Date: Fri, 9 Oct 2020 12:20:38 +1100
From: Andrew Donnellan <ajd@...ux.ibm.com>
To: oss-security@...ts.openwall.com,
linuxppc-dev <linuxppc-dev@...ts.ozlabs.org>
Subject: Linux kernel: powerpc: RTAS calls can be used to compromise kernel integrity

The Linux kernel for powerpc has an issue with the Run-Time Abstraction Services (RTAS) interface, allowing root (or CAP_SYS_ADMIN users) in a VM to overwrite some parts of memory, including kernel memory.

This issue impacts guests running on top of PowerVM or KVM hypervisors (pseries platform), and does *not* impact bare-metal machines (powernv platform).

Description
===========

The RTAS interface, defined in the Power Architecture Platform Reference, provides various platform hardware services to operating systems running on PAPR platforms (e.g. the "pseries" platform in Linux, running in a LPAR/VM on PowerVM or KVM).

Some userspace daemons require access to certain RTAS calls for system maintenance and monitoring purposes.

The kernel exposes a syscall, sys_rtas, that allows root (or any user with CAP_SYS_ADMIN) to make arbitrary RTAS calls. For the RTAS calls which require a work area, it allocates a buffer (the "RMO buffer") and exposes the physical address in /proc so that the userspace tool can pass addresses within that buffer as an argument to the RTAS call.

The syscall doesn't check that the work area arguments to RTAS calls are within the RMO buffer, which makes it trivial to read and write to any guest physical address within the LPAR's Real Memory Area, including overwriting the guest kernel's text.

At the time the RTAS syscall interface was first developed, it was generally assumed that root had unlimited ability to modify system state, so this would not have been considered an integrity violation. However, with the advent of Secure Boot, Lockdown etc, root should not be able to arbitrarily modify the kernel text or read arbitrary kernel data.

Therefore, while this issue impacts all kernels since the RTAS interface was first implemented, we are only considering it a vulnerability for upstream kernels from 5.3 onwards, which is when the Lockdown LSM was merged. Lockdown was widely included in pre-5.3 distribution kernels, so distribution vendors should consider whether they need to backport the patch to their pre-5.3 distro trees.

(A CVE for this issue is pending; we requested one some time ago but it has not yet been assigned.)

Fixes
=====

A patch is currently in powerpc-next[0] and is expected to be included in mainline kernel 5.10. The patch has not yet been backported to upstream stable trees.

The approach taken by the patch is to maintain the existing RTAS interface, but restrict requests to the list of RTAS calls actually used by the librtas userspace library, and restrict work area pointer arguments to the region within the RMO buffer.

All RTAS-using applications that we are aware of are system management/monitoring tools, maintained by IBM, that use the librtas library. We don't anticipate there being any real world legitimate applications that require an RTAS call that isn't in the librtas list, however if such an application exists, the filtering can be disabled by a Kconfig option specified during kernel build.

Credit
======

Thanks to Daniel Axtens (IBM) for initial discovery of this issue.

[0]
https://git.kernel.org/pub/scm/linux/kernel/git/powerpc/linux.git/commit/?h=next&id=bd59380c5ba4147dcbaad3e582b55ccfd120b764

--
Andrew Donnellan OzLabs, ADL Canberra
ajd@...ux.ibm.com IBM Australia Limited