

CVE-2026-98228

Microsoft mips: select CONFIG_WEAK_REORDERING_BEYOND_LLSC from CONFIG_EYEQ

In the Linux kernel, the following vulnerability has been resolved: mips: select CONFIG_WEAK_REORDERING_BEYOND_LLSC from CONFIG_EYEQ. On I6500 CPU cores, lld and scd give no ordering guarantees (same as all other instructions). To respect the assumption that arch_cmpxchg() is fully ordered, we must inject sync instructions above and below our lld/scd loops using the already in place WEAK_REORDERING_BEYOND_LLSC infrastructure.

EVRS 12 /100 Elevated priority	CVSS V3.1 7.8 High · NVD	EPSS 0.001 1.7th percentile · FIRST	CISA DEADLINE — No CISA deadline
--	--	---	---

Signals behind the score

☒ counted in the score ☐ not counted

evidence for 3 of 8 signals

☐ CISA KEY	Not currently CISA KEY-listed	☐ Vendor exploitation	Active-exploitation status not yet checked - scored at the population base rate (none) until known
☐ Ransomware	Ransomware association unavailable	☐ KEY recency	CISA KEY listing date unavailable
☐ EPSS	0.001 probability	☐ Pre-patch disclosure	Disclosure status unavailable
☒ CVSS	Base score 7.8	☐ Microsoft Exploitability Index	Not a Microsoft CVE

Remediation

Scheduled Maintenance Patch or fixed version identified
Basis: External remediation intelligence

Otherwise, bad things can happen: [34.054496] CPU 3 Unable to handle kernel paging request at virtual address 0000000000000000, epc == a80000080838e01c, ra == a80000080838dfc4 [34.054559] Oops[#1]: [34.069561] CPU: 3 UID: 0 PID: 170 Comm: pipe_race Not tainted 7.2.0-rc6-01553-gb73c35220968-dirty #103 VOLUNTARY [34.079932] Hardware name: Mobile EyeQ5 MP5 Evaluation board [34.085592] \$ 0 : 0000000000000000 0000000000000001 0000000000000000 0000000000000000 [34.093616] \$ 4 : a800000808ee2618 000000000b7a879d 0000000000001000 0000000000000000 [34.101638] \$ 8 : 0000000000e3f2c9 0000000000000000 a800000808a2a9f8 0000000000000000 [34.109660] \$12 : a8000008139ffcd8 ffffffff84080018 a80000080837fae0 7878787878787878 [34.117682] \$16 : a800000807e82940 0000000000001000 0000000000000000 0000000000000000 [34.125704] \$20 : a800000802920e00 a8000008139ffdf8 a800000802649400 0000000000e3f2c9 [34.133726] \$24 : 0000000000000006 00000001200406e0 [34.141783] \$28 : a8000008139fc000 a8000008139ffd10 0000000000e3f2c8 a80000080838dfc4 [34.149837] epc : a80000080838e01c anon_pipe_read+0xd4/0x428 [34.155697] ra : a80000080838dfc4 anon_pipe_read+0x7c/0x428 [34.161549] Status: 140000e3 KX SX UX KERNEL EXL IE [34.166551] Cause : 40800408 (ExcCode 02) [34.170574] BadVA : 0000000000000000 [34.174161] PrId : 0001b028 (MIPS I6500) [34.178183] Process pipe_race (pid: 170, threadinfo=000000005ca35720, task=00000000e1013890, tls=000000014ebbb780) [34.188568] Stack : a800000802649400 0000000000000000 0000000000000000 a8000008139ffdd0 [34.196623] 00000000000000fba a800000808ee0000 0000000000000001 a8000008130c3e80 [34.204676] a8000008080d1280 a8000008139ffd58 a8000008139ffd58 1dbd2b22ea1dd500 [34.212729] a800000802649400 a800000808ee0000 ffffffffefeffea 0000000000000001 [34.220783] 0000000000001000 0000000000000000 00000001200ae518 ffffffffefeffef [34.228836] 000000fffbef0e530 a80000080837edf4 000000fffbef0e530 0000000000000000 [34.236890] 0000000000000000 0000000000000000 000000014ebb55a0 0000000000001000 [34.244943] 0000000000000001 a800000802649400 0000000000000000 0000000000000000 [34.252996] 0000000000000000 0000400400000000 0000000000000000 1dbd2b22ea1dd500 [34.261049] 00000000140000e3 a800000802649400 a800000802649400 a800000808ee0000 [34.269103] ... [34.271568] Call Trace: [34.274026] [<a80000080838e01c>] anon_pipe_read+0xd4/0x428 [34.279533]

[<a80000080837edf4>] vfs_read+0x25c/0x318 [34.284607] [<a80000080837faac>] ksys_read+0x104/0x138 [34.289763]
[<a80000080802b9cc>] syscall_common+0x44/0x68 [34.295187] [34.296689] Code: f84000cf 02209825 de020010
<dc420000> d8400004 02002825 0040f809 02802025 f84000c3 [34.306504] [34.308099] ---[end trace
0000000000000000]--- My initial reproducer was the xdp-tools test suite. A standalone reproducer would be an lld/scd loop
that, when the read is reordered by the CPU, triggers a fault. We can achieve this from userspace by stressing an anonymous
pipe, which uses a mutex. Program used: // SPDX-License-Identifier: GPL-2.0 // pipe_race.c - reproducer for MIPS LL/SC
reordering vs fs/pipe.c // // Two userspace processes on an anonymous pipe: // parent = writer: tight write() loop // child =
reader: tight read() loop #define _GNU_SOURCE #include <assert.h> #include <errno.h> #include <sched.h> #include
<signal.h> #include <stdio.h> #include <stdlib.h> #include <string.h> #include <sys/types.h> #include ---truncated---

Affected and fixed versions

PACKAGE	AFFECTED	FIXED IN	SOURCE
Not identified by the connected sources	–	no fixed version listed	NVD

Sources

Vendor guidance	git.kernel.org/stable/c/3f017c2ca8c31a1e135dd87be647096866d6b196	Additional guidance	git.kernel.org/stable/c/bbc448c541ed90d52f9ec22f17394304e3bf170e
NVD	nvd.nist.gov/vuln/detail/CVE-2026-98228	ENISA EUVD	euvd.enisa.europa.eu/vulnerability/EUVD-2026-93156
Current data	evulnable.com/vulnerability-management/cve/CVE-2026-98228		

Snapshot at generation time. EPSS can move quickly. A public exploit module is not evidence of active exploitation.