel. How nice. Just a little bit to warm. A little bit. Or did they gave me the key card for the sauna. As no naked people opened the door, it must be really my room ... but better lock the door from my side.

Day n-5: Waking up. I'm sure the official anthem of people with sinusitis is "Pretty Paracetamol" by Fisher-Z. This song is 35 years old now yet another reason to get bad temper.

Day n-5: Where is the key of my car? Not for the rental car. Found that at exactly the same location i dropped it. For the car waiting in the airport parking lot in Hamburg. Checking travel time with my navigation device ... enough time. Doing a full search in the hotel room. Dang. It's not there. Looking into my backpack . Ouch ... cutting myself with a razor inside my backpack. Who was the idiot putting it into my backpack instead of the bag with the foam, soap and blades. Oh that was me.Dang. And no band-aid in reach. A piece of toilet paper must be sufficient. Checking travel time with my navigation device again ... Safety margin down to 5 minutes.Doing another search. Still not there. Safety margin almost eaten away. I have to leave the hotel. Looking around in the room a last time to check for forgotten things.

Looking onto the desk in the room. Who has slaughtered a pig? Oh, it was me, the pig and the pig slaughtering the pig ... back into the bathroom. Cleaning up the desk. Safety margin away. Checkout was faster than expected. Looking around in the rental car. Thinking how to get to my second car key. Driving to the Oracle office nearby. Want to get my badge out the backpack. The badge is not at the usual location. But my car key. WTF. At least i've reached the customer perfectly at the planned time.

Day n-5: Left the customer at the latest possible moment with no extra time or safety margin. Glad i refueled the car before driving to the customer. I don't know how how often I drove a car to the rental car return of Frankfurt Airport. I never took the wrong turn there because i know it gets difficult from there. However today i took the wrong turn to Terminal 2 instead of Terminal 1. I didn't knew that i can run that long with a notebook in my backpack and in a suit ... i'm really glad nobody shouted "Jumanji", going through security in record time and being at the gate at boarding time sharp ... only to learn that the plane is boarding 10 minutes later than expected ... and we waited even longer for pushback.

Day n-4: Just learned that i left my bag with all cables except the power supply of my notebook in Frankfurt. Damned. Really: ALL cables ranging from Ethernet Adapters to Charger for my mobile phone and for my iPad. Glad a friendly colleague took the bag with him and send it to me.

Day n-3: I almost hit a woman on a bicycle almost directly in front of the Oracle office. That was really a close call. Just didn't saw her, because i've concentrated on the "a**hole" bicycle driver driving that passed the road shortly before. I

stopped immediately right on the street. She said, nothing happened except the "shock" of the close call. I have explained what happened. As i stopped my car directly on the road, I had to drive my car into the garage, shouted that i will back in a few minutes, however when I got back to the location the woman wasn't there any longer. I can just hope that really nothing happened.

Day n-0: I was attacked today. By a bottle of Ketchup. I just wanted some on my fries for lunch. However i opened the bottle and instead of my fries i got the ketchup direct into my face. Think of a really naughty thing and you have the picture.

Posted by Joerg Moellenkamp in English at 22:33

Oracle Dojo

Für diejenigen unter meinen Lesern, die häufig auch mit der Datenbankseite von Oracle zu tun haben und der deutschen Sprache mächtig sind: Es gibt eine brauchbare Serie an Kompaktinformationen zu bestimmten Datenbank- und datenbanknahen Themen. Diese sind in der Oracle Dojo Serie erschienen. Man kann sie hier downloaden. Bisher sind dort folgende "Dojos" verfügbar:

#9: Die Oracle-Datenbank-Cloud mit Oracle Cloud Control

#8: Komprimierung in der Datenbank

#7: Oracle 12c - Multitenant Database Option

#6: Oracle GoldenGate 11gR2 - Aktiv-Aktiv-Replikation

#5: Real Application Testing - Testen mit Database Replay und SQL Performance Analyser

#4: Oracle Data Warehouse - Konzepte und Methoden

#3: Oracle Enterprise Manager 12c Cloud Control

#2: Big Data: Eine Einführung in Oracle NoSQL Database, Hadoop MapReduce, Oracle Big Data Connectors

#1: Oracle 11g R2 mit Real Application Clusters auf Windows Server 2008 R2

Posted by Joerg Moellenkamp in German, Oracle at 10:02

Sunday, April 27. 2014

Solaris 11.2 announcement webcast

Solaris 11.2 is a major step for Solaris, so you can expect to hear a lot of interesting news in the upcoming announcement webcast. You shouldn't miss it. The webcast takes place at April 29, 2014, starts at 01:00 PM and ends at 02:30 PM (both times are Eastern Time zone)

Posted by Joerg Moellenkamp in English, Oracle, Solaris at 22:36

Friday, April 18. 2014

Checking VXFS QIO usage

This isn't something new however given that i saw this problem at a customer just last week, i would like to point to something you have to keep in mind when using Oracle DB with the Veritas Filesystem.

When you have strange performance problem when using VXFS please ensure that the database isn't sitting around in the POSIX inode r/w-lock with it's database writes. Check if Quick-I/O (solves the issue) or ODM (solves the issue as well) is really activated. QIO is not activated by just using the FILESYSTEMIO_OPTION=setall or by mounting the filesystem with the mount option qio. You have to do more and it looks like this is sometimes forgotten by people setting up or migrating a system.

So far this test was the fastest for me because it checks if something is present that shouldn't present when QIO is in action:
`echo '::threadlist -v' | mdb -k | sed 's:^$::' | tr -d '\n' | tr '$' '\n' | grep 'vx_rwsleep_rec_lock' | tr -s ' ' | cut -d ' ' -f 10 | sort | uniq -c`
When this command line returns with oracle processes containing "dbw" you should really check if you have properly configured QIO or ODM. Please refer to the VXFS documentation in regard of the correct installation and configuration steps.

You will find the explanation for this at this rather old blog entry: "Hunting red herrings"

Posted by Joerg Moellenkamp in English, Solaris at 19:17

Heartbleed info in OTN

The information about Heartbleed in regard of Oracle Products is now available in OTN as well: [OpenSSL Security Bug - Heartbleed / CVE-2014-0160](#).

Posted by Joerg Moellenkamp in English, Oracle, Security, Solaris at 15:57

Sunday, April 13. 2014

Heartbleed (CVE-2014-0160)

I got some questions regarding Heartbleed and Oracle products from blog readers. In this regard i just want to link to the entry in the Oracle Security Blog: "Heartbleed' (CVE-2014-0160) Vulnerability in OpenSSL ". The author states: Oracle recommends that customers refer to the My Oracle Support Note Doc ID 1645479.1 for information about affected products, availability of fixes and other mitigation instructions.

Posted by Joerg Moellenkamp in English, Oracle, Security at 20:24

Saturday, April 5, 2014

Upcoming Solaris 11.2 announcement

Markus Flierl writes in "Don't miss the announcement of Solaris 11.2": It's very hard to find time for writing a blog these days: I've been quite busy working with my team on getting the final features into Solaris 11.2, making sure that we address any remaining critical defects while getting ready for the announcement of Solaris 11.2 in NYC on April 29. I find the latter particularly hard: Trying to squeeze all of the new capabilities of Solaris 11.2 into a 45 min preso feels like trying to squeeze four elephants into a VW Beetle. The current plan for April 29 is to start by ringing the bell and open up the stock market in the morning followed by the announcement event after lunch at the Metropolitan Pavilion on 639 W. 46th Street.

Posted by Joerg Moellenkamp in General at 08:39

Creating a zpool configuration out of a bunch of F40/80 cards

In a some customer situation i'm using a number of Oracle Sun Flash Accelerator F40 PCIe Cards or F80 PCIe cards to create flash storage areas inside a server. For example i had 8 F40 cards in a server by using a SPARC M10 and a PCIe Expansion Box which enables you to connect up to 11 F40/F40 cards per expansion box.

The configuration with 8 F80 cards for example is a configuration i'm using on very special occasions and for special purposes, in this case it was a self-written application of a customer needing a lot flash storage inside the server. I won't disclose more. On the other side: I'm sizing quite frequently systems with two F80 cards for "separated ZIL" purposes. Either if you use the SSD as data storage or as separated ZIL: When you do mirroring you have to ensure that mirrors are not using mirror halves on the same card.

From the systems perspective you see four disk devices per F40/F80 card with 100 respective 200 GB capacity per disk and thus you can just add them to your zpool configuration. However configuring the system was a little bit unpractical. The problem: It's not that easy to create a configuration that ensures that no mirror has it's two vdevs on a single F40/F80 card. Perhaps there is an easier way, however I didn't found it so far.

It's a little bit hard to find acceptable disk pairs when you are looking on PCI trees like
/devices/pci@8000/pci@4/pci@0/pci@8/pci@0/pci@0/pci@0/pci@1/pci@0/pci@1/pciex1000,7e@0/iport@80:scsi.
Well, at two cards it's not that hard, but still not a nice job. After doing this manually a few times, i thought that at 8 or 22 cards doing this manually is a job for people who killed baby seals, listen to Justin Bieber or equivalent horrible things.

But i didn't committed to such crimes and this problem is nothing that a little bit of shell-fu can't solve. You can do it in a single line of shell. Well ... a kind of a single line of shell.

You see the F40/F80 cards in output of `cfgadm -alv` like this.

```
unavailable scsi-sas n
/dev/devices/pci@8000/pci@4/pci@0/pci@8/pci@0/pci@0/pci@0/pci@1/pci@0/pci@1/pciex1000,7e@0/iport@80:scsi
c13::w9999999999993b58,0 connected configured unknown Client Device:
/dev/dsk/c0t9999999999993B58d0s0(sd27)
```

In most of my examples outputs i will just show a single line. The commands produce obviously a lot more output lines. The single line is just used to show what the command is doing.

At first we have to solve a problem. The output for each device is divided into two lines. We have to concatenate them.

```
Piece of cake: cfgadm -alv | awk '!(NR%2){print p " "$0}{p=$0}'. This results in
c13::w9999999999993b58,0 connected configured unknown Client Device:
/dev/dsk/c0t9999999999993B58d0s0(sd27) unavailable disk-path n
/dev/devices/pci@8000/pci@4/pci@0/pci@8/pci@0/pci@0/pci@0/pci@1/pci@0/pci@1/pciex1000,7e@0/iport@80:scsi::w99
9999999999993b58,0
```

Believe me ... every two lines are now joined into a single one.

I'm just interested in the lines containing `/dev/dsk`. So i just keep them by `cfgadm -alv | awk '!(NR%2){print p " "$0}{p=$0}' | grep "dev/dsk"`

When i do a word count i will see 32 device. Looks like i have found my 32 solid state disks:

```
cfgadm -alv | awk '! (NR%2){print p " "$0}{p=$0}' | grep "dev/dsk" | wc -l  
32
```

At first I have to get rid of the unnecessary spaces in the output lines. \$ cfgadm -alv | awk '! (NR%2){print p " "\$0}{p=\$0}' | grep "dev/dsk" | tr -s " "

```
c13::w9999999999993b58,0 connected configured unknown Client Device: /dev/dsk/c0t9999999999993B58d0s0(sd27)  
unavailable disk-path n  
/devices/pci@8000/pci@4/pci@0/pci@8/pci@0/pci@0/pci@0/pci@1/pci@0/pci@1/pciex1000,7e@0/iport@80:scsi:w99  
9999999999993b58,0
```

There are a lot of unnecessary fields in this input. I will just keep the ones I need - the eleventh and the seventh column containing the disk device and the position in the /devices tree:\$ cfgadm -alv | awk '! (NR%2){print p " "\$0}{p=\$0}' | grep "dev/dsk" | tr -s " " | awk '{print \$11,\$7;}'

```
/devices/pci@8000/pci@4/pci@0/pci@8/pci@0/pci@0/pci@0/pci@1/pci@0/pci@1/pciex1000,7e@0/iport@80:scsi:w99  
9999999999993b58,0 /dev/dsk/c0t9999999999993B58d0s0(sd27)
```

Now I will sort them. The idea is that when you sort the devices by the string representing the position in the /devices tree, the devices on the same card are directly following each other in packs of 4. Okay, the /device tree has done it's job after sorting. Let's get rid of it. \$ cfgadm -alv | awk '! (NR%2){print p " "\$0}{p=\$0}' | grep "dev/dsk" | tr -s " " | awk '{print \$11,\$7;}' | sort -k 1 | awk '{print \$2;}'

```
/dev/dsk/c0t9999999999993B58d0s0(sd27)
```

The s0 and the (sdxx) part must be cut away ... i will to this in a single step. i define s0 as the delimiter and keep just the stuff before the delimiter.\$ cfgadm -alv | awk '! (NR%2){print p " "\$0}{p=\$0}' | grep "dev/dsk" | tr -s " " | awk '{print \$11,\$7;}' | sort -k 1 | awk '{print \$2;}' | awk -F "s0" '{print \$1}'

```
/dev/dsk/c0t9999999999993B58d0
```

The /dev/dsk/ is unnecessary ... so just remove it as well.\$ cfgadm -alv | awk '! (NR%2){print p " "\$0}{p=\$0}' | grep "dev/dsk" | tr -s " " | awk '{print \$11,\$7;}' | sort -k 1 | awk '{print \$2;}' | awk -F "s0" '{print \$1}' | sed 's/\dev\dsk//'

```
c0t9999999999993CA8d0  
c0t9999999999991B7Cd0  
c0t9999999999994478d0  
c0t99999999999934B8d0  
c0t9999999999992748d0  
c0t999999999999273Cd0  
c0t9999999999993B5Cd0  
c0t9999999999993B58d0  
c0t99999999999980ECd0  
c0t9999999999997CE0d0  
c0t9999999999998548d0  
c0t999999999999860Cd0  
c0t9999999999991A6Cd0  
c0t9999999999991C7Cd0  
c0t9999999999991A84d0  
c0t9999999999991AA4d0  
c0t9999999999992CE4d0  
c0t9999999999992398d0  
c0t9999999999992358d0  
c0t9999999999993D98d0  
c0t999999999999801Cd0  
c0t9999999999998030d0  
c0t9999999999998038d0  
c0t9999999999998034d0  
c0t9999999999997DE8d0  
c0t9999999999997DFCd0  
c0t9999999999993288d0  
c0t9999999999994500d0  
c0t99999999999940F4d0  
c0t9999999999993F38d0  
c0t999999999999358Cd0  
c0t99999999999935B8d0
```

Now we have to bring this devices in a different sequence to form mirror pairs. As already stated: The assumption is that due to the sort we did before each four lines are representing the devices that are placed on the same card. So when the both mirror halves are 4 lines apart on the list before, it's on a different card.

In order to process all 32 devices, i'm using the following construct: \$ cfgadm -alv | awk '!(NR%2){print p " "\$0}{p=\$0}' | grep "dev/dsk" | tr -s " " | awk '{print \$11,\$7;}' | sort -k 1 | awk '{print \$2;}' | awk -F "s0" '{print \$1}' | sed 's/\dev\dsk\///' | tee >(awk 'NR%4==0' > 0.out) >(awk 'NR%4==1' > 1.out) >(awk 'NR%4==2' > 2.out) >(awk 'NR%4==3' > 3.out) >> /dev/null ; cat 0.out 1.out 2.out 3.out

c0t99999999999934B8d0
c0t9999999999993B58d0
c0t999999999999860Cd0
c0t9999999999991AA4d0
c0t9999999999993D98d0
c0t9999999999998034d0
c0t9999999999994500d0
c0t99999999999935B8d0
c0t9999999999993CA8d0
c0t9999999999992748d0
c0t99999999999980ECd0
c0t9999999999991A6Cd0
c0t9999999999992CE4d0
c0t999999999999801Cd0
c0t9999999999997DE8d0
c0t99999999999940F4d0
c0t9999999999991B7Cd0
c0t999999999999273Cd0
c0t9999999999997CE0d0
c0t9999999999991C7Cd0
c0t9999999999992398d0
c0t9999999999998030d0
c0t9999999999997DFCd0
c0t9999999999993F38d0
c0t9999999999994478d0
c0t9999999999993B5Cd0
c0t9999999999998548d0
c0t9999999999991A84d0
c0t9999999999992358d0
c0t9999999999998038d0
c0t9999999999993288d0
c0t999999999999358Cd0

Okay, let's do a quick check if everything went right:cfgadm -alv | awk '!(NR%2){print p " "\$0}{p=\$0}' | grep "dev/dsk" | tr -s " " | awk '{print \$11,\$7;}' | sort -k 1 | awk '{print \$2;}' | awk -F "s0" '{print \$1}' | sed 's/\dev\dsk\///' | tee >(awk 'NR%4==0' > 0.out) >(awk 'NR%4==1' > 1.out) >(awk 'NR%4==2' > 2.out) >(awk 'NR%4==3' > 3.out) >> /dev/null ; cat 0.out 1.out 2.out 3.out | xargs -l {} grep {} list.devices | awk '{print \$1;}' | awk 'ORS=NR%2?RS:RS RS' | sed 's#/devices/pci@8000/pci@4/pci@0/pci@8/pci@0/pci@0/pci@0/pci@1/pci@0#..#'/pci@0/pciex1000,7e@0/iport@80:scsi:w99999999999934b8,0
../pci@1/pciex1000,7e@0/iport@80:scsi:w9999999999993b58,0

../pci@10/pci@0/pci@0/pciex1000,7e@0/iport@80:scsi:w999999999999860c,0
../pci@10/pci@0/pci@1/pciex1000,7e@0/iport@80:scsi:w9999999999991aa4,0

../pci@10/pci@0/pci@10/pciex1000,7e@0/iport@80:scsi:w9999999999993d98,0
../pci@11/pci@0/pci@0/pciex1000,7e@0/iport@80:scsi:w9999999999998034,0

../pci@11/pci@0/pci@1/pciex1000,7e@0/iport@80:scsi:w9999999999994500,0
../pci@8/pciex1000,7e@0/iport@80:scsi:w99999999999935b8,0

../pci@0/pciex1000,7e@0/iport@10:scsi:w9999999999993ca8,0
../pci@1/pciex1000,7e@0/iport@10:scsi:w9999999999992748,0

../pci@10/pci@0/pci@0/pciex1000,7e@0/iport@10:scsi:w99999999999980ec,0
../pci@10/pci@0/pci@1/pciex1000,7e@0/iport@10:scsi:w9999999999991a6c,0

../pci@10/pci@0/pci@10/pciex1000,7e@0/iport@10:scsi:w9999999999992ce4,0
../pci@11/pci@0/pci@0/pciex1000,7e@0/iport@10:scsi:w999999999999801c,0

```
../pci@11/pci@0/pci@1/pciex1000,7e@0/iport@10:scsi::w999999999997de8,0
../pci@8/pciex1000,7e@0/iport@10:scsi::w9999999999940f4,0

../pci@0/pciex1000,7e@0/iport@20:scsi::w999999999991b7c,0
../pci@1/pciex1000,7e@0/iport@20:scsi::w99999999999273c,0

../pci@10/pci@0/pci@0/pciex1000,7e@0/iport@20:scsi::w999999999997ce0,0
../pci@10/pci@0/pci@1/pciex1000,7e@0/iport@20:scsi::w999999999991c7c,0

../pci@10/pci@0/pci@10/pciex1000,7e@0/iport@20:scsi::w999999999992398,0
../pci@11/pci@0/pci@0/pciex1000,7e@0/iport@20:scsi::w999999999998030,0

../pci@11/pci@0/pci@1/pciex1000,7e@0/iport@20:scsi::w999999999997dfc,0
../pci@8/pciex1000,7e@0/iport@20:scsi::w999999999993f38,0

../pci@0/pciex1000,7e@0/iport@40:scsi::w999999999994478,0
../pci@1/pciex1000,7e@0/iport@40:scsi::w999999999993b5c,0

../pci@10/pci@0/pci@0/pciex1000,7e@0/iport@40:scsi::w999999999998548,0
../pci@10/pci@0/pci@1/pciex1000,7e@0/iport@40:scsi::w999999999991a84,0

../pci@10/pci@0/pci@10/pciex1000,7e@0/iport@40:scsi::w999999999992358,0
../pci@11/pci@0/pci@0/pciex1000,7e@0/iport@40:scsi::w999999999998038,0

../pci@11/pci@0/pci@1/pciex1000,7e@0/iport@40:scsi::w999999999993288,0
../pci@8/pciex1000,7e@0/iport@40:scsi::w99999999999358c,0
```

Looks correct. No mirror pair has it's device on the same card. Before you ask: I've generated the file list.devices beforehand with `cfgadm -alv | awk '!(NR%2){print p " "$0}{p=$0}' | grep "dev/dsk" | tr -s " " | awk '{print $11,$7;}' > list.devices.`

Second check: None of the devices have been used twice in the configuration. `cfgadm -alv | awk '!(NR%2){print p "$0}{p=$0}' | grep "dev/dsk" | tr -s " " | awk '{print $11,$7;}' | sort -k 1 | awk '{print $2;}' | awk -F "s0" '{print $1}' | sed 's/\dev\dsk\//' | tee >(awk 'NR%4==0' > 0.out) >(awk 'NR%4==1' > 1.out) >(awk 'NR%4==2' > 2.out) >(awk 'NR%4==3' > 3.out) >> /dev/null ; cat 0.out 1.out 2.out 3.out | sort | uniq -c | grep -v "1 " | wc -l`

Okay ... now i have to transform this into a zpool configuration or at least as a easily cut n' pasteable part of a zpool configuration. By the tasks we did before, it's ensured that the device in the n-th line and the n+1 line are not on the same card. So transforming it into a zfs configuration is quite simple. `$ cfgadm -alv | awk '!(NR%2){print p "$0}{p=$0}' | grep "dev/dsk" | tr -s " " | awk '{print $11,$7;}' | sort -k 1 | awk '{print $2;}' | awk -F "s0" '{print $1}' | sed 's/\dev\dsk\//' | tee >(awk 'NR%4==0' > 0.out) >(awk 'NR%4==1' > 1.out) >(awk 'NR%4==2' > 2.out) >(awk 'NR%4==3' > 3.out) >> /dev/null ; cat 0.out 1.out 2.out 3.out | awk '{if ((NR % 2) == 1) printf("mirror "); print; }' | tr -s "\n" " "`

```
mirror c0t9999999999934B8d0 c0t999999999993B58d0 mirror c0t99999999999860Cd0 c0t999999999991AA4d0
mirror c0t999999999993D98d0 c0t999999999998034d0 mirror c0t999999999994500d0 c0t9999999999935B8d0
mirror c0t999999999993CA8d0 c0t999999999992748d0 mirror c0t9999999999980ECd0 c0t999999999991A6Cd0
mirror c0t999999999992CE4d0 c0t99999999999801Cd0 mirror c0t999999999997DE8d0 c0t9999999999940F4d0
mirror c0t999999999991B7Cd0 c0t99999999999273Cd0 mirror c0t999999999997CE0d0 c0t999999999991C7Cd0
mirror c0t999999999992398d0 c0t999999999998030d0 mirror c0t999999999997DFCd0 c0t999999999993F38d0
mirror c0t999999999994478d0 c0t999999999993B5Cd0 mirror c0t999999999998548d0 c0t999999999991A84d0
mirror c0t999999999992358d0 c0t999999999998038d0 mirror c0t999999999993288d0 c0t99999999999358Cd0
```

Now you can use it as you want for creating for the zpool itself or for the separated log for a zpool.

There is one thing this script can't do: When you have multiple PCI Expansion boxes with F40/F80 cards the configuration generated by the shell-fu doesn't ensure that the both mirror are in different PCIe Expansion boxes. But hey, there is some stuff i have to keep for myself

Posted by Joerg Moellenkamp in English, Solaris at 07:06

Friday, April 4. 2014

Event announcement: Oracle DB Tuning auf Solaris 11 in München 9.5.2014

Ich habe ja schon vor einigen Tagen geschrieben, das Anfang Mai ein Oracle Business Breakfast zum Thema Oracle DB Tuning auf Solaris 11 in München stattfindet. Es ist nunmehr die Einladung da und der Anmeldelink aktiv. Ihr könnt euch hier anmelden.

Zum Inhalt: Zwar sind die Defaults von Solaris und der Oracle Datenbank für den durchschnittlichen Anwendungsfall gut gewählt, doch manchmal möchte man seine Datenbank doch noch weiter tunen und mehr herausholen oder auf die Spezialitäten der eigenen Last anpassen. Dieser Vortrag wird eine Einführung in die Stellschrauben der Oracle Datenbank und von Solaris geben und erklären, wie sie funktionieren, warum sie funktionieren, wie sie interagieren und Hinweise für den Einsatz in der Praxis geben. Weiterhin soll der Vortrag erklären, wie man aus dem AWR-Report auf System- oder Hardwareprobleme schließen kann.

Abgerundet wird die Veranstaltung durch die Vorstellung der neuen Version des Oracle Enterprise Manager Ops Centers zur vereinfachten Administration - insbesondere für die Oracle VM auf SPARC (aka LDOMs).

Posted by Joerg Moellenkamp in English, Oracle, Solaris at 12:05