The Linux 5.13 development kernel is introducing a new "misc" cgroup controller. The misc cgroup controller is to be used for resources that are controlled by simply counting / limiting the number of resource instances in a scalar manner.

This cgroup controller can be used for purposes like tracking/limiting the number of enclaves on the system based upon hardware constraints. The initial user of this misc cgroup controller is for address space IDs used for virtual machine memory encryption.

One of the new security features in Linux 5.13 is the ability to randomize kernel stack offsets at each system call. This optional feature is now mainlined.

Randomizing the kernel stack offset per-system-call is intended to make it more challenging for rogue actors to carry out stack-based attacks on the Linux kernel. This has been in the works for over two years and was inspired by PaX's "RANDKSTACK" feature but the actual implementation has taken a different approach. Simply put though this randomizing of the kernel stack at each system call is to fend off exploits relying on kernel stack determinism.

The printk() function dates all the way back to the original Linux kernel release and even with
Linux turning thirty years old this week, work on printk is not over.

Back during Linux 5.10 last year a long overdue revamp of printk() began to land and now with Linux 5.13 more of that work is crossing the finish line. In 5.10 the ring-buffer became fully lock-less but still relying on a log buffer lock.

- Linux kernel vulnerability exposes stack memory, causes data leaks | ZDNet [5]

Disclosed by Cisco Talos researchers on Tuesday, the bug is described as an information disclosure vulnerability "that could allow an attacker to view Kernel stack memory."

The kernel is a key component of the open source Linux operating system. The vulnerability, tracked as CVE-2020-28588, was found in the proc/pid/syscall functionality of 32-bit ARM devices running the OS.

Source URL:  http://www.tuxmachines.org/node/150565

Links: