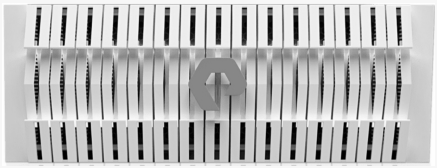
Flashblade



Blade commands

Commands	Description
pureblade list	*To list the blades in FB*
purehw list --spec	*To show the blade serial number*
hal-show -e /local | jq '.family, .id, .asy, .desc'	*To show blade type and model*
hal-eeprom -e /local/eeprom/id -r | jq .	*To show blade serial number and Node ID*
fbdiag wait-helper -v -n1.3	*To verify Process status in blade*
rpc.py blades_available | jq . | egrep "ir.*8001|cluster_id"	*To verify blade in cluster geometry*
exec.py -n$chassisNum.$bladeNum "ir_version | grep build"	*To verify the Blade Purity / Build version*
exec.py -nx -- sudo netstat -anot | grep <CLIENT_IP:PORT>	*To verify any connections to blade*
sudo supervisorctl status nfsd	*verify NFS is running on the blade*
fbdiag nfs-health-check --mgmt-vip ch1-fb1 | grep -c "running on server"	*To verify any authority running on the blade*
nfs_control.py -n$chassisNum.$bladeNum stop -v	*Stop NFS on the blade*
nfs_control.py -n$chassisNum.$bladeNum start -v	*Start NFS on the blade*
purehw setattr --identify on CH$chassisNum.FB$bladeNum	*Turn on Locator LED on blade*
purehw setattr --identify off CH$chassisNum.FB$bladeNum	*Turn off locator LED on blade*
puregrep -E "is_evacuating changed to true|start evacuation" *.fb*/nfs.log	*To check when Evac started on the blade <run from FUSE>*
puregrep -E "Calling blade removal RPC|blades observed geom change" *[af]m?/platform.log*	*To check when Evac completed on the blade <run from FUSE>*
date; fbupgrade power-ctl -v -nc.b cycle	*Powercycle the blade*
hal-show -e /local/temperature --headers name endpoint value units	*How to check Blade temperature*
lsblk	*How to check Blade filesystem usage*
sudo smartctl -i /dev/sda	*How to check Blade SSD information*
sudo dmidecode -t 17	*How to check Blade DIMM information*
date; exec.py -n1-2,4-11,13,14 -- 'time=`date +%H:%M:`; zgrep "$time.*counter.*S3.*allocated_a-us" /logs/nfs.log' | awk '{a=a+$12;b=b+$13}END{print a,b}'	To check EVAC is Progressing or not , Run twice at 10 second interval

By **ethiraj**
cheatography.com/ethiraj/

Not published yet.
Last updated 4th May, 2023.

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Blade commands (cont)

	How to evacuate the blade (there are few ways, here we are stopping the NFS)?
hal-slot -l	How to check Blade is powered ON or OFF in slot
fbdiag nfs-health-check --mgmt-vip ch1-fb1 |grep 'booting for'	commands to check Authority booted
exec.py -na -- 'zgrep -a " [AEK] " /logs/nfs.log|tail'	Check for any AEK errors on Blades
exec.py -nx.x 'sudo rsync -auv ch1-fb1:/ssd/nfs_conf.json /ssd/nfs_conf.json'	Copy tunable from one blade to another
zgrep "flash_read_uncorrectable, all Vt retry options were unsuccessful" nfs.log | perl -n -e '/(\[U\d+\]).*<(SM=\d+ BNK=\d+ CE=\d+ LUN=\d+ BLK=\d+)/&& print $1 . " " . $2 . "\n"' | sort | uniq | perl -n -e '/(\[U\d+\]).*(SM=\d+ BNK=\d+ CE=\d+ LUN=\d+) BLK=(\d+)/&& print $1 . " " . $2 . " PLANE=" . $3%4 . "\n"' | sort | uniq -c	Checking for Bad blocks and bad planes
ir@ch1-fb5:~$ /opt/ir/devcat_lookup.py device_health|grep -A10 "sketchy_block_ages_sec"	How to check Bad block rebuild progress , check for total count reached to "0"
tgrep -a "Blade nand type detected" ch*fb*/platform.log* | uniq -f4	Getting blade type information from FUSE logs
fb dump hdiag --key puresmb.status	To check SMB type configured from FUSE
fb info smb	To check SMB type configured from FUSE

SEV-1\Performance issue helpful commands

purearray monitor	show the cluster's throughput
purearray monitor – protocol nfs --client	show all nfs client's stats
Purehw list	verify all blades are healthy , XFM, NW ports are healthy and no failures
exec.py -na -- 'zgrep -a " [AEK] " /logs/nfs.log|tail'	Check for any AEK errors across blades in NFS logs
Purearray list --space	check Array got sufficient space
purealert list	verify for any active Alerts on the blade
./fbhealth --fail	Run fbhealth live on system (Always push latest FBtools to array
Pureblade list	Check all blades are healthy
exec.py -na 'zgrep -a "Vt retry" /logs/nfs.log	Check for any BAD blocks on the blade
exec.py -sa "fbdiag monitor-portstats --monitor err --once"	Check for any port errors <Single chassis>
fbdiag monitor-portstats --monitor err --once	Check for any port errors <Multi Chassis>
fbdiag monitor-portstats --once	verify for Port serving IO's
exec.py -na 'zgrep -E " E | A | K " /logs/platform.log.*	Check for any AEK errors in Platform logs
bdiag nfs-health-check --mgmt-vip ir2	check all Authorities in case any Blade evacuation in-progress or completed

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

SEV-1\Performance issue helpful commands (cont)

exec.py -na 'zgrep "Segmentation fault" /logs/nfs.log-[0-1]*'	Check for any SEGFAULTS in NFS logs (Need to understand what is SEGFAULT)
exec.py -na 'zgrep "segfault" /logs/system.log'	Check for any SEGFAULTS in System logs
exec.py -na "zgrep 'root::rpc_service.tcp_throttle_avail_slots' /logs/-nfs.log | tail -n1"	verify for any RPC slots exhaustion
atopen "log filename"	To check the process utilization

Blade tools and commands

fbhealth

fbdiag

exec.py

fbupgrade

fbadmin

supervisorctl

ir_version

HAL Commands

hal-slot

Used to list, discover hardware such as PSU, fan, QSFP, EFM and blades, is also used to start and stop blades

hal-slot -l

hal-slot --discover-all

hal-slot -e /local/slot/slot-blade-4 --start

hal-slot -e /local/slot/slot-blade-4 --stop

hal-show

Used to show the rpc query result for one specified entry, this can be used on EFM to query the info on blade or other devices shown in the list of "hal-slot -l"

hal-show -e /local/gpio

hal-show -e /local/i2c-bus

hal-show -e /local/i2c

hal-show -e /local/slot/slot-blade-4/module/gpio

hal-i2c

used to do raw i2c read/write from/to I2C device on given i2c bus, using "hal-show -e /local/i2c" to find out the I2C device address and the bus entry for target I2C device

hal-i2c -e /local/i2c-bus/bus_entry --addr xx --offset xx --read xx(number of bytes)

hal-i2c -e /local/i2c-bus/bus_entry --addr xx --offset xx --write xx xx xx

hal-eeprom

Used to read the eeprom contents

hal-eeprom -e /local/eeprom/id -r

hal-upgrade

FUSE commands

 # Failng to RA or VATS PUSH/PULL , run below commands from FUSE

export PURELOGIN_KEY_TYPE=ed25519

#How to push latest FBtools from FUSE to Customer array

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

FUSE commands (cont)

fb auto fbtools

FB Network commands

purelag list
purenetwork list
puresubnet list
transcievers.py
switch_shell.py ps
purehw list --type eth
rpc.py -p switch get_port_stats | jq .
lldpcli show neigh
bdiag net-health
switch_shell.py vlan show | grep
switch_shell.py l3 multipath show
switch_shell.py vlan translate show

rpc.py -p switch get_xcver | jq .	Check transceiver is installed in the slot
SWITCHSTATS=$(rpc.py -p switch get_port_stats);for x in {1..4}; do echo qsfp $x/1-4; for y in {1..4}; do echo $SWITCHSTATS \| jq '.["qsfp'$x'/'$y'"]' | grep 'link_status' ; done ;done	Check the Link status of individual QSFP's
lldpcli show neighbor | egrep 'qsfp|ifname'	Identify ports on the TOR switches connecting to the FlashBlade
monitor_portstats.py	Check life statistics on all interfaces
exec.py -sa "fbdiag monitor-portstats --monitor err --once"	Monitor errors on ports <Muti chassis-s>
purehw connector list --cli	To show the Ethernet port details
tgrep -aE "link status|LACP port|partner state" *[af]m?/platform* | egrep -i "up|down"	from FUSE to check any port flappings
zgrep -v " INFO " *[af]m?/platform* | egrep -B1 -A40 " FCS " | less	from FUSE to check for any FCS errors

Upgrade related Doubts

Pre-upgrade health check failures

ECMP inconsistency issue

exec.py -xa -sa "switch_shell.py l3 egress show" (check for the refcount is same across chassis , if there is +1 or -1 difference is acceptable)

Where is the Purity Images kept in FUSE?

All versions live on fuse in: /support/flashblade/releases

What are the three stages of an NDU?

"FlashBlade Upgrade Stage 1: Gather System Information"

"FlashBlade Upgrade Stage 2: Upgrade Images"

"FlashBlade Upgrade Stage 3: Software Restart and Reboot"

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Upgrade related Doubts (cont)

How to check the upgrade logs ?

Upgrade logs are recorded under /logs

** extend RA* 8

then first go to master fm and type sudo tmux

this should start a tmux session.

once the tmux session is on run puresupport disconnect ; sleep 20 ; puresupport connect ; exit

this will disable the RA and then again re-enable for you.

less -r fbupgrade.log.2022-03-27.07-17-01.gz |tail

If you are disconnected from tmux, how do you reconnect to it?

sudo tmux attach elasticity

How do you drop out of a tmux session?

Ctrl-b d

How to disconnect console session

CTRL+SHIFT+e and c and .

Extend RA

then first go to master fm and type sudo tmux

this should start a tmux session.

once the tmux session is on run puresupport disconnect ; sleep 20 ; puresupport connect ; exit

this will disable the RA and then again re-enable for you.

How to start simple http server

The "python -m SimpleHTTPServer" command is used to launch a basic HTTP server in the current working directory using Python 2.

When you run this command, Python will start a web server on port 8000 and serve the files in the current directory as static web pages.

Here's how to use it:

Open a terminal or command prompt in the directory where you want to serve files.

Run the command "python -m SimpleHTTPServer" or "python2 -m SimpleHTTPServer" (depending on your Python version).

The server will start and you'll see a message like "Serving HTTP on 0.0.0.0 port 8000".

Open a web browser and navigate to "http://localhost:8000" to see the files being served.

Note that this command is intended for testing and development purposes only and should not be used in production environments. Also, if you're using Python 3, the command has been changed to "python -m http.server" or "python3 -m http.server".

FB array commands

Log file location

Files and locations

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com