



KubeCon



CloudNativeCon

North America 2024

The Path To Helm 4

Before We Get Started



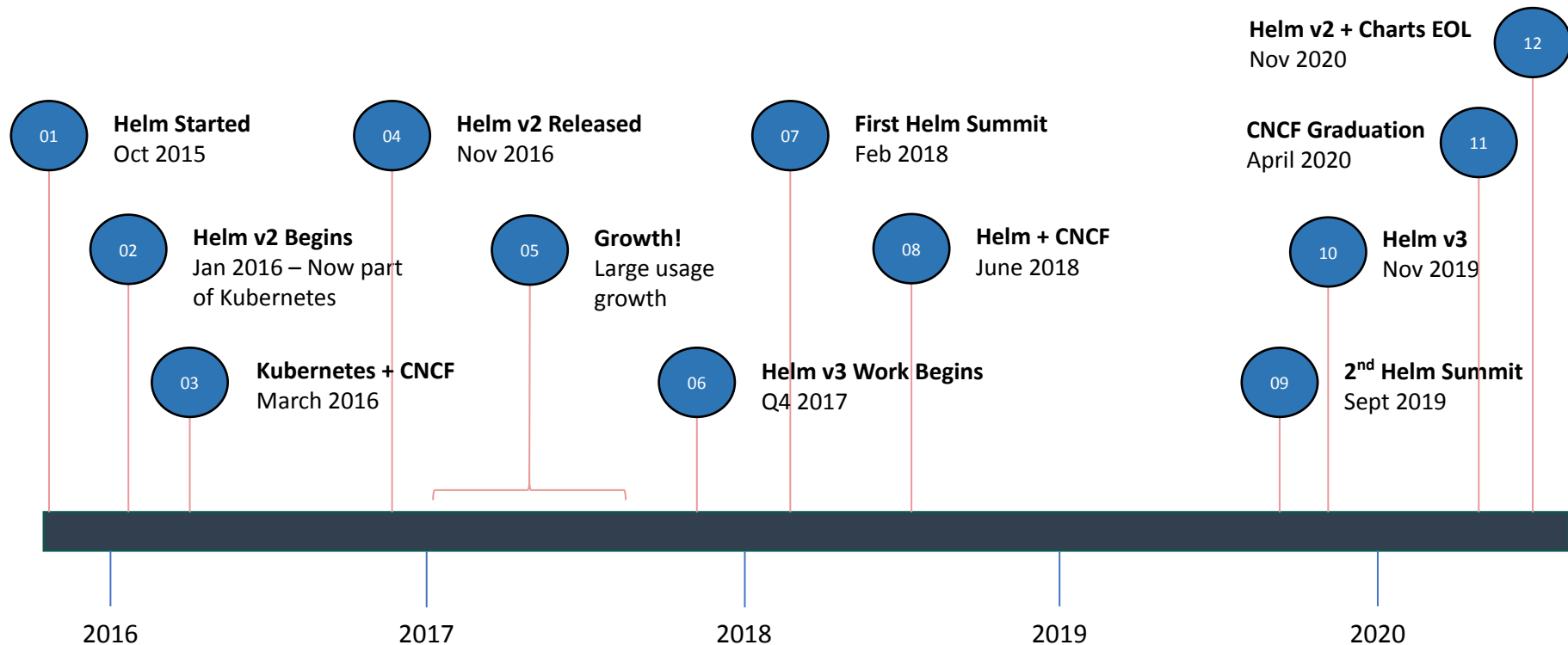
Matt Farina



Andrew Block



The Ancient Past...



Features Since Helm v3.0.0

Several features have been delivered since the initial release of Helm v3.0.0



Post Rendering

Manipulate rendered manifests before installed by Helm



JSON Index Files

JSON indexes scale better



OCI Registry Support

Storage of Helm charts as OCI artifacts in OCI registries



Advanced Autocompletion

The Helm CLI has autocompletion in new ways



KubeCon




CloudNativeCon


North America 2024


Helm 4: Its Time Has Come

Why Helm 4?

Nov 13, 2019

 bacongobbler

 v3.0.0

 e29ce2a

Compare ▾

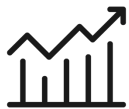
Helm v3.0.0

The Helm Team is proud to announce the first stable release of Helm 3.

Helm 3 is the latest major release of the CLI tool, focusing on simplicity, security, and usability.

Check out below what's available in this major release.

5 years, 1 day since the original release of Helm 3.0.0



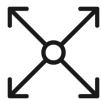
Technical Debt

Accumulation within the Helm codebase that has becoming a limiting factor for future innovation



API Versioning Policy

Strict backwards compatibility policy following semantic versioning limits types of changes and enhancements that can be made



Feature Size

Size of proposed initiatives are larger than what can/should be delivered in a minor version of a project

Take advantage of the latest capabilities of Kubernetes and the Cloud Native Community

Frameworks



CLOUD NATIVE
COMPUTING FOUNDATION



Patterns

Projects



kubernetes

Communities



KubeCon




CloudNativeCon

North America 2024

Let the Games Begin!





It All Begins With a HIP

community / hips / hip-0012.md

 robertsirc adding matt as named release engineer ... ✓ e64a1e4 · 3 days ago History

Preview Code Blame

273 lines (186 loc) · 17.3 KB

Raw    

hip	title	authors	
0012	Helm 4 Development Process	Matt Butcher <matt.butcher@fermyon.com>	George Jenkins <gvjenkins@gmail.com>

Abstract

Helm 4 is planned as the next major version release of Helm. Continuing to evolve Helm in its role as the Kubernetes package manager.

This HIP outlines the requirements and process for building Helm 4. It describes the scope of Helm 4 changes, with a focus on maintaining continuity within the Helm ecosystem.

This HIP also aims for an open and productive process that enables Helm 4 to build upon the success of Helm 3. To continue advancing Helm and its ecosystem for the benefit of the Helm community.

Helm Improvement Proposal

Design document describing a new feature or process for the Helm project

Helm v4 Plan includes the high level steps that will be needed to deliver the next version of Helm

- Rationale for Helm 4
- Prerequisites
- In Scope/Out of Scope
- Timelines

Helm v4 Plan

This document outlines a plan for the development of Helm v4 including details on Helm v3 handling, the scope of changes for v4, and other pertinent details.

Reason For v4

When KubeCon/CloudNativeCon North America goes to Utah it will have been 5 years since the release of Helm v3.0.0. Since the release of Helm v3, there have been many additions to Helm (e.g., OCI registry support) that have proven to be useful. Along the way, Helm has accumulated design debt which has continued to pile up. Kubernetes has continued to grow and change. Some of those Kubernetes changes can now be defaults in Helm but that would be a breaking change for users.

Helm follows [semantic versioning](#) and has [a policy around backwards compatibility](#). This helps to communicate the kinds of changes in Helm to users and protects helm users (especially SDK and automation users) from changes in Helm breaking anything.

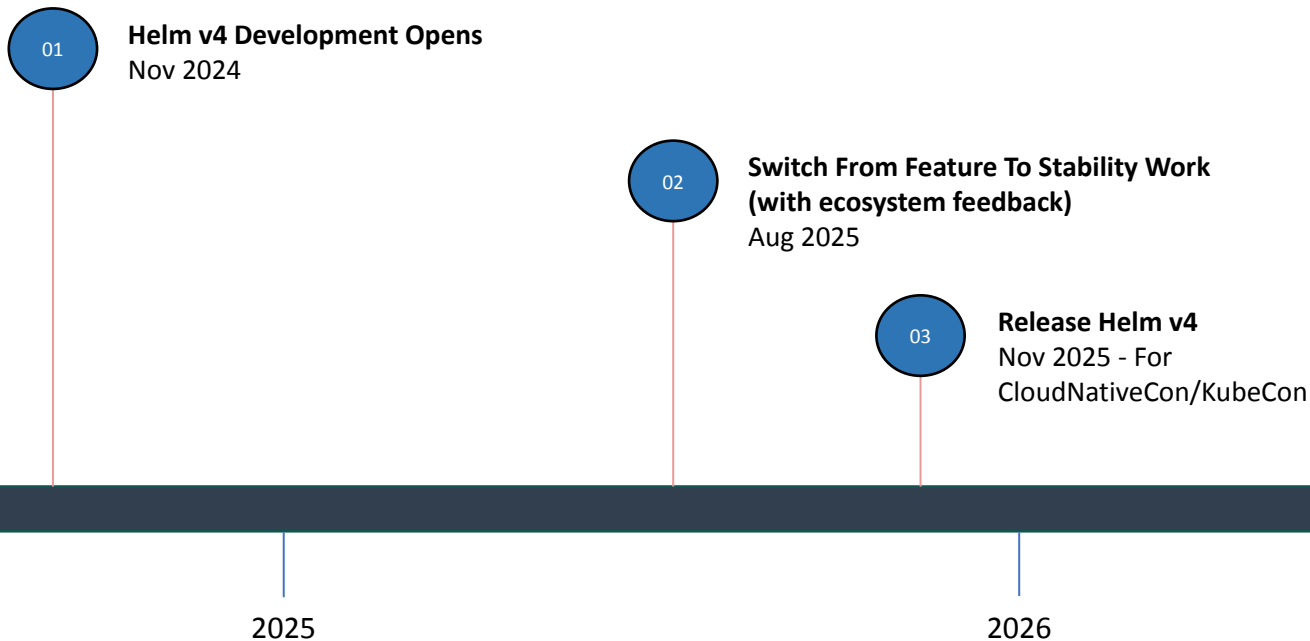
In order to reduce technical debt and leverage some newer capabilities of Kubernetes and Go, it's time for Helm to have a major release that makes API and other breaking changes.

Prerequisites

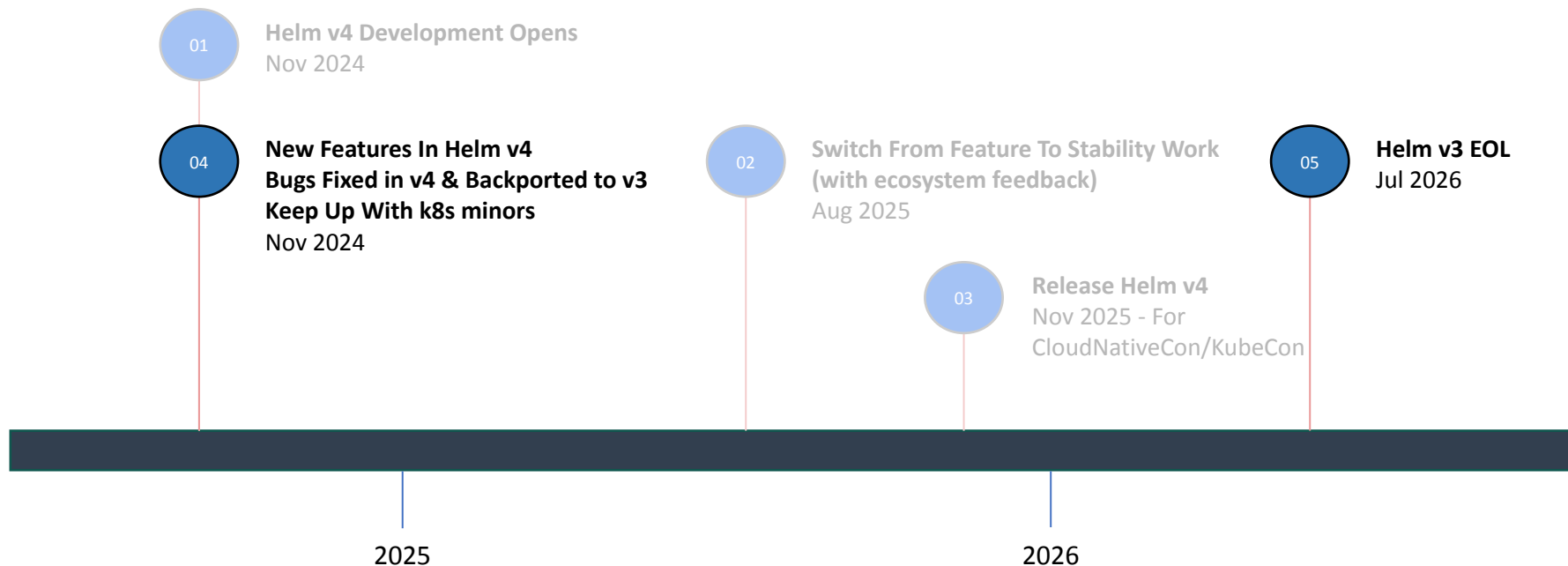
Before Helm can start the development of Helm v4 there are some things the Helm organization needs to have in order. These include:

1. **Active maintainers who will take the time to review changes.** Helm has been in maintenance mode for some time where changes have slowed down and the project needs the people capable of driving a v4.
2. **Direction on what is in and out of scope for v4.** It's important to be able to know what to say "yes" to and what to say "no" to with reasons for both.
3. **A timeline for the development of v4.** A timeline sets expectations for the time of development. If the time for development is open-ended then development could end up in a perl 6 development cycle and just take forever. At some point development needs to stop. There will be opportunities for Helm v5, v6, and so forth (as long as we don't screw up Helm).

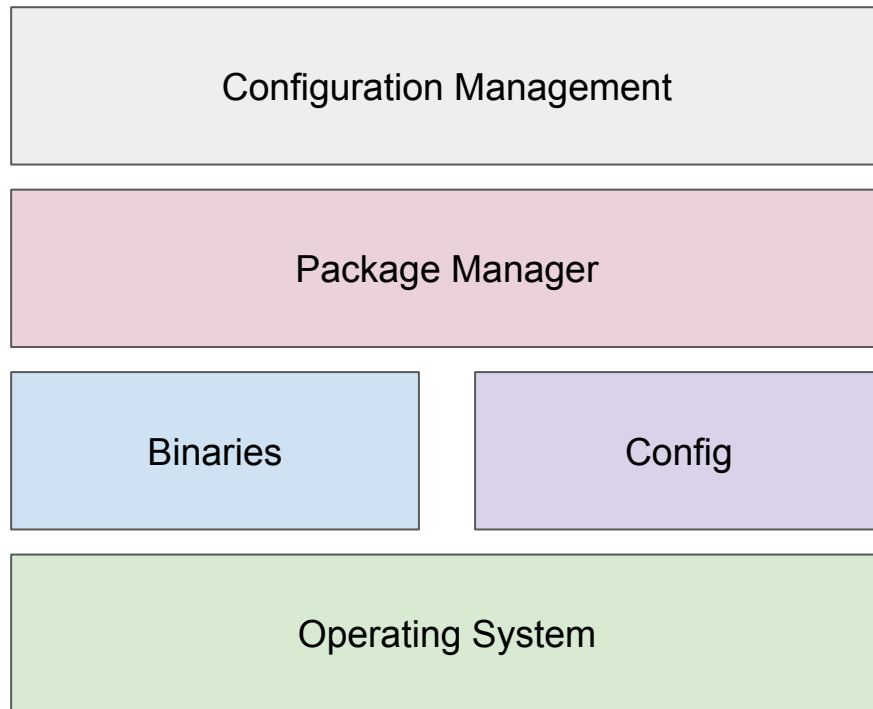
Helm v4 Timeline

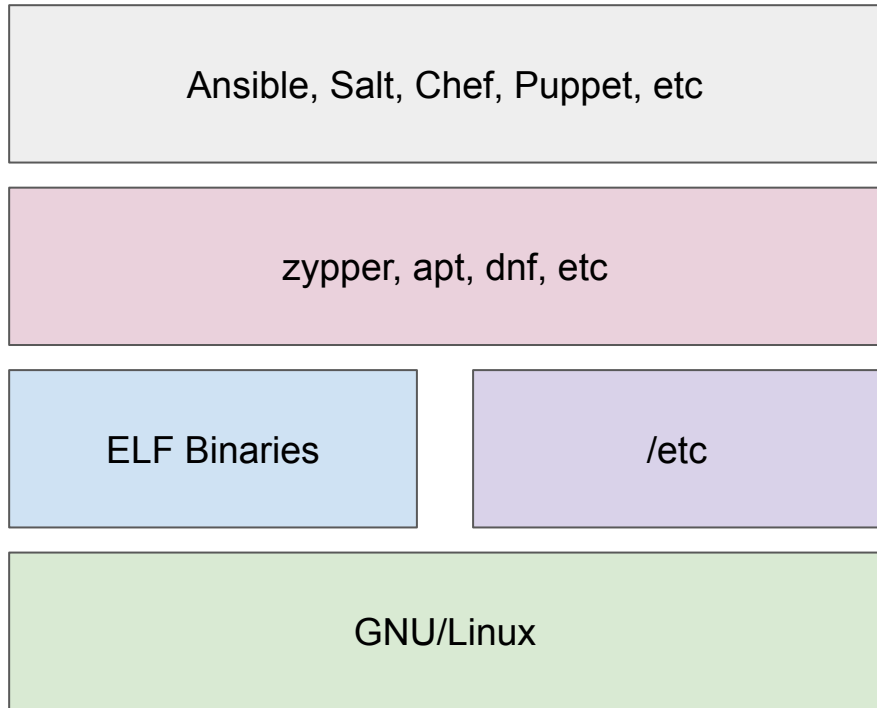


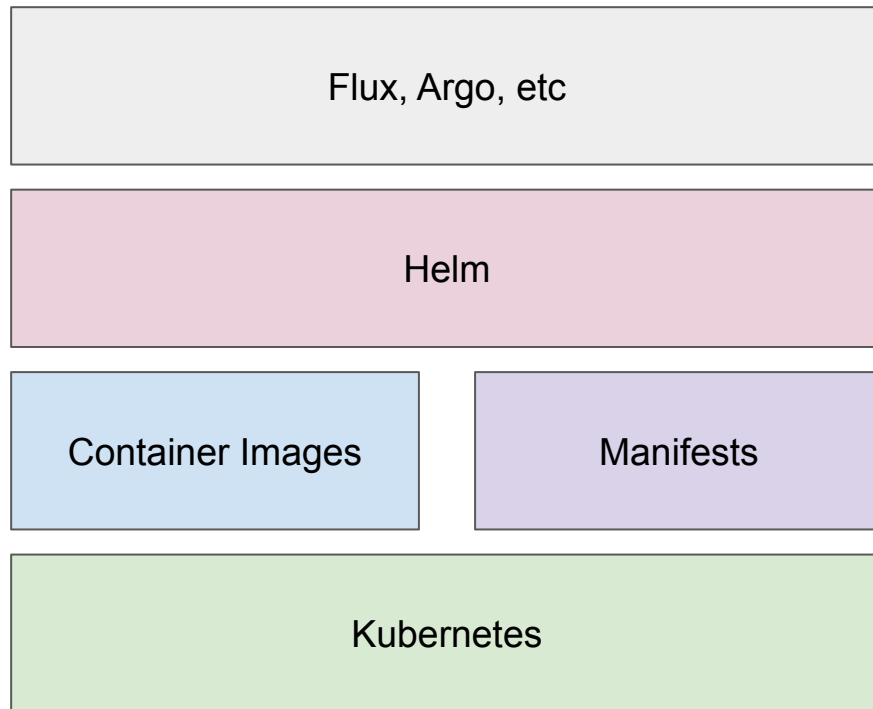
Helm v3 Impact



1. Breaking Charts - The Charts in use today **MUST** continue to work.
2. Changing The Scope of Helm - Helm is a package manager. The scope **SHALL NOT** change.









KubeCon



CloudNativeCon

North America 2024

What Might We Do?

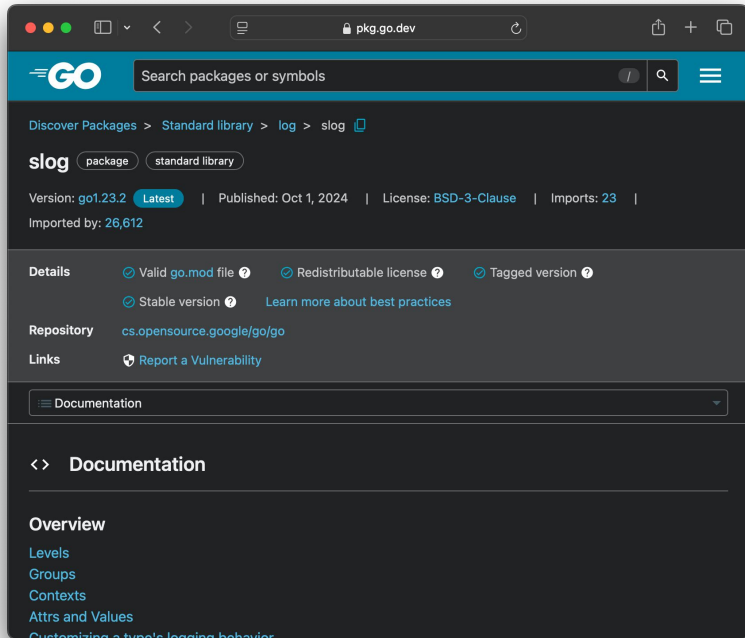
```
// InterfaceExt is introduced to avoid breaking backwards compatibility for Interface implementers.
//
// TODO Helm 4: Remove InterfaceExt and integrate its method(s) into the Interface.
type InterfaceExt interface {
    // WaitForDelete wait up to the given timeout for the specified resources to be deleted.
    WaitForDelete(resources ResourceList, timeout time.Duration) error
}

// InterfaceDeletionPropagation is introduced to avoid breaking backwards compatibility for Interface implementers.
//
// TODO Helm 4: Remove InterfaceDeletionPropagation and integrate its method(s) into the Interface.
type InterfaceDeletionPropagation interface {
    // Delete destroys one or more resources. The deletion propagation is handled as per the given deletion propagation value.
    DeleteWithPropagationPolicy(resources ResourceList, policy metav1.DeletionPropagation) (*Result, []error)
}

// InterfaceResources is introduced to avoid breaking backwards compatibility for Interface implementers.
//
// TODO Helm 4: Remove InterfaceResources and integrate its method(s) into the Interface.
type InterfaceResources interface {
    // Get details of deployed resources.
    // ...
    Get(resources ResourceList, related bool) (map[string][]runtime.Object, error)

    // BuildTable creates a resource list from a Reader. This differs from
    // ...
    BuildTable(reader io.Reader, validate bool) (ResourceList, error)
}
```

Use Modern Logging



The screenshot shows the Go Package Catalog page for the `slog` package. The URL is `pkg.go.dev`. The package is listed under the "Standard library" category. It shows the version `go1.23.2` as the latest, published on Oct 1, 2024, with a BSD-3-Clause license and 23 imports. The package is imported by 26,612 other packages. The details section includes checks for a valid `go.mod` file, a redistributable license, and a tagged version. The repository is `cs.opensource.google/go/go`. The documentation section is expanded, showing an overview with links to Levels, Groups, Contexts, Attrs and Values, and Customizing a type's logging behavior.

GO Search packages or symbols

Discover Packages > Standard library > log > slog

slog package standard library

Version: `go1.23.2` Latest | Published: Oct 1, 2024 | License: BSD-3-Clause | Imports: 23 | Imported by: 26,612

Details

- Valid `go.mod` file
- Redistributable license
- Tagged version
- Stable version
- Learn more about best practices

Repository `cs.opensource.google/go/go`

Links

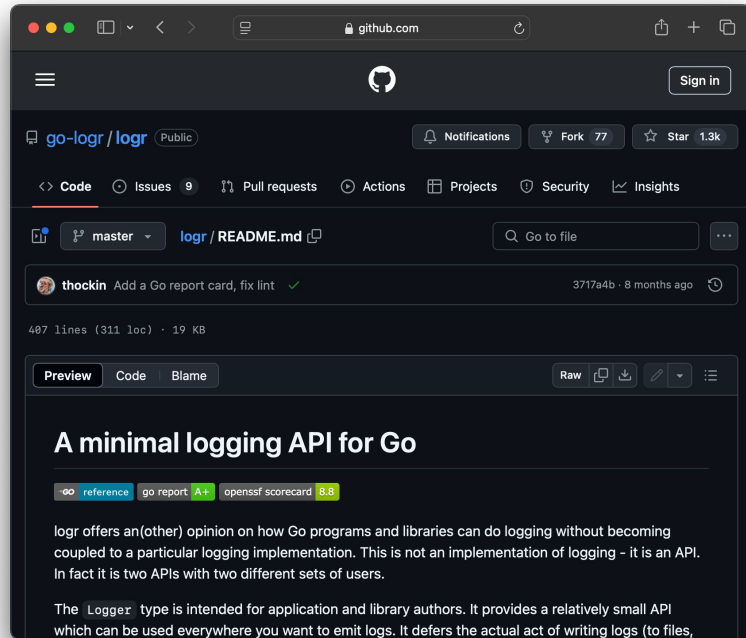
- Report a Vulnerability

Documentation

<> Documentation

Overview

- Levels
- Groups
- Contexts
- Attrs and Values
- Customizing a type's logging behavior



The screenshot shows the GitHub repository page for `go-logr/logr`. The repository is public and has 77 forks and 1.3k stars. It shows the `master` branch with the `logr / README.md` file selected. The README content is displayed, starting with "A minimal logging API for Go". It includes a Go reference badge, a Go report badge (A+), and an OpenSSF scorecard badge (5.8). The text describes the library's purpose: to provide a logging API without being coupled to a specific implementation. It mentions that the `Logger` type is intended for application and library authors and can be used everywhere to emit logs.

go-logr / logr Public

Notifications Fork 77 Star 1.3k

<> Code Issues 9 Pull requests Actions Projects Security Insights

master logr / README.md

thockin Add a Go report card, fix lint 3717a4b · 8 months ago

487 lines (311 loc) · 19 KB

Preview Code Blame Raw

A minimal logging API for Go

go reference go report A+ openssf scorecard 5.8

logr offers an(other) opinion on how Go programs and libraries can do logging without becoming coupled to a particular logging implementation. This is not an implementation of logging - it is an API. In fact it is two APIs with two different sets of users.

The `Logger` type is intended for application and library authors. It provides a relatively small API which can be used everywhere you want to emit logs. It defers the actual act of writing logs (to files,

Newer Signing

The screenshot shows the GitHub repository page for `sigstore/helm-sigstore`. The repository is public and has 13 forks and 60 stars. The main branch is `main`. The repository contains a table of files and folders, including `.github`, `cmd`, `pkg`, `scripts`, `tests`, `.gitignore`, `.golanci.yml`, `.goreleaser.yml`, `CODEOWNERS`, `CODE_OF_CONDUCT.md`, `CONTRIBUTORS.md`, `COPYRIGHT.txt`, and `LICENSE`. The right sidebar shows the repository's description, a list of files, and the latest release, `v0.3.0`, which was released on July 7.

Product Solutions Resources Open Source Enterprise Pricing

Search or jump to... Sign in Sign up

sigstore/helm-sigstore Public

Notifications Fork 13 Star 60

Code Issues 3 Pull requests Actions Security Insights

main 1 Branch 9 Tags

Go to file Code

dependabot[bot]	Bump anchore/sbom-action from 0.17.5 to 0.17.6 in the act...	b41ec07 - yesterday	281 Commits
.github	Bump anchore/sbom-action from 0.17.5 to 0.17.6 in the ac...	yesterday	
cmd	update go to 1.22.x (#239)	7 months ago	
pkg	added integration test and fixed verify (#209)	9 months ago	
scripts	Misc fixes (#276)	3 months ago	
tests	added integration test and fixed verify (#209)	9 months ago	
.gitignore	added integration test and fixed verify (#209)	9 months ago	
.golanci.yml	upgrade go to go1.21 and group dependabot updates (#1...	10 months ago	
.goreleaser.yml	update goreleaser version and golanci-lint (#273)	3 months ago	
CODEOWNERS	add artifact hub definition to claim the verified publisher (...)	3 years ago	
CODE_OF_CONDUCT.md	Project Setup	3 years ago	
CONTRIBUTORS.md	Project Setup	3 years ago	
COPYRIGHT.txt	Project Setup	3 years ago	
LICENSE	Initial commit	3 years ago	

About

Plugin for Helm to integrate the sigstore ecosystem

helm sigstore

Readme

Apache-2.0 license

Code of conduct

Security policy

Activity

Custom properties

60 stars

7 watching

13 forks

Report repository

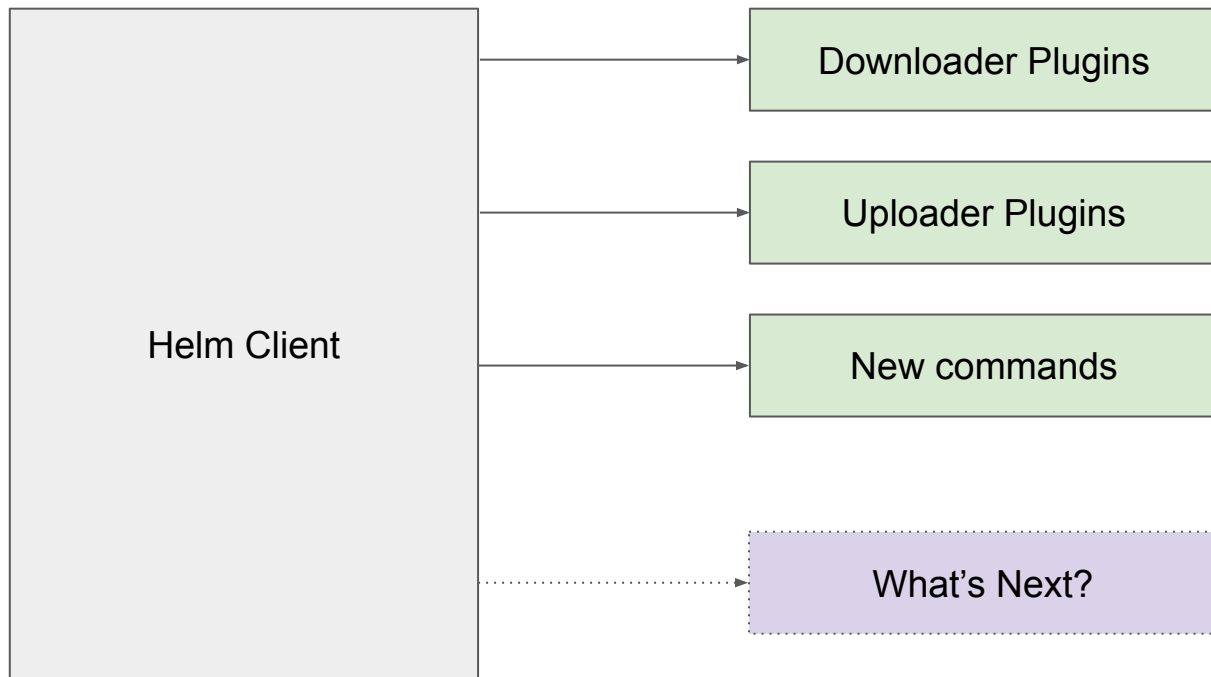
Releases 9

v0.3.0 Latest on Jul 7

+ 8 releases

Contributors 10

Extension Points (Plugins)





KubeCon



CloudNativeCon

North America 2024

What Would You Like?



KubeCon



CloudNativeCon

North America 2024

Getting Involved



Collaborate on Slack

Engage with other members of the Helm community



Create a H4HIP

A Helm Improvement Proposal (HIP) specifically focused on items related to Helm 4



Join the Community Meetings

Participate with other members of the Helm community to discuss efforts related to Helm 4



KubeCon



CloudNativeCon

North America 2024

Q&A

<https://sched.co/1hoxU>