Troubleshoot IPsec tunnels

This guide helps you diagnose IPsec tunnel issues (also called connectors in the Cloudflare dashboard), from initial establishment through ongoing operation. Use the following sections to identify your symptom and find the appropriate solution.

Tunnel never establishes (IKE negotiation fails)

Symptoms

- Tunnel status shows `Down` and never becomes healthy
- No traffic passes through the tunnel
- Tunnel endpoint logs show IKE negotiation errors or retransmissions

Possible causes and solutions

Firewall blocking IKE traffic

Your edge firewall may be blocking the traffic required for IPsec tunnel establishment. Verify your firewall permits:

- UDP port `500` (IKE)
- UDP port `4500` (IKE NAT-T)
- IP protocol `50` (ESP)

Crypto parameter mismatch

IKE negotiation fails when Phase 1 (IKE) or Phase 2 (IPsec) parameters do not match between your tunnel endpoint and Cloudflare. Common symptoms include "no proposal chosen" errors in your device logs.

Verify your parameters match Cloudflare's supported values. For the complete list, refer to Supported configuration parameters.

Pre-shared key (PSK) mismatch

Authentication failures in Phase 1 indicate a PSK mismatch. To resolve:

1. Go to **Connectors** and select your tunnel.

   Go to Connectors ↗

2. Select **Generate new PSK**.
3. Copy the new PSK exactly — do not add extra spaces or characters.
4. Update your tunnel endpoint with the new PSK.

IKE ID format mismatch

Cloudflare uses FQDN format for the IKE ID. If your tunnel endpoint expects a different peer identity format (such as an IP address), authentication fails even when the PSK is correct.

Ensure your tunnel endpoint is configured to accept an FQDN peer identity. To find your tunnel's FQDN, go to **Connectors**, select your tunnel, and check the tunnel details.

Tunnel establishes but health checks fail

Symptoms

- IKE negotiation completes successfully
- Tunnel shows `Down` or `Degraded` in the dashboard
- User traffic may still pass through the tunnel

Possible causes and solutions

Anti-replay protection enabled on tunnel endpoint

This is the most common IPsec issue. Anti-replay protection expects packets to arrive in sequence from a single sender. Cloudflare's anycast architecture means tunnel traffic originates from thousands of servers, each with its own sequence counter. This causes your tunnel endpoint to drop packets as out-of-order.

Disable anti-replay protection on your tunnel endpoint, or set the replay window to `0`. For a detailed explanation, refer to Anti-replay protection.

Health check type incompatible with stateful firewall

Stateful firewalls (such as Palo Alto Networks, Check Point, Cisco, and Fortinet) drop the default *Reply* health check packets because no matching ICMP request exists in their session table.

Change the health check type from *Reply* to *Request*. For detailed steps, refer to Troubleshoot tunnel health.

ISP blocking health check return path

With unidirectional health checks, Cloudflare sends probes through the tunnel, but responses return via the public internet (direct server return). If your ISP blocks ICMP reply packets destined for Cloudflare, health checks fail even though tunnel traffic works normally.

If you have egress traffic enabled, consider switching to bidirectional health checks so that both the probe and response traverse the tunnel. For configuration details, refer to Troubleshoot tunnel health.

Policy-based VPN health check failures

If you use a policy-based VPN (where traffic selectors define specific prefixes rather than `0.0.0.0/0`), Reply-style health checks do not work. Reply health checks are self-addressed to Cloudflare IP addresses, which fall outside your tunnel's traffic selectors.

Use Request-style health checks instead. Configure a loopback address on your tunnel endpoint as the health check target. The target must be routable and covered by the tunnel's traffic selectors (encryption domain). For more details, refer to Troubleshoot tunnel health.

Tunnel works intermittently (flapping)

Symptoms

- Tunnel alternates between healthy and unhealthy states
- Intermittent packet loss on the tunnel
- Traffic works for a period then stops without configuration changes

Possible causes and solutions

Anti-replay protection dropping out-of-order packets

Cloudflare's anycast architecture means packets arrive from many servers with different sequence counters. Anti-replay protection interprets this as a replay attack and drops packets intermittently.

Disable anti-replay protection on your tunnel endpoint, or set the replay window to `0`. For a detailed explanation, refer to Anti-replay protection.

Rekey events causing brief disruption

When your tunnel endpoint initiates an IPsec rekey, new Security Associations (SAs) must propagate across Cloudflare's network. Rekey propagation delays have been significantly reduced and are uncommon in most deployments. However, brief tunnel degradation during rekeys can still occur in some configurations.

Cloudflare never initiates rekey — only responds. All rekey attempts must come from your tunnel endpoint. If your device receives a TEMPORARY_FAILURE response during rekey, configure Dead Peer Detection (DPD) with a "restart" action so the

device re-establishes the IKE session automatically. Without DPD restart, the device can get stuck in a loop of failed rekeys.

To minimize any impact from rekeys, increase SA lifetimes on your tunnel endpoint to reduce rekey frequency. Common values are 8-24 hours for IKE SA and 1-8 hours for IPsec SA. For more details, refer to Troubleshoot tunnel health.

MTU issues

Packets exceeding the tunnel MTU are fragmented or dropped, causing intermittent connectivity issues. Verify MTU is set correctly — typically `1476` for GRE tunnels and `1400` - `1450` for IPsec tunnels. For detailed guidance, refer to MTU and MSS.

Monitor with IPsec logs

Use IPsec logs to monitor tunnel activity during the key-exchange phase of the IPsec negotiation. Configure a Logpush job to forward these logs to your preferred storage service for analysis.

Set up an IPsec Logpush job

1. Go to the **Logpush** page.

   Go to Logpush ↗

2. Select **Create a Logpush job**.

3. Select **IPsec logs** as your dataset.

Refer to the Logpush documentation for more information about features, including the available fields in the dataset.