Subject: Help Please - Kernel Keeps Crashing
Posted by Anthony Moon on Wed, 03 Aug 2011 18:48:11 GMT
View Forum Message <> Reply to Message

Can someone please help me figure out why our node keeps crashing?

I've got a crash log here:

```
crash: invalid kernel virtual address: 70755f7570635f  type: "possible"
WARNING: cannot read cpu_possible_map
crash: invalid kernel virtual address: 6e  type: "present"
WARNING: cannot read cpu_present_map
crash: invalid kernel virtual address: 776f645f7570635f  type: "online"
WARNING: cannot read cpu_online_map
crash: invalid kernel virtual address: 776f645f7570635f  type: "cpu_online_map"
crash: invalid kernel virtual address: 776f645f7570635f  type: "cpu_online_map"
      KERNEL: /usr/lib/debug/lib/modules/2.6.32-042stab015.1/vmlinux
    DUMPFILE: /var/crash/127.0.0.1-2011-07-31-04:54:18/vmcore  [PARTIAL DUMP]
        CPUS: 24
        DATE: Sun Jul 31 04:52:59 2011
      UPTIME: 9 days, 18:22:50
LOAD AVERAGE: 1.28, 1.76, 5.71
       TASKS: 2321
    NODENAME: eznode2.ezprovider.net
     RELEASE: 2.6.32-042stab015.1
     VERSION: #1 SMP Mon May 30 22:09:11 MSD 2011
     MACHINE: x86_64  (2394 Mhz)
      MEMORY: 48 GB
       PANIC: "[842341.161749] Oops: 0000 [#1] SMP " (check log for details)
         PID: 313411
     COMMAND: "cpcpan_check_in"
        TASK: ffff88017cf5e140  [THREAD_INFO: ffff880140a98000]
         CPU: 15
       STATE: TASK_RUNNING (PANIC)
```

Even after I updated to the *24.1 kernel it crashed within 24 hours, so now I'm running the *15.1 kernel which only crashes every ten days.

I can provide any other information obtainable from a kernel dump as well.

sysctl.conf:

```
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
```

```
net.ipv4.ip_forward = 1

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 1

# Controls whether core dumps will append the PID to the core filename.
# Useful for debugging multi-threaded applications.
kernel.core_uses_pid = 1

# Controls the use of TCP syncookies
net.ipv4.tcp_syncookies = 1

# Disable netfilter on bridges.
#net.bridge.bridge-nf-call-ip6tables = 0
#net.bridge.bridge-nf-call-iptables = 0
#net.bridge.bridge-nf-call-arptables = 0

# On Hardware Node we generally need
# packet forwarding enabled and proxy arp disabled
net.ipv6.conf.default.forwarding = 1
#net.ipv6.conf.all.forwarding = 0
net.ipv4.conf.default.proxy_arp = 0

# Enables source route verification
net.ipv4.conf.all.rp_filter = 1


# We do not want all our interfaces to send redirects
net.ipv4.conf.default.send_redirects = 1
net.ipv4.conf.all.send_redirects = 0


#network tweaks added by amoon on 08/06/2011
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_timestamps = 1
net.ipv4.tcp_sack = 1
net.ipv4.tcp_fack = 1
net.ipv4.tcp_fin_timeout = 15
net.ipv4.tcp_keepalive_intvl = 15
net.ipv4.tcp_keepalive_probes = 5
net.core.rmem_max = 8388608
net.core.wmem_max = 8388608
```

net.ipv4.tcp_rmem = 4096 87380 8388608
net.ipv4.tcp_wmem = 4096 87380 8388608
net.ipv4.tcp_tw_reuse = 1
net.ipv4.tcp_tw_recycle = 1
net.ipv4.ip_local_port_range = 2000 65000
net.core.netdev_max_backlog = 5000
net.netfilter.nf_conntrack_max = 1787040
net.ipv4.tcp_max_tw_buckets = 1440000
net.ipv4.tcp_max_tw_buckets_ub = 132288

# Controls the maximum size of a message, in bytes
kernel.msgmnb = 65536

# Controls the default maxmimum size of a mesage queue
kernel.msgmax = 64000

# Controls the maximum shared segment size, in bytes
kernel.shmmax = 1073741824

# Controls the maximum number of shared memory segments, in pages
kernel.shmall = 104857600

# System wide maximum number of message queues: policy dependent (on Linux, this limit can be read and modified via /proc/sys/kernel/msgmni).
kernel.msgmni = 2048

# Increase Semaphores limits due to high number of processes running
kernel.sem = 500 512000 64 2048