VPN lockdown mode breaks CLAT IPv4 connectivity on NAT64 IPv6-only networks in Android 13

+1 4 Hotlists (6) Mark as Duplicate

Comments (5) Dependencies Duplicates (0) Blocking (0) Resources (5)

Fixed Bug P2 + NeedsInfo

STATUS UPDATE No update yet. Edit

DESCRIPTION tm...@gmail.com created issue #1

IPv6 networks that rely on NAT64 / CLAT to allow access to IPv4 destinations will fail to initialize clatd when a VPN is configured as "Always on" and to "Block connections without VPN" (lockdow
Fi.

Prerequisites

- Install a VPN app, and enable the VPN app in "Always on" and "Block connections without VPN" (lockdown). It *might* be necessary for the VPN to be accessible over IPv4 as opposed to IPv6
- Have access to an IPv6 mobile or Wi-Fi network that utilizes NAT64 and does not provide direct IPv4 connectivity, therefore kicking clatd / Clat / Nat464Xlat into gear. (Example: T-Mobile-ba

Steps to reproduce

1. Start out in Airplane Mode for clean reproduction.
2. Ensure the VPN app is enabled as specified and ready to connect.
3. (Debugging) Clear and start capturing logs.
4. Connect to the NAT64 network described, disabling Airplane Mode if necessary.

Expected result

The VPN connects successfully. clatd starts successfully.

Actual result

There is no connectivity.

Logs

clatd fails to start with errors in log such as:

- `E Nat464Xlat: Error starting clatd on wlan0: java.io.IOException: no IPv6 addresses were available for clat: java.io.IOException: Unable to find global sou`
- `D ConnectivityService: Returning BLOCKED NetworkInfo to uid=1000`

Please see attached log snippets for further details. `lockdown-enabled.txt` demonstrates the problem, `lockdown-disabled.txt` demonstrates success (with VPN lockdown disabled).

Device

- Model: Google Pixel 4a (5G)
- Build Number: TP1A.221005.003.22402030 (Android 13)

Further investigation

- Why is uid=1000 being blocked from accessing the network? Usually apps with uid < 10000 are not blocked.
- Why is clatd not able to succeed here?
  - https://cs.android.com/android/platform/superproject/+/android-13.0.0_r8:packages/modules/Connectivity/service/jni/com_android_server_connectivity_ClatCoordinator.cpp;l=116
  - Failing in generateIpv6Address: https://cs.android.com/android/platform/superproject/+/master:packages/modules/Connectivity/service/native/libs/libclat/clatutils.cpp;l=128

deleted
0 B

deleted
0 B

✓ Links (4)

Links (4)

" ...etwork that utilizes NAT64 and does not provide direct IPv4 connectivity, therefore kicking clatd / Clat / Nat464Xlat into gear. (Example: T-Mobile-based networks with their default IPv6-only APN.)
"https://cs.android.com/android/platform/superproject/+/android-13.0.0_r8:packages/modules/Connectivity/service/jni/com_android_server_co... "
"Failing in generateIpv6Address: https://cs.android.com/android/platform/superproject/+/master:packages/modules/Connectivity/service/native/libs/libclat/clatutils.cpp;l=... "
"I've provided a patch for review here: https://android-review.googlesource.com/c/platform/packages/modules/Connectivity/+/2267683"

COMMENTS

su...@google.com <su...@google.com>

Assigned to su...@google.com.

su...@google.com <su...@google.com> #2

Please provide the following additional information:

Android full bug report capturing

After reproducing the issue, press the volume up, volume down, and power button simultaneously. This will capture a bug report on your device in the “bug reports” directory.

Alternate method

Navigate to “Developer options”, ensure “USB debugging” is enabled, then enable “Bug report shortcut”. Capture bug report by holding the power button and selecting the “Take bug report” op

Note: Please upload the files to google drive and share the folder to android-bugreport@google.com, then share the link here.

tm...@gmail.com <tm...@gmail.com> #3

I think the logs that I attached are enough. (-:

I've provided a patch for review here: https://android-review.googlesource.com/c/platform/packages/modules/Connectivity/+/2267683

ma...@google.com <ma...@google.com> #4

Marked as fixed, reassigned to ma...@google.com.

The above patch - after fixing up 2 tests to account for the new argument - is now merged.

If all goes according to plan this will *hopefully* roll out as part of the (rollout at 100% near end of) January 2023 tethering mainline module release to all Android 13+ (T+) devices which ta

I think that should hopefully close this bug.

ma...@google.com <ma...@google.com> #5

(unfortunately various end of year holidays push things out by a month)