



## Bug 2435258 (CVE-2026-1584) - CVE-2026-1584 gnutls: gnutls: Remote Denial of Service via crafted ClientHello with invalid PSK binder

**Keywords:**

**Reported:** 2026-01-29 12:21 UTC by OSIDB Bzimport

**Status:** NEW

**Modified:** 2026-02-09 14:34 UTC ([History](#))

**Alias:** CVE-2026-1584

**CC List:** 6 users ([show](#))

**Deadline:** 2026-03-12

**Product:** Security Response

**Fixed In Version:**

**Clone Of:**

**Environment:**

**Component:** vulnerability

**Last Closed:**

**Embargoed:**

**Version:** unspecified

**Hardware:** All

**OS:** Linux

**Priority:** high

**Severity:** high

**Target Milestone:** ---

**Assignee:** Product Security DevOps Team

**QA Contact:**

**Docs Contact:**

**URL:**

**Whiteboard:**

**Depends On:** [2437988](#) [2437989](#)

**Blocks:**

**TreeView+** [depends on](#) / [blocked](#)

**Attachments** ([Terms of Use](#))

OSIDB Bzimport 2026-01-29 12:21:16 UTC

[Description](#)

Summarysummary

A malicious TLS client can trigger a NULL pointer dereference on the server by sending a crafted ClientHello message with an invalid PSK binder value. This leads to a server crash and constitutes a remote Denial-of-Service condition.

Technical Detailstechnical-details

The issue occurs during PSK binder verification in the server-side code path.

In `pre_shared_key.c`, when the server receives a `pre_shared_key` extension, the function `_gnutls_psk_recv_params()` is invoked.

Under certain conditions, the following logic is executed:

```
pskcred = (gnutls_psk_server_credentials_t)
_gnutls_get_cred(session, GNUTLS_CRD_PSK); if (pskcred ==
NULL && (session->internals.flags & GNUTLS_NO_TICKETS))
return 0; return server_recv_params(session, data, len,
pskcred);
```

When the server issues a NewSessionTicket and the client later sends a ClientHello using the ticket identity from that message, `_gnutls_get_cred()` returns NULL. However, in this scenario the conditional check above does not return early, and `pskcred` (which is NULL) is passed to `server_recv_params()`.

Inside `server_recv_params()`, the PSK binder value is verified. If the received binder size matches the PRF MAC length but the binder value itself is incorrect, the following code path is taken:

```
if (_gnutls_mac_get_algo_len(prf) != binder_recvd.size ||
gnutls_memcmp(binder_value, binder_recvd.data,
binder_recvd.size)) { if (pskcred->binder_algo == NULL &&
mac == GNUTLS_MAC_SHA384) { mac = GNUTLS_MAC_SHA256;
_gnutls_free_key_datum(&key); goto retry_binder; }
gnutls_assert(); ret = GNUTLS_E_RECEIVED_ILLEGAL_PARAMETER;
goto fail; }
```

At this point, `pskcred` is NULL, and dereferencing `pskcred->binder_algo` results in a NULL pointer dereference and crashes the server.

#### Security Impactsecurity-impact

An unauthenticated remote client can reliably crash a gnuTLS-based TLS server by sending a malformed ClientHello with incorrect PSK binder values. This constitutes a remote Denial-of-Service vulnerability.

#### Proof of Conceptproof-of-concept

Due to ongoing research constraints, I am unable to publicly disclose the PoC at this time. However, I can provide a minimal Python-based PoC privately upon request to assist with verification and debugging.

#### Priority Argument Settingspriority-argument-settings

For completeness, the following priority string was used in my test environment: This priority configuration is specific to my research and experimental setup. However, the NULL pointer dereference does not depend on this particular priority string. In a more typical configuration, as long as the server issues a NewSessionTicket and a malicious client subsequently sends a ClientHello that references the ticket identity with an invalid PSK binder, the same NULL pointer dereference condition can still be triggered.

In other words, the issue is inherent to the server-side PSK binder handling logic and is not limited to this experimental priority configuration.

#### Note

You need to [log in](#) before you can comment on or make changes to this bug.

