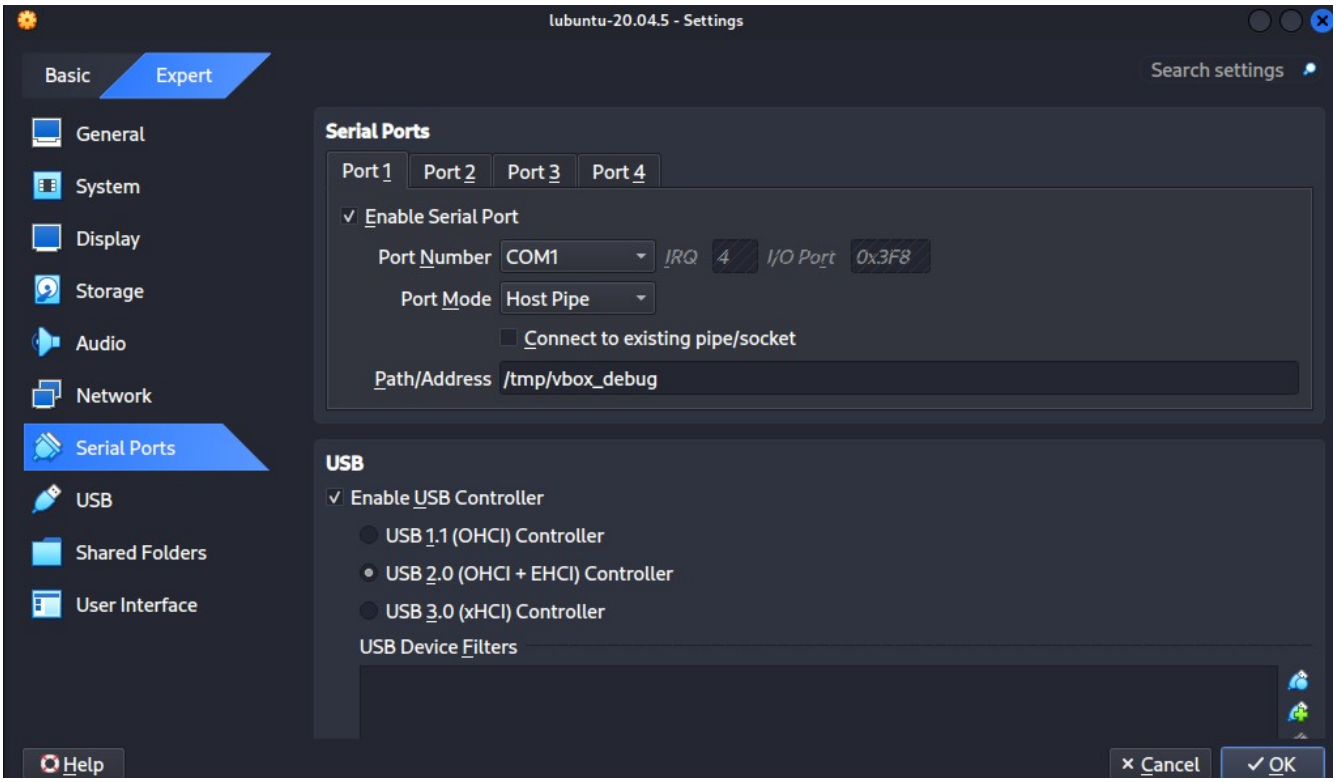
6.2. Teknik debugging Linux Kernel pada virtualbox

Debugging linux kernel diperlukan saat proses pembuatan LKM (Loadable Kernel Module) dan proses exploit development untuk linux kernel.

Untuk debugging linux kernel kita akan menggunakan virtualbox, socat, gdb, gdb-multiarch dan gef. Untuk gdb dan gef sudah diinstall sebelumnya jadi tidak perlu kita atur lagi.

Pertama tama pada virtualbox untuk guest os, pergi ke pengaturan Settings, lalu pilih Serial Ports



- Centang pada bagian Enable Serial Port
- Port Number : COM1
- Port Mode : Host Pipe
- Uncheck pada bagian Connect to existing pipe / socket
- Path/Address : /tmp/vbox_debug
- Klik ok

Selanjutnya jalankan guest os.

Agar bisa mendebug linux kernel di host os, kita perlu mendownload vmlinux dari guest os ke kali linux kita, untuk itu kita bisa upload vmlinux dari guest os ke mesin host os dengan ftp misalnya.

Pada contoh kali ini saya sudah menginstall openssh server di host os, saya menggunakan sftp untuk transfer vmlinux ke host os.

Pada guest os, buka terminal, lalu ke direktori /boot:

cd /boot

ls vmlinuz*

Misal isinya :

vmlinuz vmlinuz-5.15.0-46-generic vmlinuz.old

Yang perlu kita unggah ke host os adalah vmlinuz-5.15.0-46-generic

di sini saya menggunakan sftp :

sftp robohax@192.168.56.1

put vmlinuz-5.15.0-46-generic

Setelah vmlinuz didownload ke host os selanjutnya kita atur grub di guest os:

sudo nano /etc/default/grub

Pada bagian GRUB_CMDLINE_LINUX_DEFAULT, ganti menjadi :

GRUB_CMDLINE_LINUX_DEFAULT="quiet nokaslr kgdboc=ttyS0,115200"

Selanjutnya simpan, lalu update grub:

sudo update-grub

Selanjutnya power off guest os.

Kembali ke host os, kita akan menginstall gdb-multiarch. Gdb-multiarch ini adalah versi gdb yang bisa digunakan untuk mendebug berbagai macam arsitektur misal : x86, x64, arm64, mips,dll.

sudo apt install gdb-multiarch -y

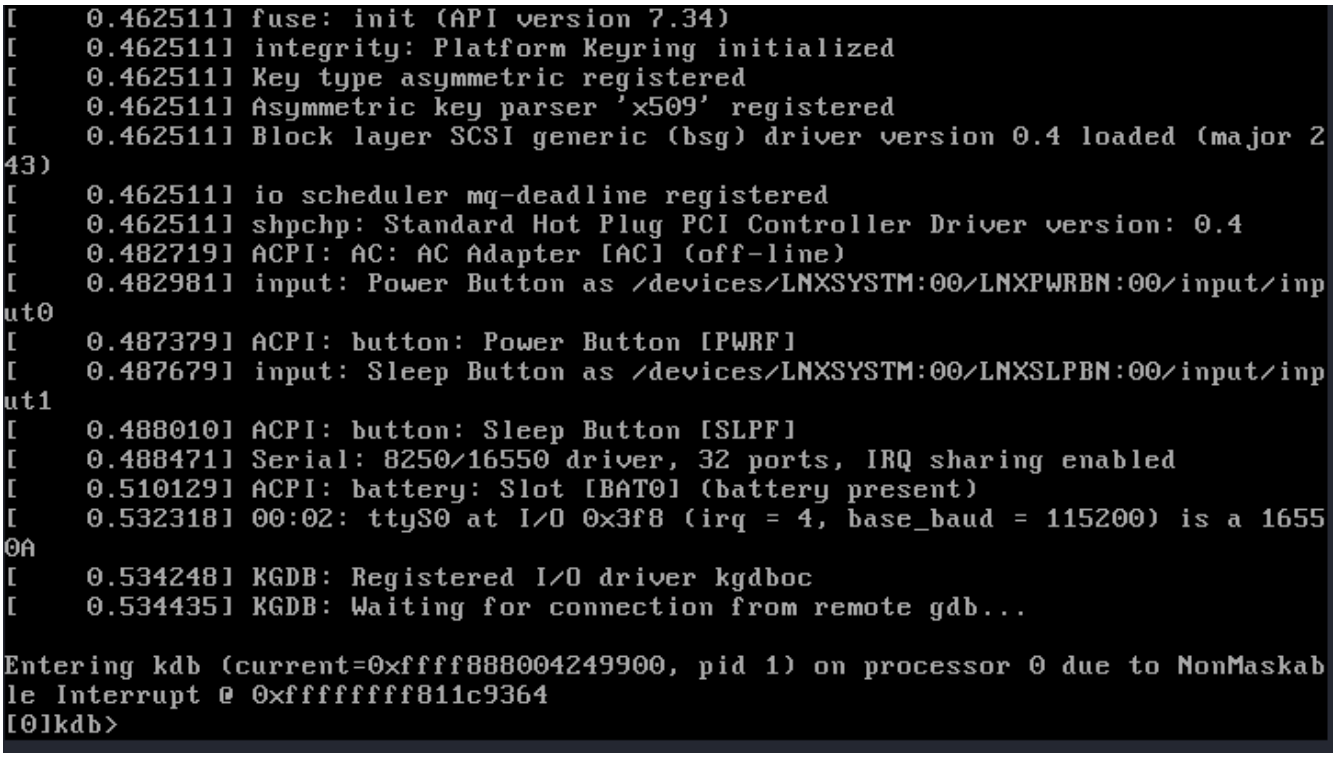
Selanjutnya untuk mempermudah menjalankan gdb-multiarch, buat symlink:

sudo ln -s /usr/bin/gdb-multiarch /sbin/gm

Install socat di host os jika belum ada !

Langkah selanjutnya start guest os.

Pada saat guest os boot, jika pengaturan kita berhasil maka kgdb di guest os akan menunggu koneksi untuk memulai sesi kernel debugging:



Pada host os jalankan socat untuk listen di port 1234 :

socat -d -d UNIX-CLIENT:/tmp/vbox_debug TCP-LISTEN:1234

Selanjutnya pergi ke direktori di mana terdapat vmlinuz yang berasal dari guest os , contoh :

cd /home/robohax/Desktop/sploit/kernelspace/SETUP/vmlinux lubuntu 20.04/

ls -lah

total 11M
drwxrwxr-x 2 robohax robohax 4.0K Jan 27 08:39 .
drwxrwxr-x 5 robohax robohax 4.0K Jan 27 08:38 ..
-rwxrwxrwx 1 robohax robohax 11M Jan 27 08:39 vmlinuz-5.15.0-46-generic

ketik sudo su

Jalakan gdb multiarch :

gm ./vmlinuz-5.15.0-46-generic

Pada console gdb, ketikkan :

target remote :1234

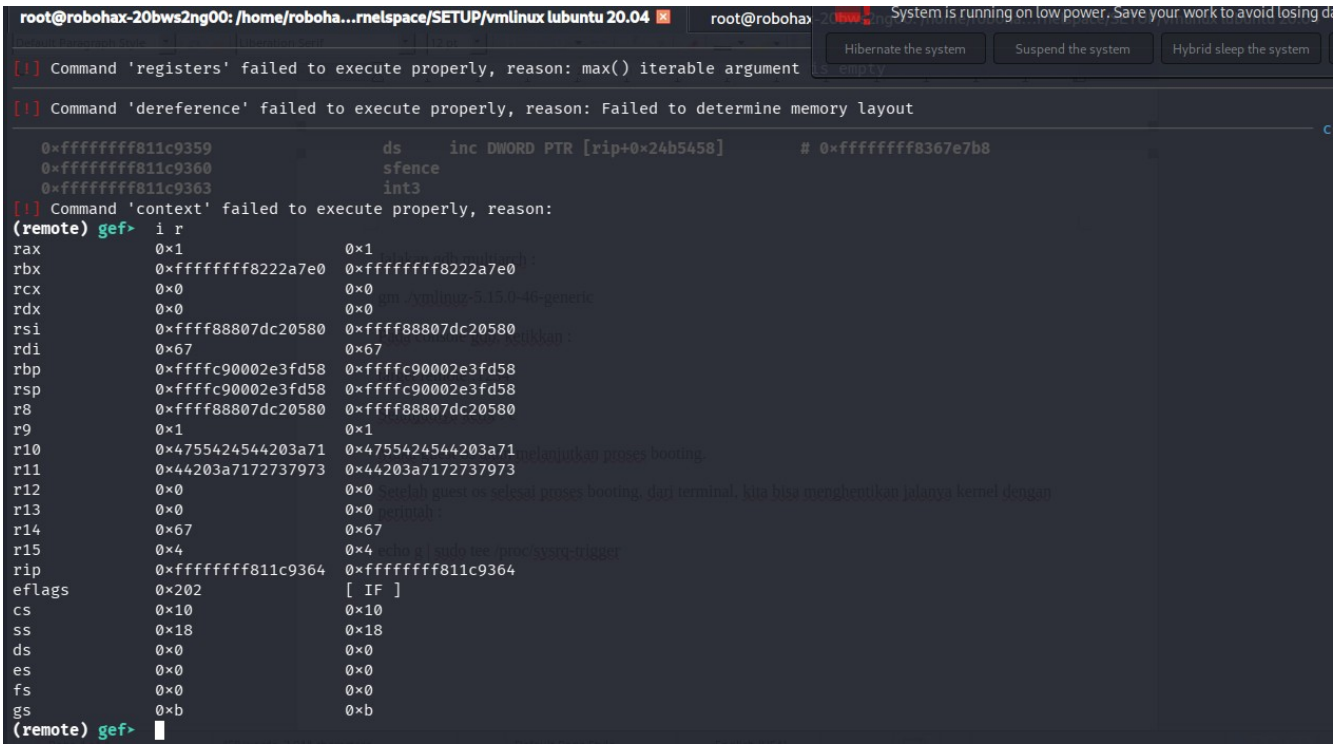
Selanjutnya ketik : c

Maka guest os akan melanjutkan proses booting.

Setelah guest os selesai proses booting, dari terminal, kita bisa menghentikan jalanya kernel dengan perintah :

echo g | sudo tee /proc/sysrq-trigger

Kembali ke jendela gdb di host os, kita bisa melihat informasi yang terdapat pada register kernel di guest os dengan perintah : info registers



Jika ingin menjalankan lagi kernel di guest os, ketik : c