Rhino Linux & Pacstall: Towards a Rolling Ubuntu
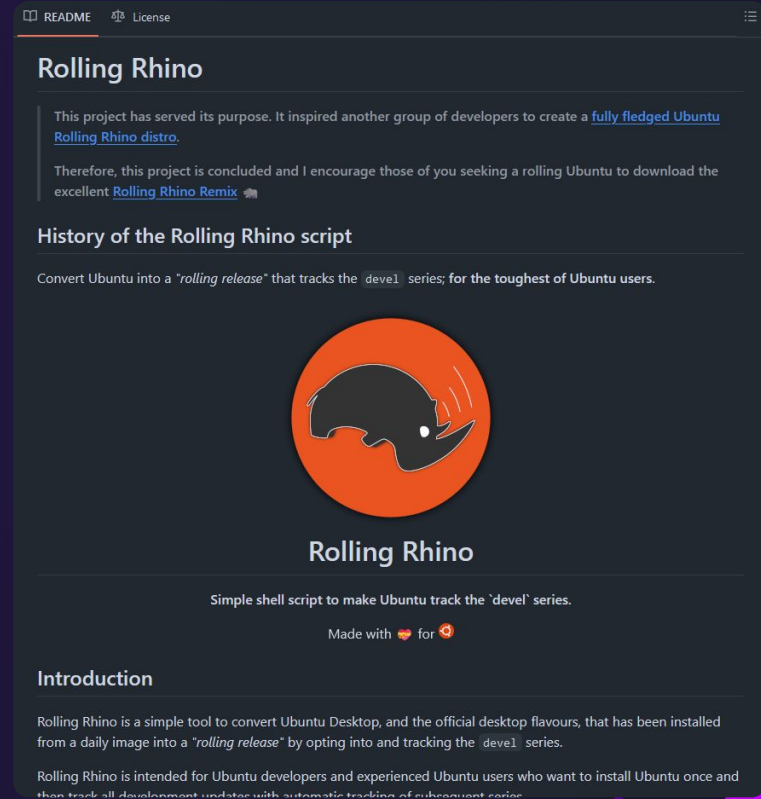
Adam Salt & Oren Klopfer

1.1 From a Script to a Remix

- Initial concept from "Rolling Rhino" script by Martin Wimpress

- But what if it was a full flavour?

1.2 The making of Rolling Rhino Remix

- Update script pulled latest kernels + modifications for the user

- Gradual introduction of more utilities

Rolling Rhino Remix

Home | Download | Documentation | Blog | Bug Tracker

Rolling Rhino Remix - Blog

Release 2022.03.26

Rolling Rhino Remix has had its first official public release! Ubuntu is now (unofficially) provided as a rolling release distribution! You are able to download the .iso file for the distribution here.

What's new?

- All packages have been updated to their latest versions in the Ubuntu repository
- The rhino utility tool has been created and packaged in the distribution. Rhino is used as a drop-in replacement for apt update
  upgrade and allows for the initialisation of the distribution upon first installation. To learn more please read the rhino
  ...ation and our Quick Start guide.
- ...eplaced the default background with our own background image featuring our mascot, a rhino! You are able to
  ...d the background image here.

- ...olling Rhino is now fully open source upon its first release, if you for any reason decide to create your own disk image
  ... the ones we have created, you can. Just read our documentation and follow the steps provided here.

...ll of our contributors who have made our first release possible! You are able to view the full list of contributors here.

... from all of our developers & contributors <3

...nonical are registered trademarks of Canonical Ltd. Rolling Rhino Remix is not affiliated with the Ubuntu Project.

🖥 RRR-builder (Public)

RRR-builder is used for the creation of Rolling Rhino Remix disk images. Automated through GitHub actions.

🟢 Shell   ⭐ 12   ⑂ 8

🖥 rhino-update (Public)

Rhino Update is a command-line utility script which will provide updates to items that do not hit the Ubuntu devel repositories, such as the latest Linux Kernel.

🟢 Shell   ⭐ 15   ⑂ 11

🖥 rhino-config (Public)

Rhino Config is a command-line utility that allows you to configure Rhino, our utility tool that allows you to upgrade the distribution. You will be able to configure Rhino to download and install ...

🟠 Rust   ⭐ 9   ⑂ 2

🖥 rhino-deinst (Public)

rhino-deinst is a command-line utility script which provides a Terminal User Interface to install desktop environments on Rolling Rhino Remix + Ubuntu-based systems

🟢 Shell   ⭐ 8   ⑂ 4

2.1 A New Source Based Package Manager

- Provide up-to-date releases on top of standard package managers
    - Originally distro agnostic, but quickly moved to Ubuntu only

- Started off with some "unique" decision making
    - Pulled and kept all build scripts gzipped in the git repo
    - Managed package files by tracking syscalls from build scripts

Check out Henry's blog here to get an in-depth history of Pacstall's origins

1. Get package tarball from repo.
2. Get individual bash scripts from repo and run them (such as `build.sh`, `install.sh`).
3. Install with (and this part changed very early from `checkinstall(8)` to) porg[1].

2.2 Transitioning to a Standard Format

- Introduction of Pacscripts, like PKGBUILDs
  - Built to special location and integrated via symlinks w/ GNU stow

- From symlinks to proper .debs

```
name="foo"
pkgname="foo"
version="1.0"
url="https://github.com/Henryws/foo/archive/refs/tags/1.0.zip"
license="GPL"
build_depends="vim gcc"
depends="neofetch plasma"
gives="libfoo"
breaks="libfoo-git"

prepare() {
        command -v $depends
}

build() {
        ./configure
        make -j$(nproc)
}

install() {
        sudo make install DESTDIR=/usr/src/pacstall
}
```

```
[+] INFO: Running functions
        [>] Running prepare
        [>] Running package
[+] INFO: Packaging pacstall
        [>] Packing control.tar
        [>] Packing data.tar
        [>] Compressing
[+] INFO: Package built at /home/rhino/pacstall_6.1.0-pacstall1_all.deb
[+] INFO: Performing post install operations
        [>] Storing pacscript
```

3.1 From a Remix to a Distro

- Accumulation of features turned RRR into something more than just an Ubuntu flavour

Rolling Rhino Remix is officially End of Life.

In October 2022 I wrote a blog post detailing that Rolling Rhino Remix would no longer be reciving any further development, in favour of its successor project, Rhino Linux. A lot of development has happened since then, and I am proud to announce that Rhino Linux is ready to be released soon!

Rhino Linux will still be a rolling release distribution based on Ubuntu. It's main advantages have already been announced, such as the AUR-like package manager: Pacstall, the rhino-pkg package manager wrapper, and more. A new development is that it will feature our own heavily-customised XFCE-desktop environment, called Unicorn, which will combine the best things about many different desktop environments. It will have an App Grid, Virtual desktops dashboard, uLauncher and much more.

Unfortunately this means Rolling Rhino Remix is officially end of life. That means that from here on out, it will recieve no major or minor developments, and we will be in the process of archiving the repositories.

The next course of action, if you are still on Rolling Rhino Remix would be to install Rhino Linux. We are currently in open beta as of the 12th May 2023, and have had users report a relatively stable experience. You will be able to update from the beta to the release version with a simple update.

The website for Rhino Linux: https://rhinolinux.org

I would like to thank everyone who has supported the development of Rolling Rhino Remix, and those that will continue to support us at Rhino Linux. It has been an honour working on this project, and I can't wait to keep providing with Rhino Linux.
- Sincerely, AJStrong - Project Lead for Rhino Linux and Rolling Rhino Remix

AJStrong

3.2 Rhino PKG and Device Ports

- rhino-pkg combined RRR's utilities into one, and more...

- Early establishment of core ports:
  - Generic ISO (AMD64 + ARM64)
  - Pine64 (PinePhones + PineTabs)
  - Raspberry Pi (Desktop + Server)
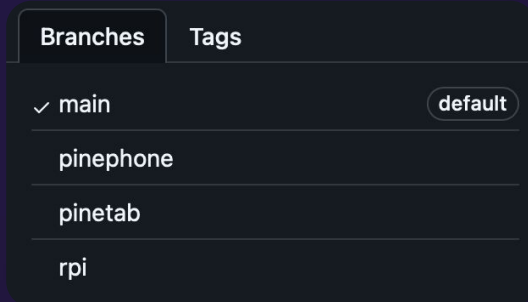

```
Terminal - rhino@rhino: ~

[~] rhino@rhino $ rpk install discord
Found packages matching 'discord':

[0]: com.discordapp.DiscordCanary (flatpak)
[1]: com.discordapp.Discord (flatpak)
[2]: libdatetime-calendar-discordian-perl (apt)
[3]: libdiscord-rpc-dev (apt)
[4]: libdiscord-rpc3 (apt)
[5]: php-symfony-discord-notifier (apt)
[6]: purple-discord (apt)
[7]: ruby-discordrb-webhooks (apt)
[8]: discord (pacstall)
[9]: discord-canary (pacstall)
[10]: discord-deb (pacstall)
[11]: discord-ptb-deb (pacstall)
[12]: discord (snap)
[13]: discord-canary (snap)
[14]: cwtlbot-discord (snap)
[15]: playstationdiscord (snap)

Select which package to install [0-15]:
```

Branches	Tags

✓ main	default
pinephone	
pinetab	
rpi	

3.3 Creating the Unicorn Desktop Experience

- Needed a more stable, yet still familiar desktop

- Modernizing XFCE

- Fear of the terminal and need for GUI
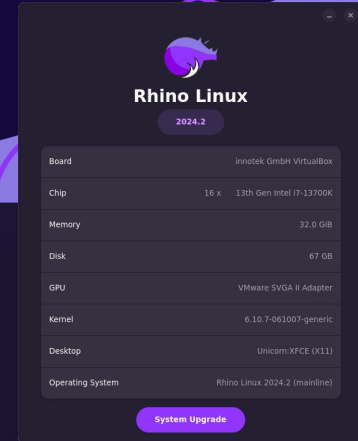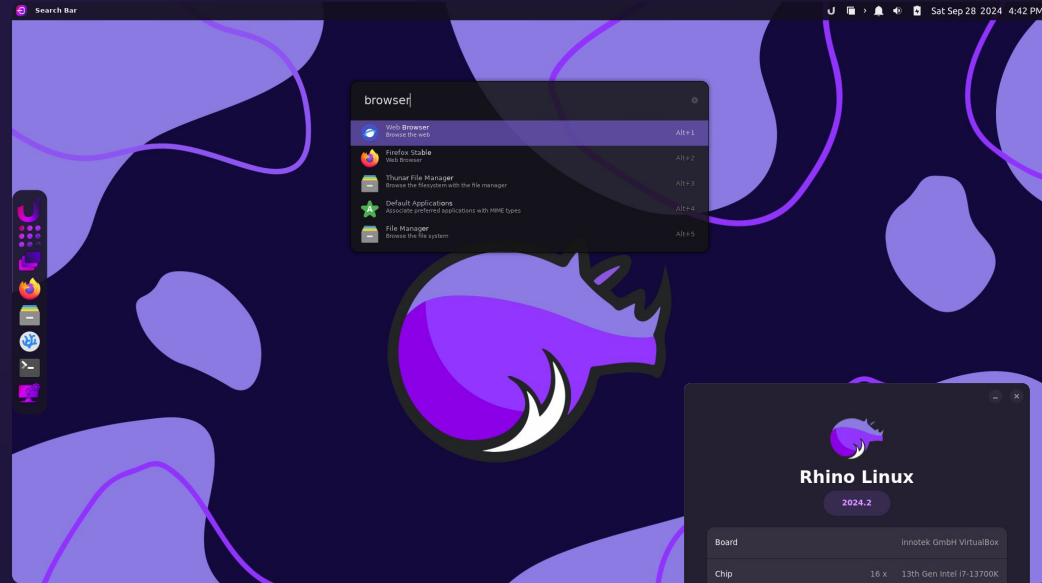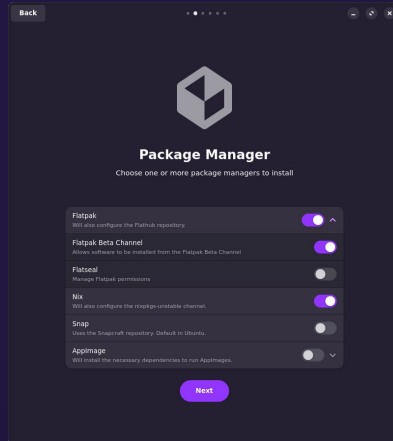
3.4 Integrating with Rhino Linux

- Rhino specific/oriented goals emerged:
    - Multi-architecture support
    - Distro version compatibility
    - Priority field

- rhino-core, rhino-pine-core, and rhino-server-core

rhino-core	Oren Klopfer	2025.1-1
rhino-pine-core	Oren Klopfer	2025.1-1
rhino-server-core	Oren Klopfer	2025.1-1
rhino-ubxi-core	Oren Klopfer	2025.1-1

- `amd64` / `x86_64`
- `arm64` / `aarch64`
- `armel` / `arm`
- `armhf` / `armv7h` *
- `i386` / `i686` *
- `mips64el`
- `ppc64el`
- `riscv64`
- `s390x`

```
 1    ubuntu:
 2       focal/20.04
 3       jammy/22.04
 4       noble/24.04
 5       oracular/24.10
 6       plucky/25.04
 7       devel
 8    debian:
 9       bullseye/11
10       bookworm/12
11       trixie/13
12       sid
```

1. `essential`
2. `required`
3. `important`
4. `standard`
5. `optional` (default)

3.5 Mediating Between Debian and PKGBUILD

- PKGBUILD parity: fn/var/arr renames

- Deb conformity: handling control fields

```
Package: rhino-server-core
Version: 2025.1-pacstall1
Status: install ok installed
Priority: required
Essential: yes
Section: Pacstall
Maintainer: Oren Klopfer <oren@taumoda.com>
Bugs: https://github.com/pacstall/pacstall-programs/issues
Installed-Size: 38.9 kB
Provides: rhino-server-core
Depends: rhino-pkg, rhino-neofetch
Conflicts: rhino-core, rhino-pine-core, rhino-ubxi-core
Replaces: rhino-core, rhino-pine-core, rhino-ubxi-core
Uploaders: Oren Klopfer <oren@taumoda.com>
Download-Size: unknown
APT-Manual-Installed: yes
APT-Sources: /var/lib/dpkg/status
Description: Transitional package to provide minimal core Rhino Linux software
```

pre_remove

This function is run before the a package is removed. It is the Debian `prerm` script.

pre_install

This function is run before the deb unpacks onto the system. It is the Debian `preinst` script.

pre_upgrade

This function works just like `pre_install`, but runs instead if the package is already on the system. If `pre_install` is provided but not `pre_upgrade`, then `pre_install` runs as `pre_upgrade`. If only `pre_upgrade` is provided, then it will not run on initial install, but will on every upgrade after.

post_remove

This function is run after the package is removed. It is the Debian `postrm` script.

post_install

This function is run after the package is installed. It is the Debian `postinst` script.

post_upgrade

This function is run after a package upgrade. The same rules from `pre_upgrade` and `pre_install` apply for `post_upgrade` and `post_install`.

4.1 Getting the First Rhino Release Rolling

- Moving on from beta releases
    - Snapshot versioning

- Handling community growth
    - Project roadmaps and kanban boards
    - Consolidating community channels

- Mirror struggles
    - Slow speeds or high costs

2023.1 Release - The making of the distro

Rhino Linux Team, Tue Aug 08 2023   Back 🌙

Rhino Linux has now officially moved out of Beta! We have released Rhino Linux 2023.1 on x86_64, ARM, Pine64 and Raspberry Pi devices.

Quite a lot has changed since our first beta release back in April 2023, and so instead of a generic changelog, I want to do something a little different and discuss what actually went into the making of the operating system.

You can download Rhino Linux 2023.1 **here**. Contact us **here**.Or maybe donate to us **here**.

⊞ **Rhino Linux 2023.4** #3 updated 15 hours ago	Development for Rhino Linux 2023.4
⊞ **Rhino Linux 2023.3** #2 updated on Jun 13, 2024	A collection of issues and changes to be addressed in the development of Rhino Linux 2023.3
⊞ **Rhino Linux 2023.2** #1 updated on May 12, 2024	A collection of issues and changes to be addressed in the development of Rhino Linux 2023.2

📁 2023.1-beta7	2023-07-09
📁 2023.1-beta6	2023-07-01
📁 2023.1-beta5	2023-05-19
📁 2023.1-beta4	2023-04-29
📁 2023.1-beta3	2023-04-12
📁 2023.1-beta2	2023-04-07
📁 2023.1-beta1	2023-04-04

4.2 Branching and Burning Out

- Attempted to expand with the Unicorn Beyond XFCE Initiative (UBXI)

- Momentum slowed, and we burned out
  - Stepped back for several months

The Unicorn Beyond XFCE Initiative

Rhino Linux Team, Thu Oct 19 2023   Back ☾

We announced the development of Rhino Linux just over one year ago today, and to celebrate our birthday we would like to officially present the "Unicorn Beyond XFCE Initiative" (UBXI).

Rhino Linux was designed with user control in mind. The user should have the right to choose, and that should extend to their choice desktop environments as well.

While we have continued to improve the Unicorn Desktop environment, our users have consistently asked us to provide alternate desktop environment spins, and have shown us consistently that they can create highly stylised Unicorn themed desktops on other environments/window managers.

UBXI, put simply, is a way for our users to maintain their own desktop environment packages and distribution spins. Unicorn is more than just the XFCE base it was built on, it's an idea that helps to improve the Rhino Linux desktop experience, and Rhino Linux is more than the desktop environment the user interacts with.

Replacing Unicorn (XFCE)

⚠ **Warning:** We are still looking for UBXI maintainers, and at this time no UBXI ports exist. This being said, the proper way to switch to any generic desktop environment available in normal repositories (GNOME, KDE, etc) is:

```
# switch to UBXI
rpk install rhino-ubxi-core # or rhino-server-core
# install some environment
rpk install kubuntu-desktop
# remove Unicorn
rpk remove unicorn-desktop-git
```

ERROR - Evaluating, Rethinking, and Rebalancing Our Rhinos

In the Linux community, there is nothing short of 'quite a few tale, of how Linus does not scale' - as our team has found out recently, our team struggles with this exact issue too. **Read More →**

Sun Feb 11 2024

4.3 Wrestling with the Rhino

- Why Ubuntu?
    - Rhino is an untried concept
    - Devel vs Sid timing and stability

- Ubuntu, Why?
    - Opinionated Snap + AppArmor patches break upstream, that we have to fix downstream

(for the next week or two) a risky move. Debian and Ubuntu are in the peak of transitioning to 64-bit time, which comes at the cost of them rebuilding virtually every package. (https://wiki.debian.org/ReleaseGoals/64bit-time)

This is causing issues, as some t64 dependencies of other t64 packages are not ready before their parents, which could result in possible system breakage if it is just the wrong packages. This breakage extends all the way to debootstrap being stuck, meaning we cannot even generate test .ISO images for development. (https://wiki.debian.org/BrainDumpT64)
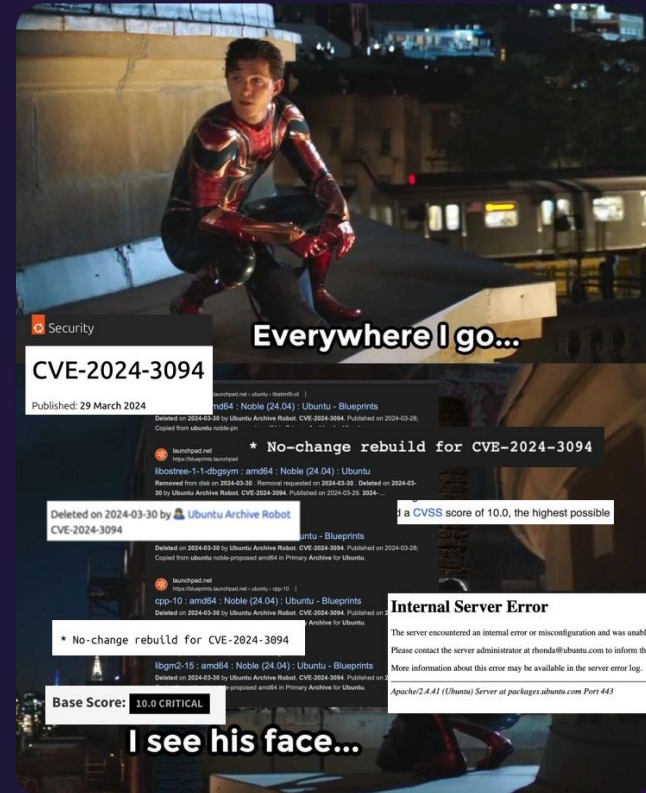
4.4 Reevaluating and Restructuring

- Coming back together as a team

- Mutual code of conduct established

- Web/Blog/Wiki overhaul

rhino

Rhino Linux Wiki

Home   Contact ↗   Search documentation...   ⌘ K

Introduction

User Docs 📚

Installation Guide

ISO/Live Image

Pine64

Raspberry Pi

Post-Install

Rhino PKG

Pacstall

Unicorn Desktop

User Docs 📚 > Installation Guide > ISO/Live Image

ISO/Live Image

GRUB Bootloader

When you boot from your Live USB you will be greeted with the GRUB Bootloader, which is the most commonly used bootloader for Linux systems. You will want to select "Try or Install Rhino Linux" - Which should be selected by default, and press enter.

• Rhino Linux currently utilizes unsigned kernels, this means that you may need to disable secureboot in your UEFI/BIOS beforehand.

Booting to the Rhino Linux Desktop

When you first boot into your live system, the installer will not automatically start, rather you are immediately given the chance to try Rhino Linux on your live USB. When you are ready to install Rhino Linux you can click on the icon on your desktop*, which will then launch our customised Calamares Installer.

Contributor Code of Conduct

Introduction

• This document should serve as a framework for the standards that we hold contributors and those on the Rhino Linux team. This document is intended to be continually updated and edited as Rhino Linux grows as an organisation.

• Last Edited: June 29th 2024.

At Rhino Linux, we are committed to fostering an inclusive and welcoming community which seeks to thrive on mutual respect and honesty. We wholeheartedly believe that every member of our community should be treated with dignity and respect, regardless of race, sexuality, gender identity, disability or any other protected characteristic. To uphold these values, we expect all contributors to adhere to the following code of conduct:

Expected behaviours

Expected behaviours are to be followed by all contributors to the Rhino Linux project in all environments. Contributors represent the

5.1 Major Shifts Towards Parity...

- url var to source array
  - source architecture + distro extensions

- Dependency version constraints

- Pkgbase splitting

i. `pkg>=version`

ii. `pkg<=version`

iii. `pkg>version`

iv. `pkg<version`

v. `pkg=version`

```
pkgbase="rhino-pkg-git"
source=("https://github.com/rhino-linux/rhino-pkg.git")
pkgver="0.1.2"
makedepends=("make" "gettext")
depends=("gettext")
pkgdesc="Rhino-pkg wrapper script"
maintainer=("Oren Klopfer <oren@taumoda.com>")
pkgname=("rhino-pkg-git" "rpk-git")

package_rhino-pkg-git() {
  gives="rhino-pkg"
  cd "${gives}"
  DESTDIR="${pkgdir}" make install
}

package_rpk-git() {
  gives="rpk"
  pacdeps=("pacstall-qa-git")
  cd "${pkgbase%-git}"
  DESTDIR="${pkgdir}" make install
  mv "${pkgdir}/usr/bin/rhino-pkg" "${pkgdir}/usr/bin/rpk"
  rm -r "${pkgdir}/usr/share/locale"
}
```

BREAKING CHANGES

- Use `bwrap` when running functions by **@D-Brox** & **@oklopfer** ([#1068](), [#1069](), [#1075]())
- `url` -> `source[@]` by **@oklopfer** ([#1038]())
- `homepage` -> `url` by **@oklopfer** ([#1051]())
- `replace` -> `replaces` by **@Elsie19** ([#1016]())
- `SRCDIR` -> `PACDIR` by **@oklopfer** ([#1045]())
- `name` -> `pkgname` by **@Elsie19** ([#1017]())
- `hash` -> `*sums[@]` by **@oklopfer** ([#1053]())
- `maintainer` -> `maintainer[@]` by **@Elsie19** ([#1022]())
- `STOWDIR` -> `STAGEDIR` by **@D-Brox** ([#1097]())
- Mark `pacdeps` as deps for `-deb` packages by **@oklopfer** ([#1062]())
- Require `cd` for `build/prepare/check/package` functions by **@oklopfer** ([#1059]())

Features

- Add version constraints to all `*depends` arrays by **@Elsie19** ([#1021]())
- Add `check()` function and `-Nc/--nocheck` optional flag by **@oklopfer** ([#1056]())
- Add `{pre,post}_upgrade` & `pre_remove` functions by **@oklopfer** ([#1055]())
- Add `conflicts` array by **@oklopfer** ([#1054]())
- Add `nosubmodules` array by **@oklopfer** ([#1060]())
- Add `license` array by **@oklopfer** ([#1063]())
- Add `var_CARCH` style arrays by **@oklopfer** ([#1073]())
- Add `checkdepends` array by **@oklopfer** ([#1072]())
- Add `-Qa/--quality-assurance` command by **@oklopfer** ([#1089]())
- Add `-Q/--quiet` optional flag by **@oklopfer** ([#1092]())
- Show if you are in a develop branch when using `-V` by **@D-Brox** ([#1081]())

5.2 ...Maintainability and Security...

- SRCINFO integration

- Build isolation with bwrap

- i18n/l10n

- Stacktrace made for Bash

```
        [>] Packing data.tar
        [>] Compressing
sudo: gzip iiii: command not found
[!] ERROR: Stacktrace (most recent call last)
  ─> source() /usr/bin/pacstall:905
        └─> if ! source "$SCRIPTDIR/scripts/package-base.sh"; then
  ─> package_pkg() /usr/share/pacstall/scripts/package-base.sh:242
        └─> package_pkg
  ─> source() /usr/share/pacstall/scripts/package-base.sh:160
        └─> if ! source "$SCRIPTDIR/scripts/package.sh"; then
  ─> makedeb() /usr/share/pacstall/scripts/package.sh:369
        └─> makedeb
  ─> createdeb() /usr/share/pacstall/scripts/build.sh:604
        └─> createdeb "${pacname}" "${full_version}" "${deb_arch}"
  ─> TRACEBACK: /usr/share/pacstall/scripts/build.sh:274
        └─> sudo "$command iiii" "${flags[@]}" "$DATA_LOCATION" "$CONTROL_LOCATION"
[+] INFO: Cleaning up
```

```
pkgbase = firefox-bin
        gives = firefox
        pkgver = 133.0.3
        pkgdesc = Free and open-source web browser developed by the Mozilla Foundation
        url = https://www.mozilla.org/en-US/firefox
        arch = amd64
        maintainer = Oren Klopfer <oren@taumoda.com>
        repology = project: firefox
        source = https://download-installer.cdn.mozilla.net/pub/firefox/releases/133.0.3/linux-x86_64/en-US/firefox-133.0.3.tar.bz2
        sha256sums = 43713e238d0153fdbf1ab46dd76c6b01ab83fae197b5dc3a95087f51907ba44d

pkgname = firefox-bin
```

5.3 ...Scalability and Reproducibility

- Package site API improved from SRCINFO data

- Introduction of the PPR
    - Major conformity sweeps
    - Long term goals

The Chaotic PPR is finally live!

Check it out at https://ppr.pacstall.dev/

Breaking Changes

- Enforce `arch` array, split `any` and `all` by @oklopfer (#1266)
- `spdx` license list update by @oklopfer (#1269)
- `pkgbase` download and function rework by @oklopfer (#1270)
- Do not allow `-U/--update` if pacstall is installed from deb by @oklopfer (#1274)
- Remove `_archive` internal var by @oklopfer (#1289)

Chaotic PPR   **Pacstall**

What is the Chaotic PPR?

The Chaotic PPR is a system where Pacstall builds debs from pacscripts and uploads them to an APT repository, meaning that you can enjoy prebuilt pacstall packages without the build times, and you can even use the Chaotic PPR without Pacstall installed!

Installation instructions:

```
$ bash -c "$(curl -fsSL https://pacstall.dev/q/ppr)"
```

6.1 Reaching Out and Reflecting

- Recognizing we've gotten ambitious
    - Several incomplete projects lie dormant

- It's okay to ask for help
    - Making a call for contributors

Call for Developers and Contributors!

Rhino Linux continues to roll along, but as a small team of student developers, our limited resources and manpower can make it challenging to address every issue and suggestion as quickly as we'd like to - and trust us, we want to address them all. We're calling on you to join us in keeping this Rhino rolling, and make it even better. Will you join us? **Read More →**

Mon Nov 18 2024

horns (rhino-import)

A script that will allow for you to import your Rhino Linux installation from a configuration script.

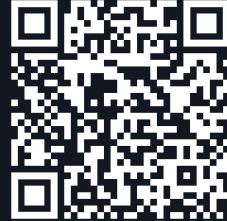## rework: switch to rhinu-pkg

#36 · oklopfer opened on Nov 14, 2023

6.2 Keeping the Rhino Rolling

- Future plans and in the works:
  - RPK2
  - Unicons, Unicorn Settings, Wayland
  - UBXI Ports, Mobile DE
  - Server ISOs

- A Rhino Foundation?

Thank You!

rhinolinux.org

We will update this blog post shortly to include our full script from this talk

Check out Henry's blog here to get an in-depth history of Pacstall's origins

pacstall.dev