



Distribution Survey Results

What we want to learn?



Real-world Keycloak deployments



Key user challenges



Currently supported deployments



Operator adoption & challenges



Alternative deployments

Characteristic



Start: **19th December**
End: **16th February**
~ 2 months



No anonymous users



27 questions
(multichoice, open)

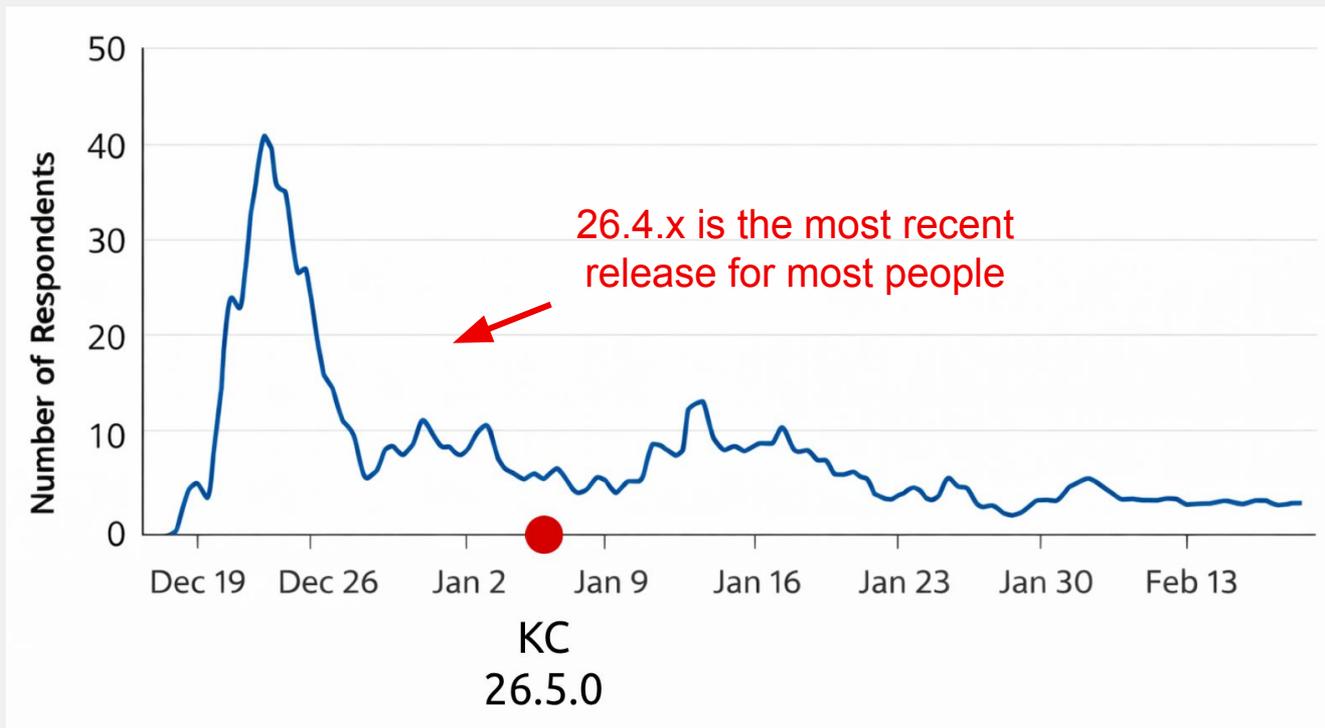


Keycloak blog
LinkedIn
CNCF Slack, ...

Responses



366 responses!



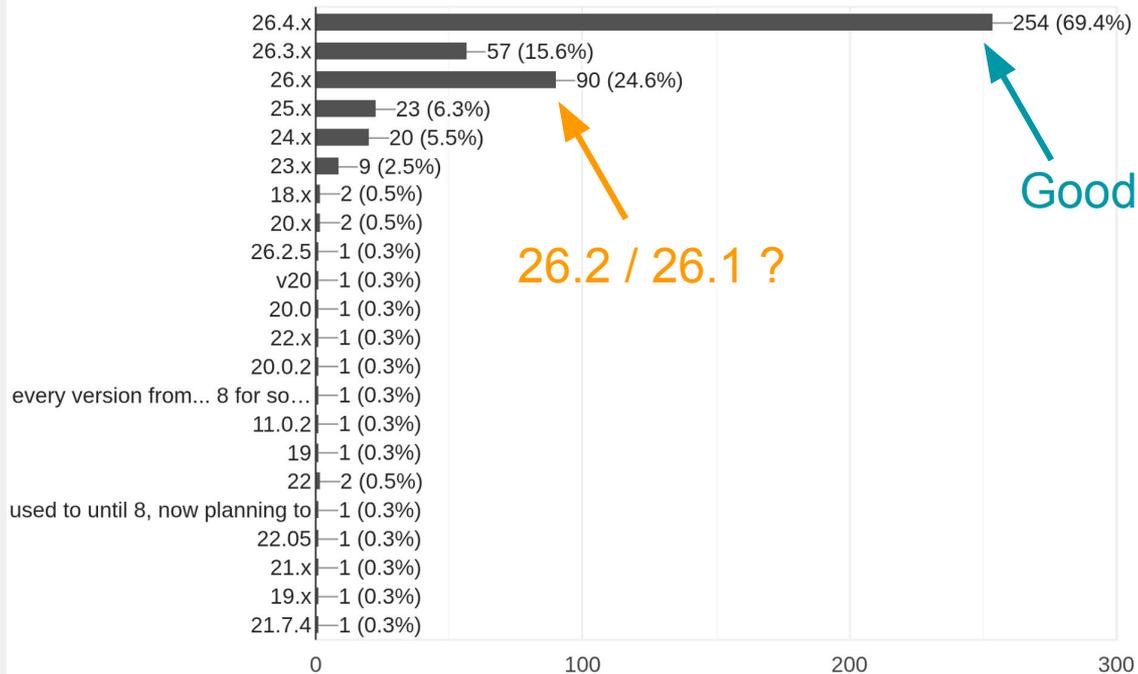
1

Real-world deployments

What Keycloak version are you currently running?

What versions of Keycloak are you currently running?

366 responses



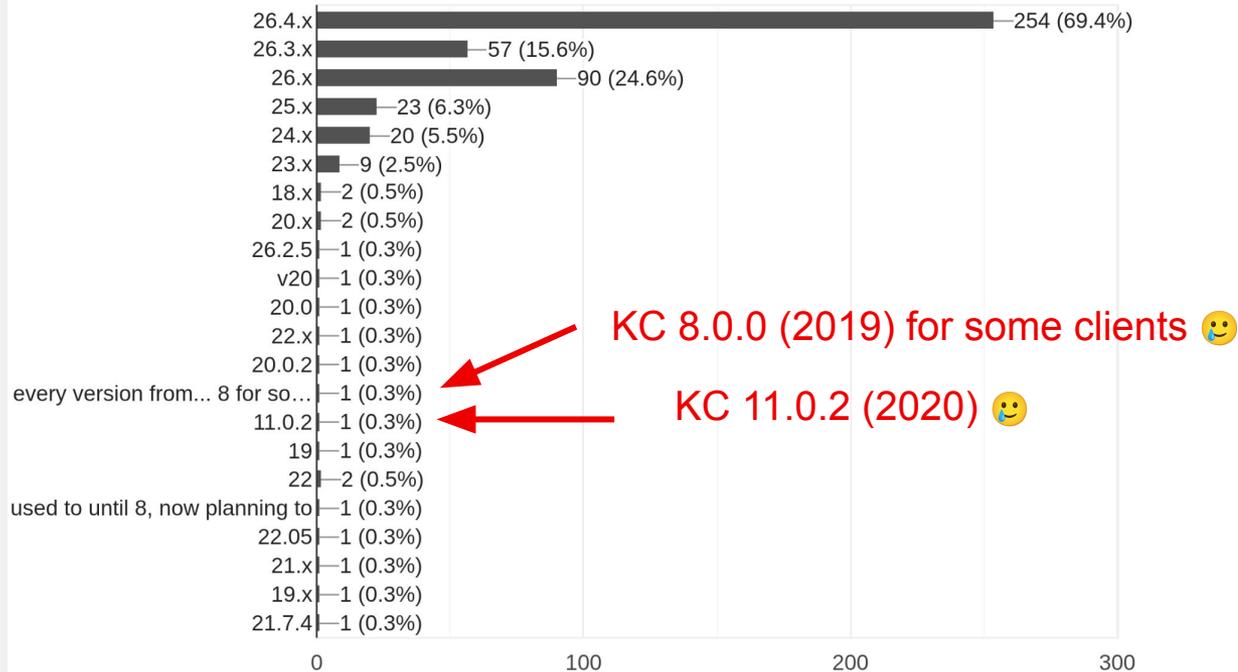
26.2 / 26.1 ?

Good

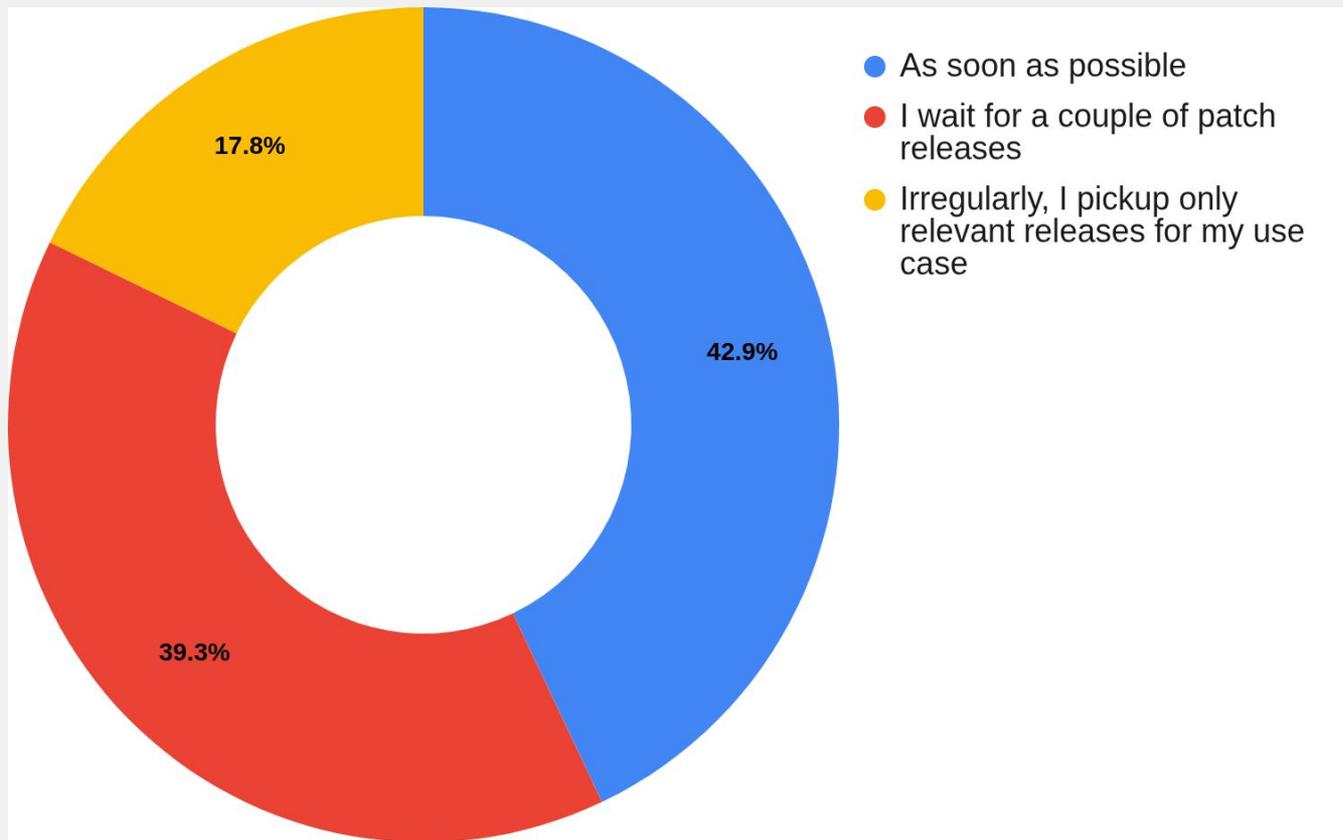
What Keycloak version are you currently running?

What versions of Keycloak are you currently running?

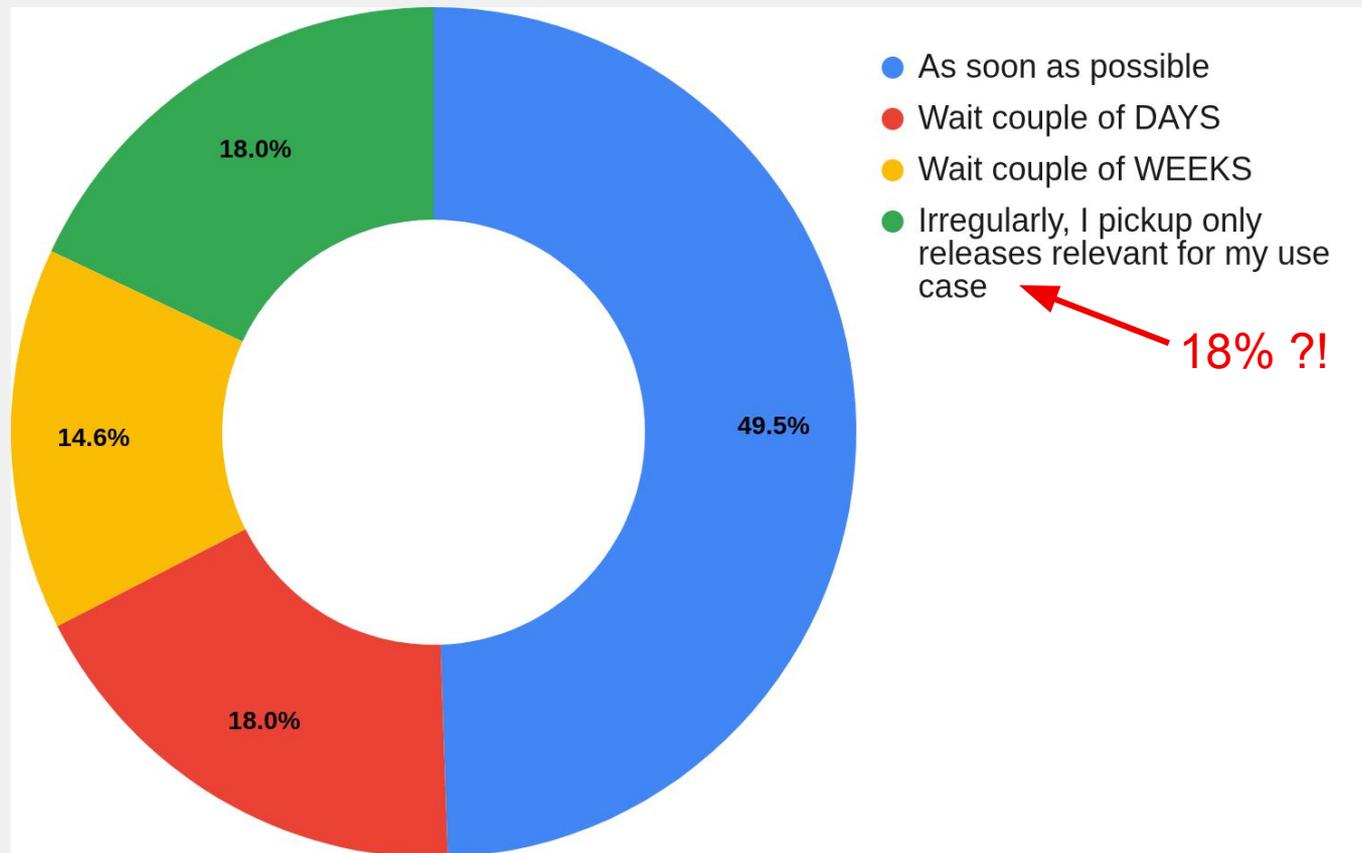
366 responses



How quickly do you adopt a new **MINOR** release?

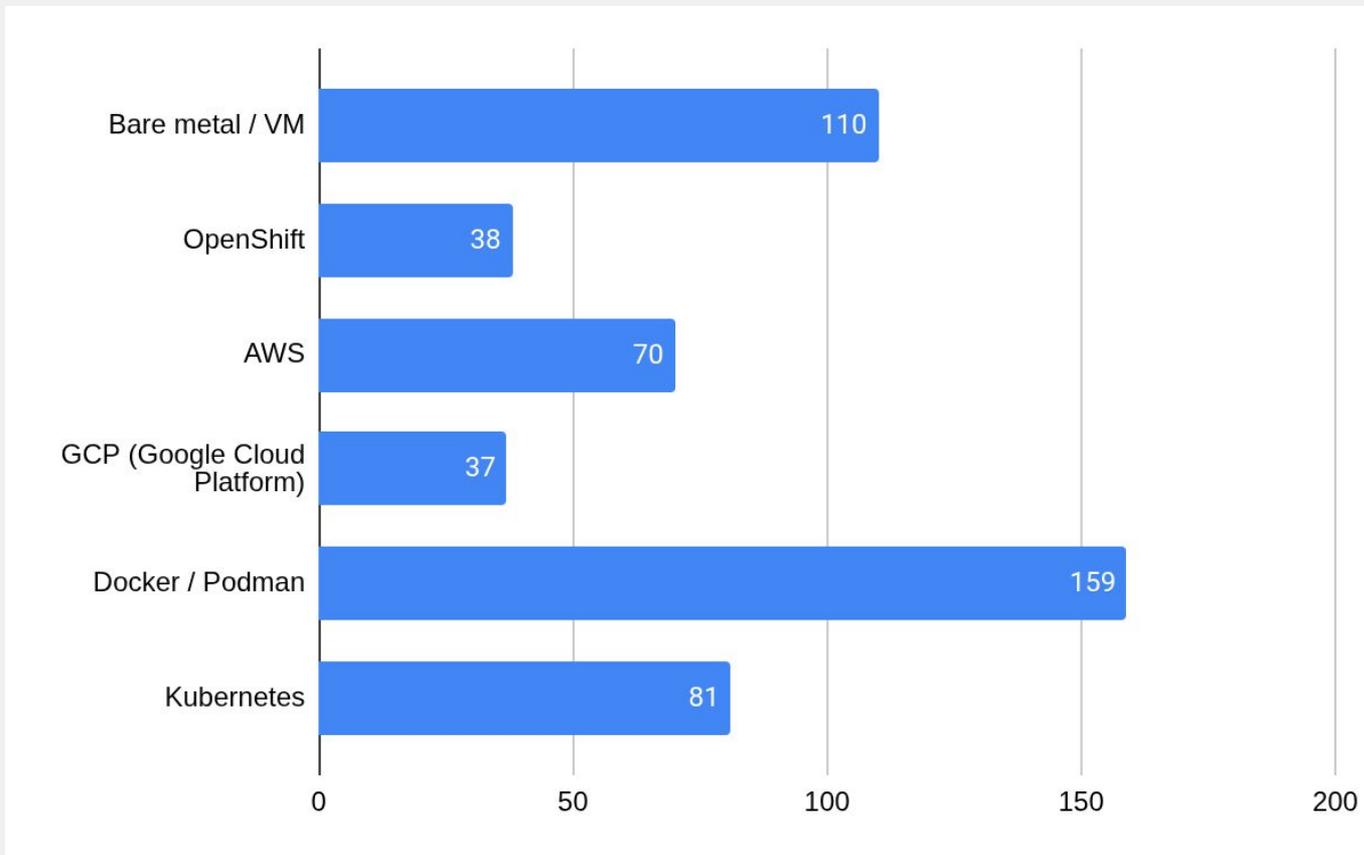


How quickly do you adopt a new **PATCH** release?



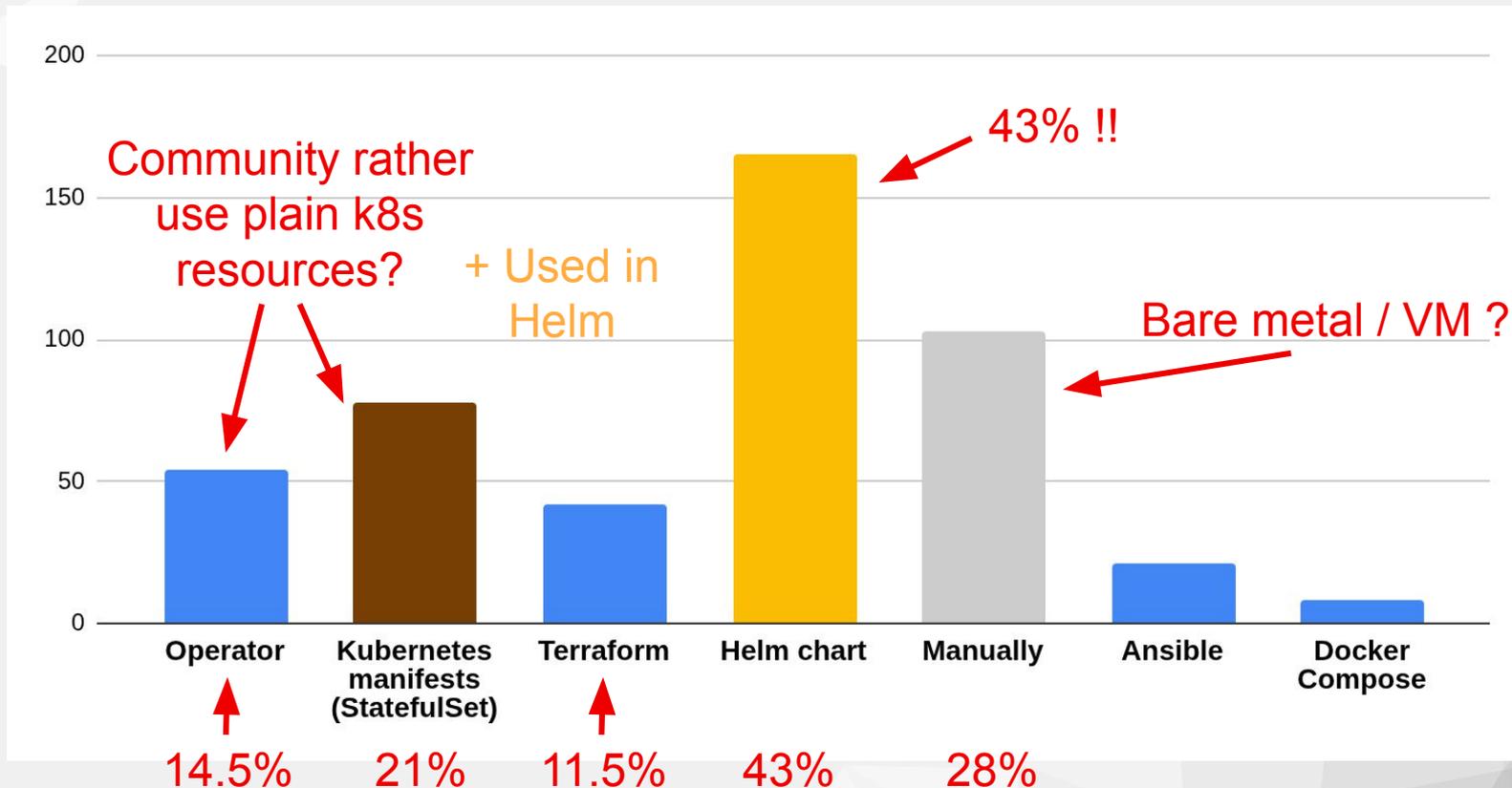
Where do you deploy Keycloak?

*multichoice



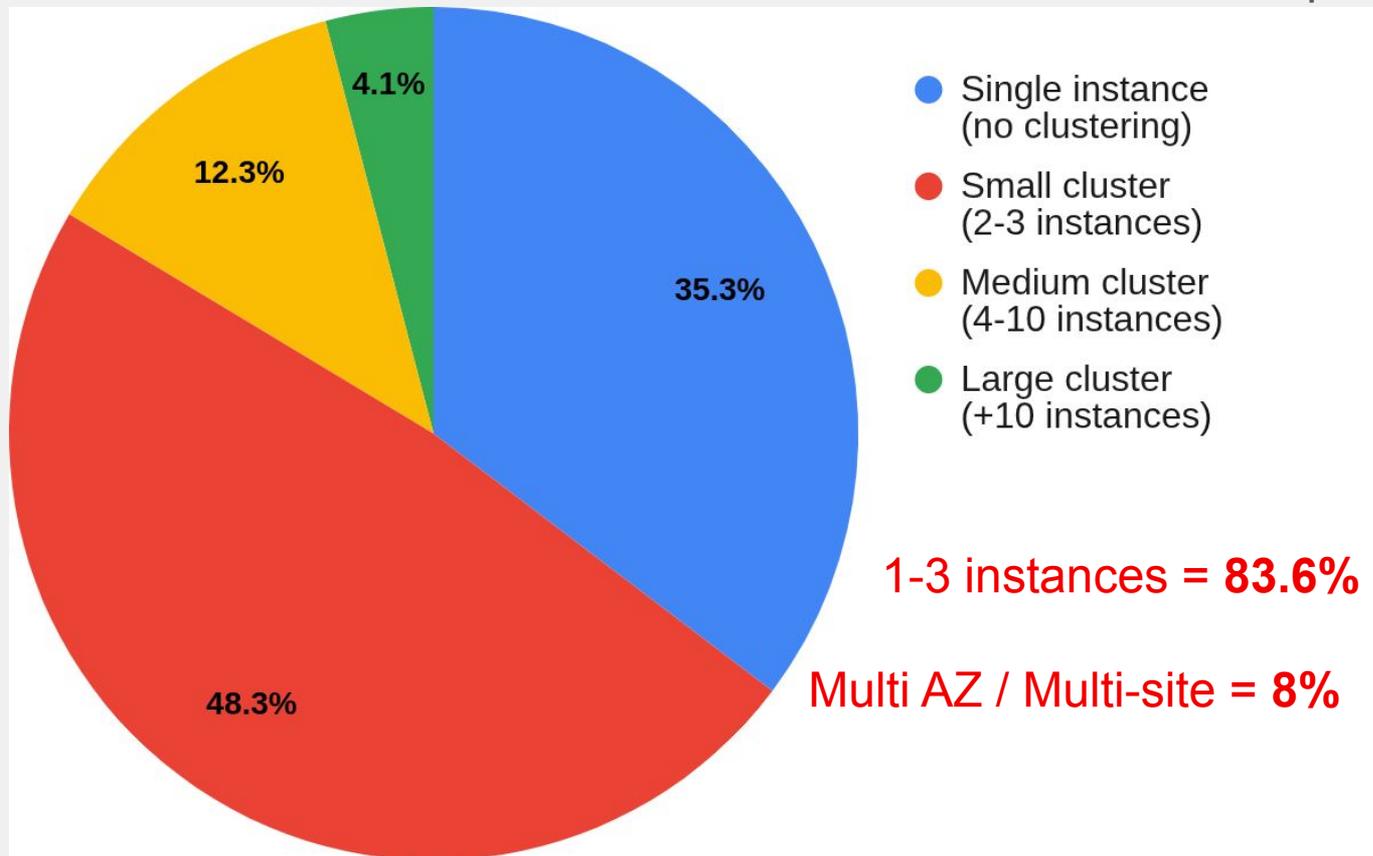
How / where

How do you **manage** your deployment?



What is the **size of your deployment?**

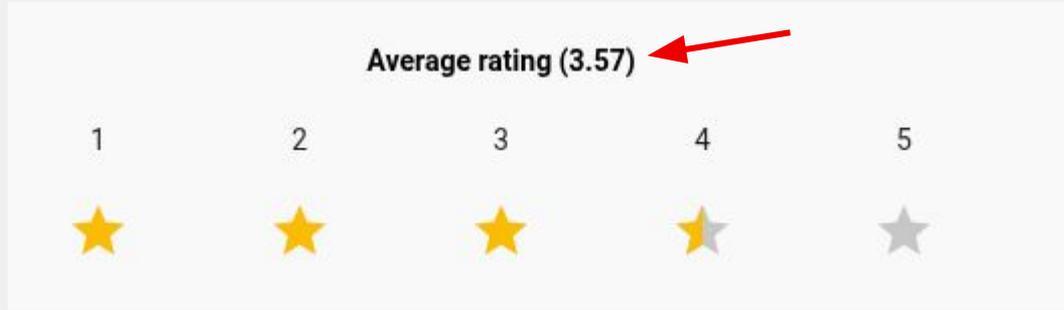
*instance/pod



2

Key user challenges

Rate the **ease of setup** and **initial configuration**



Would you go there?
Probably **NOT** 😜 

Sponsored :
Keycloak Config Restaurant
3.6  (366) ⓘ
Mediterranean restaurant · 9 One St.
Open · Closes 11pm
✓ Dine-in · ✓ Takeaway · ✓ Delivery



Rate the **ease of setup** and **initial configuration**

Average rating (3.57)

1



2



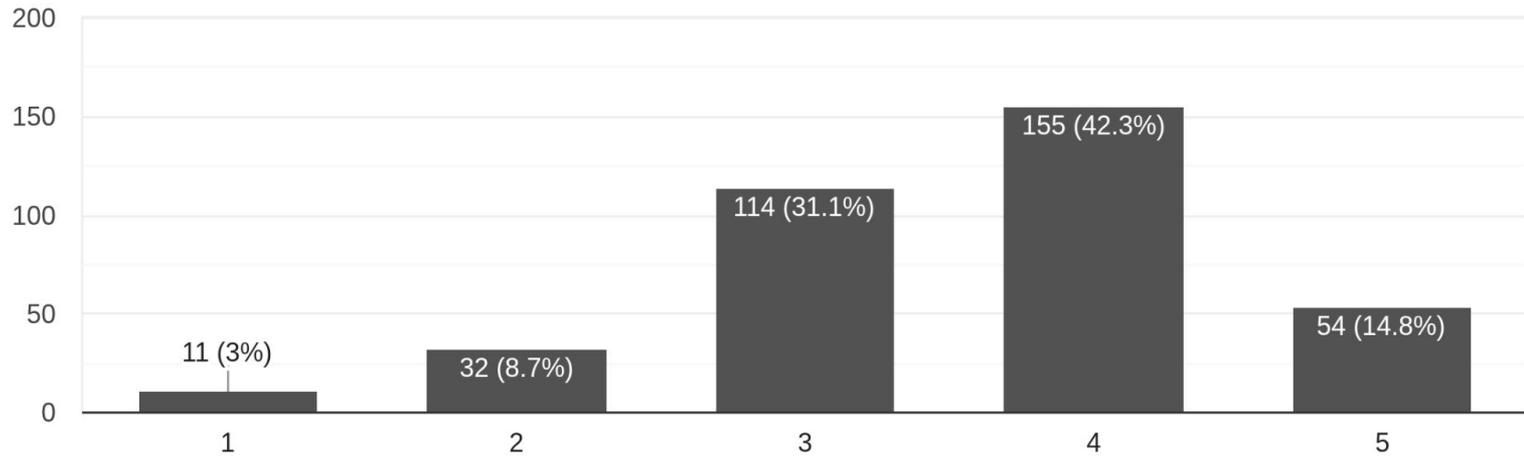
3



4



5



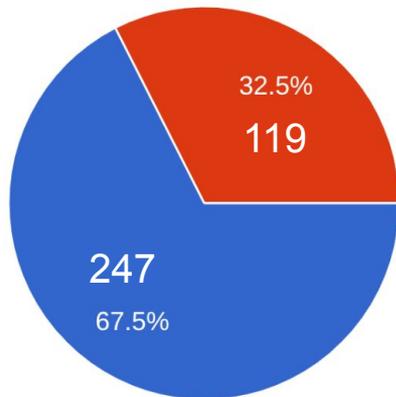
What are the **pain points of initial configuration**?

-  Understanding configuration options and defaults (52%)
-  Clustering / High-availability / performance tuning (35%)
-  Realms, clients and components setup (32%)
4. Custom Themes (31%)
5. Documentation is not clear (21%)
6. Tooling complexity (Operator, Kubernetes specifics) (20%)
7. Custom providers (19%)
8. TLS / certificates / HTTP configuration (18%)
9. Proxy / Load balancer (17%)
10. Integration with external IdPs (16%)
11. Database setup and Migration (13%)

Do you use the **official Keycloak container image**?

Do you use the official Keycloak container image?

366 responses



● Yes

● No

32.5% is a lot...

WHY?

What you're **missing from the official image?**

- Using the official builder image to have optimized image
- We don't use containers
- ...
- I'm using **custom SPIs**
- I'm using **custom themes**
- I'm using **Bitnami Helm chart** (and they use Bitnami image)

Majority of responses related to these 3

So, the only concern is with the Bitnami images, as they're attached to the Bitnami Helm chart

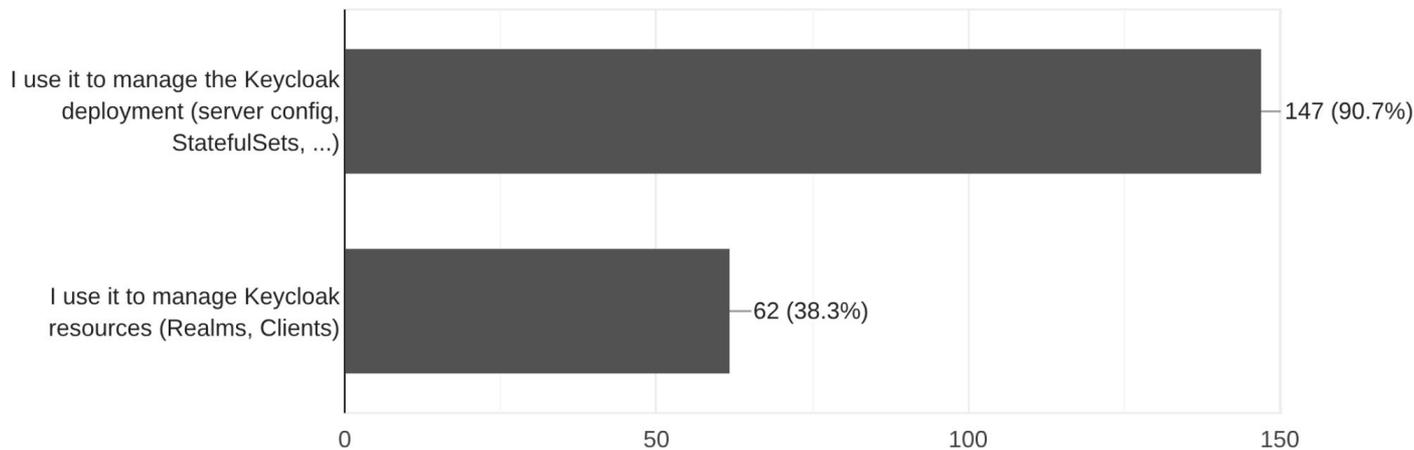
3

More **deployment** information

If you use **GitOps** based **workflows**, how do you **integrate Keycloak** with it?

If you use GitOps-based workflows (e.g. Argo CD), how do you integrate Keycloak with it?

162 responses



Can you share some **details about your deployment?**

Deployment Platforms and Tools

Mostly Kubernetes managed with GitOps like ArgoCD and FluxCD. Many use Helm charts (Bitnami, Codecentric, custom). Some are planning to the Operator, sometimes integrated into their own Helm charts.

Configuration management

Terraform provider is quite popular. Also the 'keycloak-config-cli' (adorsys) is frequently used.

Customization and Image build

Many builds optimized images with custom themes and SPIs. Often automated via CI/CD pipelines (e.g., Gitlab CI/CD, Bamboo)

Can you share some **details about your deployment?**

Migration and challenges

Many people migrating from the Bitnami Helm charts. Big desire for a better way to manage realm configuration - new Admin API, official feature-rich Operator/Helm.

Specific setups

From small to large-scale, multi cluster, HA deployments serving millions of users. Lack of clear documentation for setting up active-active HA.

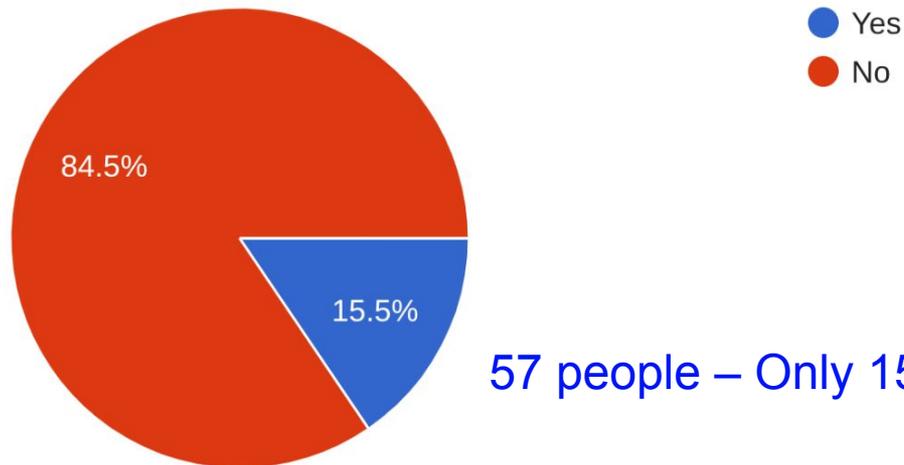
4

Operator Adoption & Challenges

Do you use the **Keycloak Operator**?

Do you use the Keycloak Operator?

368 responses



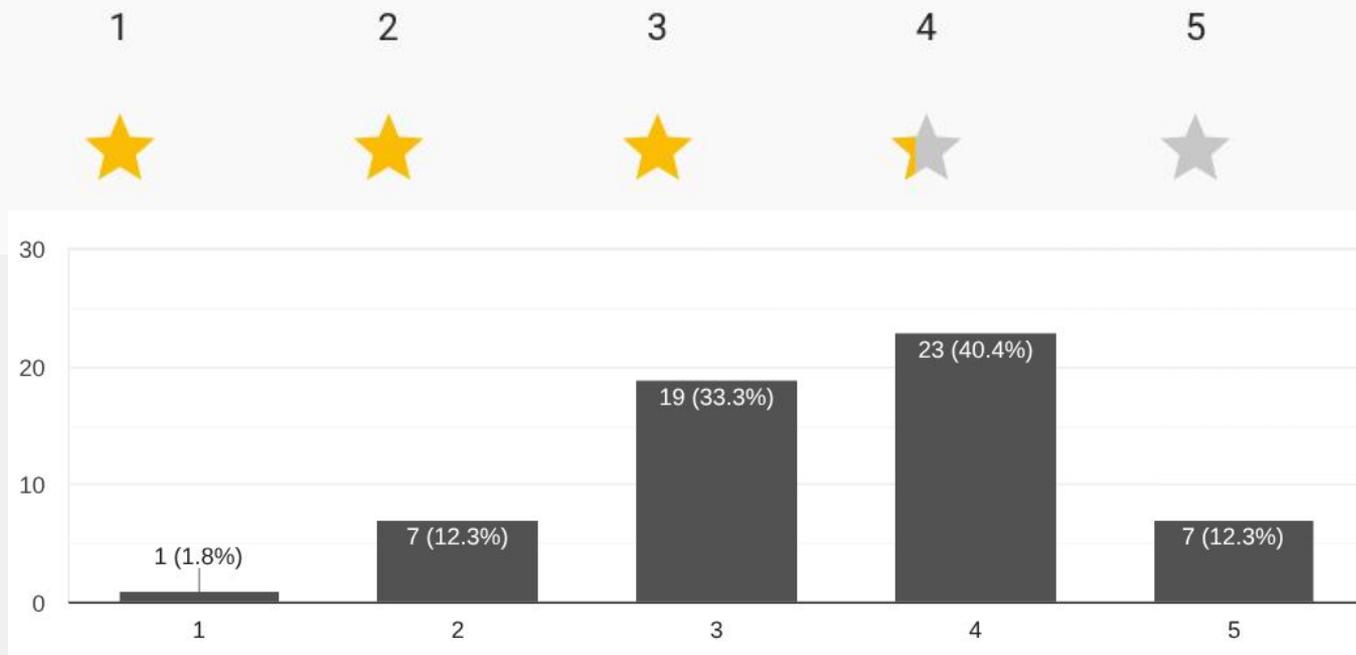
57 people – Only 15.5% **use it**

 **YES, I use Operator!**
section
(57 people - 15.5%)

Are you satisfied with the Operator?

Average rating (3.49)

Yep, not high score again :/





What **is good** about Keycloak Operator



Simplifies the initial deployment (79%)



Manages clustering, HA, StatefulSets automatically (60%)



Simplified upgrades of Keycloak (54%)

4. Integrates well with Kubernetes/OpenShift native features (52%)

5. Reduces manual configuration and operational errors (35%)

6. Handles Server and RealmImport configuration via CRs (19%)

✗ What **is bad** about Keycloak Operator



Limited documentation/examples for advanced usage (53%)



Missing CRs (Clients, Realms,...) (47%)



Too limited - we need to use unsupported approaches (35%)

4. Difficult installation of the Operator (32%)

5. Complexity in understanding CRs and parameters (30%)

6. Integration with GitOps tools (24%)

6. Debugging/troubleshooting can be challenging (24%)

8. Operator behaviour might be opaque/"magic" (21%)

8. Upgrades or migrations can be tricky (21%)

10. Not cluster wide (17%)

✘ What is **bad** about Keycloak Operator



Limited documentation/examples for advanced usage (53%)



Missing CRs (Clients, Realms,...)



Too limited - we need to use unsupported

Limited + hard to understand?!

4. **Difficult installation** of the Operator

5. **Complexity in understanding** CRs and para

6. Integration with GitOps tools

6. Debugging/troubleshooting can be **challenging** (24%)

8. Operator behaviour might be **opaque/"magic"** (21%)

8. Upgrades or migrations can be **tricky** (21%)

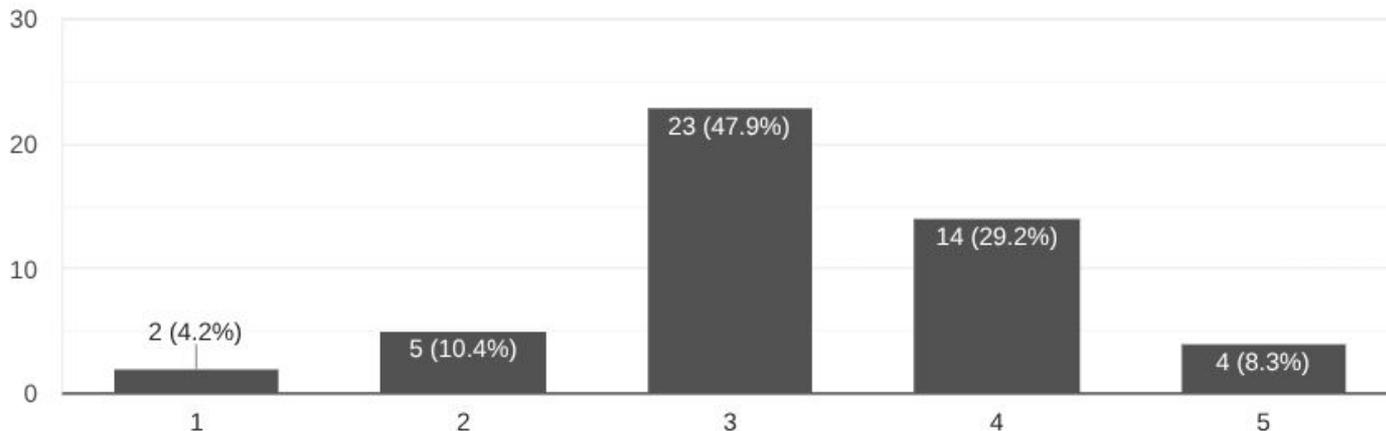
10. Not cluster wide (17%)

How well does the Operator fit into GitOps?

48 responses

Again nothing to brag about :/

Average rating (3.27)



✘ NO, I don't use Operator!
section
(313 people - 84.5%)

✘ What **is bad** about Keycloak Operator

- | | | |
|---|---|-------|
|  1 | Not using Kubernetes/OpenShift | (41%) |
|  2 | No added value in comparison to other methods | (25%) |
|  3 | Too complex for our use-case | (23%) |
| 4. | Difficult installation and/or requirements | (18%) |
| 5. | Prefer manual deployments | (14%) |
| 6. | Lack of documentation/examples | (13%) |
| 7. | Upgrade or version management concerns | (10%) |
| 8. | Too complex to understand | (9%) |

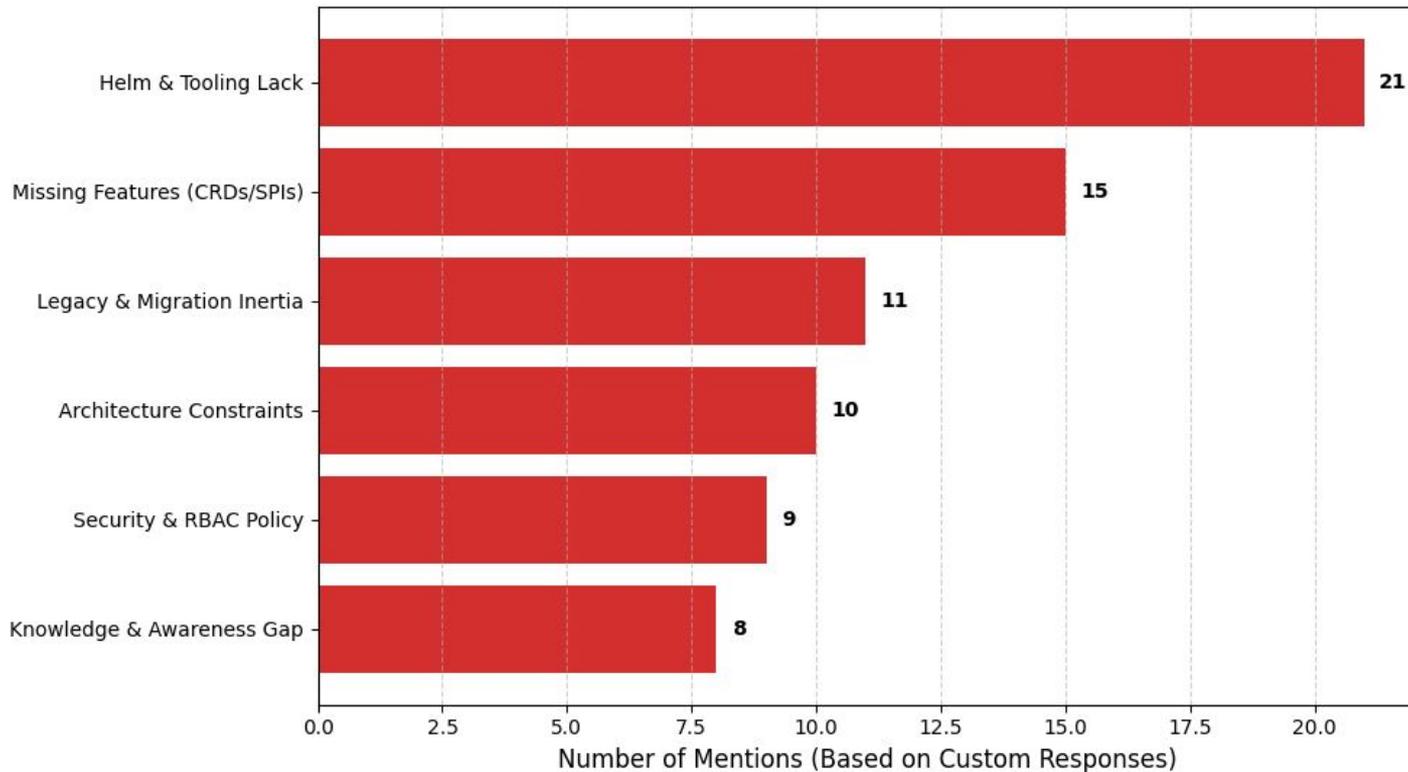
 What **is bad** about Keycloak Operator

155 custom responses!



What is **bad** about Keycloak Operator

Why Users Skip the Operator: Custom Barriers



What **is bad** about Keycloak Operator

#1 Helm & Tooling Lack

Users refuse to ditch Helm for a tool that doesn't fit their existing CI/CD automation.

#2 Missing Features

No declarative support for Realms or Clients makes the Operator feel "half-finished."

#3 Legacy & Migration inertia

"Historically grown" deployments work just fine; teams won't pay the high migration tax for no clear ROI (return of investment).

What is **bad** about Keycloak Operator

#4 Architecture Constraints

Namespace-scoping, waiting for OLMv1 to stabilize, we want Helm to manage everything,...

#5 Security & RBAC Policy

Strict enterprise rules block the ClusterRole permissions required to even install the Operator.

If the Keycloak Operator requires permissions to look at the whole cluster just to manage one instance, it gets rejected by security audits.

#6 Knowledge & Awareness Gap

Documentation explains the *how*, but fails to explain the "why" or the impact of migrating.

What would make you **reconsider to use Operator?**

- | | |
|---|-------|
|  1. Not sure / No opinion | (37%) |
|  2. Better documentation and examples | (26%) |
|  3. Stronger GitOps/automation support | (22%) |
| 4. Easier setup | (20%) |
| 5. More CRs | (19%) |
| 6. Organizational adoption of Kubernetes | (18%) |
| 7. Improved upgrade and version management | (10%) |

What would make you reconsider to use Operator?



Not sure / No opinion

(37%)

“No reason to use it for a few k8s resources”

Documentation and examples

(26%)

Tools/automation

“Cluster-wide”

(22%)

4. Easier

(20%)

5. More CRs

(10%)

6. Organization

Integration

7. Improved

Management

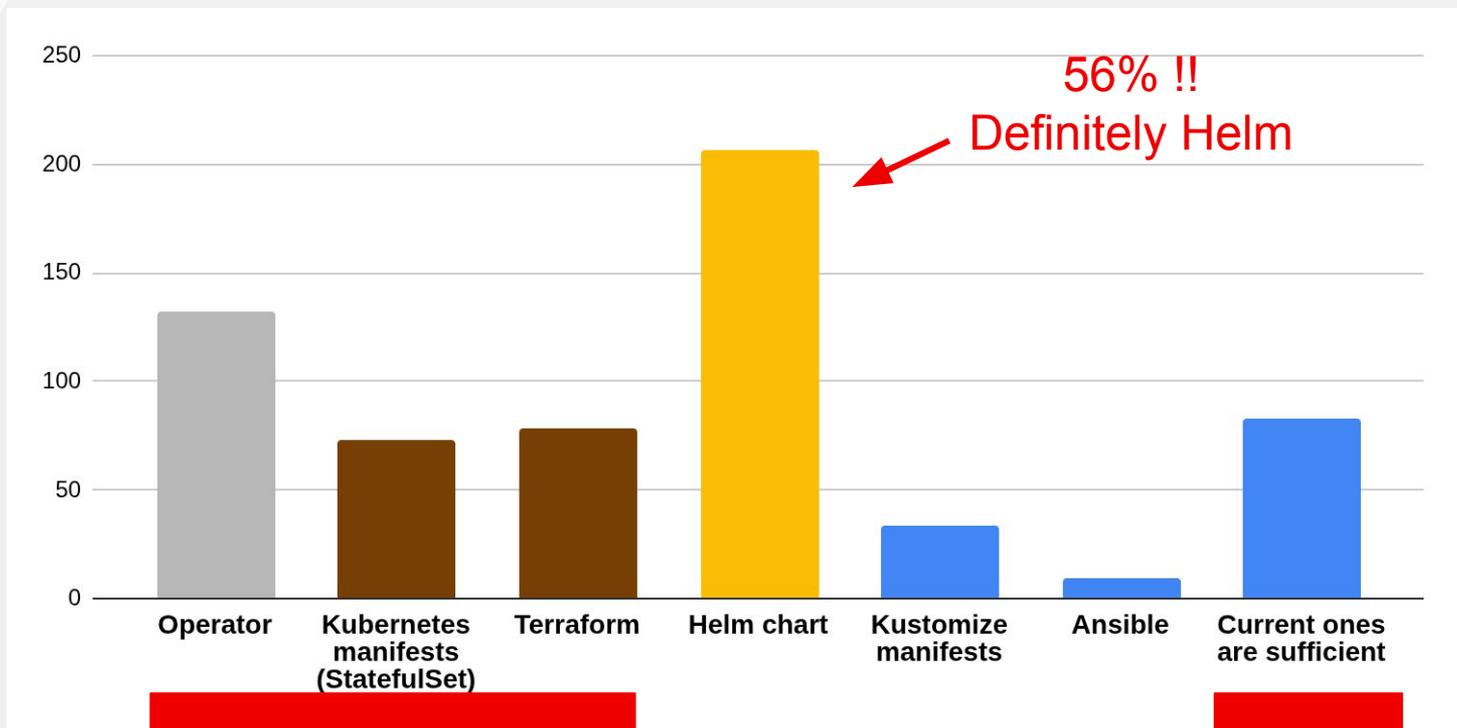
“At least deploy Operator via Helm charts”

“Why introduce additional complexity, if my setup works just fine?”

4

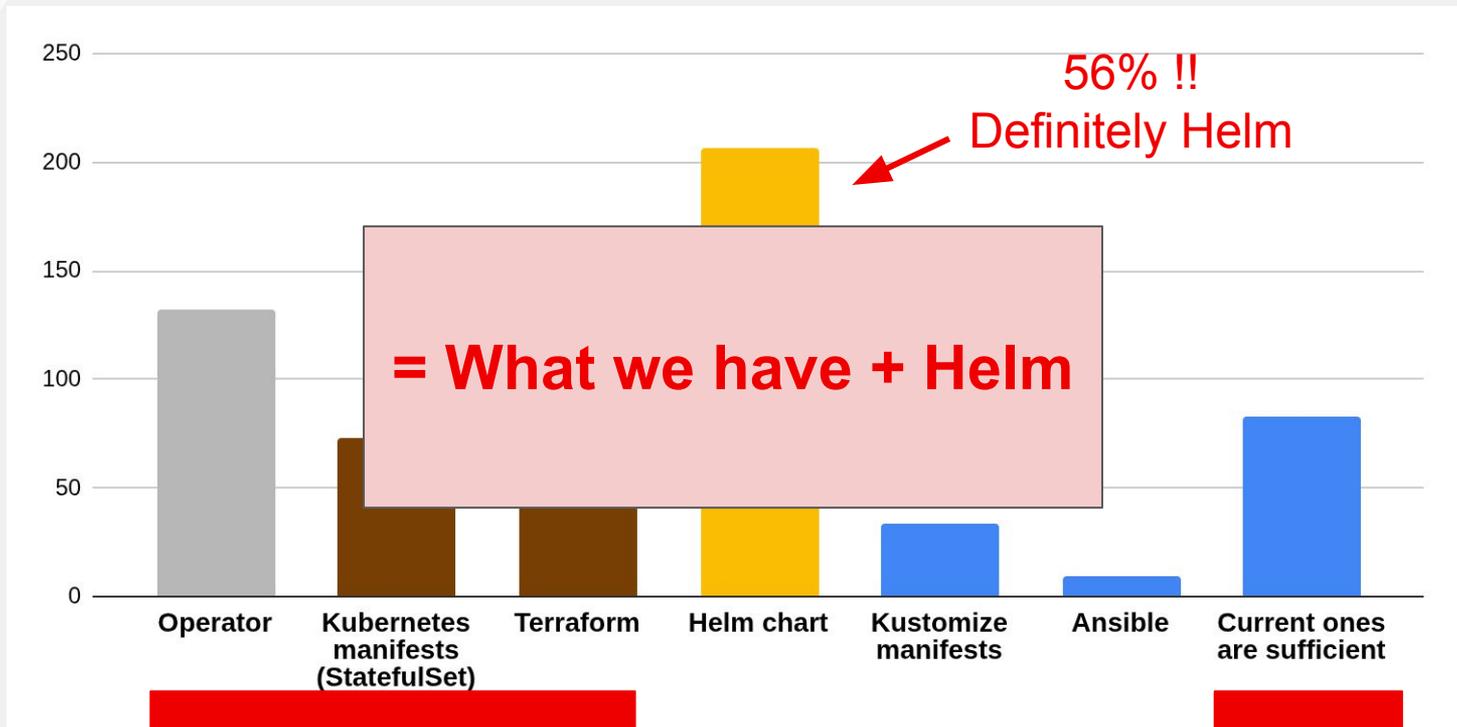
Alternative deployments

What would you wish to support?



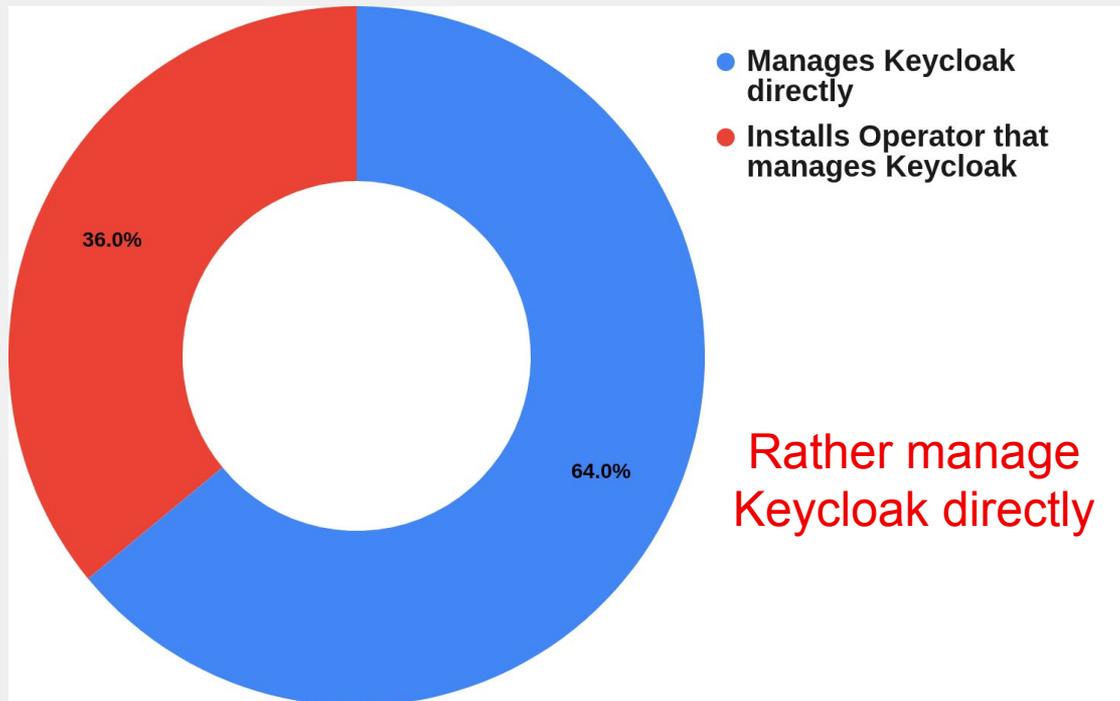
We support

What would you wish to support?



We support

If Helm charts were provided; how?



Single/multiple namespace/cluster?

*multichoice



Single namespace only

(31%)



Multiple namespaces in **multiple** clusters

(27%)



Not running Keycloak on Kubernetes

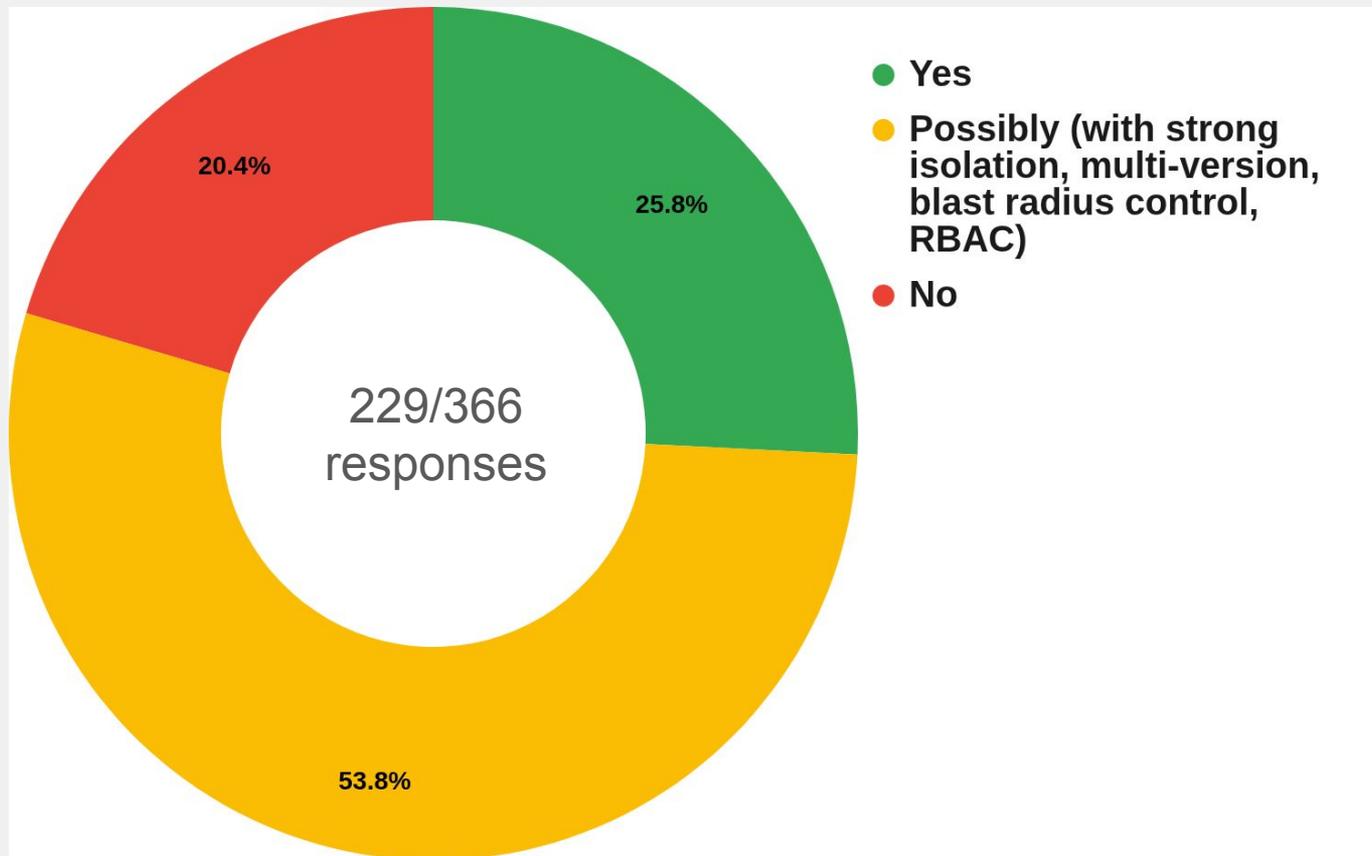
(26%)

4. **Multiple** namespaces in the **same** cluster

(22%)

Quite similar

Would a *cluster-wide* Operator be useful for you?



What would you **expect** from an ideal deployment?

-  **Zero-downtime** upgrades (73%)
-  **Easy** out-of-the-box **deployment** (65%)
-  Change **Server configuration** (53%)
4. Manage **Keycloak resources** (realms, clients, users,...) (49%)
5. **Back-up** functionality (45%)
6. Automatic **upgrades** (42%)
7. **Deploy DB** for Keycloak (17%)

What should be the **next priority for Keycloak?**

#1 Official Helm Charts

Helm Charts for Keycloak deployment, Operator, or both.

Many users migrate from Bitnami + provides easier GitOps and cluster management.

#2 Zero-downtime upgrades

Significant number of responses :)

#3 Infrastructure as Code (IaC) / GitOps

Better support for managing Keycloak resources (*Realms, Clients,..*) via IaC tools like Terraform, Operator CRs.

What should be the **next priority for Keycloak?**

#4 UX and Theming

Improvements to user-friendliness, better Admin APIs, and better built-in customization for theming (login and account are important).

#5 Organizations and SCIM

Complete the Organizations feature for better multi-tenancy + support SCIM protocol.

What would make **cloud-native Keycloak** great?

#1 Helm Charts

<we heard it all already :>

#2 High Availability and Upgrades

Zero-downtime upgrades, easier HA setup, geo-redundant deployments (active-active).

#3 Remove Infinispan/JGroups

Many respondents want to remove ISPN/JGroups to make Keycloak truly **stateless** - allow replacement with managed services like Redis.

What would make **cloud-native Keycloak** great?

#4 Documentation and UX

Better and more comprehensive documentation for REST APIs, clustering/HA without Kubernetes, or more real-life examples + Easier theme customization.

#5 Other topics

- Full import/export
- Declarative configuration
- Video tutorials
- Resource ownership
- Automatically send account activation emails

Any wishes? 🙏

Wishes

- Community really want **HELM charts** :)

“A Helm chart”

“Official Helm
chart please :)”

“Helm chart...”

“..., helm chart...”

Wishes

- Community really want **HELM charts** :)
- **Easier theming** for a simple customization

"Other than the
theme stuff,
keycloak is great!"

"themes were causing us
headaches"
(polite translation)

"Still need **great effort**
for customization for
end user"

Wishes

- Community really want **HELM charts** :)
- **Easier theming** for a simple customization
- **Feature requests**

“**Passkeys sync**
across realms”

“**Import/export**
authentication
flows”

“**LTS version**”

“**Complete**
workflows feature”

“**OTP by SMS or**
Email (disabled by
default)”

Wishes

- Community really want **HELM charts** :)
- **Easier theming** for a simple customization
- **Feature requests**
- **Documentation**

“Better
documentation and
samples also for
new technologies”

“More **docs**...”

“Making **operator docs**
more exhaustive”

Wishes

- Community really want **HELM charts** :)
- **Easier theming** for a simple customization
- **Feature requests**
- **Documentation**
- **Contributions**

“It’s quite hard to get anything reviewed in GitHub”

“Comprehensive ‘How do I get involved’ docs”

Community appreciate what we're doing

"Thanks folks!"

"Keycloak is a great addition to the open source world."

"Thank you for Keycloak's team contributions!"

"...tremendously powerful and flexible solution"

"Keycloak is a fantastic product..."

"Amazing product. Love you guys."

"Thanks for building this as open source!"

"I'm loving it"

"Thank you for the hard work!"

"Thanks for being open to the public community!"



"I love Keycloak!"

"Great product!"

"Thanks for this great software!"

"Let's make Keycloak rule the world :)"

"Thanks!"

"Keep up the great work!"

"We really like Keycloak!"

"It is great!"

Summary

Summary

- **Helm Charts!** (at least for installing Operator) - as people use it also for other services (also with ArgoCD)
- **Operator CRs** (Realms, Clients,...)
- **Stronger GitOps/automation**
- Not all people running on **Kubernetes**
- Not many people using **Operator**
- ~83% deployments run **1-3 instances**
- **Documentation!**
- **Themes customization**