Diagnose Satisfactory Server API errors and stalled joins

A Satisfactory service can appear in Server Manager yet fail when you join. The manager, gameplay traffic and reliable messaging do not all test the same network path. Start by identifying the last step that works, so a port problem does not lead to an unnecessary factory reset.

1. Save the exact symptom

Record whether the server cannot be added, reports a Server API failure, rejects a password, or remains on a loading screen after joining. Include the UTC time and whether every player is affected. Copy the displayed wording rather than describing all four cases as “offline”. Keep the service address and server ID together to avoid investigating another instance.

Check the game-panel console separately. An unavailable web console can be a panel connection fault; it does not establish that the Satisfactory process has crashed. Read a fresh startup log if one is available and note whether it names the expected session.

2. Compare the connection paths

Check the standard-port allocation and reliable TCP allocation against the startup configuration and current server log. Being visible to the manager does not prove the reliable channel is reachable. A TCP-only external checker cannot establish that UDP gameplay traffic works, either.

If the service recently moved, have support compare the public mapping with the advertised reliable port. Do not repeatedly increase client timeouts to hide a missing route. Use assigned ports; opening a broad range or changing the default port without updating the template can introduce another fault.

3. Separate version, credentials and session state

1. Compare client and dedicated-server versions and release branches. Record a mismatch before planning an update.
2. Confirm that the entered password is the required player or administrator password for that action.
3. Check whether the manager reports the intended factory as loaded, rather than merely showing a reachable server.
4. Try one other authorised player on a different connection to distinguish a local issue from an instance-wide failure.

Do not share an administrator password as a workaround for ordinary joining. If only one player fails, collect that player's exact error and platform details before changing a working server for everyone.

4. Investigate load or crash evidence

For a join that stalls while the factory loads, compare console progress, memory use and the last useful log entry. A large factory can take time; a repeated crash or unchanged error requires investigation. Avoid rapid stop/start attempts, which can interrupt the same load repeatedly and obscure the original sequence.

If a configuration change preceded the incident, compare it with your saved copy and reverse only that change through a controlled maintenance step. Do not clear the save directory, remove server identity files or reinstall to test whether the network works. Keep a recoverable checkpoint before any repair.

Confirm recovery

A useful result is a successful manager connection followed by an ordinary player joining the correct factory, moving normally and reconnecting. Send support the failing stage, timestamp, assigned ports, versions and redacted startup excerpt if the distinction remains unclear. Coffee Stain's support page points operators to its dedicated-server wiki and issue-reporting route; hosting-specific mapping still needs CloudExa investigation.

Related CloudExa guides

- Diagnose a panel console connection
- Keep a game backup

References

- Developer-recommended dedicated-server troubleshooting
- Satisfactory configuration reference
- Coffee Stain support and issue reporting

Revision #1
Created 2026-11-10 10:57:20 UTC by CloudExa Hosting Admin
Updated 2026-11-10 10:57:20 UTC by CloudExa Hosting Admin