Episode Book

EPISODE #6183

Private Podcast Feeds: Tokens, Buckets, and Basic Auth

PUBLISHED	21 October 2026 at 15:31 UTC
DURATION	24:09
LISTEN	myweirdprompts.com/episode/private-podcast-feeds-privacy-05420/
HOSTS	Corn, Herman, Dorothy, Hilbert
SCRIPT MODEL	anthropic/claude-sonnet-5.5
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.11
BOOK GENERATED	22 October 2026 at 07:46 UTC

SYNOPSIS

Making a private podcast feed is easy; keeping it private is where things fall apart. This episode walks the whole ramp — from paid hosts like Transistor and Castos that hand each subscriber a tokenized feed address, down to a storage bucket and a Worker that signs every enclosure on the fly. Along the way: why "private, not secure" is the honest label, why pre-signed URLs go stale in seven days, why a password in the URL is an obscure URL with extra steps, and why podcast apps can't send the custom headers Cloudflare Access wants. Also, Dorothy calls in with a recipe for unbuttered bread.

TRANSCRIPT



Corn

How do you stop a podcast that exactly one person wants from being found by everyone with an internet connection? That's Daniel's question. He wants to make something like this show: a personal, AI-generated one, for himself or a handful of friends. And the part that trips people up is the privacy. Making a feed is easy. Making a feed that keeps its mouth shut is where everybody gets lost.



Herman

There's a ramp here, from paying somebody, to building it yourself out of a storage bucket, to maybe not needing a feed at all. The paid hosts are the easy end, so I'll take those first. Most of them do privacy exactly one way. Every listener gets their own feed address with a long random token in it, and there's a database row somewhere saying whether that address still works.



Corn

That's the whole trick? A long address and a row.



Herman

Transistor is the clearest about it. They separate "private" from "unlisted". Unlisted is one shared feed that nobody advertises, so you can't cut off one person without breaking it for everyone. Private is a separate feed per subscriber, emailed in a welcome message, and revoking someone is switching off that one address. You even get play analytics per person.



Corn

Per-person analytics for an audience of four. A detailed report on who skipped the intro.



Herman

Money, then. Transistor's Starter plan is nineteen dollars a month for fifty private subscribers. Professional is forty-nine for five hundred. Pay yearly and it's twelve months for the price of... no, ten. You pay for ten and get twelve. The cap is per account, shared across all your private shows, and one email address counts once.



Corn

Nineteen dollars a month for an audience of me.



Herman

Castos is in the same neighbourhood, nineteen for a hundred private subscribers, but that's from their own comparison page, so call it a vendor number. Supercast is built around paid membership, one feed per subscriber, and the address stays the same if somebody cancels and comes back. Patreon, Memberful and Substack all hand members a private link too, and Pocket Casts has a whole help page about pasting those in. Spotify's paid podcasts put one in the confirmation email.



Corn

Does anyone notice when my cousin pastes her link into a group chat?



Herman

Hello Audio reportedly disables the link shortly after you first subscribe, which limits the damage. I couldn't confirm that from their own page, though. For Supercast I haven't seen a statement either way. Buzzsprout is the blunt one. People in their community describe the private feed as private, not secure. Obscure address, nothing behind it.



Corn

Private, not secure. They should print that on the box.



Herman

It's fine if you know that's what you bought. Apple's own subscriptions are the exception, built into the app so the listener never sees a feed address, but you're then in Apple's creator program, nineteen ninety-nine a year, and from what I've read secondhand they keep a cut of what members pay. All of it is priced for people selling subscriptions. A hundred and ninety dollars a year, which is what that cheapest Transistor tier comes to if you pay up front, is a lot for one listener.



Corn

A hundred and ninety dollars a year to host a podcast for four people. That's how people end up putting an XML file in a bucket.



Herman

A feed is a text file and some audio files, and anything that serves bytes over HTTPS can hold both. S three, R two, Backblaze B two, Wasabi. The problem you've signed up for has two halves and most people only notice one. The feed has to be protected, and then every audio file it points to has to be protected, because the enclosure is just another URL. Lock the feed, leave the MP three public, and anyone with the file address walks in the side door.



Corn

And the app has to get through both doors without me typing a password every morning.



Herman

Ideally automatically, yes. The cheapest answer is the obscure URL. You put the whole show under a path with forty random characters in it, unguessable, and you don't link to it from anywhere. A crawler never finds it and nobody guesses it.



Corn

Then I'm done. Nobody is going to guess forty random characters.



Herman

Nobody will guess it, no. The guess was never the risk. The address leaks. It sits in your app's backup, in a screenshot, in a message you sent to a friend, in somebody's browser history. Whoever has it has everything, forever, and you can't revoke one person. You change the whole path and tell everyone again. For a show that's only for you, honestly, that's a fair trade.



Corn

You sound like you're about to say but.



Herman

The other leak is directories. If you submit the feed anywhere public, or an app indexes it, the show is out. Pocket Casts has you submit the feed and tick Private so it stays out of search. And the itunes block tag asks Apple to leave you out, but I haven't pinned down how every crawler treats it, and it's a request, not a lock. Same with the podcast locked tag. That only stops other hosting services importing your feed, if they choose to honour it.



Corn

A polite sign on an unlocked door.



Herman

The standard upgrade is HTTP basic auth. The server asks for a username and password and the app supplies them. Support is patchy. Overcast takes a username and password in the app when you add a show by URL. AntennaPod supports it, including credentials written into the address, according to a maintainer, though almost nobody uses it. Apple Podcasts prompts you for them when you follow a show by URL... hold on, I should narrow that. It prompts. Whether it accepts a username and password written into the address itself, I haven't seen confirmed anywhere.



Corn

And Pocket Casts?



Herman

Their help page says they don't formally support password-protected feeds. Their word is "limited". The documented method is to put the username and password into the address itself, and special characters have to be written in their encoded form, so an at sign in your password becomes a percent code. Player FM, Castbox and the old Google Podcasts reportedly don't do auth at all, but the guide saying so is undated.



Corn

So the password is in the URL. I've built an obscure URL with extra steps.



Herman

Not quite, because the server can say no properly. You can change the password, you can have several, and the audio sits behind the same wall. Though the help page itself says never to share a feed with credentials embedded, which tells you where the leak went. What I don't know is how each app treats credentials on the audio requests, not just the feed. Does the app send them along when the enclosure lives on a different host?



Corn

You don't know, and I don't know. Anyone who does, I'd like to meet.



Herman

The object stores have their own answer, pre-signed URLs. You compute a signature with your secret key and the URL says, this object, for this long. Anyone holding it can fetch the file until it expires. On R two the longest it can last is seven days, which is six hundred and four thousand eight hundred seconds. The signing happens on your machine, no call to R two at all.



Corn

Seven days. So the feed I publish on Sunday has dead links by the next Sunday.



Herman

A static feed full of signed enclosures goes stale within a week. That's the reason a small piece of code has to sit in the middle. A Worker, or a Lambda on the Amazon side. Each time an app fetches the feed, the code builds the XML on the spot and signs every enclosure fresh. Or the enclosure address points at the code, which checks a token and then either streams the file itself or answers with a three oh two redirect to a freshly signed URL.



Corn

Do podcast apps follow a redirect like that? My doubt is the app that doesn't.



Herman

I haven't seen it tested across Podcast Addict, Overcast and AntennaPod. Redirects are routine on public feeds, because analytics services bolt prefixes on, so I'd expect most to follow. That's an expectation, not a result. Then there's the R two wrinkle. Pre-signed URLs don't work on a custom domain, only on the long account address. If you want your own domain with auth, Cloudflare points you at HMAC validation in its firewall rules, and that needs a Pro plan or above. Which is why people let the Worker do the checking itself.



Corn

And if the Worker streams the audio instead of redirecting?



Herman

Then it has to handle range requests properly, or seeking and resuming break. The app asks for the file from the middle, and the Worker has to answer from the middle. There's a reference implementation of that floating around. People forget that part and then wonder why the scrub bar does nothing.



Corn

Skip back thirty seconds is a feature somebody has to build.



Herman

The simplest Worker is the token in the path. A request comes in, the Worker looks at a secret segment of the address, compares it to what it holds, and serves the feed or serves nothing. Give each listener their own token and revoking one is deleting one line. That's what the hosted services do. You've rebuilt them, for pennies. Corn, one token or one per listener, if it were yours?



Corn

One. I'm one person. If I add my brother I'll get a second, and I'll rotate the first when I feel like it.



Herman

You'd defend the one-line config even with five people listening, and call it a principle.



Corn

It's repairable. A principle is just a repair nobody has needed yet.



Herman

I think that covers the token pattern, but we seem to have a caller on the line. Hello there, you are on the air.

D

Dorothy

Hello dear, it is your mother. It is Dorothy's Kitchen, and I am ringing with the recipe.



Corn

Hello, Mum.

D

Dorothy

Today it is a slice of bread. Unbuttered.



Herman

Unbuttered. We're taking notes, Dorothy.

D

Dorothy

The ingredients are bread, and nothing. That is the whole list. It is a short list, which is the only mercy.



Corn

Nothing and bread. Got it.

D

Dorothy

You take the bread. You watch it. You watch it the whole time, sweetheart, because the moment you look away it starts getting ideas. The butter is not involved and I would like to keep it that way. The butter has turned on me before and I would rather not say how.



Herman

How long do we watch it, Dorothy?

D

Dorothy

Timing is everything. If the bread sits out too long it goes sad, and if it does not sit out at all it is simply bread on a plate, which is what we are having, so do not overthink it.



Herman

Understood.

D

Dorothy

There. You see. Nothing burned. Wear a cardigan, it is chilly. Bye bye, dear.



Corn

Bye, Mum.



Herman

Cloudflare Access is the tempting wall for all of this. Put your bucket or your Worker domain behind it and make people sign in. The machine-friendly flavour is a service token, which means two extra headers on every request, a client ID and a secret. Podcast apps can't send custom headers. They have no setting for it.



Corn

So the app knocks on the door and can't say the password the door wants.



Herman

The workaround, as I read the docs, is a bypass policy on the feed and audio path. Bypass switches Access off for that traffic, and it doesn't log it either, and you can't use an email identity rule in a bypass. So the real security becomes a token in the URL that a Worker checks. Cloudflare doesn't publish a podcast recipe for this. It's my reading of how the pieces fit. And their bot protection can block feed readers, separately, so test from the actual app.



Corn

A podcast app is a bot with one very narrow hobby.



Herman

Now the wrinkle I'd flag for anybody who wants to keep a feed on the home network. Pocket Casts' own help pages say their servers poll private feeds, and keep checking after a subscription is canceled or expired. That means the fetching happens on their machines, not your phone.



Corn

So a feed that only exists on my LAN doesn't work in Pocket Casts.



Herman

It can't, because their server can't reach your house. And the credentials in the address sit on their infrastructure, since it's their server that needs them. A forum user said the fetches arrive from American Amazon addresses, so people allow-listed those. That's community, not official. For Overcast, the app takes a username and password, but whether its servers then fetch the feed with them, I haven't seen confirmed. Whether any given app fetches from the phone or from the cloud decides whether your home server is even an option.



Corn

Which is a strange reason to choose a podcast app.



Herman

For the self-hosted people there's Audiobookshelf. You click Open RSS Feed on a book, series, collection or podcast, choose a custom slug, and it serves a feed. Progress doesn't sync for listening through the feed. And the feed only works at the address you used when you made it, so a local IP and a public domain are different feeds. Then there are folder-to-feed generators. dir two cast is a PHP script that reads the tags inside the MP three files and caches the feed until the folder changes. Dropcaster is a Ruby command line tool that writes an RSS file you drop on a static host, with the channel described in a settings file. There's a Docker image, though whether anyone still maintains it, I couldn't say. folderpodgen is a Python package you install with pip.



Corn

And drive_rss, the one with Drive in the name.



Herman

A Python script that builds a feed from a Google Drive folder. But the folder has to be publicly shared, because it reads it with a Google API key, and you host the XML yourself.



Corn

A privacy tool whose first step is making the folder public.



Herman

Anyone-with-the-link, so it's the obscure URL again, one more time. Daniel's real fascination is the app that treats a cloud folder like a podcast, though.



Corn

He saw it on Android, he's smitten, and I went looking for it. I couldn't find that app.



Herman

Neither could I, as described. No mainstream Android app subscribes to a Drive folder as a feed. What exists is two things stitched together. Podcast Addict has Virtual Podcast. You tap plus, pick a local folder, and the files show up as episodes with progress tracking and playlists. It's device-local, doesn't sync between devices, and reads local folders only, not Drive or WebDAV directly. AntennaPod has a local folder feature that does a similar thing. Its maintainers say deliberately that they don't sync local folders. Sync there is gPodder or Nextcloud only.



Corn

So the cloud part is whatever else keeps the folder filled.



Herman

The Drive app, or Dropbox, or FolderSync, or Syncthing, running on the phone and mirroring a folder. The podcast app never knows there's a cloud. I'd say that's a perfectly good arrangement, because the privacy is your Drive login, which is real authentication, and no address-based scheme matches that. That's my view, not something I read. Would you take it, Corn, given there's a second app doing the carrying?



Corn

Yes. Files on the phone, so the commute works with no signal, and nothing hosted anywhere. I'd miss the feed's tidy episode notes, but I'd survive that.



Herman

The players that talk to the cloud directly are audiobook-style. CloudBeats on iOS connects to Dropbox, Google Drive, OneDrive, Box, pCloud and WebDAV, downloads for offline and remembers where you stopped. Audiobook Reader on iOS imports from Files, iCloud Drive and WebDAV, and probably copies things onto the device. On Android there's Shelf, a free open-source player that streams from Google Drive folders. It was created in March, it's at version one, and it has no stars.



Corn

No stars. Not even the author's.



Herman

Would you trust it with your listening, though?



Corn

With one folder and nothing in it I couldn't replace, sure. I'd like to see somebody else's name in the commit history first. And Smart AudioBook Player?



Herman

A twenty seventeen blog post describes a workaround, pointing the library at Dropbox's local sync folder. That's the same sync-plus-folder idea, not native cloud support. I also never found a Podcast Addict WebDAV option. And Captivate has a Zapier automation that makes draft episodes from new files in a Drive folder, but that runs the other way, it publishes. So the idea Daniel loves mostly lives as sync plus local folder, or as a script like drive_rss, with a public link and an unguessable folder name standing in for a wall.



Hilbert

Unguessable is not a word I accept. Dennis Pallister guessed ours in four minutes. Forty-one characters. He says he did not guess it, he intuited it, and the Circle has now spent three meetings on the difference. The Jerusalem Pastry Mould Preservation Circle, eleven members, Thursdays, in the bus shelter on Hebron Road. The minutes go out as audio, members only, which is Rule Fourteen, carried seven votes to six. Dennis was expelled on the ninth. He has until the thirty-first to appeal, and he is appealing by listening.



Herman

Forty-one random characters can't be guessed by anyone. So the address came from somewhere. A copy leaked.



Hilbert

It came off the membership card. We laminate them. I have three hundred and twelve laminators, and the Circle needs one. The address is printed on the card so members can type it in at the shelter. Dennis kept his card. His position is that expulsion does not reach back to laminate, and the committee has been asked to rule on whether a laminated address can be revoked. The motion is Rule Twenty-two. It has been tabled twice.



Corn

So the minutes of the meeting about Dennis listening to the minutes are being listened to by Dennis.



Hilbert

He has attended every meeting. The attendance sheet has eleven members and Dennis, marked present in absentia. The minutes are read aloud by the chairman's sister, Pauline, in what she calls a neutral voice, into a dictaphone chained to the shelter's timetable. Pauline does not use the word members. She says persons.



Herman

Then the file has to get from the dictaphone to the bucket somehow, and every handoff is somewhere the address can get out.



Hilbert

Pauline carries the tape to the chairman's flat in a biscuit tin. The chairman is, for tax reasons, two of the members. He uploads it as the other one. The bucket is on the Backblaze account in his late mother's name, which has been paid up through the end of next year.



Corn

Two suspects, one chair.



Hilbert

Oriel Katz is the secretary. She changes the address every Thursday and posts the new one to each member in a sealed envelope. Dennis gets one as well, because Rule Nine says expelled members receive correspondence for forty days. Rule Nine was written by Dennis. He also does the envelopes. Nobody has removed him from envelopes, because there is no rule for removing a man from envelopes. He licks them at his own kitchen table. He has the new address before anyone.



Herman

A separate address for each member would fix the whole thing. Every envelope gets a different token, and when the address gets fetched, you see whose copy it was. If it's Dennis's, you switch that one off and nobody else notices.



Hilbert

That is a good arrangement. One address each, and the fetch tells you who. I would like to see it run. I did two winters in a sorting office, so I know what an envelope tells you about the man who sealed it.



Corn

Then Dennis opens the minutes of the meeting that expels him, and the audio stops at the word motion.



Hilbert

Twelve envelopes, then. Pauline's counts separately. She says reading it and hearing it are different jobs, and she wants her own.



Herman

The ladder depends on who you are. If it's just you, the lowest rung is the sync-and-local-folder route. Nothing to host, nothing to leak, and the wall is your cloud login. Podcast Addict or AntennaPod on the phone, a sync client keeping the folder full.



Corn

One rung up is the bucket and the long address, never submitted anywhere. Fine if the only person who could leak it is you and you've stopped sending screenshots. I'd stay there.



Herman

Above that, you want more than one listener, so each person needs their own token. A Worker, a token in the path, and either signed URLs or streamed audio with proper range handling. It costs an afternoon, and you own the revocation.



Corn

And the top rung is paying somebody. Nineteen dollars a month, and a stranger's database remembers who's allowed.



Herman

For a family of four that's the least work. Corn, what do you do for the person who won't install anything, who just wants the show in whatever app they already have?



Corn

Then they get the Worker and a link. Private feeds are a link by design, so the app doesn't matter, only whether it fetches from the phone or from its servers. I'd ask them which app first.



Herman

Basic auth sits between the rungs, for anyone whose app takes it, and you check that against the app before you promise anything.



Corn

People think that a private feed means nobody else can listen.



Herman

A private feed means a separate address per person, and anyone holding an address can listen, and can share the audio. Transistor says outright that it isn't for confidential material. Private describes who got the link, not who can hear it.



Corn

Forty characters were never the weak part. The weak part is wherever somebody wrote them down so they wouldn't forget them, a card, an envelope, a note on the phone. So the sensible thing to build is a feed where you can tell whose copy it was, and cut that one off.



Herman

Thanks to our producer, Hilbert Flumingtop, for keeping the levels where they belong. This has been My Weird Prompts.



Corn

Every episode is at my weird prompts dot com.



Herman

See you tomorrow.