How to identify linux kernel issue?

Kernel Panic
Kernel Hang
Performance issue

Kernel Panic

what's kernel panic

```
/**
 *  panic - halt the system
 *  @fmt: The text string to print
 *
 *  Display a message, then perform cleanups.
 *
 *  This function never returns.
 */
void panic(const char *fmt, ...)
{
```

Linux Guest OS p roblems from customer

1.Guest OS rebooting unexpectedly
  Possible reasons:
    1.Rebooted mannually. <---/var/log/message
    2.Guest OS panic and kdump enabled. <---/var/crash/vmcore
3.Guest OS hung
  Possible reasons:
    1.Guest OS kernel panics and stops there. <-----Do not reboot Guest OS, collect vmss
    2.Guest OS hung. <-----Do not reboot Guest OS, collect vmss
3.Low performance, debug inside guest OS

What is a vmcore?

A vmcore (Virtual Memory Core) is a dump of all memory contents on a system at a point in time.
Can be debugged by crash tool

Why might I want to capture a vmcore?

Vmcores can be useful in many situations:

- They capture system state at a particular point in time. This is often useful at the time of a cras h, or at a user-defined time when crashed manually by the customer.
- Can be used to see why a system might be slow. Can analyse what the system is doing at the ti me.
- <there are many more reasons>

How do I capture a vmcore?

Config kdump and capture vmcore.

Suspend or take a snapshot on VM, collect vmss/vmsn

1. Kdump

1.reserve physical memory for second kernel

2. run kexec system call to start second kernel

3.collect vmcore by second kernel.

2. Suspend or take a snapshot on VM, collect vmss/vmsn

#vmss2core -N6 xxx.vmss

If RHEL6 kernel, to reduce the vmss.core size.
#makedumpfile -f -d 31 -x vmlinux vmss.core vmcore

How to open a vmcore?

1. find kernel version:
#strings vmcore|grep "2.6."

2.extrace debuginfo rpm
#rpm2cpio kernel-debuginfo-2.6.18-128.el5.x86_64.rpm | cpio -idv

3.crash vmlinux vmcore

Environment

host: cybertron.eng.vmware.com
user/passwd: gss/vmware

1. Kernel Panic

kernel Panic analysis

1. Find kernel oops in the log
crash> log

2.Print backtrace of panic thread.
crash>bt

Other commands:
ps - list processes
set <pid> switch process
set -p switch back to panic process
sym - source code file and line number
dis - disassemble the address
rd - read memory

2. Kernel Hang

Hung Caused by D state process

1.First find UNINTERRUPTABLE process

ps -ef|grep UN

2.switch to UN process and print call stack

set <pid>
bt or   foreach UN bt

3. Find the lock owner by reading stack data.

rd xxxxx -e xxxxx -S
file dentry...

Hung caused by CPU softlock up

#sysctl -w kernel.softlockup_panic=1
1. find soft lockup process in dmesg
#log
2. switch to softlock up process
#set <pid>

3. print stack
#bt

Others

Out of Memory
#kmem -i
reached max pid
#ps -ef|wc -l

3.Low performance, debug inside guest OS

3.1 sysrq.

enable sysrq
sysctl -w kernel.sysrq=1
1.to panic guest OS
echo c >/proc/sysrq-trigger
2.dump stacks
echo t >/proc/sysrq-trigger

3.dump memory
echo m >/proc/sysrq-trigger

3.2. perf

#perf top
#perf top -p <pid>

3.3 vmcore

#runq
#foreach RU bt
#ps|grep RU

Useful links

Linux coredump analyzer:
http://cybertron.eng.vmware.com

LCSA:
http://lcsa.eng.vmware.com:5001/

kernel oops analyzer:
https://access.redhat.com/labs/kerneloopsanalyzer

rhel kernel source code:
https://access.redhat.com/lab/psb