

# KUBERNETES-KÉNYYSZEREK MEGVALÓSÍTÁSA MAGASSZINTŰ DSL ALAPJÁN

MSc Diplomatervezés 1

Készítette: Tieger Balázs

Konzulense: Elekes Márton

Külső konzulense: Dr. Szatmári Zoltán



**Ktor**

# Kubernetes



Nyílt forráskódú konténerkezelő platform

Alkalmazások hatékony üzemeltetése és skálázása

Konténerek kezelése: önállóan futó alkalmazások és függőségeik

Üzemeltetés deklaratív leírás alapján



**kubernetes**

# Példák kényszerekre

Céges policy, vagy kubernetes best practice kikényszerítése

Minden frontend pod két példányban fusson, különböző node-okon a magas rendelkezésre állás miatt

Minden podnak kell legyen resources limit megszabva

Horizontálisan skálázott komponenseknél a resources limit és request meg kell egyezzen.

## Nehéz

- Megfigyelni
- Számontartani
- Betartatni
- Módosítani

# Kényszer betartatási módszerek

## Események alapján

- Validálás:  
allow / deny
- Beavatkozás,  
módosítás

## Állapot vizsgálata alapján

- Diagnosztika: jelenlegi állapot vizsgálata
- **Meglévő erőforrások  
automatikus  
módosítása, javítása**

# Elkészült rendszer

Nyelv – Kotlin DSL

Management rendszer

- DSL fordítás
- Monitorozó és beavatkozó ágensek indítása

Egyszerű WebUI

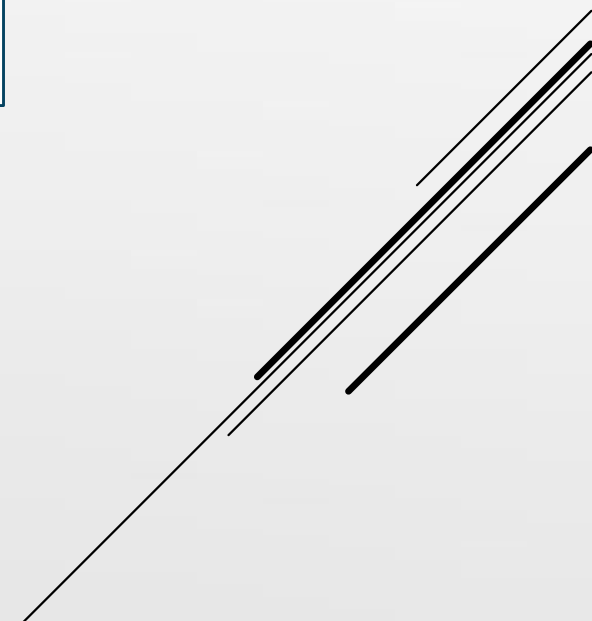
- DSL feltöltés
- Ágens indítás
- Diagnosztika vizualizáció

Esettanulmány

- Alap diagnosztika
- Alap beavatkozások

# Nyelv – Kotlin DSL

```
val server = server("my-server", permissions) {  
    // Diagnosztika  
    monitor("monitor-name", { kubectl.pods().list() }) {  
  
    }  
    // Engedélyezés, módosítás  
    webhook("webhook-name", configDefaults) {  
  
    }  
}
```



# Nyelv – diagnosztika

```
monitor("Deployments", {kubectl.apps().deployments().inAnyNamespace().list()}) {  
    mark( "No node selector") {  
        spec.template.spec.nodeSelector.isEmpty()  
    }  
    mark( "No probes") {  
        spec.template.spec.containers.any {it.livenessProbe == null}  
    }  
    mark("Has long history") {  
        spec.revisionHistoryLimit > 4  
    }  
}
```

# Nyelv – diagnosztika

Servers Monitors

basic-diagnostics

Deployments

deployments.apps/basic-diagnostics.konstrainer-ns

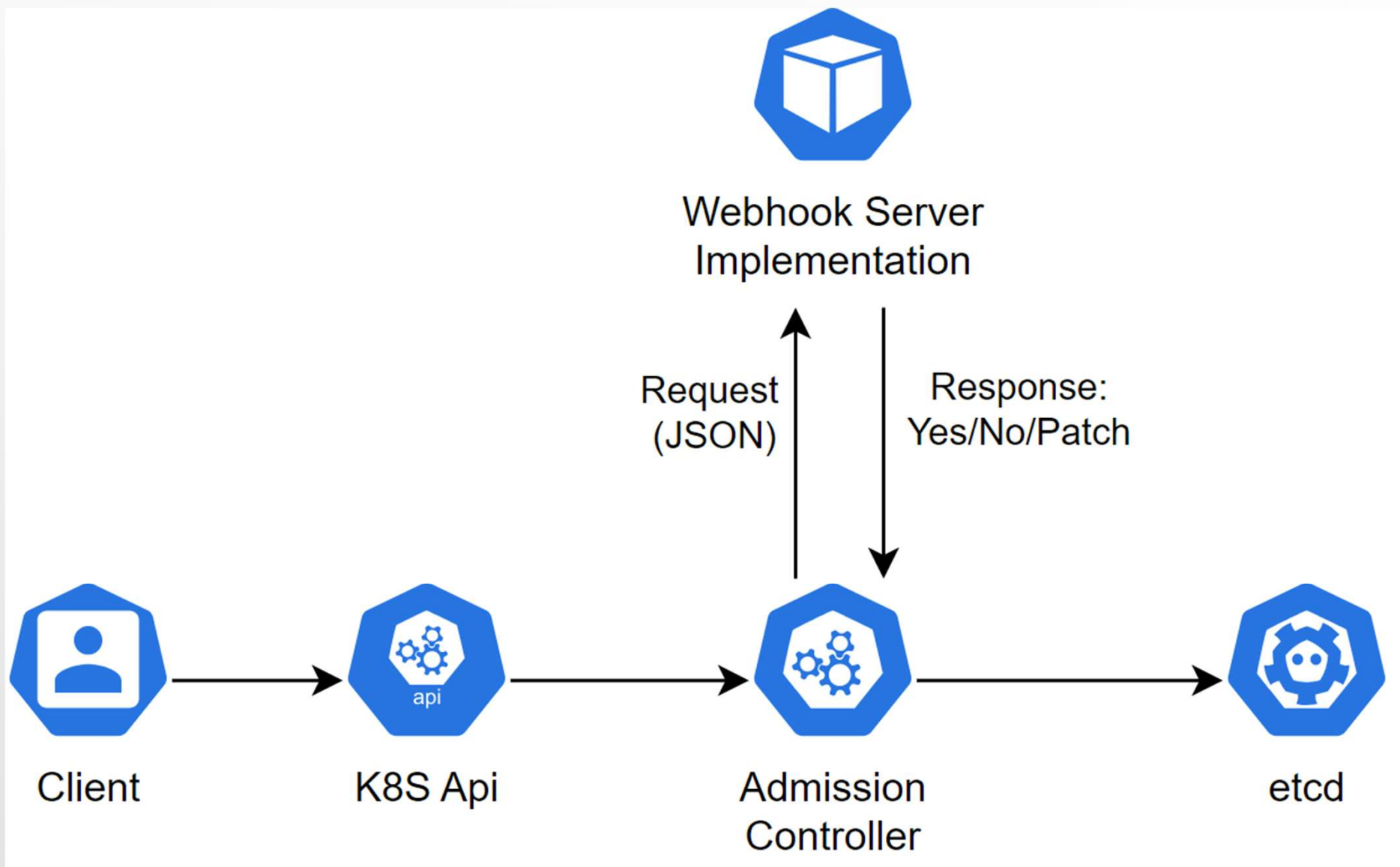
Has no resources, No node selector, No probes, Has long history

deployments.apps/konstrainer-core.konstrainer-ns

No node selector



# Események figyelése - Webhookok



# Nyelv – Esemény validálás

```
webhook("deny-no-resources", defaults) {  
  operations(CREATE, UPDATE)  
  resources(DEPLOYMENTS)  
  behavior {  
    val containers = (request jqx "/object/spec/template/spec/containers").jsonArray  
    allowed {  
      containers.all { (it.jsonObject jqx "/resources/").jsonObject.isNotEmpty() }  
    }  
    status {  
      message = "Deployment must have resource definitions!"  
    }  
  }  
}
```

# Nyelv – Esemény módosítás

```
webhook("cut-history", defaults) {  
  operations(CREATE, UPDATE)  
  resources(DEPLOYMENTS)  
  behavior {  
    val limit = (request jqx "/object/spec/revisionHistoryLimit" parseAs int) ?: 10  
    warnings {  
      if (limit > 4) warning("RevisionHistoryLimit was set to 4, from original: $limit")  
    }  
    patch {  
      if (limit > 4) replace("/spec/revisionHistoryLimit", 4)  
    }  
  }  
}
```

# Validálás és beavatkozás

```
webhook("deny-no-resources", defaults) {  
  operations(CREATE, UPDATE)  
  resources(DEPLOYMENTS)  
  behavior {  
    val containers = (request jqx "/object/spec/template/  
    allowed {  
      containers.all { (it.jsonObject jqx "/resources/") .js  
    }  
    status {  
      message = "Deployment must have resource defini  
    }  
  }  
}
```

```
apiVersion: apps/v1  
kind: Deployment  
metadata: ...  
spec:  
  selector: ...  
  template:  
    metadata: ...  
    spec:  
      containers:  
        - name: demo-myapp  
          image: debian:11
```

```
x btieger ~/Uni/KonstrainR/main } main ± k apply -n demo-ns -f test/deployment.yaml  
Warning: RevisionHistoryLimit was set to 4, from original: 10  
Error from server: error when creating "test/deployment.yaml": admission webhook "deny-no-resources.  
btieger.me" denied the request: Deployment must have resource definitions!
```

# Validálás és beavatkozás

```
webhook("cut-history", defaults) {  
  behavior {  
    val limit = (request jqx "/object/spec/revisionHistoryLimit")  
    warnings {  
      if (limit > 4)  
        warning("RevisionHistoryLimit was set to 4, from original: 10")  
    }  
    patch {  
      if (limit > 4)  
        replace("/spec/revisionHistoryLimit", 4)  
    }  
  }  
}
```

```
apiVersion: apps/v1  
kind: Deployment  
metadata: ...  
spec:  
  selector: ...  
  template:  
    metadata: ...  
    spec:  
      containers:  
        - name: demo-myapp  
          image: debian:11  
          resources:  
            limits:  
              cpu: 100m  
              memory: 250M
```

```
❌ btieger ~/Uni/KonstrainR/main } main ± k apply -n demo-ns -f test/deployment.yaml  
Warning: RevisionHistoryLimit was set to 4, from original: 10  
deployment.apps/demo-myapp created
```

```
btieger ~/Uni/KonstrainR/main } main ± k get -n demo-ns deployment.apps/demo-myapp  
-o yaml | yq '.spec.revisionHistoryLimit'  
4
```

# Validálás és beavatkozás

```
webhook("cut-history", defaults) {  
  behavior {  
    val limit = (request jqx "/object/spec/revisionHistoryLimit"  
    warnings {  
      if (limit > 4)  
        warning("RevisionHistoryLimit was set to 4, from orig  
    }  
    patch {  
      if (limit > 4)  
        replace("/spec/revisionHistoryLimit", 4)  
    }  
  }  
}
```

```
apiVersion: apps/v1  
kind: Deployment  
metadata: ...  
spec:  
  revisionHistoryLimit: 3  
  selector: ...  
  template:  
    metadata: ...  
    spec:  
      containers:  
        - name: demo-myapp  
          image: debian:11  
resources: ...
```

```
btieger ~/Uni/KonstrainR/main % main ± k apply -n demo-ns -f test/deployment.yaml  
deployment.apps/demo-myapp created
```

```
btieger ~/Uni/KonstrainR/main % main ± k get -n demo-ns deployment.apps/demo-myapp  
-o yaml | yq '.spec.revisionHistoryLimit'  
3
```

# Demo

Servers Monitors

Choose file...

Browse Submit

basic-webhook-rules

Build status: Ready  
Server status: Down

View Details Delete Start Server

Fordítás: 20~40 másodperc

Indítás: 2~5 másodperc

Implementálással együtt: <30 perc

# Webhook kézzel

## Tudás elsajátítás

- AdmissionWebhooks: <https://kubernetes.io/docs/reference/access-authn-authz/extensible-admission-controllers/>
- AdmissionWebhooks gyakorlat: <https://www.baeldung.com/java-kubernetes-admission-controller>
- RBAC: <https://kubernetes.io/docs/reference/access-authn-authz/rbac/>
- JSON Patch: <https://jsonpatch.com/>
- PKI
- Általános programozás
- Webszerver keretrendszer

## Webszerver implementálás

### HTTPS konfigurálás

- Kulcs menedzsment
- Tanúsítványok kiállítása

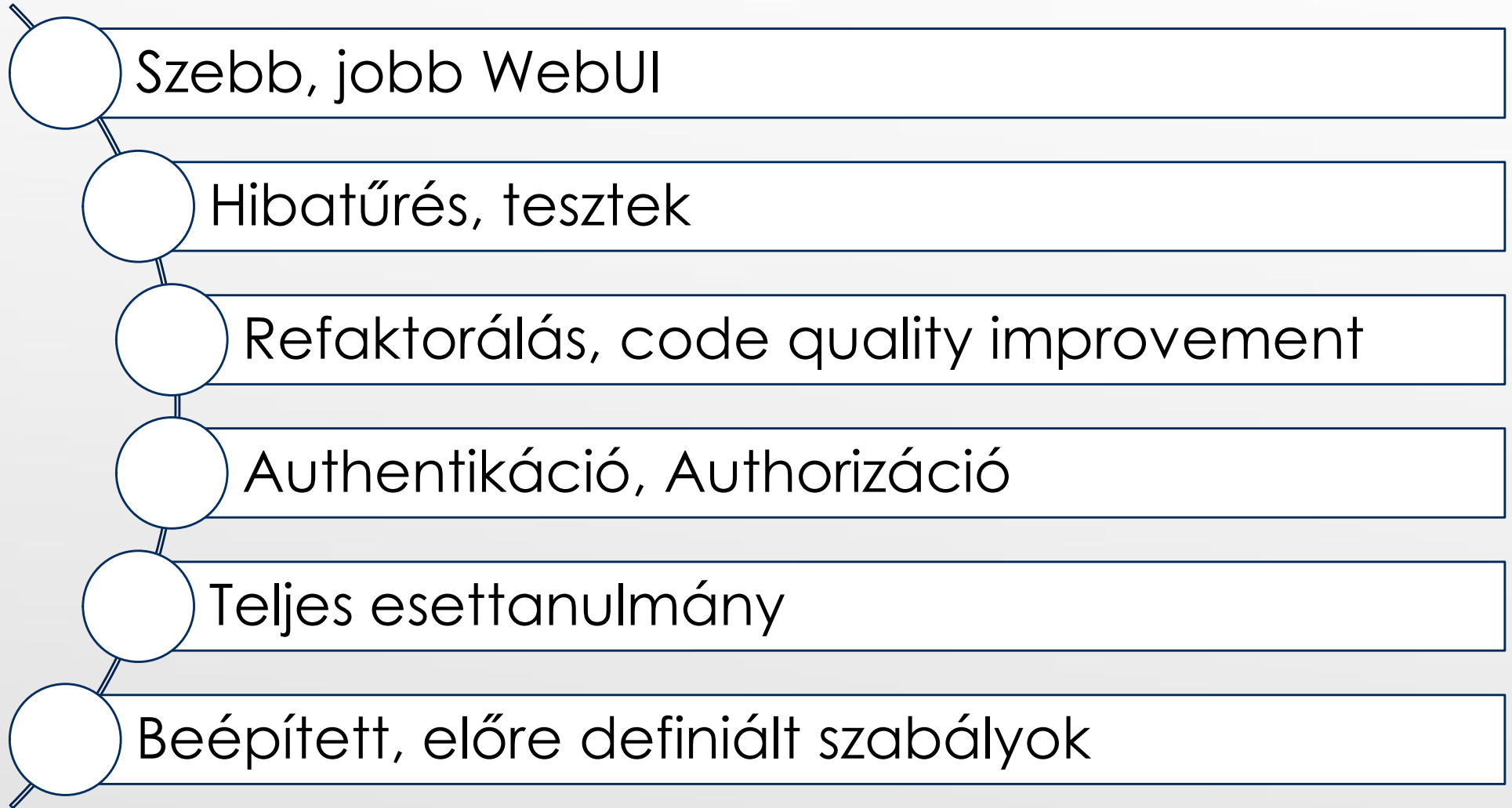
### Fordítás, konténerizálás, image feltöltés

### Telepítés, konfigurálás

- Deployment
- MutatingWebhookConfiguration
- Service



# Továbbfejlesztési lehetőségek



# Köszönöm a figyelmet!

```
monitor("Deployments", {kubectl.apps().deployments().inAnyNamespace().list()}) {  
  mark("No node selector") {  
    spec.template.spec.nodeSelector.isEmpty()  
  }  
  mark("No probes") {  
    spec.template.spec.containers.any {it.livenessProbe == null}  
  }  
  mark("Has long history") {  
    spec.revisionHistoryLimit > 4  
  }  
}
```

## Validálás és beavatkozás

```
apiVersion: apps/v1  
kind: Deployment  
metadata: ...  
spec:  
  selector: ...  
  template:  
    metadata: ...  
    spec:  
      containers:  
        - name: demo-myapp  
          image: debian:11
```

```
btieger ~/Uni/KonstrainsR/main k apply -n demo-ns -f test/deployment.yaml  
Warning: RevisionHistoryLimit was set to 4, from original: 10  
Error from server: error when creating "test/deployment.yaml": admission webhook "deny-no-resources.btieger.me" denied the request: Deployment must have resource definitions!
```

## Webhook kézzel

### Tudás elsajátítás

- AdmissionWebhooks: <https://kubernetes.io/docs/reference/access-authn-authz/extensible-admission-controllers/>
- RBAC: <https://kubernetes.io/docs/reference/access-authn-authz/rbac/>
- JSON Patch: <https://jsonpatch.com/>
- PKI
- Általános programozás
- Webszerver keretrendszer

### Webszerver implementálás

#### HTTPS konfigurálás

- Kulcs menedzsment
- Tanúsítványok kiállítása

#### Fordítás, konténerizálás, image feltöltés

#### Telepítés, konfigurálás

- Deployment
- MutatingWebhookConfiguration
- Service