KubeCon | CloudNativeCon

Europe 2022

WELCOME TO VALENCIA

KubeCon CloudNativeCon Europe 2022

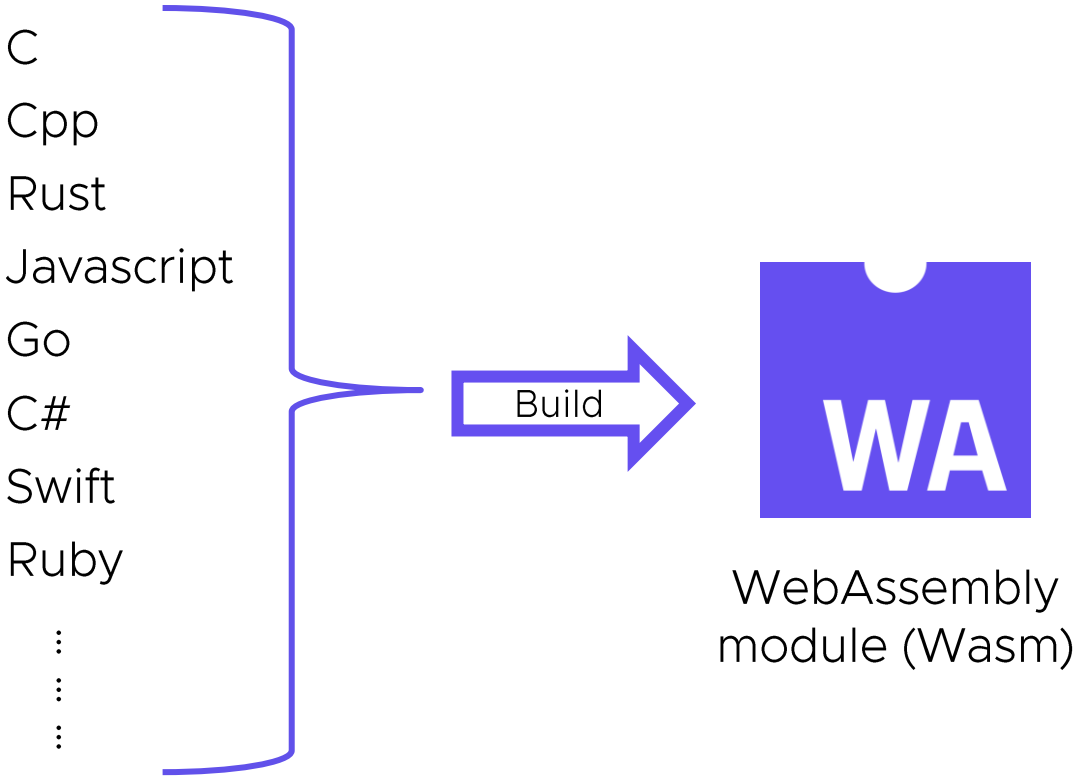
What if... kube-apiserver could be extended via WebAssembly?

Flavio Castelli, SUSE

What is WebAssembly?

WebAssembly
module (Wasm)

What is WebAssembly? Polyglot

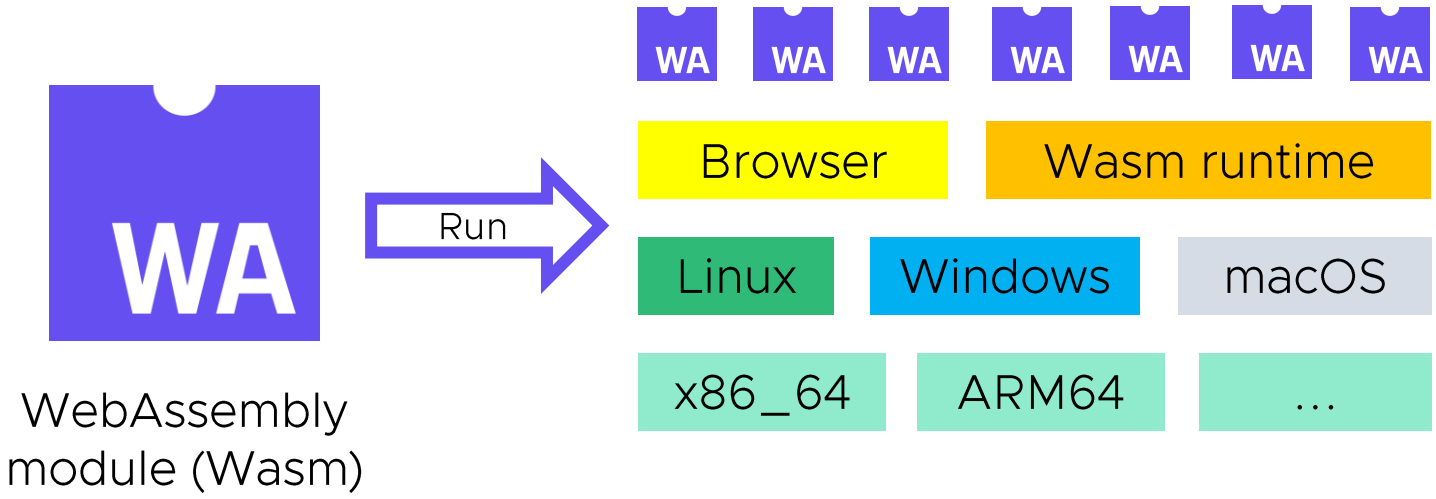


What is WebAssembly? Small

WebAssembly
module (Wasm)

Go 300 Kb
Rust 1.5 Mb

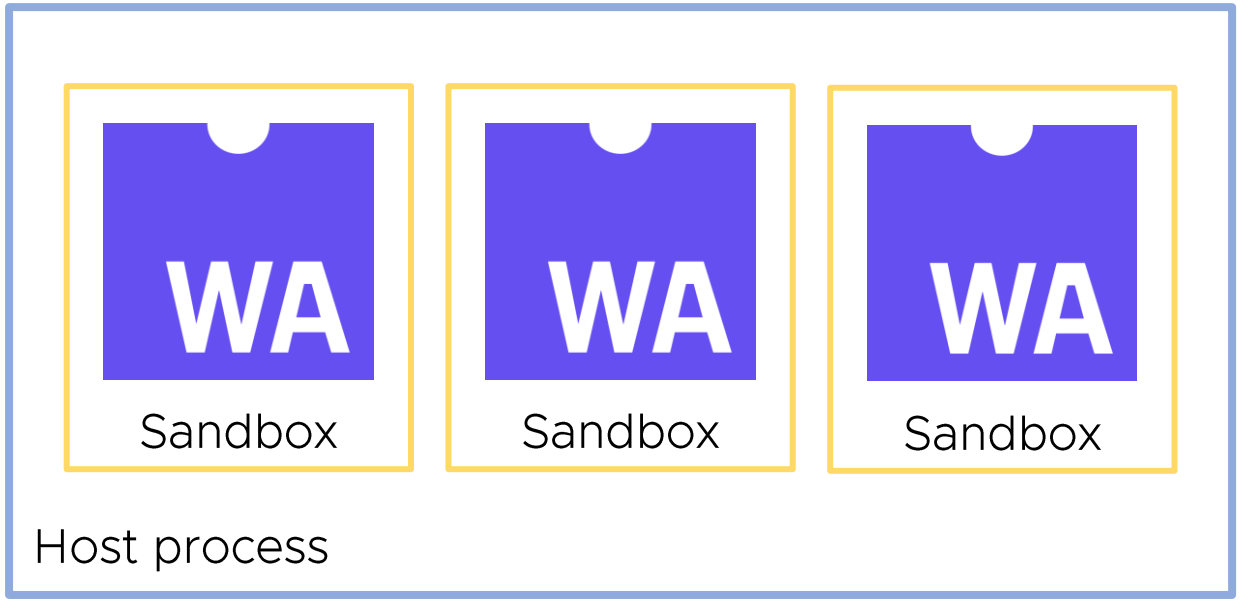
What is WebAssembly? Portable



What is WebAssembly? Secure

WebAssembly
module (Wasm)

What is WebAssembly? Secure



- Memory safety
- Control-flow integrity
- Runtime isolation

More details here

WebAssembly Outside of the Browser

- A new way to build and distribute applications
- Implement plugin systems

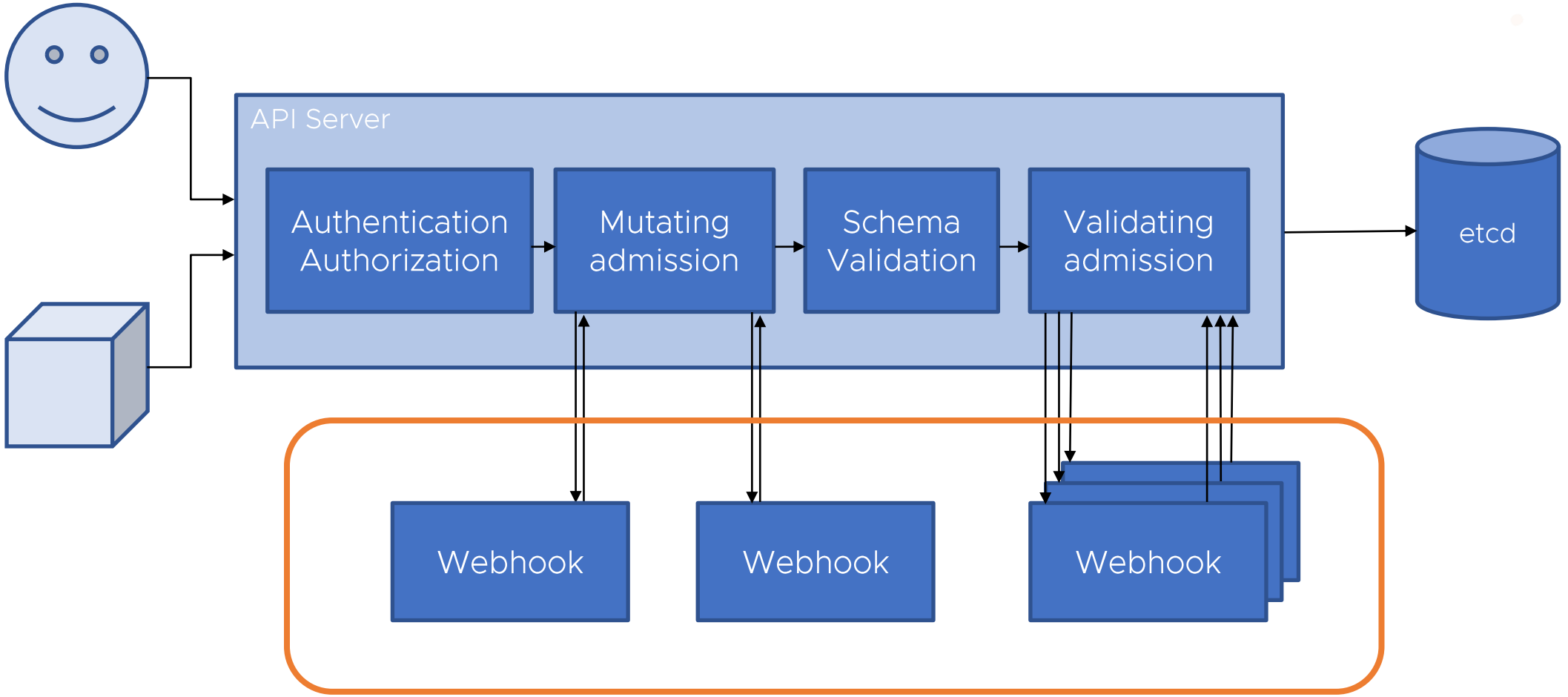
Kubernetes Control Plane Extensibility

- Authentication and Authorization
- Scheduler
- Dynamic Admission Controllers

Kubernetes Control Plane Extensibility

- Authentication and Authorization
- Scheduler
- Dynamic Admission Controllers

Dynamic Admission Controller



Introducing Kubewarden

A policy engine for Kubernetes.

Its mission is to simplify the adoption of Policy As Code.

Kubewarden Policies

- Written using:
  - Rust, Go, AssemblyScript, Swift
  - Rego
- Compiled to WebAssembly
- Distributed using container registries
- Signed and verified using Sigstore

The Idea

- Define admission rules using WebAssembly modules
- Extend the API server to make use of WebAssembly-based rules

What do we gain?

Remove Uncertainty

- Webhooks rely on the network
- The network introduces many types of failures
- The network increases attack surface

```yaml
apiVersion: admissionregistration.k8s.io/v1
kind: ValidatingWebhookConfiguration
...
webhooks:
- name: my-webhook.example.com
  timeoutSeconds: 15
  failurePolicy: Fail
  ...
```

Limit Resource Usage

- A set of Kubernetes Custom Resource Definitions
- The Webhook server
- The Controller that reconciles the Custom Resources

Limit Resource Usage

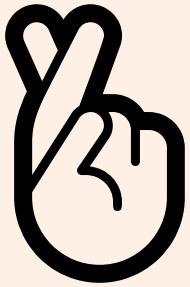
- A set of Kubernetes Custom Resource Definitions
- ~~The Webhook server~~
- ~~The Controller that reconciles the Custom Resources~~

Great for Edge environments!

The POC

- Leverage Kubewarden policies
- Introduce a new Kubernetes Feature Gate capable of running them

Demo Time



The Biggest Challenge

- Major WebAssembly runtimes are not written in pure Go
- Initial approach: patch Kubernetes build system
- Iteration: custom build of wasmtime-go with musl libc
- Final solution, rely on the wazero runtime: pure Go, no usage of CGO

POC: the Missing Details

- No support for Mutating policies
- Not implemented Kubewarden features offered to policy authors:
  - Policy tracing
  - Context aware information
  - Interactions with container registries
  - Sigstore primitives
  - Execute Rego policies built as WebAssembly modules
- Verify pulled policies using Sigstore
- Performance testing

What's Next?

- Kubernetes community: are you interested about this idea?
- Kubernetes developers: can we talk about other areas where we can leverage WebAssembly?

Links

- Kubernetes fork - based on vanilla 1.23.5
- k3s:
  - Kubernetes fork - based on vanilla k3s 1.23.5+k3s1
  - k3s fork - based on vanilla k3s 1.23.5+k3s1
  - k3s experimental build
- krew-wasm: kubectl plugins built with WebAssembly
- Kubewarden main website: https://kubewarden.io
- Slack: "kubewarden" channel on Kubernetes workspace