CREATE & DISTRIBUTE
A PLUGIN FOR KUBERNETES.
EASY!

Aurélie Vache - @aurelievache / Gaëlle Acas - @Gaelleacas
Aurélie Vache
@aurelievache
Cloud Engineer at Stack Labs
Google Developer Expert on Google Cloud
Conferences organizer
Duchess France Leader & Mentor
Technical articles & sketchnotes writer
... & Retrogaming lover
GAËLLE ACAS
@Gaelleacas
SRE at Stack Labs
CNCF Nantes
Co-organizer
Technical articles writer

Cloud addict

<Mum who code>
<Kids coding>

SPEAKER

Sketchnote reviewer
Why this talk?

- $ kubectl CLI is awesome but not exhaustive
- Painful to extend kubernetes/kubectl CLI
- Easy to create your own plugins
- Possible to spread your plugin worldwide!
Kubectl > pronunciation fight

Disclaimer

Pronounce “kubectl” as you want :-)
Kubectl > Kubernetes tool/cli
Kubectl > a very logical CLI!

I want the list of Pods in my-namespace namespace

$ kubectl get pods -n my-namespace
Kubectl > a very logical CLI!

I want to scale a deployment `my-deploy` to 5 replicas

```bash
$ kubectl scale deploy my-deploy --replicas=5
```
**Kubectl > a very logical CLI!**

$ kubectl get pods -n my-namespace
$ kubectl describe pod your-pod
$ kubectl logs my-pod
$ kubectl delete super-pod
$ kubectl create ns my-ns
$ kubectl run tmp --image=my-img:tag
$ kubectl exec a-pod -it -- sh
Kubectl > cool, but can I extend it?

```
$ kubectl get pods,deploy
NAME                   READY STATUS    AGE
pod/my-pod-c9cb9cbf6-r2gnc 1/1 Running 2d6h
pod/my-pod-75897978cd-f28j9 1/1 Running 3d1h
```
Kubectl plugins
**Kubectl: tied to core release cycle**

Wants to implement a new feature in Kubectl?

So … Forget about it and think about plugins instead ;-)
Plugins > K[tool] or Too many tools?

Too many opinions about one need
Plugins > K[tool] or Too many tools?

Not easily understandable

ksort 😞
kubectl sort-manifest 😎

No common way to install them

apt-get install
brew install

And… it’s not easy to know them!
Kubernetes, an extensible world
Plugins > Why it’s useful?

Extend `kubectl` with extra functionality that still feels like `kubectl`

- e.g. `kubectl service-catalog [...]`

Encapsulate repetitive workflow

- e.g. checking a secret resource, check logs
Plugins > language agnostic!
Kubectl plugin > How it works?

1. Create a file named kubectl-mypugin

$ vi kubectl-mypugin
**Kubectl plugin > How it works?**

1. Create a file named `kubectl-myplugin`
2. Make it executable

```bash
$ chmod +x kubectl-myplugin
```
Kubectl plugin > How it works?

1. Create a file named kubectl-myplugin
2. Make it executable
3. Put it in your PATH

$ cp kubectl-myplugin /usr/local/bin/kubectl-myplugin
Kubectl plugin > How it works?

1. Create a file named `kubectl-myplugin`
2. Make it executable
3. Put it in your PATH
4. Run "`kubectl myplugin`"

https://kubernetes.io/docs/tasks/extend-kubectl/kubectl-plugins/
Let's create a plugin!
What do we want?

- Display seasonal emojis in “kubectl get” command
- In a simple and easy plugin
- In BASH
Let’s do it, in LIVE!
Our plugin > steps

1. Create the script

$ vi kubectl-season
Our plugin > steps

2. Add executable permission

$ chmod +x kubectl-season
Our plugin > steps

3. Initialize our script

#!/bin/bash
Our plugin > steps

4. Define a list of emojis

```bash
NOW=`date +%m`
case $NOW in
  01|02|03)
    EMOJIS=(😭😭😭)
  ;;
  04|05)
    EMOJIS=(😭😭😭)
  ;;
  #...
  12)
    EMOJIS=(😭😭😭)
  ;;
esac
```
Our plugin > steps

5. Get a randomized emoji

SELECTED_EMOJI=${EMOJIS[$( ( RANDOM % ${#EMOJIS[@]} ) )]};
Our plugin > steps

6. Define & display an usage/help

```bash
# Usage
n="\n"
usage="Display seasonal emoji randomly just before resources name.$n$Usage:$n kubectl season KIND$n$Examples:$n kubectl season pod$n kubectl season pod,svc$n kubectl season deploy -A"

# MAIN
# Usage
if [[ "$1" == "" || "$1" == "-h" || "$1" == "--help" ]]
then
echo "$usage"
exit 1
fi
```
Our plugin > steps

7. Execute `kubectl get` command & add seasonal emojis

```bash
# add an emoji at the beginning of the line except for blank line and line beginning by NAME
kubectl get "$@" | sed -E "/^\(NAME\|\$/ ! s/^/\{SELECTED_EMOJI\} /"
```

execute `kubectl` get command
Our plugin > steps

7. Execute `kubectl get` command & add seasonal emojis

```
# add an emoji at the beginning of the line except for blank line and line beginning by NAME
kubectl get "@$" | sed -E "^/NAME/ ! s/\/^$\{SELECTED_EMOJI\}/"
```

add randomized emoji at the beginning of the line
Our plugin > steps

7. Execute kubectl get command & add seasonal emojis

```bash
# add an emoji at the beginning of the line except for blank line and line beginning by NAME
kubectl get "$@" | sed -E "/^\$(NAME|\$)/ ! s/^\$/\${SELECTED_EMOJI} /"
```

except for line starting by NAME & blank line
Our plugin > final! (1/2)

```bash
#!/bin/bash

NOW=`date +%m`

case $NOW in
  01|02|03)
    EMOJIS=(京津
    04|05)
    EMOJIS=(京津
    06|07|08)
    EMOJIS=(京津
    09)
    EMOJIS=(京津
    10)
    EMOJIS=(京津
    11)
    EMOJIS=(京津
    12)
    EMOJIS=(京津
esac
```

https://github.com/scraly/kubectl-season/
Our plugin > final! (2/2)

```bash
SELECTED_EMOJI=${EMOJIS[$(( RANDOM % ${#EMOJIS[@]} ) ))]};

## Usage
n="'\n' usage="Display seasonal emoji randomly just before resources name.$(n)$(n)Usage:$n  kubectl season KIND$(n)$(n)Examples:$n  kubectl season pod$(n)  kubectl season pod,svc$(n)  kubectl season deploy -A"

# MAIN
if [[ "$1" == "" || "$1" == "-h" || "$1" == "--help" ]];
then
  echo "$usage"
  exit 1
fi

# add an emoji at the beginning of the line except for blank line and line beginning by NAME
kubectl get "$@" | sed -E "/^\(NAME\.cloudflare\)/ ! s/\/^$\{SELECTED_EMOJI\}/"
```

https://github.com/scraly/kubectl-season/
Our plugin > in kubectl?

Cool... but, how can I add it in `kubectl` command?
Our plugin in kubectl!

```bash
$ cp kubectl-season /usr/local/bin/kubectl-season
```
Our plugin > results

$ kubectl season -h
Display seasonal emoji randomly just before resources name.

Usage:
  kubectl season KIND

Examples:
  kubectl season pod
  kubectl season pod,svc
  kubectl season deploy -A
<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>pod</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3h10m</td>
</tr>
<tr>
<td>a-pod-f9fd979d6-6tnbs</td>
<td>1/1</td>
<td>Running</td>
<td>2</td>
<td>26d</td>
</tr>
</tbody>
</table>

```
$ kubectl season pod
```
Our plugin > results

```
$ kubectl season pod,deploy

<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>pod/pod</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3h12m</td>
</tr>
<tr>
<td>pod/a-pod-f9fd979d6-6tnbs</td>
<td>1/1</td>
<td>Running</td>
<td>2</td>
<td>26d</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>UP-TO-DATE</th>
<th>AVAILABLE</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>deployment.apps/my-deploy</td>
<td>2/2</td>
<td>2</td>
<td>2</td>
<td>26d</td>
</tr>
<tr>
<td>deployment.apps/a-deploy</td>
<td>1/1</td>
<td>1</td>
<td>1</td>
<td>26d</td>
</tr>
</tbody>
</table>
```
Our plugin > how to share it?

Top!
But I want to share my awesome plugin!
What is krew?

- A package manager for kubectl plugins
  - Like “brew” but for kubectl :)
  - Developed summer of 2018 by Google intern Luk Burchard (@lbb)

- Krew is open sourced, try it:
  https://github.com/kubernetes-sigs/krew
Krew > useful for users

- Discover kubectl plugins
- Easy way to install plugins
- Update easily plugins
Krew > Useful for developers

- Make their plugins discoverable
- Easy way to package their plugins for multiplatforms (Win, Linux, MacOS)
Krew

$ kubectl krew -h
krew is the kubectl plugin manager.
You can invoke krew through kubectl: "kubectl krew [command]..."

Usage:
kubectl krew [command]

Available Commands:
  help  Help about any command
  index Manage custom plugin indexes
  info  Show information about an available plugin
  install Install kubectl plugins
  list  List installed kubectl plugins
  search Discover kubectl plugins
  uninstall Uninstall plugins
  update Update the local copy of the plugin index
  upgrade Upgrade installed plugins to newer versions
  version Show krew version and diagnostics

Flags:
  -h, --help help for krew
  -v, --v Level number for the log level verbosity

Use "kubectl krew [command] --help" for more information about a command.
<table>
<thead>
<tr>
<th>NAME</th>
<th>DESCRIPTION</th>
<th>INSTALLED</th>
</tr>
</thead>
<tbody>
<tr>
<td>access-matrix</td>
<td>Show an RBAC access matrix for server resources</td>
<td>yes</td>
</tr>
<tr>
<td>advise-psp</td>
<td>Suggests PodSecurityPolicies for cluster.</td>
<td>no</td>
</tr>
<tr>
<td>apparmor-manager</td>
<td>Manage AppArmor profiles for cluster.</td>
<td>no</td>
</tr>
<tr>
<td>auth-proxy</td>
<td>Authentication proxy to a pod or service</td>
<td>no</td>
</tr>
<tr>
<td>bd-xray</td>
<td>Run Black Duck Image Scans</td>
<td>no</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>view-utilization</td>
<td>Shows cluster cpu and memory utilization</td>
<td>yes</td>
</tr>
<tr>
<td>virt</td>
<td>Control KubeVirt virtual machines using virtctl</td>
<td>no</td>
</tr>
<tr>
<td>warp</td>
<td>Sync and execute local files in Pod</td>
<td>no</td>
</tr>
<tr>
<td>who-can</td>
<td>Shows who has RBAC permissions to access Kubernetes</td>
<td>no</td>
</tr>
<tr>
<td>whoami</td>
<td>Show the subject that is currently authenticated</td>
<td>no</td>
</tr>
<tr>
<td>PLUGIN</td>
<td>VERSION</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>access-matrix</td>
<td>v0.4.5</td>
<td></td>
</tr>
<tr>
<td>graph</td>
<td>v0.1.1</td>
<td></td>
</tr>
<tr>
<td>krew</td>
<td>v0.4.0</td>
<td></td>
</tr>
<tr>
<td>tree</td>
<td>v0.4.0</td>
<td></td>
</tr>
<tr>
<td>view-secret</td>
<td>v0.6.0</td>
<td></td>
</tr>
<tr>
<td>view-utilization</td>
<td>v0.3.3</td>
<td></td>
</tr>
</tbody>
</table>
Krew > install

$ kubectl krew install view-utilization
Krew > install not published plugins

Your plugin is not already published/available on krew index?

Not problem, you can also install it through Krew!

```bash
$ kubectl krew install --manifest=.krew.yaml --archive=dist/kubectl-mypugin_v0.0.1.tar.gz
```
KreW > Install not published plugins

Your plugin is not already published/available on krew index?

Not problem, you can also install it through Krew!

$ kubectl krew install --manifest-url=<URL>
krew > Local Architecture

```
$ tree ~/.krew
/Users/gacas/.krew
    bin
    ├── kubectl-krew -> /Users/gacas/.krew/store/krew/v0.4.0/krew
    │          └── kubectl-neat -> /Users/gacas/.krew/store/neat/v2.0.1/kubectl-neat
    └── index
        └── default
            └── plugins.md
    receipts
    ├── krew.yaml
    │    └── neat.yaml
    └── store
        ├── krew
        │    └── neat
```
Publishing on Krew

1. Develop a plugin
2. Create a Release
3. Write a Krew plugin manifest
4. Make a Pull Request to Krew index
apiVersion: krew.googlecontainertools.github.com/v1alpha2
kind: Plugin
metadata:
  name: season
spec:
  homepage: https://github.com/scraly/kubectl-season
  shortDescription: Display seasonal emoji randomly before resources name
  version: v0.0.4
  description: |
    This plugin allows you to add randomly a seasonal (halloween, christmas, easter...) emoji just before the resources name. It's a special seasonal mode.
  platforms:
  - selector:
    matchExpressions:
    - key: os
      operator: In
      values:
      - darwin
      - linux
    uri: https://github.com/scraly/kubectl-season/archive/v0.0.4.tar.gz
    sha256: 7331819379ca14da988d2537904381b6186c989d527719cb1e0fd212293d749e
    bin: kubectl-season-0.0.4/kubectl-season
apiVersion: krew.googlecontainertools.github.com/v1alpha2
kind: Plugin
metadata:
  name: season
spec:
  homepage: https://github.com/scraly/kubectl-season
  shortDescription: Display seasonal emoji randomly before resources name
  version: v0.0.4
  description:
    This plugin allows you to add randomly a seasonal (halloween, christmas, easter...) emoji just before the resources name. It's a special seasonal mode.
  platforms:
    - selector:
        matchExpressions:
          - key: os
            operator: In
            values:
              - darwin
              - linux
  uri: https://github.com/scraly/kubectl-season/archive/v0.0.4.tar.gz
  sha256: 7331819379ca14da988d2537904381b6186c989d527719cb1e0fd212293d749e
  bin: kubectl-season-0.0.4/kubectl-season
kubectl-season/.krew.yaml

apiVersion: krew.googlecontainer.tools.github.com/v1alpha2
kind: Plugin
metadata:
  name: season
spec:
  homepage: https://github.com/scraly/kubectl-season
  shortDescription: Display seasonal emoji randomly before resources name
  version: v0.0.4
  description: |
    This plugin allows you to add randomly a seasonal (halloween, christmas, easter...) emoji just before the resources name. It's a special seasonal mode.
  platforms:
    - selector:
      matchExpressions:
        - key: os
          operator: In
          values:
          - darwin
          - linux
      uri: https://github.com/scraly/kubectl-season/archive/v0.0.4.tar.gz
      sha256: 7331819379ca14da988d2537904381b6186c989d527719cb1e0fd212293d749e
      bin: kubectl-season-0.0.4/kubectl-season
---

**kubectl-season/.krew.yaml**

```yaml
apiVersion: krew.googlecontainertools.github.com/v1alpha2
kind: Plugin
metadata:
  name: season
spec:
  homepage: https://github.com/scraly/kubectl-season
  shortDescription: Display seasonal emoji randomly before resources name
  version: v0.0.4
  description: |
    This plugin allows you to add randomly a seasonal (halloween, christmas, easter...) emoji just before the resources name. It's a special seasonal mode.
  platforms:
    - selector:
        matchExpressions:
          - key: os
            operator: In
            values:
              - darwin
              - linux
  uri: https://github.com/scraly/kubectl-season/archive/v0.0.4.tar.gz
  sha256: 7331819379ca14da988d2537904381b6186c989d527719cb1e0fd212293d749e
  bin: kubectl-season-0.0.4/kubectl-season
```

---

Platforms[1]
**kubectl-season/.krew.yaml**

```yaml
apiVersion: krew.googlecontaintools.github.com/v1alpha2
kind: Plugin
metadata:
  name: season
spec:
  homepage: https://github.com/scraly/kubectl-season
  shortDescription: Display seasonal emoji randomly before resources name
  version: v0.0.4
  description: |
    This plugin allows you to add randomly a seasonal (halloween, christmas, easter...) emoji just before the resources name. It's a special seasonal mode.
  platforms:
  - selector:
      matchExpressions:
      - key: os
        operator: In
        values:
        - darwin
        - linux
  uri: https://github.com/scraly/kubectl-season/archive/v0.0.4.tar.gz
  sha256: 7331819379ca14da988d2537904381b6186c989d527719cb1e0fd212293d749e
  bin: kubectl-season-0.0.4/kubectl-season
```
**kubectl-season/.krew.yaml**

```yaml
apiVersion: krew.googlecontainertools.github.com/v1alpha2
kind: Plugin
metadata:
  name: season
spec:
  homepage: https://github.com/scraly/kubectl-season
  shortDescription: Display seasonal emoji randomly before resources name
  version: v0.0.4
  description: |
    This plugin allows you to add randomly a seasonal (halloween, christmas, easter...) emoji just before the resources name. It's a special seasonal mode.
  platforms:
    - selector:
        matchExpressions:
          - key: os
            operator: In
            values:
            - darwin
            - linux
  uri: https://github.com/scraly/kubectl-season/archive/v0.0.4.tar.gz
  sha256: 7331819379ca14da988d2537904381b6186c989d527719cb1e0fd212293d749e
  bin: kubectl-season-0.0.4/kubectl-season
```

Download 61
apiVersion: krew.googlecontainer.tools.github.com/v1alpha2
kind: Plugin
metadata:
  name: season
spec:
  homepage: https://github.com/scraly/kubectl-season
  shortDescription: Display seasonal emoji randomly before resources name
  version: v0.0.4
  description: |
    This plugin allows you to add randomly a seasonal (halloween, christmas, easter...) emoji just before the resources name. It's a special seasonal mode.
  platforms:
    - selector:
        matchExpressions:
          - key: os
            operator: In
            values:
              - darwin
              - linux
  uri: https://github.com/scraly/kubectl-season/archive/v0.0.4.tar.gz
  sha256: 7331819379ca14da988d2537904381b6186c989d527719cb1e0fd212293d749e
  bin: kubectl-season-0.0.4/kubectl-season
Publishing on Krew > Pull Request

https://github.com/kubernetes-sigs/krew-index
**Krew > Create your private index & publish into it**

New feature: you can create your private index

```bash
$ kubectl krew index add scraly https://github.com/scraly/krew-index

$ kubectl krew index list
INDEX   URL
default https://github.com/kubernetes-sigs/krew-index.git
scraly   https://github.com/scraly/krew-index

$ kubectl krew search scraly
NAME      DESCRIPTION                                INSTALLED
scraly/halloween Display resources with Halloween emojis   yes
scraly/season   Display seasonal emoji randomly before resource...  no

$ kubectl krew install scraly/season
```
Krew > update index

$ kubectl krew update
Krew > upgrade plugins

// Upgrade all plugins
$ kubectl krew upgrade

// OR Upgrade only one plugin
$ kubectl krew upgrade season
Krew > our fav plugins

> view-secret

$ kubectl view-secret mysecret -a
Krew > our fav plugins

> view-secret
> view-cert

$ kubectl view-cert -A
Krew > our fav plugins

> view-secret
> view-cert
> view-utilization

$ kubectl view-utilization
Krew > our fav plugins

> view-secret
> view-cert
> view-utilization
> neat

$ kubectl get deploy mydeployment -o yaml | kubectl neat
BEST PRACTICES
Plugin creation > best practices

Naming is important

- **DON’T**: kubectl-version

  Not possible to create plugins that overwrite existing kubectl commands
Plugin creation > best practices

Naming is important

- **DON’T**: kubectl service
  Unclear what the plugin is doing with service

- **DON’T**: kubectl open
  Unclear what it is opening

- **DO**: kubectl open-svc
  The plugin will open a service :-)
PLUGIN CREATION > BEST PRACTICES

Golang

https://github.com/kubernetes/sample-cli-plugin
Plugin creation > best practices

Golang  krew-release-bot

https://github.com/rajatjindal/krew-release-bot
PLUGIN CREATION > BEST PRACTICES

Golang

krew-release-bot

Go releaser

https://goreleaser.com/
Pod

Smallest unit deployable & reusable in a cluster

Can contain several containers

Ova: IP address per Pod

Understanding in a visual way

http://gumroad.com/aurelievache & http://dev.to/aurelievache
Understanding in a visual way

https://www.youtube.com/c/AurelieVache
Conclusion

$ KUBECTL THANK-YOU

@aurelievache | @Gaelleacases
$ kubectl apply -f questions.yaml