CMPE 283

Virtualization

Assignment 2: Modifying instruction behavior in KVM

Question 2: Steps to follow

1. Install git using following commands
   1. sudo apt-get update
   2. sudo apt-get upgrade
   3. sudo apt --fix-broken install ( I had to run this command to fix problem which was occurring during installation of git)
   4. sudo apt-get install git
2. Clone linux kernel tree from github using following command
   1. git clone <https://github.com/torvalds/linux.git>
3. Change directory to linux
   1. cd linux
4. Reset linux source code repo to stable Version 4.10
   1. git checkout 569dbb88e80deb68974ef6fdd6a13edb9d686261
   2. git reset --hard
5. Enter following command to see git all commit and save latest git commit id
   1. git log
   2. commit id: 569dbb88e80deb68974ef6fdd6a13edb9d686261
6. Go to folder and edit file arch/x86/kvm/cpuid.c
   1. Add a global variable toggle: int toggle = 0;
   2. Change kvm\_emulate\_cpuid function as follows:

int kvm\_emulate\_cpuid(struct kvm\_vcpu \*vcpu)

{

u32 eax, ebx, ecx, edx;

if (cpuid\_fault\_enabled(vcpu) && !kvm\_require\_cpl(vcpu, 0))

return 1;

eax = kvm\_register\_read(vcpu, VCPU\_REGS\_RAX);

ecx = kvm\_register\_read(vcpu, VCPU\_REGS\_RCX);

if(eax==0x0){

if(toggle){

eax=0xd;

ebx=0x45504d43;

ecx=0x45504d43;

edx=0x3338325f;

}

else{

kvm\_cpuid(vcpu, &eax, &ebx, &ecx, &edx);

}

}

else if(eax==0x4fffffff){

toggle = toggle?0:1;

kvm\_cpuid(vcpu, &eax, &ebx, &ecx, &edx);

}

else{

kvm\_cpuid(vcpu, &eax, &ebx, &ecx, &edx);

}

//kvm\_cpuid(vcpu, &eax, &ebx, &ecx, &edx);

kvm\_register\_write(vcpu, VCPU\_REGS\_RAX, eax);

kvm\_register\_write(vcpu, VCPU\_REGS\_RBX, ebx);

kvm\_register\_write(vcpu, VCPU\_REGS\_RCX, ecx);

kvm\_register\_write(vcpu, VCPU\_REGS\_RDX, edx);

return kvm\_skip\_emulated\_instruction(vcpu);

}

1. Run make menuconfig command
2. After updating code run following commands
   1. Sudo make -j4
   2. Sudo make modules -j4
   3. Sudo make modules\_install -j4
   4. Sudo make install -j4
3. Reboot into newly built kernel
4. Install virt-manager
   1. Sudo apt install virt-manager
5. Start linux into a new virtual machine
   1. Test functionality using same VM

12. Clone previous VM in KVM and start that VM with functionality enabled.

12.1 Answer question 3 based on provided testcases.

13. Commit linux source tree modification and take a diff using following command.

Git diff HEAD~1 > cmpe283-2.diff

Question 3: Answer the following questions with the assignment functionality enabled, boot a second VM

* What happens during boot?
  + In the dmesg output following shows up:

CPU: vendor\_id ‘CMPE\_283CMPE’ unknown, using generic init.

CPU: your system may be unstable.

* Does the system behave differently?
  + I have used system for browsing and documentation for 5 to 10 minutes. I didn’t notice anything out of place. I don’t think system behaves differently.
  + May be longer use of system can reveal some abnormalities if there are any.
* Does the content of /proc/cpuinfo change when functionality is enabled vs disabled?
  + No, the content of /proc/cpuinfo remains same even though functionality is disabled. The vendor\_id remains ‘CMPE\_283CMPE’ unchanged. I think the VM only fetches these information when it starts and then provide same output every time.
* What happens if you disable the functionality and restart the test VM?
  + After disabling the functionality when i restarted second test VM.
  + Dmesg output is changed and the error above mentioned is gone.
  + Also output of cat /proc/cpuinfo is changed and vendor\_id is now changed to ‘GenuineIntel’