

Report – MP5

Elias Moreno

November 6th, 2020

1. What is the version of the kernel you compiled?
 - 5.9.4
2. Run the following command from a terminal: `uname -a`. What is the output from your old kernel? What's the output from the new kernel?
 - `Linux parallels-Parallels-Virtual-Platform 5.4.0-42-generic #46-Ubuntu SMP Fri Jul 10 00:24:02 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux`
 - `Linux parallels-Parallels-Virtual-Platform 5.9.4-42-generic #1 SMP Tue Jan 24 23:42:05 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux`
3. Go to the `/boot` directory and issue an `ls -rlt` command (this puts newly created files at the bottom). You should see four new files: `vmlinuz-version`, `System.map-version`, `config-version` and `initrd.img-version`, where *version* is the kernel version you compiled. Give a short description of the purpose of each of these four files.
 - `vmlinuz-version`
 - the name of the linux executable
 - `System.map-version`
 - Is a symbol table used by the kernel
 - `config-version`
 - contains the configurations for the kernel
 - `initrd.img-version`
 - *initrd provides the capability to load a RAM disk by the boot loader.*
4. What is the size of the kernel binary (e.g., your `vmlinuz` file) in MB? You can use the `ls -lh` command to determine this.
 - 8.7Mb

5. Find the "Timer frequency" configuration option (under "Processor type and features"). What does this configuration option control? What performance differences do you expect to observe with the different options?
 - Timer frequency configuration controls how long each task can execute before it is interrupted by the CPU
 - You'd expect for more responsiveness with higher timer frequency and less responsiveness with a lower timer frequency
6. What does the configuration menu say about a 100Hz timer interrupt vs a 1000Hz timer interrupt?
 - 1000Hz should be used when you're wanting to improve framerate when gaming
 - 100Hz should be used when you're using a heavy I/O workload.
7. Look for the "Network packet filtering framework (Netfilter)" option and describe what Netfilter is. You may use Netfilter later on (e.g., can be an option for the final project).
 - Netfilter is a filtering framework that filters and mangles network packets that enter your linux machine. This is usually used as a firewall to filter out bad packets of data
8. Look for the "Randomize the address of the kernel image (KASLR)" option. What does this option do?
 - This option randomized the physical address where the kernel image is decompressed and the virtual address where the kernel image is mapped
9. Ensure that you have the "64-bit kernel" option enabled and look under the "Security options" category for the "Remove the kernel mapping in user mode" option. What does the help message say about this option?
 - What the menu page says
10. Is the kernel compiled against a C library (like regular user-space programs)? Why or why not?
 - No because it's only implemented with a user mode and not a kernel mode.

Reference:

1. https://cateee.net/lkddb/web-lkddb/RANDOMIZE_BASE.html
2. <http://www.linfo.org/vmlinuz.html>
3. <https://en.wikipedia.org/wiki/System.map>
4. <https://wiki.debian.org/Initrd>
5. <https://cateee.net/lkddb/web-lkddb/NETFILTER.html#:~:text=Netfilter%20is%20a%20framework%20for,local%20network%20from%20the%20Internet.>
6. <https://stackoverflow.com/questions/13485271/why-cant-we-use-c-standard-library-functions-in-kernel-development>