Troubleshoot VPS DNS failures, blocked ports and timeouts

Start with one failing connection: which application, source machine, destination hostname or IP, port and protocol? A timeout can occur at several points. These checks help narrow the cause without replacing network configuration or disabling security rules.

Confirm the service and keep recovery access

Open the VPS panel and compare its assigned IP with the address you use. Check the power state and open the browser console if SSH fails. Keep an existing SSH session open while investigating. Do not reinstall the VPS to test connectivity.

> CloudExa VPS browser console example for checking a machine when SSH is unavailable
> Sanitised example of the VPS console. Your server and login details will differ. Select the image to enlarge it.

On Linux, inspect addresses with `ip -br address` and routes with `ip route`. For IPv6, inspect `ip -6 route`. Compare these with the assigned network details; do not guess a gateway or copy another customer's configuration.

Separate DNS from connectivity

Run `getent ahosts example.com` on the machine where the connection fails, replacing the example with the destination hostname. This checks that machine's configured name-resolution path. Check for old addresses, unexpected IPv6 results or an entry in `/etc/hosts`.

If the name does not resolve, capture the exact error and resolver configuration. Some systems use a local resolver service, and others use a generated `/etc/resolv.conf`; editing that file may not persist. Do not replace DNS settings before identifying what manages them.

Check whether the application listens

On the destination VPS, run `sudo ss -lntup`. Find the intended port and process. A TCP listener on `127.0.0.1` accepts local connections only. An application inside Docker also needs the correct container networking and published port. Change binding only when remote access is intended and authenticated.

If the process is absent, check its service status and application logs. Opening a firewall port cannot start a missing service. Avoid exposing a private database or management interface to everyone as a diagnostic shortcut.

Test the exact transport

Where OpenBSD netcat is installed, `nc -vz -w 5 example.com 443` performs a bounded TCP connection test. Replace the host and port with your own authorised destination. Success proves that a TCP connection opened, not that login or the application works. A refusal often means a closed listener or an explicit reject; a timeout needs further investigation.

This TCP test does not verify UDP games or voice services. Use the actual UDP application and its logs. Ping failure also does not prove a server is offline because networks may filter ICMP.

Review rules and send evidence

Check the guest firewall and any applicable panel or application rules for the precise source, destination, protocol and port. Keep SSH and recovery access intact. Do not flush firewall rules or disable the firewall to make a test pass.

Send support your service ID, timestamp with timezone, exact error and whether one or several sources fail. Redact secrets and unrelated addresses from logs. See VPS access recovery if you cannot reach the guest console.

Further reading: Debian networking; ss options; OpenBSD netcat.

Revision #2
Created 2026-11-08 21:03:24 UTC by CloudExa Hosting Admin
Updated 2026-11-08 21:03:24 UTC by CloudExa Hosting Admin