

2025-06-26 Lix NixOS Module

- Date: 2025-06-26
- Participants: Niko, k900, piegames, Raito
- Topic: The current state of the Lix NixOS module (<https://git.lix.systems/lix-project/nixos-module/>)

Conclusion / Action items

- ☑ piegames will reach out in *Lix on main* asking for volunteers
- Primary goal is to get rid of the module by integrating its overlay-style functionality into `pkgs.lixPackageSets`
- Stretch goal is to have a `lix` module in the NixOS name space for additional configuration

Discussion

Selective transcript from a [Matrix discussion](#)

- niko: Is lix-module something that's planned to be gotten rid of? I'm asking because since Lix now requires cgroups for auto-allocate-uids, whereas cppnix doesn't (unless I'm wrong) then it'd be nice for lix-module to extend the nix module and add checks for stuff like whether cgroups is enabled if auto-allocate-uids is, and similarly to check whether nix-daemon service unit has cgroups configured, and other potential differences that come up that nixpkgs doesn't necessarily take into account
 - raito: answer is yes
- k900: I think we can upstream all of that to nixpkgs tbh
- piegames: Even the lix overlay?
- piegames: This functionality should still be available for non-NixOS though. The module does little more than setting the nix package option and applying an overlay that does all the nix lix substitutions in dependents
 - k900: But we can have things in lixPackageSets that are lix-overlay-shaped
 - raito: lixPackageSets does it too - lixPackageSets is the Nix-dependent universe reinstated with Lix
- raito: If someone wants to take the lead on Nixpkgs packaging, I can provide guidance, review time and help, but I cannot spearhead this until we landed the start of RPC in Lix I'd say
 - piegames: Do we want to publish a larger call-for-participation on this? Maybe on *Lix on main* for example. Raito would you be open to mentor such a thing?
 - raito: Agreed
- k900: What are we missing on the nixpkgs side?
 - raito: Double checking there's nothing missing between the module, implementing a deprecation notice on nixos-module side, I'd say. Instructions & docs

- k900: Oh you mean not nixpkgs packaging of lix, but absorbing nixos-module to nixpkgs
 - raito: I meant the nixos-module replacement inside nixpkgs, yep
 - raito: Possibly, designing for `lix.enable = true;`, etc. We are increasingly diverging from CppNix, the `nix.*` namespace will feel constrained at some point
 - raito: `import <nixpkgs> { config.nixImplementation = "lix"; }` something like this, which dumbs it down to an overlay or anything idc
-

Revision #2

Created 2025-06-28 01:57:11 UTC by piegames

Updated 2025-07-26 22:28:22 UTC by piegames