SERIES 0-DAY NO CVE HIGH PUBLIC

Jellyfin 12.2, Suggestions omit-userId hidden-library dump

Jellyfin Server 12.2 GET /Items/Suggestions with no userId queries the library as user=null. A restricted account dumps hidden-library Path and IDs. GHSA-9×85 taught GetUserId; Suggestions did not. No CVE yet.

05 SERIES Jellyfin 12.2 leftovers

00 The pin

00 Jellyfin, EncodingUtils -i leftover of GHSA-wwwm

00 Jellyfin, public playlist ItemIds leftover of GHSA-9x85

00 Jellyfin 12.2, Suggestions omit-userId hidden-library dump

00 Jellyfin 12.2, gzip data:image/png leftover of GHSA-cf3c

← PREVIOUS
Phase 0: Jellyfin, public playlist ItemIds leftover of GHSA-9×85

NEXT →
Phase 0: Jellyfin 12.2, gzip data:image/png leftover of GHSA-cf3c

NAME	SERIES
Jellyfin 12.2, Suggestions omit-userId hidden-library dump	Jellyfin 12.2 leftovers
PHASE	TYPE
0	0-day analysis
CVE	CVE RISK
n/a	high

DISCLOSURE STATUS	VENDOR
public	Jellyfin
AFFECTED	**PUBLISHED**
Jellyfin Server 12.2 (ca0f16eb7c195f720a1493ed469a27c4657db0c6), image jellyfin/jellyfin:12.2. Any authenticated non-admin. [Authorize] only.	17 Oct 2026
UPDATED	**TAGS**
17 Oct 2026	0-day, jellyfin, idor, cwe-639, cwe-862, authenticated

TL;DR

Lab files and proof-of-concept here.

01 Empty userId is not "me"

I am @abraxas_null. The proof of concept is on GitHub: **abraxas/jellyfin-suggestions-idor** (loopback client). The lab stack is `lab/`: docker-compose.yml, run.sh, poc.py. Authorized lab only. It talks to loopback.

This is leftover #1 of the Jellyfin 12.2 leftovers series. Operator order. Suggestions omit-userId dumps the hidden library, so it goes first. Same pin, other packs: jellyfin-svg-gzip (GHSA-cf3c leftover gzip `data:image/png`), jellyfin-playlist-itemids (GHSA-9×85 leftover playlist `ItemIds` plus unauth stream), jellyfin-ffmpeg-colonslash (GHSA-wwwm leftover `.strm` URL quotes). Those are siblings. This post is the catalog dump.

This is **Jellyfin Server v12.2** (`ca0f16eb7c195f720a1493ed469a27c4657db0c6`), image `jellyfin/jellyfin:12.2`. No CVE yet. CWE-639 / CWE-862. 6.5 High (`CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N`). **Reach:** Remote. **Class:** IDOR.

Authenticated. Any non-admin, including a child / hidden-library account. `[Authorize]` only.

`GET /Items/Suggestions` looks like "suggest something I can watch." When `userId` is omitted, `SuggestionsController.GetSuggestions` sets `user = null` and never calls `RequestHelpers.GetUserId`. `InternalItemsQuery(null)` does not `SetUser`. `LibraryManager.GetItemsResult` skips `AddUserToQuery`. `new DtoOptions()` includes `ItemFields.Path`. `GetBaseItemDtos(..., user: null)` skips `IsVisible`. A restricted token gets titles, IDs, overviews, provider IDs, and filesystem Path for libraries it is not assigned.

GHSA-9×85 taught `GetUserId`: omit goes to claims, a foreign GUID is 403. `ItemsController.GetItems` always uses it. Suggestions did not. Legacy `GET /Users/{userId}/Suggestions` always has a route GUID, so that sibling **does** go through `GetUserId`. The live route is the one without a GUID in the path.

Those GUIDs play on unauth `GET /Videos/{id}/stream`. VideosController has no class `[Authorize]`. `StreamingHelpers` loads `GetItemById<BaseItem>(id)` without a user. That stream-by-GUID model is longstanding HTML5/DLNA behavior. This leftover is what enumerates the IDs. The stream itself is labbed in the playlist leftover, not here.

02 Why Jellyfin

I wanted a self-hosted media server people actually run, with a household model that pretends a kid account cannot see the adult library. Jellyfin is that product. `EnableAllFolders=false` plus `EnabledFolders` is the whole security story for a lot of installs. Hide a folder from a user, they should not get Name, Id, or Path. `/Items` honors that. Suggestions does not.

12.0 had a bad month. VirtualFolders `Path.Combine`, SVG Skia external refs, playlist Items `userId`, HLS traversal, FFmpeg argument injection, LiveTV M3U, subtitle

`.strm`. Then 12.1, then 12.2. People upgraded. The changelog said the year was over. I sat on **v12.2** anyway.

I only wanted Critical or High. Remote preferred. Medium stays on the floor unless it chains. jellyfin-web-only XSS, Wizarr, Seerr, and jellyfin-ios are other trees. This pin is `jellyfin/jellyfin` tag **v12.2**.

Vendor intake is security@jellyfin.org. Unpaid. No bounty. I am not opening a public issue on jellyfin/jellyfin.

03 Pinning v12.2

Latest stable on the day I cloned. Tag object **v12.2**, commit `ca0f16eb7c195f720a1493ed469a27c4657db0c6`. `SharedVersion.cs` says 12.2.0. Image `jellyfin/jellyfin:12.2`. Stay on **12.2+** for the disclosed 12.0 GHSAs. Do not treat "I patched playlist Items" as "every query that returns `BaseItemDto` is scoped."

The lab binds **only** `127.0.0.1:18790`. Image tag 12.2. Two movie libraries, one named with the witness. A restricted user who can see PublicB and nothing else.

04 The skip list is half the hunt

Before you look for leftovers, you write down what is already public so you do not "discover" a CVE with a press release. On v12.2 the interesting rows were already closed:

- GHSA-9×85-gx46-6522 playlist Items `userId` IDOR. `GetPlaylistItems` now `GetUserId`s, then `IsVisible`. **This is the sanitizer Suggestions missed.** The leftover of that patch is this post plus the playlist `ItemIds` sibling, not a re-file of the disclosed route.

- GHSA-cf3c-8m59-2vhx SVG Skia external refs. Direct `http` / `file` hrefs on the outer SVG are denied. The leftover of that validator is gzip `data:image/png`, not this post.
- GHSA-wwwm-px48-fpvq FFmpeg subtitle path argument injection. PR 17685 escaped the `file:` `-i` branch. The leftover of that escape is non-File `.strm` quotes, not this post.
- GHSA-6828-c7cx-hvqm VirtualFolders `Path.Combine` rooted second arg. Closed.
- GHSA-4vx8-xhc9-qg6x session remote-control BAC. Closed.
- GHSA-hw73-62vp-9gxf legacy HLS traversal. Closed.
- GHSA-c39p-cgf8-866g plugin image traversal. Closed.
- GHSA-jg92-mrxq-vv75 / GHSA-9p5f-5×8v-x65m ClientLog traversal. Closed.
- GHSA-f47c-m7gr-q92j MKV attachment filename. Closed.
- GHSA-jh22-fw8w-2v9x unauth StreamOptions ffmpeg injection. Closed.
- GHSA-v2jv-54xj-h76w SyncPlay group-name DoS (CVE-2026-35034). Closed.
- GHSA-8fw7-f233-ffr8 LiveTV M3U tuner SSRF. Closed.
- GHSA-j2hf-x4q5-47j3 subtitle upload path + `.strm` RCE. Closed.
- GHSA-rrr6-mvwg-9pg9 branding splash DoS. Closed.
- GHSA-qcmf-gmhm-rfv9 `X-Forwarded-For` LAN policy. Default `KnownProxies`

empty means `ForwardedHeaders.None`. Closed.

- GHSA-2c3c-r7gp-q32m / GHSA-866x-wj5j-2vf4 FFmpeg codec-parameter injection. Closed.
- GHSA-vcmh-9wx9-rfqh SVG upload privilege. Closed as XSS (`Content-Disposition: attachment`).
- GHSA-rr9h-w522-cvmr custom FFmpeg binary path. Admin. Closed.
- GHSA-rgjw-4fwc-9v96 unauth remote-image GET. Closed.
- GHSA-wg4c-c9g9-rxhx unauth library download. Closed.

GHSA-9×85 is the interesting skip. It scoped playlist Items. It taught a helper. It did not walk every controller that still had a `user = null` branch.

05 Surfaces that returned nothing

I split the tree the way you split a media server: first-run wizard, path-combine, session remote-control, plugins, LiveTV M3U, subtitle/`.strm`, QuickConnect, backup, DLNA, branding splash, SyncPlay, transcode/trickplay, WebSocket, system/config/encoder path, library download, ClientLog, authz/`userId`.

Most of that is Jellyfin doing what the 12.0/12.2 changelogs said it would do.

Wizard cannot be re-run unauth on a finished server. `Path.Combine` on VirtualFolders / ClientLog / HLS / plugin image / attachments / subtitles / backup zip stays under the intended root on this pin. Session `AddAdditionalUser` is admin-only. QuickConnect `Authorize` uses `GetUserId`. Password/policy go through `AssertCanUpdateUser` / `RequiresElevation`. Devices are elevation. Sessions for a non-admin are own-session. `GET /Items?userId=` is claims. `GET /Playlists/{id}/Items?userId=` is the patched GHSA-9×85 route.

`POST /Items/{id}/RemoteImages/Download?imageUrl=` is still a raw `NamedClient.Default.GetAsync` with no private-IP deny. It is `RequiresElevation`. Artwork download of an admin-chosen URL is the feature. GHSA-rgjw closed the unauth siblings. I dropped it.

`POST /LiveTv/Programs` still takes a raw body `UserId`. Live TV EPG is a shared guide, not a hidden user's full library. Same Moderate class as the disclosed playlist Items IDOR. Not this leftover.

`GET /Users` lists the household, including hidden. `GET /Users/{id}` returns policy. No hashes, no library contents. Household enum. Not High.

The temperature changed on Suggestions sitting next to Items. Items always `GetUserId`s. Suggestions has a branch that does not.

06 The empty-userId branch is the whole argument

`GetSuggestions` is `[Authorize]`. No `Policies.RequiresElevation`. `userId` is optional.

PLAIN TEXT Copy

```
User? user;
if (userId.IsNullOrEmpty())
{
    user = null;
}
else
{
    var requestUserId = RequestHelpers.GetUserId(User, userId);
    user = _userManager.GetUserById(requestUserId);
}

var dtoOptions = new DtoOptions();
var result = _libraryManager.GetItemsResult(new InternalItemsQuery(user)
{
    OrderBy = new[] { (ItemSortBy.Random, SortOrder.Descending) },
    // ...
```

```
    Recursive = true
});

var dtoList = _dtoService.GetBaseItemDtos(result.Items, dtoOptions, user);
```

When `userId` is omitted or empty:

01 The branch never calls `GetUserId`, so claims are ignored.

02 `InternalItemsQuery(null)` does not `SetUser`.

03 `LibraryManager.GetItemsResult` skips `AddUserToQuery` when `query.User` is null, so it never fills `TopParentIds` from the caller's views. That fill is what scopes a user to their folders.

04 Recursive query with no parent / ancestor / top-parent filter is the whole item table. `query.User is null` in the repository only disables presentation-key grouping. It does not add a folder ACL.

05 `new DtoOptions()` defaults to **all** `ItemFields` including `Path`.

06 `GetBaseItemDtos(..., user: null)` skips `IsVisible` (`skipVisibilityCheck || user is null`).

Passing another user's GUID still hits `GetUserId` and 403s for a non-admin. The hole is the empty-userId path.

Contrast, patched sibling. `ItemsController.GetItems` always:

PLAIN TEXT Copy

```
userId = RequestHelpers.GetUserId(User, userId);
```

Omit becomes claims. A hidden parent 401s `IsVisible`. That is the rule Suggestions did not get.

`RequestHelpers.GetUserId` is not subtle:

PLAIN TEXT Copy

```
if (userId.IsNullOrEmpty())
{
    return authenticatedUserId;
}
if (!userId.Value.Equals(authenticatedUserId) && !isAdministrator)
{
    throw new SecurityException("Forbidden");
}
```

Empty means me. Foreign GUID means 403 unless admin. Suggestions implemented "empty means nobody," which in this stack means everybody's library.

Legacy `GET /Users/{userId}/Suggestions` forwards to the same method **with** a route GUID, so it cannot take the null branch. The live client route is `GET /Items/Suggestions`. That is the one the web UI hits when it wants a suggestion and does not bother to stamp a user.

I am not going to print a walk of a live host's hidden Path. The missing `GetUserId` is the useful part.

07 The lab

Source of truth is `lab/`. Image `jellyfin/jellyfin:12.2`. Port **18790**. Bind it to loopback.

`docker-compose.yml`:

YAML Copy

```
name: jellyfin-suggestions-idor

services:
  jellyfin:
    image: ${JELLYFIN_IMAGE:-jellyfin/jellyfin:12.2}
    ports:
      - "127.0.0.1:18790:8096"
    volumes:
      - jellyfin-config:/config
      - jellyfin-cache:/cache
      - ./media/private-a:/media/private-a:ro
      - ./media/public-b:/media/public-b:ro
```

`run.sh` pulls the image, seeds two movie libraries (PrivateA named with the witness, PublicB), brings compose up, waits for `GET /Startup/User`, then `poc.py` finishes the wizard, creates `labuser` with `EnableAllFolders=false` and `EnabledFolders=PublicB` only, and compares `GET /Items` against `GET /Items/Suggestions`.

PLAIN TEXT Copy

```
git clone https://github.com/abraxas/jellyfin-suggestions-idor
cd jellyfin-suggestions-idor
python3 jellyfin-suggestions-idor-Abraxas-Labs.py
```

or `cd lab && ./run.sh`. The loopback client is `lab/run.sh` with a banner on it. Target **only** `http://127.0.0.1:18790`. Do not publish the port off loopback.

Last lab run, trimmed:

PLAIN TEXT Copy

```
IOC user-items http=200 witness=0
IOC user-suggestions http=200 witness=1
IOC impersonate-suggestions http=403
SUCCESS jellyfin-suggestions-idor items-hidden=yes suggestions-has-witness=yes im
```

Restricted `labuser` cannot see PrivateA on `/Items`. The same token sees the witness on `/Items/Suggestions` with no `userId`. Passing the admin GUID still 403s. That is the whole argument. Folder ACL works on the route that calls `GetUserId`. It does not work on the route that sets `user = null`.

Witness string: `JELLYFIN-SUGGESTIONS-IDOR-WITNESS`. It is a folder name under PrivateA, so it shows up in Name and Path when the query is unscoped.

08 Wrong turns, in the order I made them

Treating `GET /Items` as the leak. `/Items` is scoped. The PoC fails closed if the restricted user can already see the witness there (`items-leaked-witness`). If your fixture leaks, the run is FAIL, not SUCCESS.

Treating `Suggestions?userId=<admin>` as the leak. That still hits `GetUserId`. Non-admin is 403. Lab last line requires `impersonate-403=yes`. The leftover is omit, not impersonate.

Treating the legacy `/Users/{userId}/Suggestions` route as the hole. It has a GUID in the path, so it cannot take the `user = null` branch. Wrong route. The live route is `GET /Items/Suggestions`.

Claiming unauth stream as this pack's bug. VideosController has no class `[Authorize]`. `StreamingHelpers.GetItemById` has no user. That is real, and it is how the GUIDs become bytes. It is labbed in jellyfin-playlist-itemids. This pack's witness is the catalog dump. I kept that split in the privilege table.

Expecting Path to be opt-in. `new DtoOptions()` is all fields. `ItemFields.Path` is on by default. There is no `Fields=` query to forget. The leak is not "the client asked for Path." The handler always asks.

Planting the witness only as a filename under PublicB. Then `/Items` would see it and the run would FAIL. The witness has to live in the hidden library. PrivateA is named with it so the DTO Name is enough. Path is extra.

A reverse shell. Theatre. The oracle is a witness string in JSON. I am not claiming plugin-DLL RCE. GHSA-866x's old chain is not this leftover.

Client bugs that look like product bugs. A helper that shadows `http.client` never sends a packet. `.recv()` on an `HTTPResponse` is not `.read()`. I mention that once because it cost time and looked, for a minute, like 12.2 had finished the job.

09 What an attacker can do

Any authenticated non-admin, including a child / hidden-library / `MaxParentalRating` account. Needs a normal user token. No admin, no API key. They do not change passwords. They do not mint admin. They inventory libraries they were not assigned, including filesystem Path, then play those GUIDs.

Privilege table (exhaustive). Same bar as Signal-Server Dropwizard admin: method + path, what that call grants, recon vs integrity vs availability, lab-proven vs inferred, adjacent surface that stays closed.

Recon (lab-proven)

METHOD + PATH	GRANTS
`GET /Items/Suggestions` **with no** `userId` **(optional** `limit` **/** `mediaType` **/** `type` **)**	Unscope recursive item list: Name, Id Overview provider IDs, **filesyste Path**, image metadat of librari the calle

METHOD + PATH	GRANTS
	is not assigned including folders given on to a hidden user. Folder ACL, parental rating, a allowed-tags are not applied.

Recon (source)

METHOD + PATH	GRANTS
Same with `limit` omitted	`InternalItemsQuery.Limi` stays unset; the repository returns the full match set (entire item table, random order)
unauth `GET /Videos/{itemId}/stream` with a GUID from that list	Bytes of that file as the jellyfin UID

A Path of `/media/private-a/JELLYFIN-SUGGESTIONS-IDOR-WITNESS/...` is a host filesystem layout. Combine it with the sibling stream leftover and the GUID is enough to pull the file without a session cookie. I am not reprinting a live Path.

Integrity / availability

None on Suggestions. No password, policy, favorite, or session write. No DoS beyond ordinary query cost.

Adjacent surface that stays closed

METHOD + PATH	WHY IT IS CLOSED
`GET /Items?userId=` and `GET /Items?parentId=<hiddenFolder>`	Always `GetUserId` + `IsVisible` (`ItemsController.cs:265` `350`)
`GET /Playlists/{id}/Items?userId=`	`GetUserId` (`PlaylistsController.cs` GHSA-9×85 class, patche
`GET /Items/Suggestions?userId=<other>`	Foreign GUID still hits `GetUserId` and **403**s for admin. Lab-proven (`impersonate-403=yes`)
legacy `GET /Users/{userId}/Suggestions`	Route GUID is required, s **does** call `GetUserId`
`POST /Users/Password?userId=`	`AssertCanUpdateUser`
`POST /Users/{id}/Policy`	`RequiresElevation`
devices, sessions, `AddAdditionalUser`	elevation or own-session
`GET /Users` / `GET /Users/{id}`	UserDto + policy (folder C admin bit), not library con

METHOD + PATH	WHY IT IS CLOSED
	or hashes. Household enu this leftover

No integrity. No availability. Confidentiality of hidden-library catalog and Path: High. That is why it is in bar.

A Path is not a password. It is still every title, GUID, provider ID, and on-disk location the household thought it had hidden from this account. Combine it with playlist `ItemIds` plus unauth stream and the GUID is playable without a token. Combine it with gzip SVG and you have a different class on the same pin. Combine it with `.strm` quotes and the plant is a scanned folder, not this IDOR.

10 What this is not

It is not "GHSA-9×85 never shipped." Playlist Items is scoped. This Suggestions omit is not. It is not unauthenticated. It is not a password change. It is not admin mint. It is not plugin-DLL RCE. It is not a reverse shell.

It is not the unauth stream as a standalone finding in this pack. Stream-by-GUID is how Jellyfin HTML5 has worked for years. Suggestions is what makes the hidden IDs enumerable from a kid account.

If the instance was ever **< 12.0**, rotate and patch for the disclosed 12.0 wave anyway. That is VirtualFolders, SVG, playlist Items, HLS, FFmpeg, LiveTV, `.strm`, not this lookup.

11 The fix

Call `RequestHelpers.GetUserId` when `userId` is empty, the way `ItemsController` already does. Keep the 403 on a foreign GUID.

PLAIN TEXT Copy

```
var requestUserId = RequestHelpers.GetUserId(User, userId);
user = _userManager.GetUserById(requestUserId);
```

Drop the `userId.IsNullOrEmpty() → user = null` branch. `new DtoOptions()` can stay; `IsVisible` will run once `user` is claims. `AddUserToQuery` will fill `TopParentIds` from the caller's views.

Re-run the loopback client against a patched build: restricted `labuser` Suggestions must match `/Items` (witness=0). Impersonate must stay 403.

Until then, do not treat "I scoped `/Items`" as "I scoped every query that returns `BaseItemDto`." Empty `userId` plus `user = null` is an instance-wide catalog.

I am not going to print a walk of `/Items/Suggestions` aimed at a live host. The missing `GetUserId` is the useful part. If you own the box, run the proof of concept against loopback.

Same product, other unpublished labs on this tag: gzip `data:image/png` SVG · playlist `ItemIds` plus unauth stream · `.strm` URL quotes. Series hub: Jellyfin 12.2 leftovers.

12 References

- Proof of concept: abraxas/jellyfin-suggestions-idor · jellyfin-suggestions-idor-Abraxas-Labs.py
- Lab: `lab/` · docker-compose.yml · run.sh · poc.py
- Series: Jellyfin 12.2 leftovers · jellyfin-svg-gzip · jellyfin-playlist-itemids · jellyfin-ffmpeg-colonslash

- @abraxas_null · github.com/abraxas · abraxaslabs.tech · abraxas.null@proton.me

- CWE-639 · CWE-862

- Tree: jellyfin v12.2 (`ca0f16eb7c195f720a1493ed469a27c4657db0c6`) · `SuggestionsController.cs` · `RequestHelpers.cs` · `ItemsController.cs` · `LibraryManager.cs` · `DtoOptions.cs` · `DtoService.cs`

- Nearby patched (not this lookup): GHSA-9×85-gx46-6522

- Product: Jellyfin

05 SERIES Jellyfin 12.2 leftovers

00 The pin

00 Jellyfin, EncodingUtils -i leftover of GHSA-wwwm

00 Jellyfin, public playlist ItemIds leftover of GHSA-9x85

00 Jellyfin 12.2, Suggestions omit-userId hidden-library dump

00 Jellyfin 12.2, gzip data:image/png leftover of GHSA-cf3c

← PREVIOUS
Phase 0: Jellyfin, public playlist ItemIds leftover of GHSA-9×85

NEXT →
Phase 0: Jellyfin 12.2, gzip data:image/png leftover of GHSA-cf3c

Abraxas Labs

Independent security research by abraxas.

Analysis of 0-days, n-days, malware, and the

leftovers that still matter.

GitHub **@abraxas_null** **abraxas.null@proton.me** **abraxaslabs.tech**

Home Research Open Source Reading Room About GitHub @abraxas_null Console

Independent research. Private analysis by arrangement. No ads. abraxas.null@proton.me · For defensive awareness and authorized testing only.

© 2026 Abraxas Labs