

Alpha soak checklist (LAN demo)

From ALPHA_SOAK.md

Table of contents

[Before you start](#)

[Happy path \(~15 min\)](#)

[Stress \(optional, ~10 min\)](#)

[Pass criteria](#)

[If something fails](#)

[Not in alpha soak scope](#)

Source: ALPHA_SOAK.md — section links work in PDF viewers that support internal anchors.

Manual QA on **two devices** on the same Wi-Fi. Goal: stable show path without experimental transports or stream protection.

Before you start

Setting	Sender	Receiver
Transport	UDP	UDP (must match)
Stream protection	None	None
Video codec	H.264 (AUTO)	H.264 (AUTO)
Resolution (sender)	720p or Auto for first pass	—
Multi-receiver	One receiver selected	—

PIN default: 1234 (change in settings if needed).

Happy path (~15 min)

1. Install the same debug/release APK on both devices.
2. **Receiver:** Main → **Receive** → **Start listening** (notification shows *Listening*).
3. **Sender:** Main → **Send** → refresh discovery → pick receiver → enter PIN → approve screen capture.
4. Confirm sender notification: *Casting via UDP · WxH*.
5. Confirm video and audio on receiver for ≥ **5 minutes**.
6. Rotate sender once; confirm stream recovers or stop/restart cleanly.
7. Press **Stop** on sender notification; receiver returns to awaiting state.

Stress (optional, ~10 min)

- Background sender app for 30s, return — stream continues or fails with clear status (no silent hang).
- Lock/unlock receiver once during cast.
- Second cast session without reinstall (stop → start again).

Pass criteria

- No crash on either device during happy path.
- Video appears within **10s** of capture consent.
- Stop ends services and notifications on both sides.

If something fails

Symptom	Check
No video	Transport match; receiver listening first; open receiver UI once for surface attach
Auth failed	PIN; same Wi-Fi; firewall on LAN
Screen capture data missing	Reinstall APK; Android 13+ consent intent
Choppy only on UDP	Retry with TCP on both sides (fallback test)

Not in alpha soak scope

- QUIC / WebRTC / USB-tether transports
- FEC / NACK stream protection
- 1:N multi-receiver (unless explicitly testing)
- OTA / crash upload (backend track)