

GETTING HELM TO BE ENTERPRISE READY

ABOUT ME

THAT'S WHAT I DO:
I DRINK AND
I KNOW THINGS.



@jbaruch

Ambassadors

Cloud Native Ambassadors (CNAs) are individuals who are passionate about Cloud Native Computing Foundation technology and projects, recognized for their expertise, and willing to help others learn about the framework and community.

Successful ambassadors are people such as bloggers, influencers, evangelists who are already engaged with a CNCF project in some way including contributing to forums, online groups, community events, etc.

The Cloud Native Ambassador program exists to empower community members with tools and resources needed to:

- Promote cloud native projects and technology
- Educate a local community on the CNCF mission
- Contribute to CNCF projects

Requirements for becoming an official Cloud Native Ambassador are:

- Affiliated with a CNCF member **OR** is a contributor to a CNCF project.
- Already engaged with the project in some way whether that's as a contributor, blogger, speaker, etc.
- Willing to either speak at community events **OR** write technical content such as blog posts.
- Host a cloud native community meetup. See <http://www.meetup.com/pro/cncf/> for areas already covered.

Are you interested in becoming an official Cloud Native Ambassador?

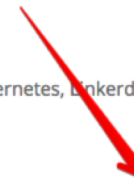
[APPLY NOW!](#)

**** We are currently looking for Ambassadors who organize/can organize Meetups around containerd, CoreDNS, Fluentd, gRPC, Kubernetes, Linkerd, OpenTracing, Prometheus, and/or rkt in countries/states not already represented.**



CLOUD NATIVE
 COMPUTING FOUNDATION
 AMBASSADOR

Weeeeee!!!



Animesh Singh
IBM

Animesh is an STSM and Lead for IBM Cloud Platform. He has been with IBM for more than 10 years and currently works with customers in designing cloud computing solutions on OpenStack, Cloud Foundry, Docker and Serverless Architecture, and is the lead for IBM Bluemix.

[Read More](#)


Ariel Jatib
StackPointCloud

Ariel is Principal Designer at StackPointCloud, Inc. He founded and organizes the New York, Seattle and San Francisco Kubernetes Meetups, regularly hosting the New York event.

[Read More](#)


Ayrat Khayretdinov
CloudOps

Ayrat is a Cloud Architect and Container Practice Lead at CloudOps.

[Read More](#)


Baruch Sadogursky
JFrog

Baruch Sadogursky (a.k.a JBaruch) is the Developer Advocate at JFrog. His passion is speaking about technology. Well, speaking in general, but doing it about technology makes him look smart, and 17 years of hi-tech experience sure helps

[Read More](#)


POLL TIME!

Let's calibrate the level first

WHAT THE HELM IS HELM?

Dependency manager for Kubernetes

HOW TO DEPLOY ANYTHING TO K8S

- Copy YAML
- Paste YAML
- Fix indents
- Repeat

KUBERNETES RESOURCE

```
{  
  "kind": "Deployment",  
  "apiVersion": "extensions/v1beta1",  
  "metadata": {  
    "name": "my-release-docker-app-chart"  
  },  
  "spec": {  
    "containers": [  
      {  
        "name": "docker-app-chart",  
        "image": "docker.artifactory/docker-app:1.0",
```

LET'S BUILD A NEW ONE!

```
> docker build -t docker.artifactory/docker-app:1.1
```

ONE LAST THING...

```
> sed -i.bak  
`s#docker.artifactory/docker-app:1.1#{imageTag}#`  
deployment.yaml
```



@jbaruch

OR JUST USE :LATEST

```
"image": "docker.artifactory/docker-app:latest"
```



ENTER HELM

ENCAPSULATED PACKAGES OF KUBERNETES DEPLOYMENTS

All this...

xrayxray-analysis

xray-event

xray-indexer

xray-nfs-server

xray-persist

Becomes this

xray

POWERFUL TEMPLATING FOR DESCRIPTOR FILES

```
{
  "kind": "Deployment",
  "apiVersion": "extensions/v1beta1",
  "metadata": {
    "name": "{{ template \"docker-app.fullname\" . }}"
  },
  "spec": {
    "containers": [
      {
        "name": "{{ template \"docker-app.name\" . }}",
        "image": "{{ .Values.image.repository }}:
                  {{ .Values.image.tag }}"
      }
    ]
  }
}
```

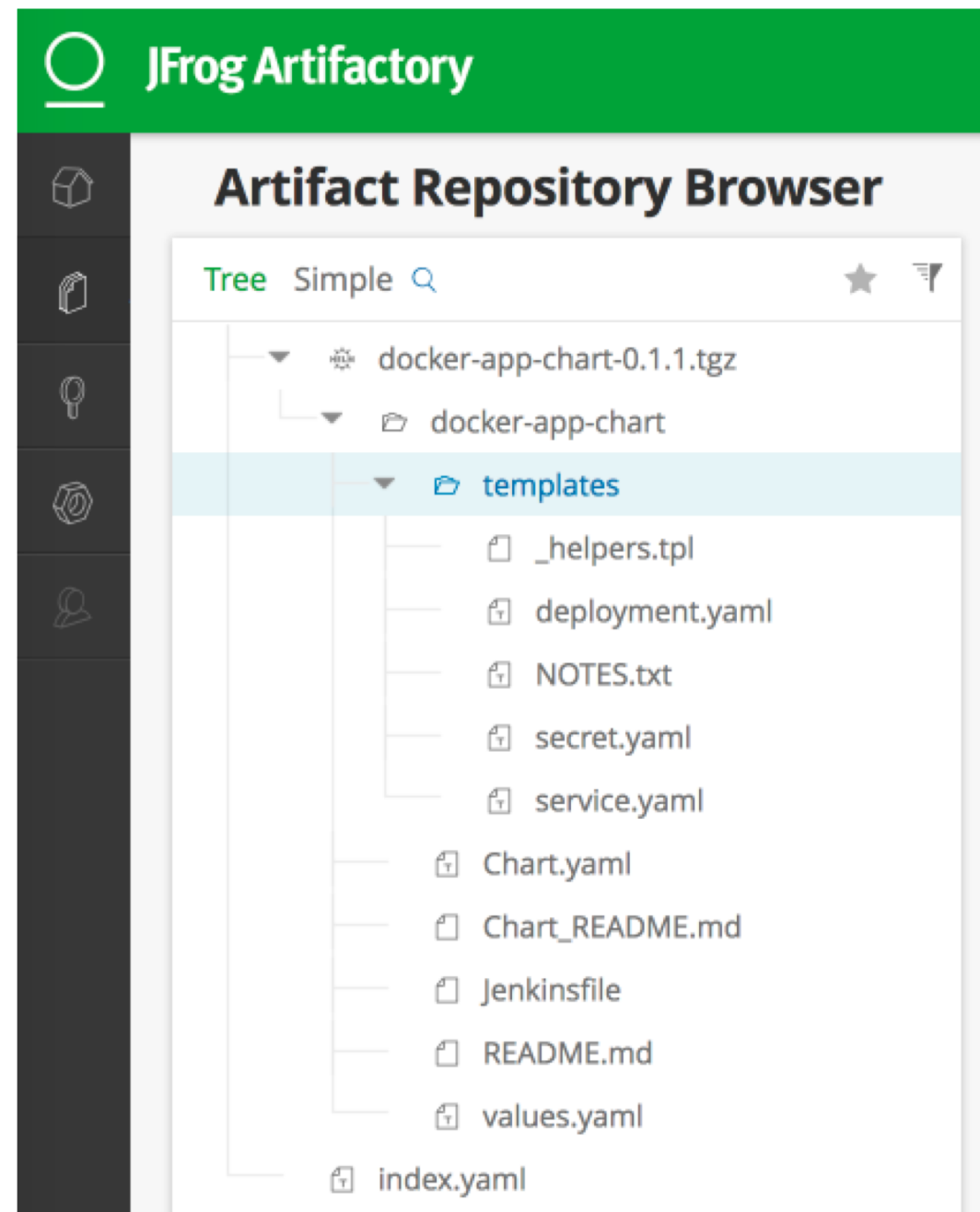
VALUES:

*# Default values for docker-app.
This is a YAML-formatted file.
Declare name/value pairs to be passed into your templates.*

image:
 repository: docker.artifactory/docker-app
 tag: 1.1
 secretName: regsecret
 pullPolicy: Always

SIMPLE!

- Templates
- Values
- Metadata

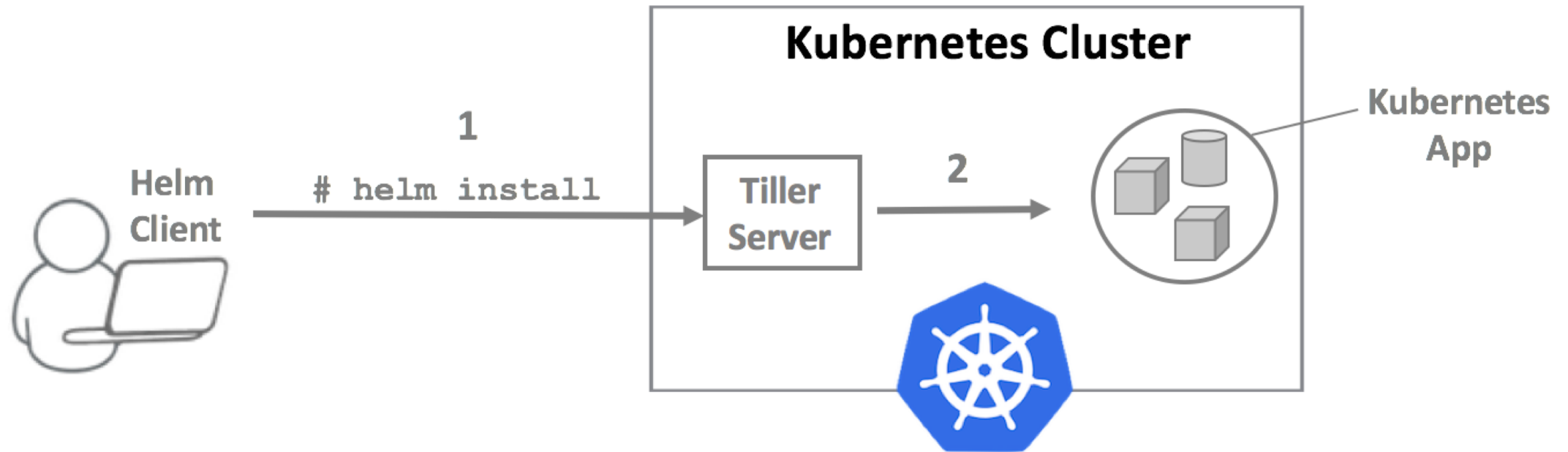


@jbaruch

CHART <=> IMAGE RELATIONSHIP

- Using templates we can reuse charts for multiple image versions
- Chart versions != Image versions

KUBERNETES CLUSTER CONTROL



TWO PARTS

Helm client

- Local chart development
- Managing repositories
- Interacting with the Tiller server

Tiller Server

- Listening for incoming requests from the Helm client
- Combining a chart and configuration to build a release
- Installing charts into Kubernetes, and then tracking the subsequent release
- Upgrading and uninstalling charts by interacting with Kubernetes

HELM COMMANDS

>helm init

>helm search

>helm install

>helm status

>helm repo

HELM REPOSITORIES

- Official repository - kubernetes.io/helm/

Discover & launch great Kubernetes-ready apps

Search charts...

159 charts ready to deploy



acs-engine-autoscaler

2.1.1 stable ☆ 0



aerospike

v3.14.1.2 stable ☆ 0

anchore

anchore-engine

0.1.6 stable ☆ 0



artifactory

5.8.3 stable ☆ 17



buildkite

3 stable ☆ 1



burrow

0.17.1 incubator ☆ 0



cassandra

incubator ☆ 3



centrifugo

1.7.3 stable ☆ 0

@jbaruch

HELM REPOSITORIES

- Official repository - kubernetes.io/helm/
- Get a local one!
- Option 1: Create your own:
 - Run an http server with index.yaml
 - Run `helm repo index` to generate one the index
- Option 2: Use JFrog Artifactory (or others)
 - Kubernetes Registry which supports Helm, containers, and everything inside them



SOFTWARE I LIKE



**SOFTWARE I
KNOW REALLY
WELL**

**WHAT DEPENDENCY MANAGERS AND
PRINTERS HAVE IN COMMON?**

Why I Believe

Printers



Were Sent From

Hell

To Make Us Miserable

By The Oatmeal <http://theoatmeal.com>

@jbaruch

So you want to write a package manager

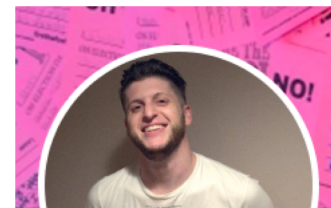
You woke up this morning, rolled out of bed, and thought, “Y’know what? I don’t have enough misery and suffering in my life. I know what to do—I’ll write a language package manager!”

...

Package management is awful, you should quit right now

Package management is a nasty domain. Really nasty. On the surface, it *seems* like a purely technical problem, amenable to purely technical solutions. And so, quite reasonably, people approach it that way. Over time, these folks move inexorably towards the conclusion that:

1. software is terrible
2. people are terrible
3. there are too many different scenarios
4. nothing will really work for sure
5. it’s provable that nothing will really work for sure
6. our lives are meaningless perturbations in a swirling vortex of chaos and entropy



sam boyer

@sdboyer

systems | people

📍 Detroit metro

📅 Joined December 2008

7 DEADLY SINS OF PACKAGE MANAGERS

1. Over-architecture
2. Not thinking of enterprise scenarios
3. Having downloadable index
4. Cross-site dependency resolution loopholes
5. Author authentication done wrong
6. Version management (or lack of thereof)
7. Using wrong place for central registry (and hardcoding it!)

HELM SUMMIT

SAVE YOUR SEAT

Join us for the inaugural Helm Summit in Portland, Oregon on February 21-22! Helm Summit provides an opportunity for new and existing users and contributors to Helm and Charts to share stories and best practices.

Day 1: All Things Helm - a wide range of in depth talks from community members for users and developers Evening: Happy Hour + Speed Networking event (details to follow)

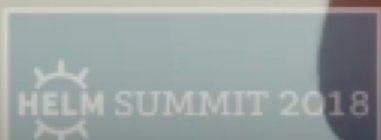


@jbaruch

Charts More People

ha – Senior Senior Developer

8



@jbaruch

7 DEADLY SINS OF PACKAGE MANAGERS

1. ~~Over-architecture~~
2. Not thinking of enterprise scenarios
3. Having downloadable index
4. ~~Cross-site dependency resolution loopholes~~
5. ~~Author authentication done wrong~~
6. ~~Version management (or lack of thereof)~~
7. ~~Using wrong place for central registry (and hardcoding it!)~~

ENTERPRISE SCENARIOS

- In-house registry
- User authentication and authorization
- org/project/team segmentation with central management
- Promotion pipelines



This repository

Search

Pull requests

Issues

Marketplace

Explore



kubernetes / helm

Watch

259

Star

4,848

Fork

1,703

Code

Issues

395



Pull requests

85



Projects

0



Wiki



Insights

Authentication support for remote charts repositories #3206



Merged

bacongobbler merged 33 commits into `kubernetes:master` from `eyalbe4:master` 14 days ago



Conversation

72



Commits

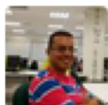
33



Files changed

25

+256 -108



eyalbe4 commented on Nov 27, 2017

Contributor



This pull request adds support for sending credentials (username and password) to remote charts repositories. This will allow vendors of charts repositories to restrict access to privileged users only. This new functionality is optional of course.

Please let us know what you think.
We'd be happy to add more commits as needed.
Thank you in advance.



19

Reviewers



bacongobbler



michelleN



seriousben



Assignees

No one assigned

Labels

enhancement

DOWNLOADABLE INDEX?

- Pros:
 - Simple server
 - Offline search
 - Offload the search

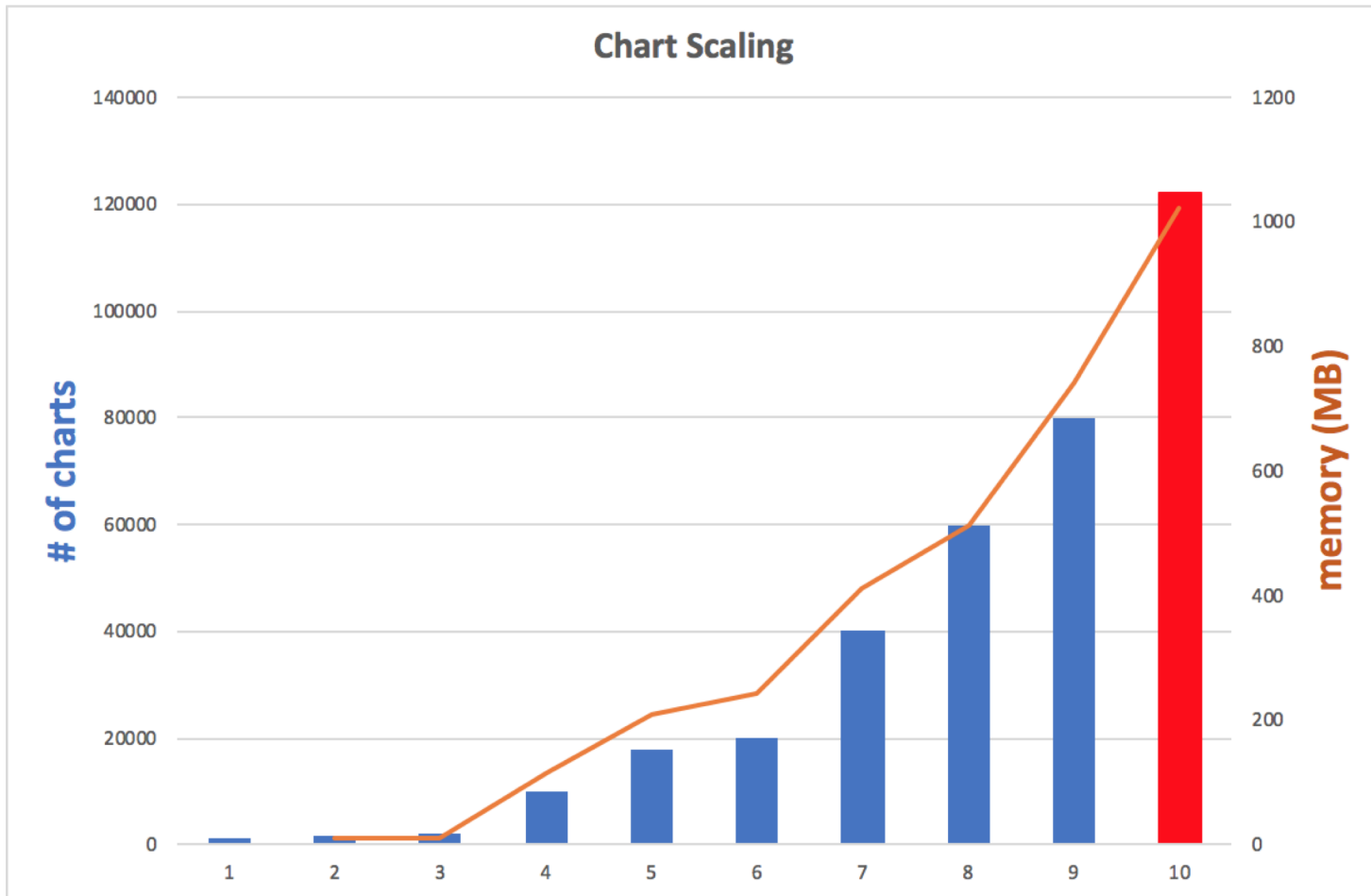


er power

DOWNLOADABLE INDEX?

- Pros:
 - Simple server
 - Offline search (not sure how it's useful?)
 - Offload the search off the server to spare computer power
- Cons:
 - Trends to become obsolete
 - Index is a bottleneck (specially if done wrong)

INDEX BOTTLENECKS



```
docker run -m=512m -it alpine-helm
```

```
helm repo add helm-prod $helm-repo
```


HOW CAN WE FIX IT?

- gzip index in transit
 - Fixes **some** of the issues
- Distribute the index

artifactory:

Entry per app

```
- apiVersion: v1
  appVersion: 5.8.3
  created: 2018-01-29T21:34:25.891969615Z
  description: Universal Repository Manager supporting all major packaging formats,
    build tools and CI servers.
  digest: c66011c6837d1fc27ffe261e8e0205188a398570b30cc61f32dcfcfb4f9f19a6
  home: https://www.jfrog.com/artifactory/
  icon: https://raw.githubusercontent.com/JFrogDev/artifactory-dcos/master/images/jfrog_med.png
  keywords:
  - artifactory
  - jfrog
  maintainers:
  - email: jainishs@jfrog.com
    name: jainishshah17
  - email: eldada@jfrog.com
    name: eldada
  name: artifactory
  sources:
  - https://bintray.com/jfrog/product/JFrog-Artifactory-Pro/view
  - https://github.com/JFrogDev
  urls:
  - https://kubernetes-charts.storage.googleapis.com/artifactory-6.2.5.tgz
  version: 6.2.5
```

Entry per version

```
- apiVersion: v1
  appVersion: 5.6.2
  created: 2017-12-18T20:03:55.777635493Z
  description: Universal Repository Manager supporting all major packaging formats,
    build tools and CI servers.
```

DIVIDE AND CONQUER!

- Main index: list of apps (with latest version)
 - artifactory:5.8.3
- App index: list of versions (and app-level metadata)
 - description
 - maintainers
 - keywords
 - sources
- Version index: the details of the version
 - appVersion
 - created
 - digest
 - url

STRUCTURED INDEX REQUIRES STRUCTURED REPOSITORY

- Layout!
- Repo
 - App
 - ver1
 - ver2
- This complicates the push!
 - We used to just upload the file!
 - See debs (or don't)

LET'S TALK ABOUT PUSH

Pushing Charts to a Repository

TODO: notes on usage of `helm push`

Several registries have emerged with support for Helm charts including JFrog Artifactory, ChartMuseum and Quay (via Application Registries) along with extras built on top of other systems, such as object storage (e.g., see the [S3 Plugin](#)). These registries are asking for and could benefit from a capability to push packages from the Helm client.

To support registries there will be a `helm push` command that works in a similar manner to `helm fetch`. It will support different systems via the URI/URL scheme and provide a default implementation for HTTP(S). Other systems, such as S3, can implement an uploader via a plugin for schemes Helm does not provide an implementation for.

JFROG SWAMPUP

May 16-18

64
Days

13
Hours

23
Minutes

46
Seconds

SWAMPUP 2018 PROMISES TO BE THE MOST
MEMORABLE EVENT YOU ATTEND IN 2018.

Keynotes, training sessions and technical workshops will
take you to the edge of DevOps.

[GET TICKETS](#)[WATCH 2017 RECAP](#)

SWAMPUP CNCF CONTENT:

- Openshift and K8S
- Rancher and K8S
- JFrog's lessons learned with K8S
- Azure, K8S and Helm
- AWS, Serverless and K8S (keynote)
- CI/CD with Helm
- etc...