Unravelling ACR on Samsung TVs

May 30, 2026

Every frame a Samsung S90D shows, from any source, is fingerprinted and sent to Samsung about once a minute, on a schedule Samsung can change remotely. The evidence is in the firmware.

Status: finished

TL;DR

- The S90D runs **Automatic Content Recognition (ACR)**: it samples the picture ~2× per second and uploads perceptual **fingerprints** to Samsung's cloud about **every 60 seconds**.
- Capture happens at the **display compositor inside TrustZone**, *before* HDCP, so it is **source-agnostic** and captures **DRM-protected streaming** (Netflix included) in the clear.
- Confirmed in-scope sources: **Samsung TV+**, **third-party streaming apps**, **HDMI inputs**, **antenna/broadcast (tuner)**; screen mirroring has its own recognition path too. The one exemption: **no fingerprinting at power-on**. The upload engine is `on-boot="false"` and wakes when a source becomes active.
- Fingerprints upload to **`acr-us-prd.samsungcloud.tv`**, an address the TV is **handed at runtime** by the control plane **`osb-v2.samsungqbe.com/service/acr/v2`**, so Samsung controls *where* the data goes, not just what's collected.
- The query **interval is server-controlled**. The binary literally logs `Query Int from server(60 is set)`. Samsung can dial the cadence up or down without a firmware update.
- Fingerprints are uploaded **whether or not the server recognizes the content**. Samsung receives the data either way.
- The same process holds your **Samsung Account identity**, **ad PSID**, **and ad targeting group**, so viewing data can be joined to who you are.

ACR ain't anything new

ACR turns your TV into a measurement device. It periodically samples what's on screen, reduces each frame to a compact **fingerprint**, and sends those fingerprints to a cloud service that matches them against a reference database to identify the show, channel, or ad. The result feeds viewing analytics and ad targeting.

This isn't hypothetical. In 2017 the FTC and New Jersey settled with **Vizio** for **$2.2M** over ACR that tracked viewing without clear consent. The question for any given TV is no longer *whether* it has ACR, but **exactly what it captures, where it sends it, how often, and how it's tied to your identity**. This post answers those four questions for the Samsung S90D with firmware-level evidence.

1. Every source is in scope

The capture itself is **source-agnostic**: the S90D grabs the final composited image at the **display compositor** through a **TrustZone (TEE)** secure call (`TZCAPTURE_Capture_screen` in `libtzcapturec.so`), at **960×540 YUV**. Because this happens *after* rendering but *before* HDCP output encryption, it sees **whatever is on the panel**, regardless of source and regardless of DRM. Separate logic then *labels* what produced that image, and the firmware has explicit handling for each input type:

Source	In scope?	What the firmware shows
Samsung TV+ (first-party)	**Yes**	`org.tizen.tv-viewer` in the binary; live `source_APP_Id` of `org.tizen.tv-viewer(tvplus)` with a 7-stream match
Streaming apps (Netflix, Max, Peacock, YouTube…)	**Yes**	captured live during Netflix playback (labeled `ACR_SOURCE_EXT_1`); IACR ships per-app metadata (e.g. YouTube)
HDMI inputs (console, PC, cable box)	**Yes**	`IsHDMISource`, `IsHDMIPort`, `HDMIport : %d` `DeviceName : %s Device Type : %d`
Antenna / broadcast (tuner, STB)	**Yes**	`IsCUDSSTBAvailable`, `IsStateSTBContent`, tuner libraries
Screen mirroring / cast	**Likely**	dedicated `mirroring-recognition-manager` (boot-resident service)
At power-on (boot)	**No**	upload service manifest `on-boot="false"`; nothing is fingerprinted until a source is active

Table 1: The input sources the firmware handles, and the evidence for each.

It tracks **up to six concurrent external-source slots** (`ACR_SOURCE_EXT_1…6`). During testing, a third-party streaming app showed up as `ACR_SOURCE_EXT_1`, confirming these slots cover external apps and HDMI inputs alike.

2. Where does it phone home?

The `acr` binary doesn't fix the server it ultimately uploads to. It **asks a control server at runtime** and is handed the target. The binary contains the bootstrap endpoint and the machinery for this (`GetAcrInfo`, `GetControlServerAddress`, `MatchingServerURL` / `IAMatchingServerURL`, `m_RequestDataToControlServer`). On the live device, the hosts actually in use were:

Endpoint	Role	Where it was found
`acr-us-stg.internetat.tv/getacrinfo`	Bootstrap "GetAcrInfo" endpoint	**Hardcoded** in the `acr` binary
`osb-v2.samsungqbe.com/service/acr/v2`	**Control plane**: config, app list, server URLs, query intervals	Live process memory
`acr0.samsungcloudsolution.com`	Control/status server (`ControlServer`)	Live process memory
`acr-us-prd.samsungcloud.tv:443`	**Production fingerprint upload / matching** server	Live process memory (delivered as `IAMatchingServerURL`)

Table 2: The hosts involved in an upload, and where each was found.

The control-plane request identifies the exact device and turns the pipeline on:

```
https://osb-v2.samsungqbe.com/service/acr/v2
  ?countryCode=US&modelid=24_PTM_QD_T09
  &infolink=T-PTMDAKUC-0090-2123.9&internalacr=on
```

Because the upload target is **delivered by the control server rather than fixed in firmware**, Samsung can change *where* fingerprints go, just as it can change *how often* (next section). The `acr-us-*` / `countryCode=US` naming indicates US-region servers.

3. Frequency and Control Granularity

- The TEE captures frames frequently, **as often as ~every half-second** (the analyzer logs `Waiting for 480.333 ms` and `497.181 ms`; the interval varies).
- Fingerprints are bundled (**2–3+ frames**) and uploaded on a **main query interval of 60 s by default**.
- Two more pipelines run in parallel: an **ad query** (**~120 s**) and a **broadcast query** (**~300 s**).

Crucially, **these intervals are server-controlled**. The binary's own log line is `Query Int from server(60 is set)`, and the control-plane response carries `queryIntervalSec`, `adQueryIntervalSec`, and `minQueryCount`/`maxQueryCount`. Samsung can change how aggressively your TV fingerprints **remotely, without a firmware update**.

4. Anatomy of a query

Each frame is reduced to **4 Regions of Interest**, each a set of four 32-bit integers (512 bits/frame), and every frame is stamped with **millisecond UTC plus your timezone offset**:

```json
{ "fps": [ {"raws": [u32,u32,u32,u32], "roi": 0}, … {"roi": 3} ],
  "tvt": {"utc": "1779224946340", "tz": -300} }
```

Two outcomes, both observed live:

- **Netflix (Arrested Development)**: every frame came back `matching:false`. Samsung's database didn't recognize it. **The fingerprints were uploaded anyway**. The engine does *not* stop on unrecognized content.
- **Samsung TV+ (live news)**: a single query matched **seven streams at once**, each with channel, **program title, genre, and a `contentOffset`**, e.g. an offset of ~68m55s, which combined with the program's start time reveals *exactly* when you tuned in and how long you'd been watching.

So matching is a *server-side* step. Whether or not your content is in Samsung's database, **Samsung receives the fingerprints**.

5. It captures DRM-protected content

Because capture is **pre-HDCP, at the compositor, inside TrustZone**, content protection doesn't stop it. The live Netflix fingerprints are the proof: a **Widevine-DRM** stream was sampled and uploaded in cleartext YUV, invisible to the Netflix app. ACR sees what the panel shows, not what the app "allows."

6. Tied to who you are

ACR isn't anonymous telemetry. The recognition process reads, and holds in memory alongside the upload logic:

- your **ad identifier** (`db/adagent/psid`) and **ad targeting group** (`db/comss/targetgrouptag`),
- your **Samsung Account** state and a live **SSO token** (the same token also appears in the TV+ daemon, it's reused across subsystems),
- your **country, device ID, timezone, and entered ZIP code**.

That means a viewing fingerprint can be joined to your account and your advertising profile. The viewing record and the identity that makes it *targetable* live in the same place.

7. Consent is one checkbox

All of this is gated behind a **single binary disclaimer flag** (`HOMEDATACONTROL_DISCLAIMERAGREES`). There's no separate control for the three pipelines (content / ad / broadcast). You can't, for example, allow program-recognition while refusing ad-break measurement. It's all-or-nothing, and the *availability* of ACR per app is decided by the server/scenario logic, not a per-app user setting.

8. Why it matters

- **Scope vs. expectation**. Most people expect ACR (if they know it exists) to watch broadcast TV. Here it watches *everything on the panel*, including third-party DRM streaming and HDMI devices.
- **Remote-tunable surveillance.** A server-controlled cadence means the data-collection rate isn't a fixed, auditable property of the device.
- **Regulatory exposure**. Continuous, identity-linked viewing capture with a single coarse consent raises questions under **GDPR** (specific, informed

consent; right to object) and **CCPA/CPRA** (sharing/sale of personal information). The Vizio precedent is directly on point.

How this was observed

These findings come from **static analysis of the extracted firmware** and **memory forensics on a device under my control**. The fingerprint structure, match results, and the production upload/control hosts were read directly from the live ACR process; the bootstrap control endpoint and the server-controlled query-interval logic are present as strings in the shipped `acr` binary, which fetches its actual upload target from a control server at runtime. **Security-specific details of how device access was obtained are deliberately withheld pending coordinated disclosure** and are out of scope for this privacy write-up.

What you can do

On current Samsung TVs, the ACR-related consent typically lives under **Settings → General & Privacy → Terms & Privacy** (look for **Viewing Information Services / Customization Service**). Disabling it withdraws consent for *use* of the data. One caveat here is that toggling consent is not the same as proving capture has stopped; there could be a situation where samsung still stores the content in a SQlite DB on the TV and then siphons all significant identifiers if the user were to re-enable ACR agreement.

This is independent security/privacy research on a device the author owns. No user data other than the author's own was accessed. Live credentials and identifiers have been redacted. Citation and contact: Sagar Mohan.