Advisory - Kerberos (krb5) Two Unauthenticated Network Vulnerabilities: CVE-2026-40355 & CVE-2026-40356

Apr 27, 2026

Management Summary

Two unauthenticated network vulnerabilities were identified in MIT Kerberos (krb5). A crafted packet could cause a null pointer dereference and terminate the process, and a separate crafted packet could trigger a read overrun of up to 52 bytes. The issues were fixed upstream in commit `2e75f0d`.

Vendor / Project Description

MIT Kerberos (krb5) is an open-source network authentication protocol and software implementation originally developed at MIT. According to the official MIT Kerberos FAQ, Kerberos is built into all major operating systems, serves as the authentication mechanism for Microsoft Active Directory, and is conservatively estimated to be used by well over 100 million people worldwide. The FAQ also describes Kerberos as one of the most widely adopted authentication technologies in computer networking, and states that one organization alone uses the MIT Kerberos software for over 50 million unique logons per month.

The project is also integrated into Google's OSS-Fuzz continuous fuzzing service. The public krb5 OSS-Fuzz project directory and its project configuration show fuzzing coverage with libFuzzer, AFL, and Honggfuzz, together with address, memory, and undefined-behavior sanitizers.

Technical Details

The vulnerable code is in:

`src/lib/gssapi/spnego/negoex_util.c`

CVE-2026-40355: Null pointer dereference

In `parse_nego_message()`, the result of the second `vector_base()` call was not checked before being dereferenced.

Impact: an unauthenticated remote attacker can trigger a null pointer dereference and cause the process to terminate.

CVE-2026-40356: Read overrun

In `parse_message()`, a short `header_len` could cause an integer underflow while calculating the remaining message length.

Impact: an unauthenticated remote attacker can trigger a read overrun of up to 52 bytes, possibly causing the process to terminate. According to the upstream commit message, exfiltration of the bytes read does not appear possible.

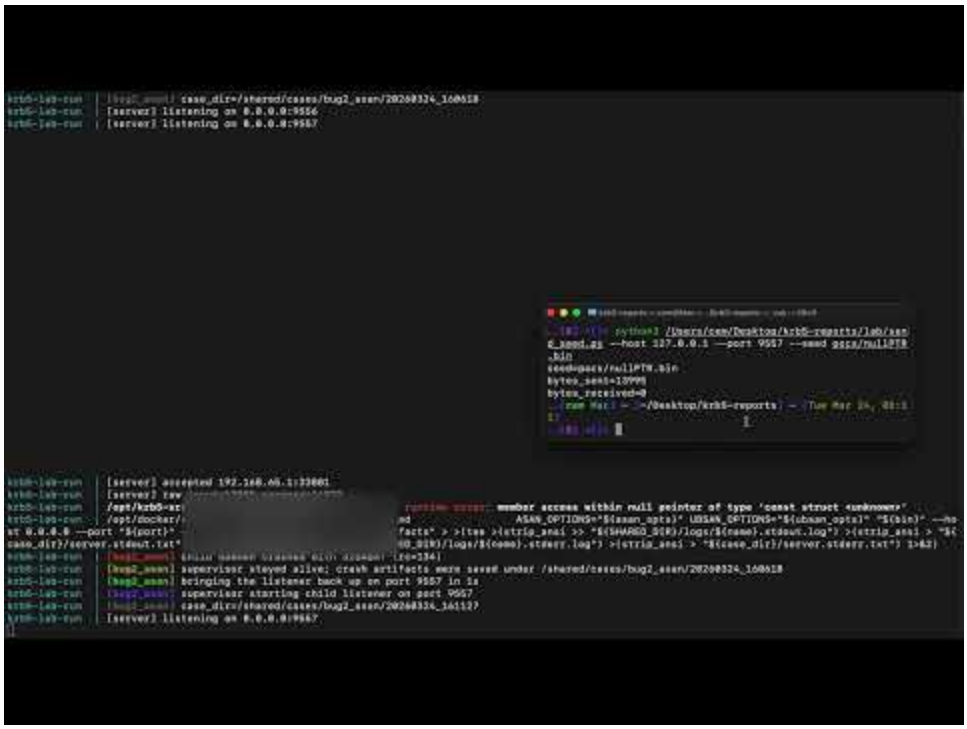
Fix

Apply the upstream patch or update to a version containing the fix.

Proof of Concept

The following video demonstrates the issues in a Dockerized krb5 lab environment compiled with AddressSanitizer (ASan), showing memory-safety failures triggered over the network:

In the PoC video, the service is supervised so that a child listener is started again after the server crashes.



Credit

Reported by Cem Onat Karagun.

Cem's Blog

Cem's Blog

cemonatk

cemkaragun

cemonatk

Mostly about security