Read Palworld crash and memory evidence before changing resources

A Palworld server disappearing does not prove it ran out of RAM. Application exceptions, container limits, host pressure, storage failures and a planned restart can all look like an abrupt disconnect to players. Collect the first useful evidence before changing resources or repeatedly starting the same failing process.

Preserve a short incident record

Write down the disconnect time and timezone, player count, recent update or configuration changes, and whether the panel showed a normal shutdown or an unexpected stop. Save output from before the failure and the next startup. The final line alone may omit the event that caused the failure several minutes earlier.

If logs include player identifiers, addresses or access tokens, keep the full copy private. Share a redacted excerpt with the relevant timestamp. Do not post complete configuration files publicly. Ask support for host or container evidence when it is not available in your account; customers do not need host-root access to report a useful incident.

Recognise the evidence boundaries

- **Application exception:** a crash message or call stack may identify an engine or mod failure. Start with the first error and recent changes.
- **Memory pressure:** steadily rising memory and a recorded limit event support a memory diagnosis. A displayed exit code of 137 alone is not proof of an out-of-memory kill; other forced terminations can produce similar symptoms.

- **Storage problem:** save-write errors, a full quota or insufficient temporary space can interrupt normal operation. Compare actual usage with the service limit.
- **Clean stop:** an orderly shutdown sequence suggests an action, schedule or controlled restart rather than an unexplained crash.

Choose one safe test

Preserve a stopped checkpoint before altering a live world. If failures began after an update or mod change, record the previous combination and ask whether an isolated reproduction is possible. Never delete the world, reset player files or disable all protections as an initial troubleshooting step.

For suspected memory pressure, compare idle usage after a completed load with usage during a repeatable gameplay session. Record the peak, limit and duration. Increasing configured base or creature limits can increase load; review recent changes rather than assuming every world requires the same RAM. A restart may temporarily reduce usage but does not establish the underlying cause.

Know when to stop retrying

If each start immediately fails at the same save-load step, stop the restart loop and keep the checkpoint intact. Repeated forced exits can add noise and risk while providing no new evidence. Give support the build, affected world, first error, timing, resource graph and last known working state.

After an agreed fix, test loading, joining, normal gameplay and a save cycle. Observe long enough to cover the original failure pattern, then check the next clean restart. Call the issue resolved only when the evidence matches the proposed cause and the server remains healthy under comparable use.

Related guides

- Prepare a reversible Palworld update
- Review Palworld settings safely

Further reading

- Pocketpair server requirements
- Pocketpair configuration parameters

Looking for another game? Browse game server setup and troubleshooting guides.

Revision #2
Created 2026-11-09 09:21:21 UTC by CloudExa Hosting Admin
Updated 2026-11-09 09:21:21 UTC by CloudExa Hosting Admin