

Firewall rules and encryption (DSM)

Use this guide to diagnose connectivity problems between a device and DSM. Most connection failures trace back to blocked outbound traffic, DNS/NTP issues, or overly restrictive network filtering.

Quick Checklist

Before digging deeper, confirm the basics:

- ☐ Device has general outbound internet access
- ☐ Required ports below are open outbound
- ☐ DNS resolution is working
- ☐ Device clock is in sync (NTP)
- ☐ No IP/domain whitelisting is restricting traffic
- ☐ MDM-specific firewall rules are in place
- ☐ Any third-party apps on the device have their own required ports open

1. Check Core DSM Firewall Rules

DSM requires the following outbound rules. If any of these are blocked, the client will fail to connect.

Address	Protocol	Port	Purpose	Symptom if blocked
api2.kuori.com	TCP	443	HTTPS REST calls	Client can't reach API, requests time out or fail
api.kuori.com	TCP	443	WebSocket (WSS)	Client connects briefly but loses real-time/live connection

Address	Protocol	Port	Purpose	Symptom if blocked
DNS servers	UDP/TCP	53	Name resolution	Client can't resolve <code>kuori.com</code> hostnames at all
NTP servers	UDP	123	Time synchronization	TLS handshake fails or auth errors due to clock drift

How to check: Test outbound connectivity from the device/network to each address and port (e.g. `curl`, `telnet`, or a port-check tool). If DNS resolution fails, verify port 53 is open and a working DNS server is reachable. If TLS handshakes fail intermittently, check the device clock against NTP.

2. Check MDM-Specific and android rules

The MDM solution in use has its own required firewall openings, separate from the DSM rules above. If the MDM agent itself can't connect, DSM enrollment/management may also fail.

- **MDM firewall rules:** If you are using some other MDM than Kuori Mobile Management(KMM) you need to check their firewall rules, but for KMM they can be found from here: [KMM Firewal rules](#)

Also confirm outbound access is allowed per Android Enterprise requirements, since MDM device management relies on Google-managed services:

- [Android Enterprise Network Requirements](#)

3. Rule Out Overly Restrictive Network Filtering

A common root cause of intermittent or unexplained connection failures is IP-based whitelisting.

IP address-based whitelisting is **not a viable approach** for Android-based devices, and is a frequent cause of connectivity problems that "used to work" and then broke.

Why this happens:

- Devices depend on Google Play and other Google-managed infrastructure, which runs on large, dynamic, globally distributed cloud networks.
- These services' IP addresses change frequently and cannot be reliably predicted or restricted to a fixed range.
- If an allowlist of IPs was configured, it will inevitably go stale as addresses rotate — causing app updates, authentication, or background services to silently fail.

If you suspect this: Check whether any firewall/proxy rule is restricting outbound traffic to a specific IP range rather than by port/protocol. If so, replace it with rules based on the ports and domains above, and allow general outbound internet access rather than an IP allowlist.

4. Check General Network Configuration

- Outbound traffic should be allowed as per the DSM, MDM, and Android Enterprise requirements above.
- Outbound NTP requests must be allowed.
- Inbound traffic should be blocked except where explicitly required (DSM does not need inbound access).
- Devices should sit within a dedicated IoT VLAN or physically isolated network — confirm the device hasn't been placed on a segment with different (more restrictive) rules.
- If egress filtering is in place, check its logs for blocked outbound attempts from the device — this is often the fastest way to spot the exact rule causing the failure.

5. Check Encryption / TLS Requirements

- TLS 1.3 is required for all WSS and HTTPS connections. Older TLS versions or a network appliance forcing a TLS downgrade/inspection can break the handshake.
- DSM uses API key and device ID pairing for authentication — confirm both are correctly provisioned on the device; an invalid pairing can look like a "connection" failure even when the network is fine.

6. Check Third-Party Applications on the Device

If other applications on the device also seem to have network issues, they may have their own port/endpoint requirements that are separate from DSM's.

Example — TeamViewer:

Type	Port(s)
Outbound TCP	5938 (primary), 443, 80
Optional UDP	5938

TeamViewer (and similar apps) use dynamic cloud servers, so IP whitelisting will likely break them too. Evaluate each third-party application individually and allow its required outbound ports rather than restricting by IP.

Summary: Most Likely Causes, In Order

1. A required port (443, 53, or 123) is blocked outbound.
2. IP-based whitelisting is in place and has gone stale or is blocking dynamic Google/cloud IPs.
3. MDM-specific firewall rules haven't been applied (see internal KMM link above).
4. Device clock is out of sync (NTP blocked), breaking TLS handshakes.
5. Device is on the wrong network segment/VLAN with different rules applied.