Hewlett Packard
Enterprise

linuxKI

Basic Commands

Prerequisite
Install Linux kernel header package(s) if you want to collect data using the LiKI DKLM module

Installation:
```
rpm --install --nodeps linuxki-4.2-1.x86_64.rpm
dpkg --install linuxki_4.2-1_amd64.deb
```

Collect KI Dump:
```
cd /dev/shm                 # if memory is available
runki [-d <secs>]           # default is 20 secs
```

Post-process KI Dump data
```
kiall [-r]
```

Mounting the debug filesystem (debugfs)
```
mount -t debugfs debugfs /sys/kernel/debug
```

Curses-base live mode
```
mount -t debugfs debugfs /sys/kernel/debug
kiinfo -live [-a <secs>]    # default is 5 secs
```

Clean-up KI files in cwd
```
kiclean -p -v -f
```

KI dump collection Examples

KI dump help
```
runki -h
```

Collect for 60 seconds
```
runki -d 60
```

Collect with all subsystems
```
runki -e all -R
```

Collect using ftrace with all subsystems
```
runki -f -e all
```

Collect with all subsystems, excluding some system calls
```
echo “time” >sysignore
echo “futex” >>sysignore
echo “gettimeofday” >>sysignore
runki -e all -I sysignore
```

Collect using PID filtering for 2 minutes
```
runki -P 26814 -d 120
```

Collect and include scsi subsys
```
runki -e default -s scsi
```

Collect with Collect/MW, sar, perf data and Java stacks
```
runki -M -U -X -j
```

Collect data and add a comment
```
runki -c “slow run, only 30 MB/sec”
```

KI Dump Post Processing Examples

Post-processing help
```
kiall -h
```

Create <hostname>/<timestamp> subdir structure
```
kiall -r
```

Create Visualization charts and graphs
```
kiall -V
```

Create Kparse Report in text format
```
kiall -x
```

Online kiinfo Examples

kiinfo help
```
kiinfo -h
```

PID Analysis Report with multiple PID filters for 30 seconds
```
kiinfo -kipid pid=6814,pid=6815,scdetail -a 30
```

Disk Analysis report to show top 10 pids accessing a specific dev every 5 seconds for a minute
```
kiinfo -kidsk dev=0x00800010,npid=10 -a 5 -p 12
```

CPU/RunQ Analysis with IRQ events
```
kiinfo -kirunq events=default,subsys=irq -a 5
```

Show just SCSI Synchronize Cache commands for 1 minute
```
kiinfo -kitrace subsys=scsi -a 60 | grep -i
synchronize_cache
```

KI Dump kiinfo Examples

PID Analysis Report with System Call details, Cooperating Task details, and per-pid RunQ Histogram
```
kiinfo -kipid scdetail,coop,rqhist -ts
0416_1523
```

PID Analysis Report with Time filtering
```
kiinfo -kipid scdetail -start 3.0 -end 4.0 -ts
0416_1523
```

Disk Analysis Report with modify I/O Histogram
```
echo “2 5 10 20 50 100” >bkfname
kiinfo -kidsk bkfname=bkfname -ts 0416_1523
```

Kitrace Report with formatted time and the per-cpu sequence counter for a specific CPU
```
kiinfo -kitrace seqcnt,fmttime,cpu=4 -ts
0416_1523
```

```
runki [-h] [-L] [-M] [-U] [-X] [-a] [-j] [-J path] [-f | -p] [-n interface] [-v] [-d duration] [-t maxrun] [-P pid] [-G tgid] [-C cpu] [-D dev]
[-R] [-e vent] [-s subsys] [-I sysignore] [-T timestamp] [-c "comment"]
                -h              Help
                -L              Gather minimal data for local analysis...no gzip/tar
                -M              Include Collectl/MeasureWare data collection
                -U              Include userspace profile (perf) data collection
                -X              Include sar data collection
                -a              Execute 'perf annotate' on userspace profile (perf) data
                -j              Collect Java stacks
                -J <path>       Location of Java jstack command
                -n <interface>  Gather tcpdump trace data
                -v              Skip vxfs, misc disc data collection
                -d <secs>       Duration of KI data collection. Defaults to 20 secs if not specified.
                -t <secs>       Maximum time for data collection tools to run. Defaults to 120 secs if not specified.
                -p              Use liKI DLKM instead of kiinfo/ftrace to dump trace data (DEFAULT).
                -f              Use ftrace tracing instead of liKI DLKM tracing to dump trace data.
                -P <pid>        Filter collection on Task ID or PID (LiKI only)
                -G <tgid>       Filter collection on Task Group ID or TGID (LiKI only)
                -C <cpu>        Filter collection on cpu (LiKI only)
                -D <dev>        Filter collection on device (LiKI only)
                -R              Advance CPU stats (Freq, CPI, LLC Hit%) using MSR registers (LiKI only)
                -e <event>      Identify events to dump. Possible values are: default | all | <event>
                -s <subsys>     Identify subsys to dump. For example: irq, scsi, block, etc.
                -I <sysignore>  File to specify ignored system call (liki only)
                -T <timestamp>  Should be of the form MMDD_HHMM
                -c "comment"    echo comment into comment.$tag file
```

```
kiall [-h] [-l] [-m] [-r] [-f] [-c] [-x] [-t timestamp]
            -h                Help
            -l                Lite version - bypasses some options
            -m                Do not process collectl or MWA data
            -r                restore into created subdir - <host>/<MMDD_HHMM>
            -f                Find KI files in pwd and any directories below
            -c                Cluster-wide processing
            -x                Generate Kparse text format instead of html format
            -V                Use visualization options where possible in reports
            -t <timestamp>    Only process KI budle with matching timestamp
```

```
kiclean [-h] [-r] [-f] [-p] [-v]
            -h                Help
            -r                Recursively traverse sudirs looking for KI data to archive
            -p                Remove PIDS subdirectory
            -v                Remove VIS subdirectory & rel. sh/php/html files
            -f                Force remove misc/tmp files
```