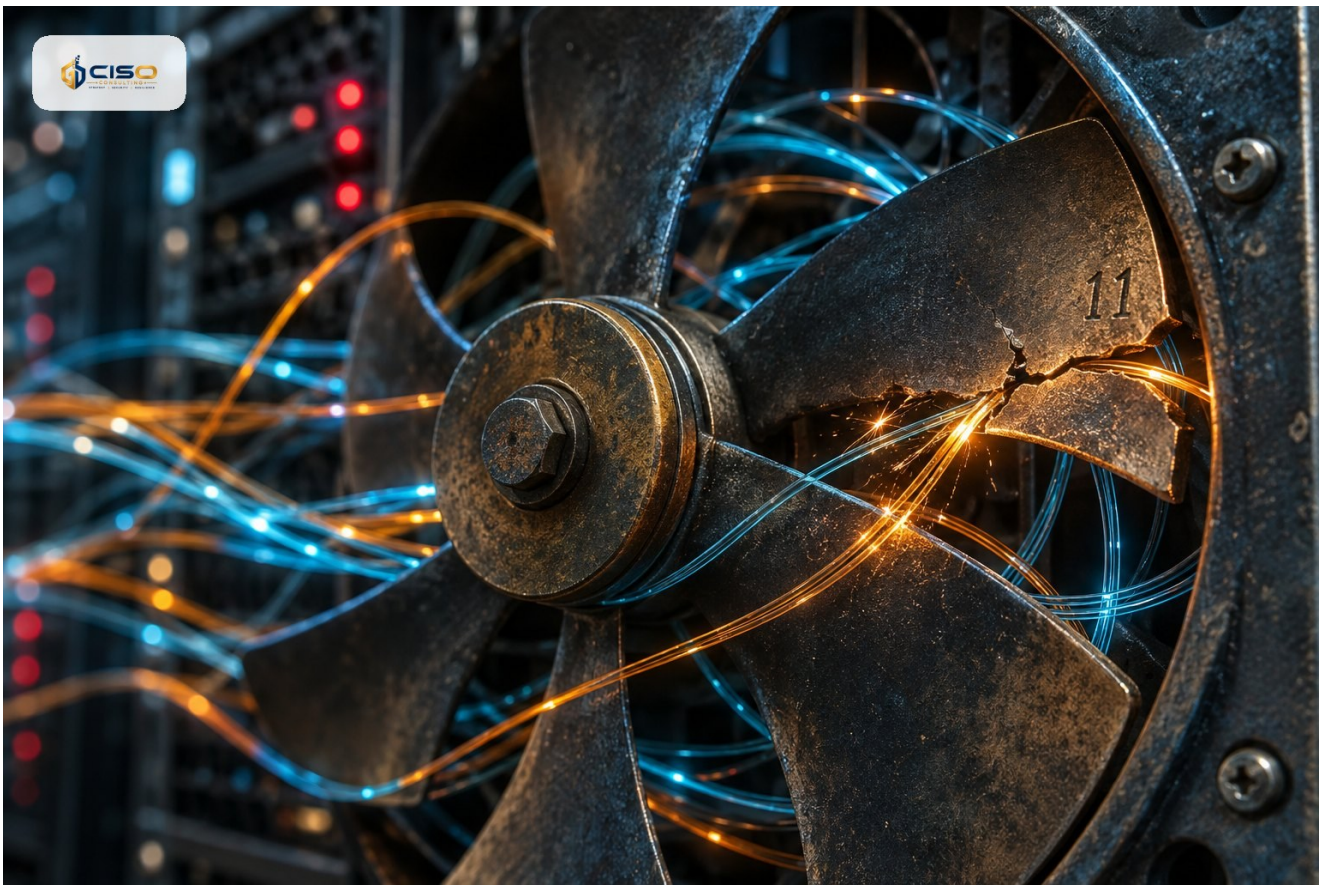
NoisyTCP Linux kernel flaw (CVE-2026-99288) can crash servers remotely



An 11-year-old Linux kernel flaw, NoisyTCP (CVE-2026-99288), lets unauthenticated attackers crash servers with crafted TCP packets. A patch exists; interim mitigations carry trade-offs.

- CVE-2026-99288 is a NULL pointer dereference in the Linux kernel TCP subsystem, affecting almost all distributions.
- It is exploitable remotely without authentication and causes a kernel panic, taking down the host.
- The source says it does not allow data alteration or arbitrary code execution.
- The race window is narrow, so exploiting a quiet bare-metal server can take a long time.
- A fix was merged into the mainline kernel two weeks before publication; interim options are disabling SYN cookies or using a single RX queue.

What was reported

A Linux kernel vulnerability named **NoisyTCP**, tracked as CVE-2026-99288, has been disclosed by researcher Hyunwoo Kim. It has existed for 11 years and affects almost all Linux distributions. Systems that receive TCP packets directly and run Nginx, Apache or an SSH server in default configurations can be crashed.

The flaw is a NULL pointer dereference in the kernel's TCP networking subsystem. It arises from a race between SYN cookie handling and socket hashing. An unauthenticated remote attacker can send crafted TCP packets to trigger a kernel panic and bring down the host.

Impact and limits

The impact is availability only. According to the source, exploitation does not let attackers alter data or execute arbitrary code. The race window is narrow, so success against a quiet bare-metal server may take a long time. Triggering it requires packet processing to be spread across contexts, which applies to servers with two or more RX queues or where RPS distributes packets across several CPUs.

Remediation

A patch is available, and the fix was merged into the mainline Linux kernel two weeks before the report. Where patching or upgrading cannot happen immediately, the researcher suggests disabling SYN cookies, which reduces protection against SYN flood attacks, and serializing traffic to a single RX queue to prevent the race.

For regulated institutions, the main concern is the availability of internet-facing Linux services. Prioritise exposed hosts and weigh the mitigation trade-offs before applying them.

What to do now

1. Inventory Linux servers that receive TCP traffic directly, especially those running Nginx, Apache or SSH.
2. Obtain the kernel update from your distribution vendor and schedule patching, prioritising internet-facing hosts.
3. If patching must wait, assess the interim options: disabling SYN cookies or using a single RX queue, and record the reduced SYN flood protection as an accepted risk.
4. Review high-availability and failover for critical services so a host crash does not cause an outage.
5. Monitor for unexplained kernel panics or reboots on exposed servers and escalate them through incident handling.