[security] Go 1.26.2 and Go 1.25.9 are released

anno...@golang.org
to golan...@googlegroups.com

Apr 7,

Hello gophers,

We have just released Go versions 1.26.2 and 1.25.9, minor point releases.

These releases include 10 security fixes following the security policy:

- os: Root.Chmod can follow symlinks out of the root on Linux

  On Linux, if the target of Root.Chmod is replaced with a symlink while the chmod operation is in progress, Chmod could operate on the target of the symlink, even when the target lies outside the root.

  The Linux fchmodat syscall silently ignores the AT_SYMLINK_NOFOLLOW flag, which Root.Chmod uses to avoid symlink traversal. Root.Chmod checks its target before acting and returns an error if the target is a symlink lying outside the root, so the impact is limited to cases where the target is replaced with a symlink between the check and operation.

  On Linux, Root.Chmod now uses the fchmodat2 syscall when available, and an workaround using /proc/self/fd otherwise.

  Thanks to Uuganbayar Lkhamsuren for reporting this issue.

  This is CVE-2026-32282 and Go issue https://go.dev/issue/78293.

- html/template: JS template literal context incorrectly tracked

  Context was not properly tracked across template branches for JS template literals, leading to possibly incorrect escaping of content when branches were used.

  Additionally template actions within JS template literals did not properly track the brace depth, leading to incorrect escaping being applied.

  These issues could cause actions within JS template literals to be incorrectly or improperly escaped, leading to XSS vulnerabilities.

  This only affects templates that use template actions within JS template literals.

  This is CVE-2026-32289 and Go issue https://go.dev/issue/78331.

- crypto/x509: excluded DNS constraints not properly applied to wildcard domains

  When verifying a certificate chain containing excluded DNS constraints, these constraints are not correctly applied to wildcard DNS SANs which use a different case than the constraint.

  For example, if a certificate contains the DNS name "*.example.com" and the excluded DNS name "EXAMPLE.COM", the constraint will not be applied.

  This only affects validation of otherwise trusted certificate chains, issued by a root CA in the VerifyOptions.Roots CertPool, or in the system certificate pool.

  This issue only affects Go 1.26.

  Thank you to Riyas from Saintgits College of Engineering, k1rnt, @1seal for reporting this issue.

  This is CVE-2026-33810 and Go issue https://go.dev/issue/78332.

- cmd/compile: no-op interface conversion bypasses overlap checking

  Previously, the compiler failed to unwrap pointers contained within a no-op interface conversion leading to an incorrect determination of a non-overlapping move.

  To prevent unsafe move operations, the compiler will now unwrap all such conversions before considering a move non-overlapping.

  Thank you to Jakub Ciolek - https://ciolek.dev/ for reporting this issue.

  This is CVE-2026-27144 and Go issue https://go.dev/issue/78371.

- cmd/compile: possible memory corruption after bound check elimination

  Previously, slices and arrays accessed using induction variables were sometimes incorrectly proved in-bound. If the induction variable used for indexing were to overflow or underflow, it could allow access to memory beyond the scope of the original slice or array.

Thank you to Jakub Ciolek - https://ciolek.dev/ for reporting this issue.

This is CVE-2026-27143 and Go issue https://go.dev/issue/78333.

- archive/tar: unbounded allocation when parsing old format GNU sparse map

  tar.Reader could allocate an unbounded amount of memory when reading
  a maliciously-crafted archive containing a large number of sparse
  regions encoded in the "old GNU sparse map" format.

  We now limit both the number of old GNU sparse map extension blocks,
  and the total number of sparse file entries, regardless of encoding.

  Thanks to Colin Walters (wal...@verbum.org) who initially reported this issue.
  Thanks also to Uuganbayar Lkhamsuren (https://github.com/uug4na) and Jakub Ciolek
  who additionally reported this issue.

  This is CVE-2026-32288 and Go issue https://go.dev/issue/78301.

- crypto/tls: multiple key update handshake messages can cause connection to deadlock

  If one side of the TLS connection sends multiple key update messages
  post-handshake in a single record, the connection can deadlock, causing
  uncontrolled consumption of resources. This can lead to a denial of service.