

CMPE283 : Virtualization

Assignment 2: KVM Statistics

Name: Ananya Praveen Shetty

Student id: 017553276

Date: 11/19/2024

Professor : Prof Mike Larkin

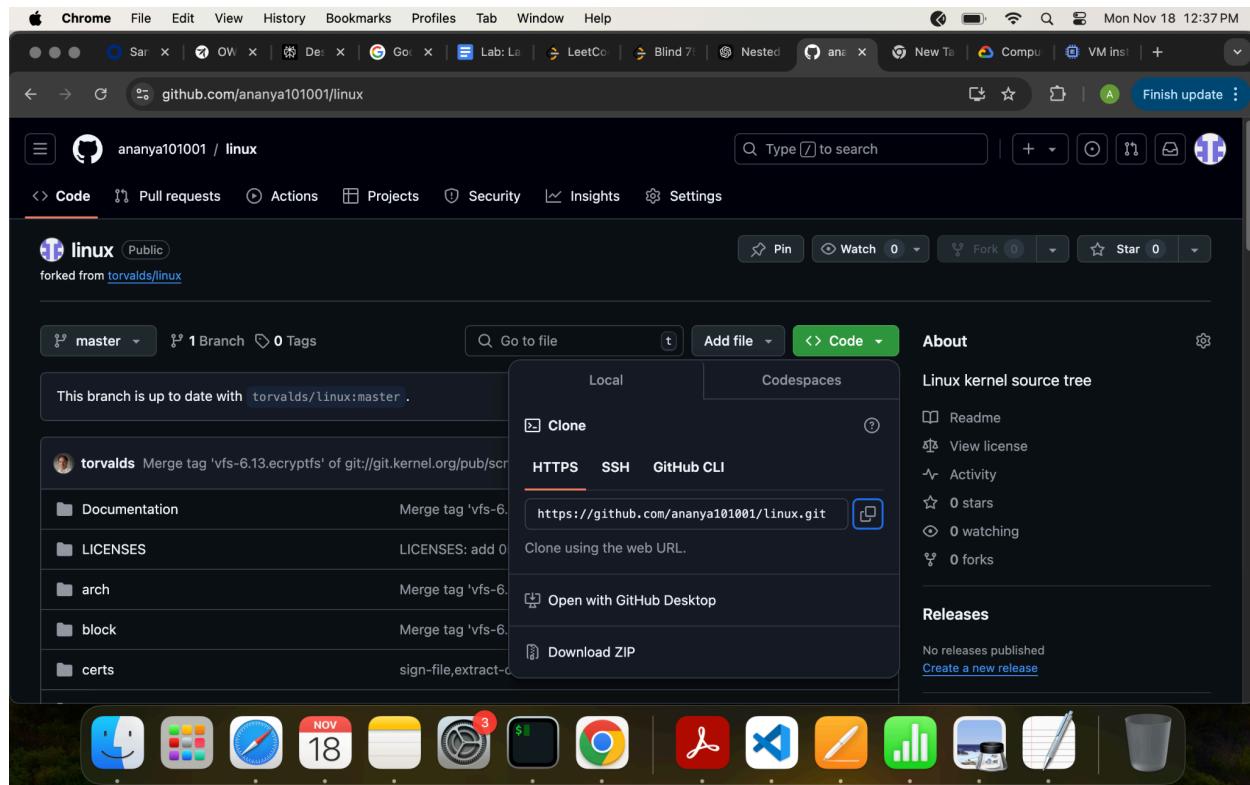
Status: Doing Individually

The Assignment - Your assignment is to modify the KVM source code in your “outer” VM and add statistics for each type of exit (how many of each type have occurred since boot). Every 10,000 total exits, you should dump (print) the total counts for each type of exit. You can omit exits that have a zero count from the printed output.

Step 1

To do the assignment, you should first perform a kernel build test:

- Fork the master linux Github repository at <https://github.com/torvalds/linux> into your own Github



account.

- Clone your own fork created above into the outer VM

→

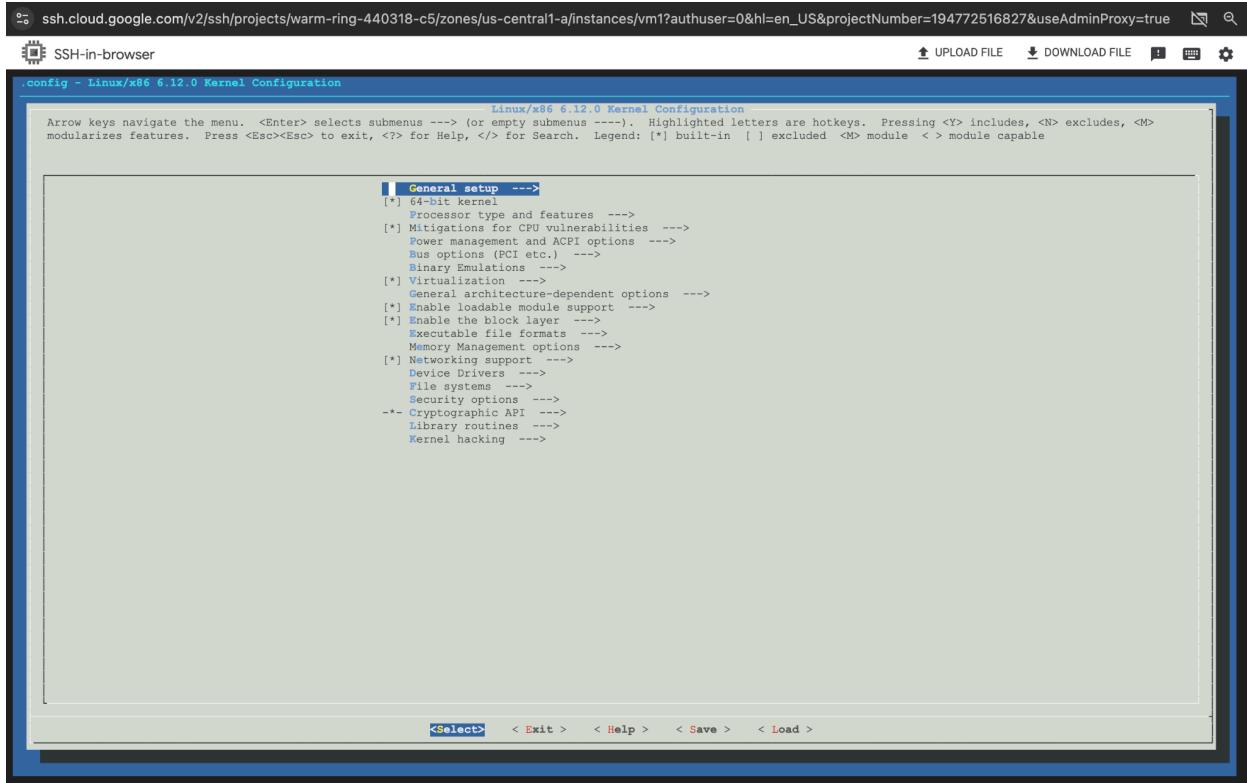
```
last login: Fri Nov 10 20:58:10 2021 from 85.233.218.150
ananyapraveen_shetty@vm1:~$ git clone https://github.com/ananya101001/linux.git
Cloning into 'linux'...
remote: Enumerating objects: 10461724, done.
remote: Counting objects: 100% (389/389), done.
remote: Compressing objects: 100% (203/203), done.
Receiving objects: 100% (10461724/10461724), 5.02 GiB | 22.90 MiB/s, done.
remote: Total 10461724 (delta 268), reused 227 (delta 186), pack-reused 10461335 (from 1)
Resolving deltas: 100% (8523258/8523258), done.
Checking objects: 100% (33554432/33554432), done.
Updating files: 100% (86696/86696), done.
ananyapraveen_shetty@vm1:~$ █
```

- Configure a linux kernel build (make config / make menuconfig, etc) using that cloned tree

```
ananyapraveen_shetty@vm1:~/linux$ make menuconfig
HOSTCC scripts/basic/fixdep
HOSTCC scripts/kconfig/mconf.o
HOSTCC scripts/kconfig/lxdialog/checklist.o
HOSTCC scripts/kconfig/lxdialog/inputbox.o
HOSTCC scripts/kconfig/lxdialog/menubox.o
HOSTCC scripts/kconfig/lxdialog/textbox.o
HOSTCC scripts/kconfig/lxdialog/util.o
HOSTCC scripts/kconfig/lxdialog/yesno.o
HOSTCC scripts/kconfig/mnconf-common.o
HOSTCC scripts/kconfig/confdata.o
HOSTCC scripts/kconfig/expr.o
LEX scripts/kconfig/lexer.lex.c
YACC scripts/kconfig/parser.tab.[ch]
HOSTCC scripts/kconfig/lexer.lex.o
HOSTCC scripts/kconfig/menu.o
HOSTCC scripts/kconfig/parser.tab.o
HOSTCC scripts/kconfig/preprocess.o
HOSTCC scripts/kconfig/symbol.o
HOSTCC scripts/kconfig/util.o
HOSTLD scripts/kconfig/mconf
#
# using defaults found in /boot/config-5.10.0-33-cloud-amd64
#
.config:837:warning: symbol value '0' invalid for BASE_SMALL
.config:1893:warning: symbol value 'm' invalid for PVPANIC
.config:2964:warning: symbol value 'm' invalid for VFIO_VIRQFD
.config:3303:warning: symbol value 'm' invalid for FSCACHE
.config:3719:warning: symbol value 'm' invalid for CRYPTO_BLAKE2S_X86
configuration written to .config

*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.

ananyapraveen_shetty@vm1:~/linux$ █
```



```
ananyapravesh@shetty:~/linux$ make -j$(nproc)
make[1]: Entering directory '/root/linux'
make[1]: Nothing to be done for 'install'.
make: Nothing to be done for 'install'.
SYSHDR arch/x86/include/generated/uapi/asm/unistd_32.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_64.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_x32.h
SYSTBL arch/x86/include/generated/asm/syscalls_32.h
SYSHDR arch/x86/include/generated/asm/unistd_32 ia32.h
SYSHDR arch/x86/include/generated/asm/unistd_64 x32.h
SYSTBL arch/x86/include/generated/asm/syscalls_64.h
HYPERCALLS arch/x86/include/generated/asm/xen-hypercalls.h
HOSTCC arch/x86/tools/relocs_32.o
HOSTCC arch/x86/tools/relocs_64.o
HOSTCC arch/x86/tools/relocs_common.o
HOSTLD arch/x86/tools/relocs
HOSTCC scripts/gensyms/gensyms.o
YACC scripts/gensyms/parse.tab.[ch]
HOSTCC scripts/gensyms/parse.tab.o
LIBGENSYMS scripts/gensyms/lex.lex.c
HOSTCC scripts/gensyms/lex.lex.o
HOSTLD scripts/gensyms/gensyms
HOSTCC scripts/sellinux/genheaders/genheaders
HOSTCC scripts/sellinux/mdp/mdp
HOSTCC scripts/kallsyms
HOSTCC scripts/sortable
HOSTCC scripts/asnl_compiler
HOSTCC scripts/sign-file
WRAP arch/x86/include/generated/uapi/asm/bpf_perf_event.h
WRAP arch/x86/include/generated/uapi/asm/errno.h
WRAP arch/x86/include/generated/uapi/asm/fcntl.h
WRAP arch/x86/include/generated/uapi/asm/locl.h
WRAP arch/x86/include/generated/uapi/asm/loctls.h
WRAP arch/x86/include/generated/uapi/asm/ipcbuf.h
WRAP arch/x86/include/generated/uapi/asm/param.h
WRAP arch/x86/include/generated/uapi/asm/poll.h
WRAP arch/x86/include/generated/uapi/asm/resource.h
WRAP arch/x86/include/generated/uapi/asm/socket.h
WRAP arch/x86/include/generated/uapi/asm/sockios.h
WRAP arch/x86/include/generated/uapi/asm/termkits.h
WRAP arch/x86/include/generated/uapi/asm/termios.h
WRAP arch/x86/include/generated/uapi/asm/types.h
WRAP arch/x86/include/generated/asm/early_ioremap.h
WRAP arch/x86/include/generated/asm/mcs_spinlock.h
WRAP arch/x86/include/generated/asm/mmzone.h
WRAP arch/x86/include/generated/asm/irq_regs.h
WRAP arch/x86/include/generated/asm/kmap_size.h
WRAP arch/x86/include/generated/asm/local64.h
WRAP arch/x86/include/generated/asm/mniowb.h
WRAP arch/x86/include/generated/asm/module_ids.h
WRAP arch/x86/include/generated/asm/rwonce.h
GEN arch/x86/include/generated/asm/orc_hash.h
UPD include/config/kernel.release
UPD include/generated/uapi/linux/version.h
```

```

Setting up patch (2.7.6-7) ...
Setting up libgdbm-compat4:amd64 (1.19-2) ...
Setting up libfl12:amd64 (2.6.4-8) ...
Setting up libperl5.32:amd64 (5.32.1-4+deb11u4) ...
Setting up libubsan1:amd64 (10.2.1-6) ...
Setting up libnsl-dev:amd64 (1.3.0-2) ...
Setting up libcrypt-dev:amd64 (1:4.4.18-4) ...
Setting up git-man (1:2.30.2-1+deb11u3) ...
Setting up libbinutils:amd64 (2.35.2-2) ...
Setting up libis123:amd64 (0.23-1) ...
Setting up libc-dev-bin (2.31-13+deb11u11) ...
Setting up libcc1-0:amd64 (10.2.1-6) ...
Setting up liblocale-gettext-perl (1.07-4+b1) ...
Setting up liblsan0:amd64 (10.2.1-6) ...
Setting up cpp-10 (10.2.1-6) ...
Setting up libitm1:amd64 (10.2.1-6) ...
Setting up libc-devtools (2.31-13+deb11u11) ...
Setting up libtsan0:amd64 (10.2.1-6) ...
Setting up libctf0:amd64 (2.35.2-2) ...
Setting up libgcc-10-dev:amd64 (10.2.1-6) ...
Setting up m4 (1.4.18-5) ...
Setting up perl (5.32.1-4+deb11u4) ...
Setting up libdpkg-perl (1.20.13) ...
Setting up cpp (4:10.2.1-1) ...
Setting up bison (2:3.7.5+dfsg-1) ...
update-alternatives: using /usr/bin/bison.yacc to provide /usr/bin/yacc (yacc) in auto mode
Setting up libc6-dev:amd64 (2.31-13+deb11u11) ...
Setting up binutils-x86-64-linux-gnu (2.35.2-2) ...
Setting up flex (2.6.4-8) ...
Setting up libncurses-dev:amd64 (6.2-20200114-2+deb11u2) ...
Setting up libstdc++-10-dev:amd64 (10.2.1-6) ...
Setting up libfile-fcntllock-perl (0.22-3+b7) ...
Setting up libalgorithm-diff-perl (1.201-1) ...
Setting up binutils (2.35.2-2) ...
Setting up libfl1-dev:amd64 (2.6.4-8) ...
Setting up dpkg-dev (1.20.13) ...
Setting up liberror-perl (0.17029-1) ...
Setting up git (1:2.30.2-1+deb11u3) ...
Setting up gcc-10 (10.2.1-6) ...
Setting up zlib1g-dev:amd64 (1:1.2.11.dfsg-2+deb11u2) ...
Setting up libalgorithm-diff-xs-perl (0.04-6+b1) ...
Setting up libalgorithm-merge-perl (0.08-3) ...
Setting up g++-10 (10.2.1-6) ...
Setting up gcc (4:10.2.1-1) ...
Setting up libelf-dev:amd64 (0.183-1) ...
Setting up g++ (4:10.2.1-1) ...
update-alternatives: using /usr/bin/g++ to provide /usr/bin/c++ (c++) in auto mode
Setting up build-essential (12.9)
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for libc-bin (2.31-13+deb11u11) ...
ananyapraveen_shetty@vm1:~$ 

```

```

Com | VM | x ssh | Nest | linux | Goo | CMF | [ 56 ] | linu | Step | Goo | Unti | anar | New | + 
← → G ssh.cloud.google.com/v2/ssh/projects/warm-ring-440318-c5/zones/us-central1-a/instances/vm1?authuser=0&hl=en_US&projectN... 🔍 ⌂ 
SSH-in-browser
ananyapraveen_shetty@vm1:~/linux$ sudo make modules_install
INSTALL /lib/modules/6.12.0+/modules.order
INSTALL /lib/modules/6.12.0+/modules.builtin
INSTALL /lib/modules/6.12.0+/modules.builtin.modinfo
SYMLINK /lib/modules/6.12.0+/build
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/events/amd/power.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/events/intel/intel-uncore.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/events/intel/intel-cstate.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/events/rapl.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/events/thermal.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/intel/cpuid.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/twofish-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/twofish-x86_64-3way.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/twofish-avx-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/serpent-sse2-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/serpent-avx-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/serpent-avx2.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/serpent-avx2-3way.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/camellia-aesni-avx2-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/camellia-aesni-avx2_ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/blowfish-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/cast5-avx-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/cast5-avx-x86_64.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/aegis128-aesni.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/aegis128-aesni_ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/aesni-intel.ko
INSTALL /lib/modules/6.12.0+/kernel/arch/x86/crypto/aesni-intel_ko
INSTALL /lib/modules/6.12.0+/kernel/fs/nfs/common/nfs_acl.ko
INSTALL /lib/modules/6.12.0+/kernel/fs/nfs/quota/quota_v1.ko
INSTALL /lib/modules/6.12.0+/kernel/fs/quota/quota_v2.ko
INSTALL /lib/modules/6.12.0+/kernel/fs/quota/quota_tree.ko
INSTALL /lib/modules/6.12.0+/kernel/fs/nls/nls_cp437.ko
INSTALL /lib/modules/6.12.0+/kernel/fs/nls/nls_cp737.ko
INSTALL /lib/modules/6.12.0+/kernel/fs/nls/nls_cp775.ko

```

```

ananyapraveen_shetty@vm1:~/linux$ sudo make install
    INSTALL /boot
run-parts: executing /etc/kernel/postinst.d/apt-auto-removal 6.12.0+ /boot/vmlinuz-6.12.0+
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 6.12.0+ /boot/vmlinuz-6.12.0+
update-initramfs: Generating /boot/initrd.img-6.12.0+
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 6.12.0+ /boot/vmlinuz-6.12.0+
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 6.12.0+ /boot/vmlinuz-6.12.0+
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.12.0+
Found initrd image: /boot/initrd.img-6.12.0+
Found linux image: /boot/vmlinuz-6.12.0+.old
Found initrd image: /boot/initrd.img-6.12.0+
Found linux image: /boot/vmlinuz-5.10.0-33-cloud-amd64
Found initrd image: /boot/initrd.img-5.10.0-33-cloud-amd64
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Adding boot menu entry for UEFI Firmware Settings ...
done
ananyapraveen_shetty@vm1:~/linux$ 

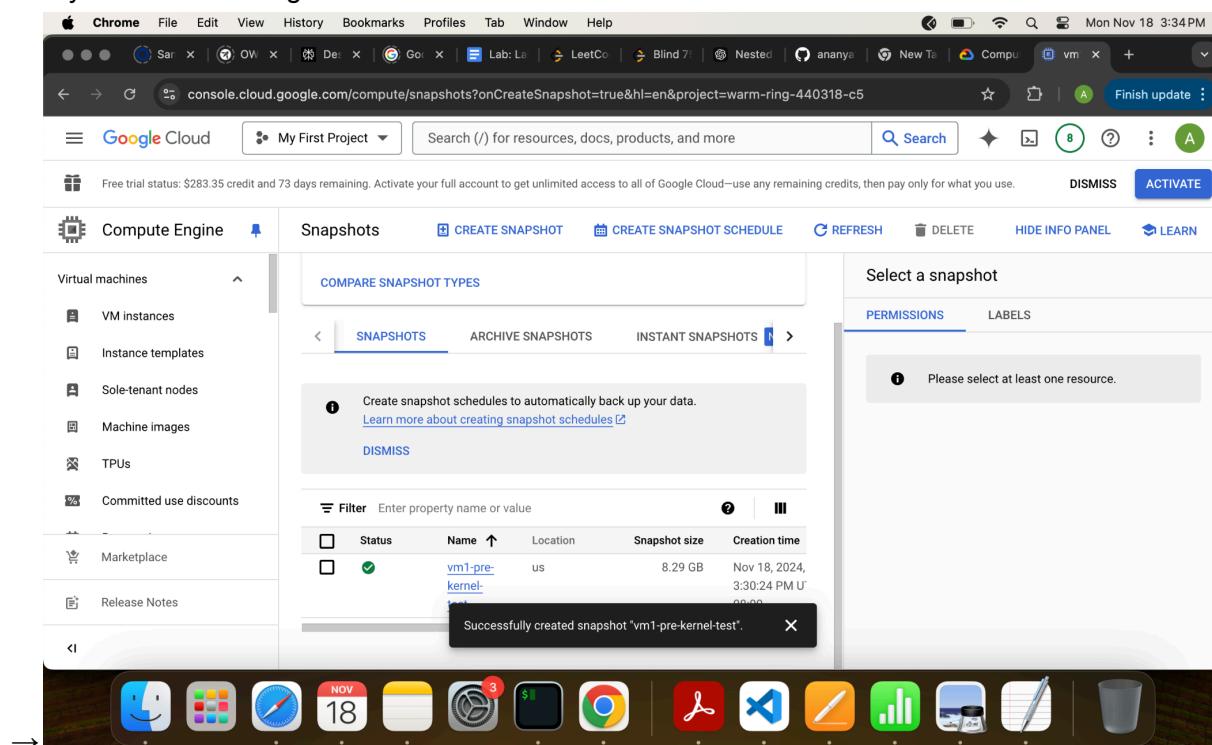
```

```

done
ananyapraveen_shetty@vm1:~/linux$ sudo update-grub
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.12.0+
Found initrd image: /boot/initrd.img-6.12.0+
Found linux image: /boot/vmlinuz-6.12.0+.old
Found initrd image: /boot/initrd.img-6.12.0+
Found linux image: /boot/vmlinuz-5.10.0-33-cloud-amd64
Found initrd image: /boot/initrd.img-5.10.0-33-cloud-amd64
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Adding boot menu entry for UEFI Firmware Settings ...
done
ananyapraveen_shetty@vm1:~/linux$ 

```

At this point, I'd advise taking a snapshot of your VM, in case the newly built kernel is incompatible with your installed image.



The screenshot shows the Google Cloud Compute Engine interface. On the left, a sidebar lists options like Reservations, Migrate to Virtual Machine, Storage (Disks, Storage Pools), Snapshots (selected), Images, Acme Integration, Marketplace, and Release Notes. The main panel displays 'Snapshot details' for 'vm1-pre-kernel-test'. It includes a 'Properties' table with columns for Source disk (vm1), Disk size (50 GB), Snapshot size (8.29 GB), Architecture (x86/64), Snapshot type (Standard snapshot), Location (us (United States)), Labels (None), and Tags (empty). Below this is a 'Creation date' (Nov 18, 2024, 3:30:24 PM UTC-08:00), Encryption type (Google-managed), and Confidential Computing services (Disabled). At the bottom, there's an 'EQUIVALENT REST' section with various icons.

```
ananyapraveen_shetty@vm1:~/linux$ sudo reboot
```

Step 2:

The next step involves editing the kernel KVM source code and adding the counters for the various types of exits.

```
done
```

```
ananyapraveen_shetty@vm1:~/linux$ nano ~/linux/arch/x86/kvm/vmx/vmx.c
```

- Locate the exit handler function in KVM
- At the start of the function, increment a per-exit-type counter for the exit type number as well as a global “total exits” counter.

```

#include <linux/atomic.h>

// Global variables to count exits
static atomic_t total_exits = ATOMIC_INIT(0);
static atomic_t exit_counters[65] = { [0 ... 64] = ATOMIC_INIT(0) };

static const char *exit_reason_str(int exit_type) {
    switch (exit_type) {
        case 0: return "EXCEPTION_NMI";
        case 1: return "EXTERNAL_INTERRUPT";
        case 2: return "TRIPLE_FAULT";
        case 7: return "INTERRUPT_WINDOW";
        case 10: return "CPUID";
        case 12: return "HLT";
        case 16: return "RDTSC";
        case 28: return "CR_ACCESS";
        case 30: return "I/O_INSTRUCTION";
        case 31: return "RDMSR";
        case 32: return "WRMSR";
        case 48: return "EPT_VIOLATION";
        case 49: return "EPT_MISCONFIG";
        case 54: return "WBINVND";
        default: return "UNKNOWN";
    }
}

int vmx_handle_exit(struct kvm_vcpu *vcpu, fastpath_t exit_fastpath)
{
    printk(KERN_INFO "KVM: VMX exit occurred\n");
    struct vcpu_vmx *vmx = to_vmx(vcpu);
    int exit_type = vmx->exit_reason.basic;
    long total = atomic_inc_return(&total_exits);
    if (exit_type < 65)
        atomic_inc(&exit_counters[exit_type]);
    if (total % 10000 == 0) {
        int i;
        for (i = 0; i < 65; i++) {
            int count = atomic_read(&exit_counters[i]);
            if (count > 0) {
                printk(KERN_INFO "KVM exit %d (%s): %d\n", i, exit_reason_str(i), count);
            }
        }
    }
}

```

File menu: Help, Write Out, Where Is, Cut, Execute, Location, Undo, Set Mark, To Bracket, Previous, Exit, Read File, Replace, Paste, Justify, Go To Line, Redo, Copy, Where Was, Next.

- Every 10,000 total exits, print (via printk) one line for each exit type containing the exit number, a human-readable name of that exit type (eg “RDTSC Exit”), and the total count of exits for that type of exit.

```

int vmx_handle_exit(struct kvm_vcpu *vcpu, fastpath_t exit_fastpath)
{
    printk(KERN_INFO "KVM: VMX exit occurred\n");
    struct vcpu_vmx *vmx = to_vmx(vcpu);
    int exit_type = vmx->exit_reason.basic;
    long total = atomic_inc_return(&total_exits);
    if (exit_type < 65)
        atomic_inc(exit_counters[exit_type]);
    if (total % 10000 == 0) {
        int i;
        for (i = 0; i < 65; i++) {
            int count = atomic_read(&exit_counters[i]);
            if (count > 0) {
                printk(KERN_INFO "KVM exit %d (%s): %d\n", i, exit_reason_str(i), count);
            }
        }
    }
}

int ret = __vmx_handle_exit(vcpu, exit_fastpath);

/*
 * Exit to user space when bus lock detected to inform that there is
 * a bus lock in guest.
 */
if (to_vmx(vcpu)->exit_reason.bus_lock_detected) {
    if (ret > 0)
        vcpu->run->exit_reason = KVM_EXIT_X86_BUS_LOCK;

    vcpu->run->flags |= KVM_RUN_X86_BUS_LOCK;
    return 0;
}
return ret;
}

/*
 * Software based L1D cache flush which is used when microcode providing
 * the cache control MSR is not loaded.
 *
 * The L1D cache is 32 KiB on Nehalem and later microarchitectures, but to
 * flush it is required to read in 64 KiB because the replacement algorithm
 * is not exactly LRU. This could be sized at runtime via topology
 * information but as all relevant affected CPUs have 32KiB L1D cache size
 * there is no point in doing so.

```

^C Help ^O Write Out ^W Where Is ^R Cut ^T Execute ^C Location M-U Undo
 ^X Exit ^R Read File ^I Replace ^U Paste ^J Justify ^G Go To Line M-P Redo
 M-A Set Mark M-J To Bracket M-Q Previous
 M-G Where Was M-W Next

- Build and install the new kernel containing your changes
- Reboot the outer VM

```

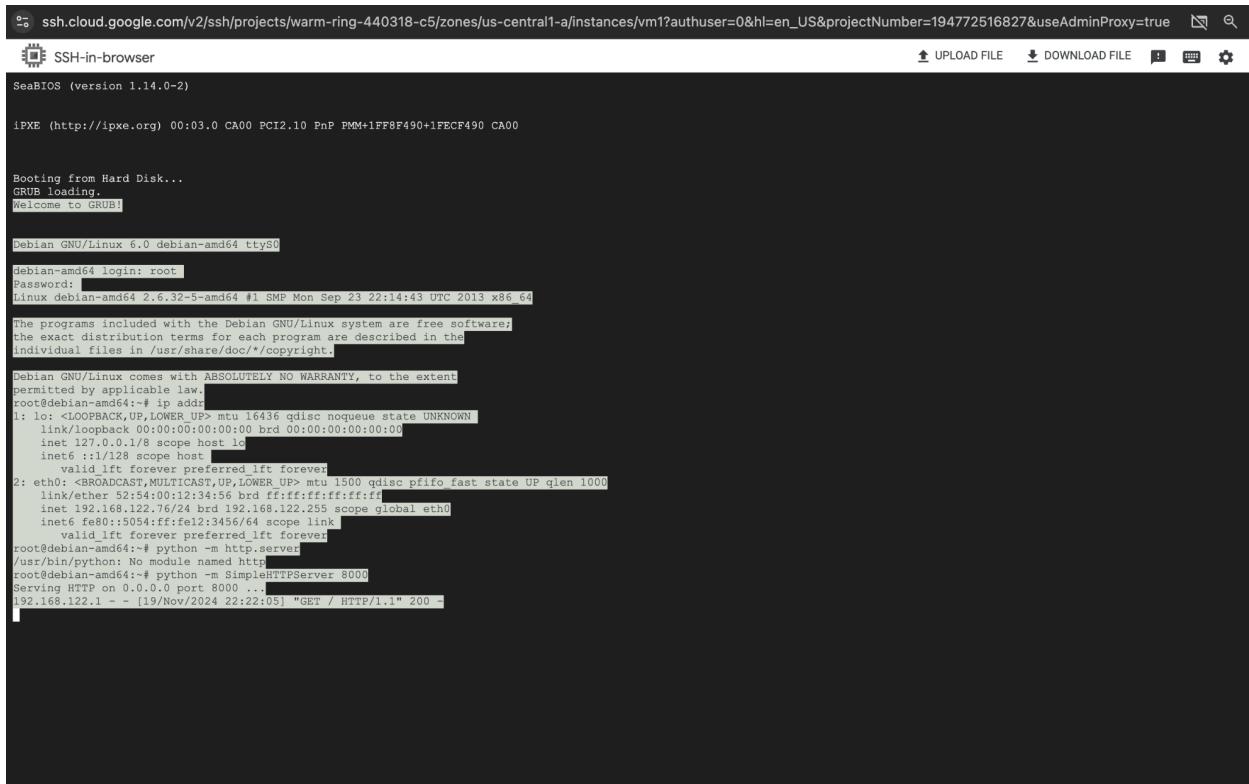
cd ~/linux
make -j$(nproc)
sudo make modules_install
sudo make install
sudo update-grub
sudo reboot

```

Step 3

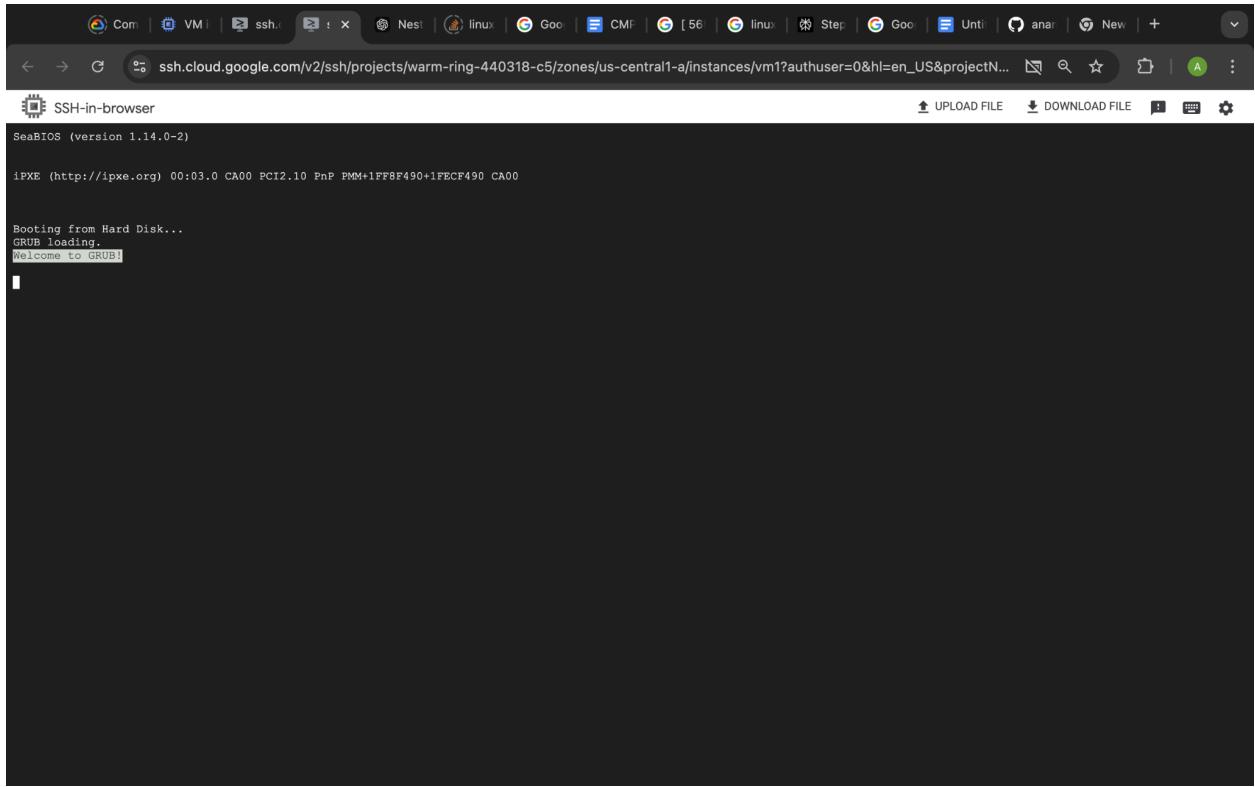
Finally, boot the inner VM using your modified KVM code.

```
→ sudo qemu-system-x86_64 -enable-kvm -hda  
debian_squeeze_amd64_standard.qcow2 -m 512 -net nic -net tap,ifname=tap0,script=no  
-nographic
```



The screenshot shows an SSH session in a browser window titled "SSH-in-browser". The session is connected to a Debian system. The terminal output is as follows:

```
SeabIOS (version 1.14.0-2)  
iPXE (http://ipxe.org) 00:03.0 CA00 PCI2.10 PnP FMM+1FP8F490+1FECF490 CA00  
  
Booting from Hard Disk...  
GRUB loading.  
Welcome to GRUB!  
  
Debian GNU/Linux 6.0 debian-amd64 ttyS0  
debian-amd64 login: root  
Password: [REDACTED]  
Linux debian-amd64 2.6.32-5-amd64 #1 SMP Mon Sep 23 22:14:43 UTC 2013 x86_64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
root@debian-amd64:~# ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
      inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000  
    link/ether 52:54:00:12:34:56 brd ff:ffff:ffff:ff:  
    inet 192.168.122.76/24 brd 192.168.122.255 scope global eth0  
      inet6 fe80::5054:ff:fe12:3456/64 scope link  
        valid_lft forever preferred_lft forever  
root@debian-amd64:~# python -m SimpleHTTPServer  
/usr/bin/python: No module named http  
root@debian-amd64:~# python -m SimpleHTTPServer 8000  
Serving HTTP on 0.0.0.0 port 8000 ...  
192.168.122.1 - [19/Nov/2024 22:22:05] "GET / HTTP/1.1" 200 -
```



You should periodically check the printk output via “dmesg” in the outer VM to see what type(s) of exits are occurring.

```
ananyapraveen_shetty@vm1:~$ cd linux
ananyapraveen_shetty@vm1:~/linux$ sudo dmesg | grep "KVM exit"
ananyapraveen_shetty@vm1:~/linux$ sudo dmesg | grep "KVM exit"
[ 216.050294] KVM exit 7 (UNKNOWN) : 2831
[ 216.055651] KVM exit 10 (CPUID) : 270
[ 216.060760] KVM exit 28 (UNKNOWN) : 417
[ 216.066037] KVM exit 30 (UNKNOWN) : 14860
[ 216.071481] KVM exit 31 (UNKNOWN) : 1
[ 216.076583] KVM exit 32 (UNKNOWN) : 32
[ 216.081780] KVM exit 48 (UNKNOWN) : 529
[ 216.087071] KVM exit 49 (UNKNOWN) : 912
[ 216.092320] KVM exit 54 (UNKNOWN) : 2
```

```
ananyapraveen_shetty@vm1:~$ sudo dmesg | grep "KVM exit"
ananyapraveen_shetty@vm1:~$ cd ~/linux
ananyapraveen_shetty@vm1:~/linux$ sudo dmesg | grep "KVM exit"
[ 761.595182] KVM exit 0 (EXCEPTION_NMI): 11
[ 761.600799] KVM exit 1 (EXTERNAL_INTERRUPT): 501
[ 761.607087] KVM exit 7 (INTERRUPT_WINDOW): 7728
[ 761.613324] KVM exit 10 (CPUID): 336
[ 761.618431] KVM exit 28 (CR_ACCESS): 1097
[ 761.623975] KVM exit 30 (I/O_INSTRUCTION): 58400
[ 761.630310] KVM exit 31 (RDMSR): 1
[ 761.635273] KVM exit 32 (WRMSR): 32
[ 761.640319] KVM exit 48 (EPT_VIOLATION): 980
[ 761.646107] KVM exit 49 (EPT_MISCONFIG): 912
[ 761.651992] KVM exit 54 (WBINVD): 2
ananyapraveen_shetty@vm1:~/linux$ █
```

```
[ 3461.249430] KVM: VMX exit occurred
[ 3461.254284] Hello: Exit Type 2, Count: 469934
[ 3461.260469] KVM: VMX exit occurred
[ 3461.265209] Hello: Exit Type 2, Count: 469936
[ 3461.271587] KVM: VMX exit occurred
[ 3461.276391] Hello: Exit Type 2, Count: 469938
[ 3461.293237] KVM: VMX exit occurred
[ 3461.297405] KVM: VMX exit occurred
[ 3461.302195] Hello: Exit Type 2, Count: 469940
[ 3461.316855] KVM: VMX exit occurred
[ 3461.321306] Hello: Exit Type 2, Count: 469942
[ 3461.327341] KVM: VMX exit occurred
[ 3461.332203] KVM: VMX exit occurred
[ 3461.337131] KVM: VMX exit occurred
[ 3461.341960] Hello: Exit Type 2, Count: 469944
[ 3461.348000] KVM: VMX exit occurred
[ 3461.352850] Hello: Exit Type 2, Count: 469946
[ 3461.358947] KVM: VMX exit occurred
[ 3461.363919] KVM: VMX exit occurred
[ 3461.369088] KVM: VMX exit occurred
[ 3461.373924] Hello: Exit Type 2, Count: 469948
[ 3461.382193] KVM: VMX exit occurred
[ 3461.386170] Hello: Exit Type 2, Count: 469950
[ 3461.392263] KVM: VMX exit occurred
[ 3461.397192] KVM: VMX exit occurred
[ 3461.402137] KVM: VMX exit occurred
[ 3461.406963] Hello: Exit Type 2, Count: 469952
[ 3461.415061] KVM: VMX exit occurred
[ 3461.419260] Hello: Exit Type 2, Count: 469954
[ 3461.428455] KVM: VMX exit occurred
[ 3461.432965] KVM: VMX exit occurred
[ 3461.437876] KVM: VMX exit occurred
[ 3461.442717] Hello: Exit Type 2, Count: 469956
[ 3461.448751] KVM: VMX exit occurred
[ 3461.453536] Hello: Exit Type 2, Count: 469958
[ 3461.461799] KVM: VMX exit occurred
[ 3461.465732] KVM: VMX exit occurred
[ 3461.470755] KVM: VMX exit occurred
[ 3461.475558] Hello: Exit Type 2, Count: 469960
[ 3461.483894] KVM: VMX exit occurred
[ 3461.487835] Hello: Exit Type 2, Count: 469962
[ 3461.495894] KVM: VMX exit occurred
[ 3461.500155] KVM: VMX exit occurred
[ 3461.505053] KVM: VMX exit occurred
[ 3461.509910] Hello: Exit Type 2, Count: 469964
[ 3461.515917] KVM: VMX exit occurred
[ 3461.520812] Hello: Exit Type 2, Count: 469966
ananyapraveen_shetty@vm1:~/linux$ sudo dmesg | grep "KVM exit"
ananyapraveen_shetty@vm1:~/linux$ sudo dmesg | grep "KVM exit"
```

Thank You !