

Android Cast — alpha release & QA

From ALPHA.md

Field	Value
Author	Anton Afanasyeu
Revision	R1
Creation date	2026-05-24
Last modification date	2026-05-24
Co-authored	
Severity	medium
State	scheduled
Document type	technical

Table of contents

Alpha scope (what we ship)
Recommended defaults (both devices)
Feature freeze checklist (engineering)
Alpha app sign-off (manual QA)
 A. Smoke (required — all must pass)
 B. Resilience (required)
 C. Settings matrix (pick one row for official alpha demo)
 D. Regression guards (quick)
 E. Known limitations (not alpha blockers)
Log capture (when something fails)
Alpha backend (optional for LAN-only alpha)
 F. Remote access — RSSH (required for full alpha sign-off)
 G. Email + registration / 2FA (required for full alpha sign-off)
Ending feature freeze

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

Alpha scope (what we ship)

In scope	Out of scope (frozen)
LAN discover → PIN → cast → stop	Internet relay, TLS on stream
UDP or TCP transport (must match on both devices)	QUIC, WebRTC, USB-tether transport in UI
H.264 + AAC (Android 10+ audio)	Passthrough video, Opus/Speex as default
Single receiver per cast session	1:N multi-receiver (disabled in CastConfig)
Stream protection None (default)	FEC/NACK soak required for alpha sign-off
Screen capture (full / user-choice on API 34+)	Secondary-display capture, Miracast in-app
Camera cast (API 29+, best-effort)	Live sender mirror while casting
Receiver playback + notification stop	Play Store / AdMob production
On-demand remote debug (RSSH, hidden) — developer settings + BE	WireGuard-only remote access as alpha deliverable

Code gate: CastConfig.ALPHA_FEATURE_FREEZE = true hides experimental transports and turns off multi-receiver selection.

Recommended defaults (both devices)

Set in **Settings** (or use fresh install defaults):

Setting	Sender	Receiver
Transport	UDP	UDP
Stream protection	None	None
Quality	Medium	Medium
Resolution	Adaptive (or HD 720p if rotation is shaky)	—
Video codec	Auto	Auto
Audio	On (if API 29+)	Play incoming audio: On
Capture mode	Full screen (or User choice on Android 14+)	—
PIN	Same on both (default 1234)	Same

Rotation soak: On Mediatek / Freeme senders (e.g. BL6000), prefer **system rotation lock ON** during first alpha pass; then repeat with lock **OFF** to validate debounced reconfigure.

Feature freeze checklist (engineering)

Item	Status
ALPHA_FEATURE_FREEZE — UDP/TCP only in settings	Done
Multi-receiver disabled for alpha	Done
Stored QUIC/WebRTC transport clamped to UDP on load	Done
Camera strings (permission vs unavailable)	Done
README matches CastConfig / CastSettings defaults	Done
Receiver: keyframe idle grace + debounced sender resize	Done (verify on soak)
Preview paused during cast (single MediaProjection)	Done
Developer-only: FEC, QUIC, USB-tether, immersive presets	Gated (dev settings)

Alpha app sign-off (manual QA)

Run on **two physical devices**, same Wi-Fi subnet. Record build: About / versionName from APK, git commit, date.

A. Smoke (required — all must pass)

#	Step	Pass
A1	Install same debug/release APK on sender + receiver	■
A2	Receiver: Receive → Start listening → notification *Listening* / *Ready*	■
A3	Sender: Send → discover receiver → correct PIN → approve capture	■

A4	Video appears on receiver within 15 s of auth	■
A5	Audio audible (if enabled and API 29+) within 20 s	■
A6	Sender notification shows *Casting* with resolution	■
A7	Stop from sender notification → stream ends, receiver returns to idle	■
A8	Stop from receiver notification → sender stops cleanly	■
A9	Second cast without reinstall (A2→A7 again)	■

B. Resilience (required)

#	Step	Pass
B1	Sender home → cast 5 min, no reboot / ANR	■
B2	Receiver app to background 2 min → return → video recovers or clean idle	■
B3	Receiver rotate once → video continues or recovers within 10 s	■
B4	Sender rotate with rotation lock ON → 5 min stable	■
B5	Sender rotate with lock OFF (optional / device-specific) → no permanent freeze; note if reboot	■

C. Settings matrix (pick one row for official alpha demo)

Profile	Transport	Protection	Use
Alpha default	UDP	None	Primary sign-off
Fallback	TCP	None	If UDP glitches on your AP

#	Step	Pass
C1	UDP + None: 10 min soak	■
C2	(Optional) TCP + None: 10 min soak	■

D. Regression guards (quick)

#	Step	Pass
D1	Wrong PIN → clear error, no crash	■
D2	Mismatched transport (sender TCP / receiver UDP) → fails fast with message	■
D3	Camera mode (if used): permission grant → cast 2 min	■
D4	Calibration test mode → pattern on receiver	■

E. Known limitations (not alpha blockers)

- No TLS; LAN cleartext.
- UDP may drop frames under congestion.
- No live preview on sender while casting.
- Some apps silent under playback capture policy.
- Heavy rotation on some OEMs may still stress system WFD/WM (watchdog) — document device + lock state.

LAN alpha approved when: all **A**, **B1–B4**, and **C1** pass on your reference sender + receiver pair. **Full alpha** additionally requires backend deploy and **F1–F5** (RSSH).

Log capture (when something fails)

From repo root:

```
./scripts/alpha-qa-logcat.sh SENDER_SERIAL RECEIVER_SERIAL
```

Or manually:

```
adb -s SENDER logcat -s ScreenCastService:* ReceiverCastService:* AndroidCast:* SenderScreenPreview:*
adb -s RECEIVER logcat -s ReceiverCastService:* AndroidCast:*
```

Alpha backend (optional for LAN-only alpha)

LAN cast demo does **not** require OTA/crash backend. For **full alpha** (owner agreement) with updates, crash upload, and **production remote debug**:

Item	Notes
------	-------

Host examples/ota/v0/	See OTA.md
Deploy crash reporter	See CRASH_REPORTER.md
Configure URLs	Developer settings or local.properties
RSSH remote access	Alpha essential: hidden mode (no VPN consent); device polls BE, opens outbound reverse SSH to bastion; operator connects via BE UI whitelist + session. WireGuard v1 is lab-only for alpha sign-off. See REMOTE_ACCESS_IMPL.md
Email + 2FA auth	Alpha essential: info@/admin@ forward to Gmail; user register + email verify + TOTP; see 20260607-2FA-email-mobile-auth-flow.md

F. Remote access — RSSH (required for full alpha sign-off)

#	Step	Pass
F1	Developer settings → RSSH (not WireGuard) → no Android VPN permission dialog	■
F2	Device polls BE (type: ra); operator whitelists device + opens session in dashboard	■
F3	Reverse tunnel active; operator reaches device debug endpoint on bastion (shell or agreed file path)	■
F4	Session expiry / disable tears down tunnel; BE notified	■
F5	No persistent VPN key icon while remote debug is active	■

Full alpha approved when: LAN sign-off (**A, B1–B4, C1**) and backend items above and **F1–F5** and **G1–G6** pass on reference environment.

G. Email + registration / 2FA (required for full alpha sign-off)

#	Step	Pass
G1	Inbound mail to info@apps.f0xx.org arrives at configured Gmail (forward)	■
G2	BE sends verification email (SMTP); link opens on mobile browser	■
G3	New user registers → verifies email → enrolls TOTP (QR)	■
G4	Login requires password + TOTP; lockout after repeated failures	■
G5	User can set recovery email; admin/root can clear auth lockouts	■
G6	Register/login UI matches crashes console theme (EN/RU, mobile layout)	■

Ending feature freeze

1. Complete sign-off table above; note devices + APK build id in git tag message.
 2. Tag release on master per GIT_FLOW.md.
 3. To re-enable experiments on a dev branch: set CastConfig.ALPHA_FEATURE_FREEZE = false and restore MULTI_RECEIVER_ENABLED as needed.
- Post-alpha priorities: rotation-stable encode path, encoder thumbnail on sender, secondary-display capture (roadmap D).