KubeCon | CloudNativeCon

North America 2019

KubeCon | CloudNativeCon

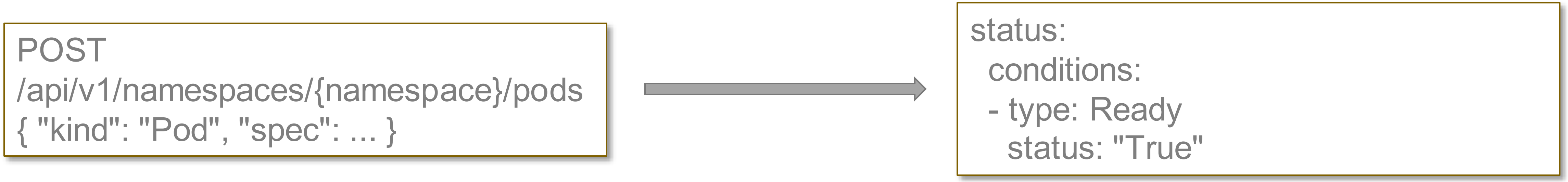
North America 2019

Ready to Serve!

Speeding-Up Startup Time of Istio-Powered Workloads

Michal Malka, Etai Lev-Ran, Zvi Cahana
IBM Research

Why care about startup time?



```
POST
/api/v1/namespaces/{namespace}/pods
{ "kind": "Pod", "spec": ... }
```

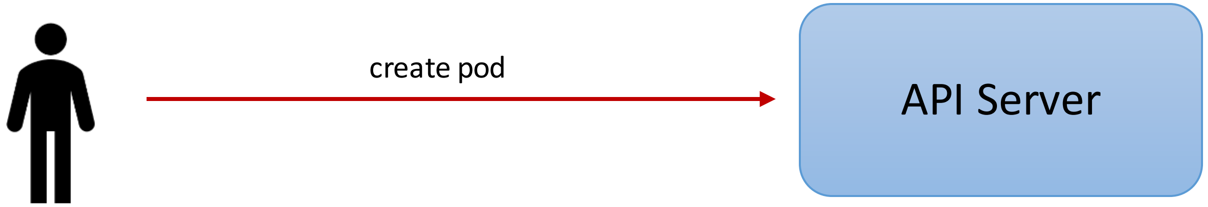
```
status:
  conditions:
  - type: Ready
    status: "True"
```

Relevant use cases:

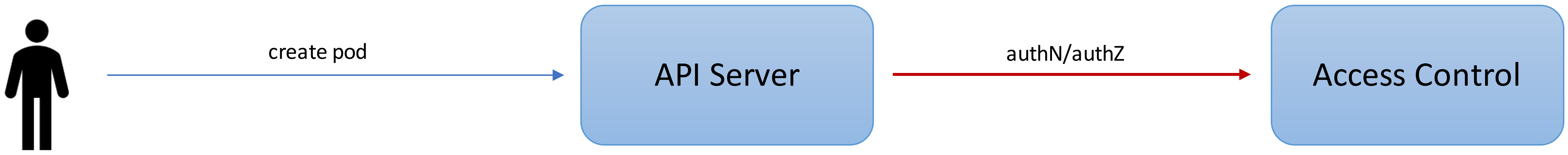
✓ Serverless
✓ Failure recovery
✓ Node eviction

✓ Scale to (from) zero
✓ Rolling upgrades
✓ Autoscaling

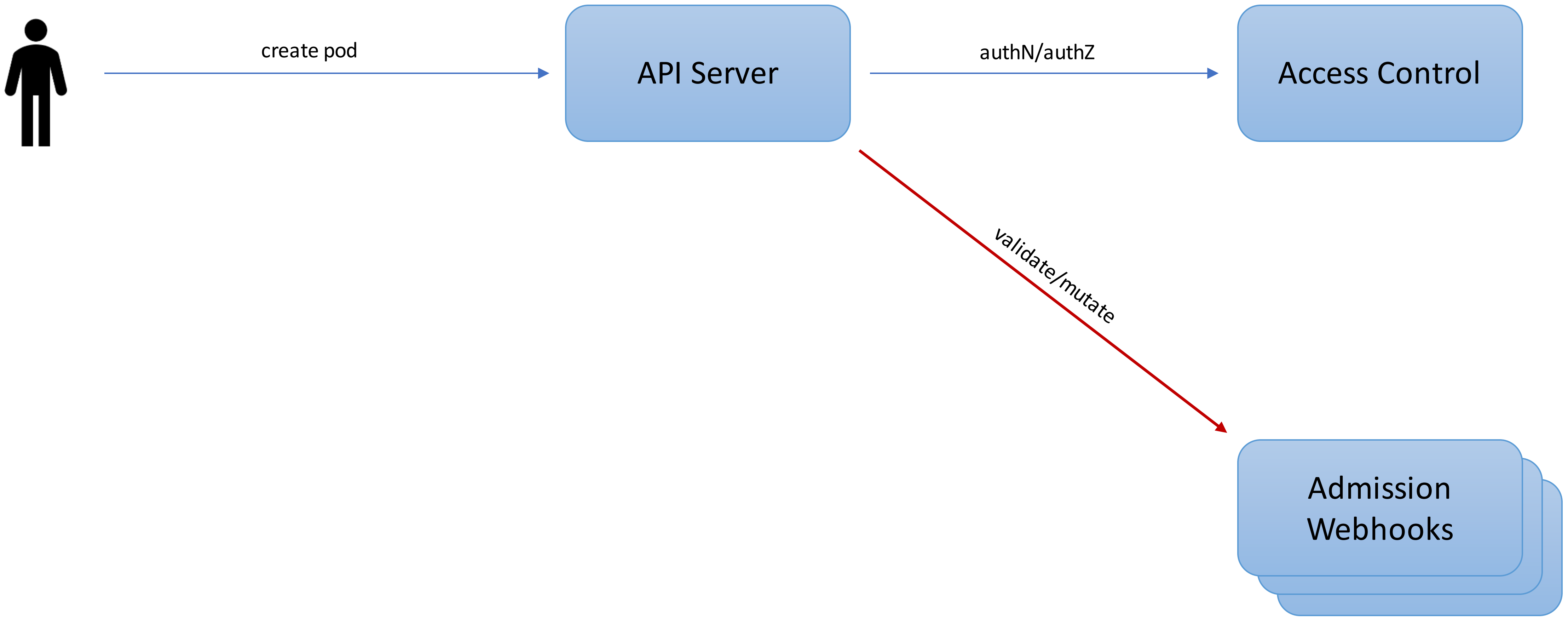
What actually happens when a pod starts?



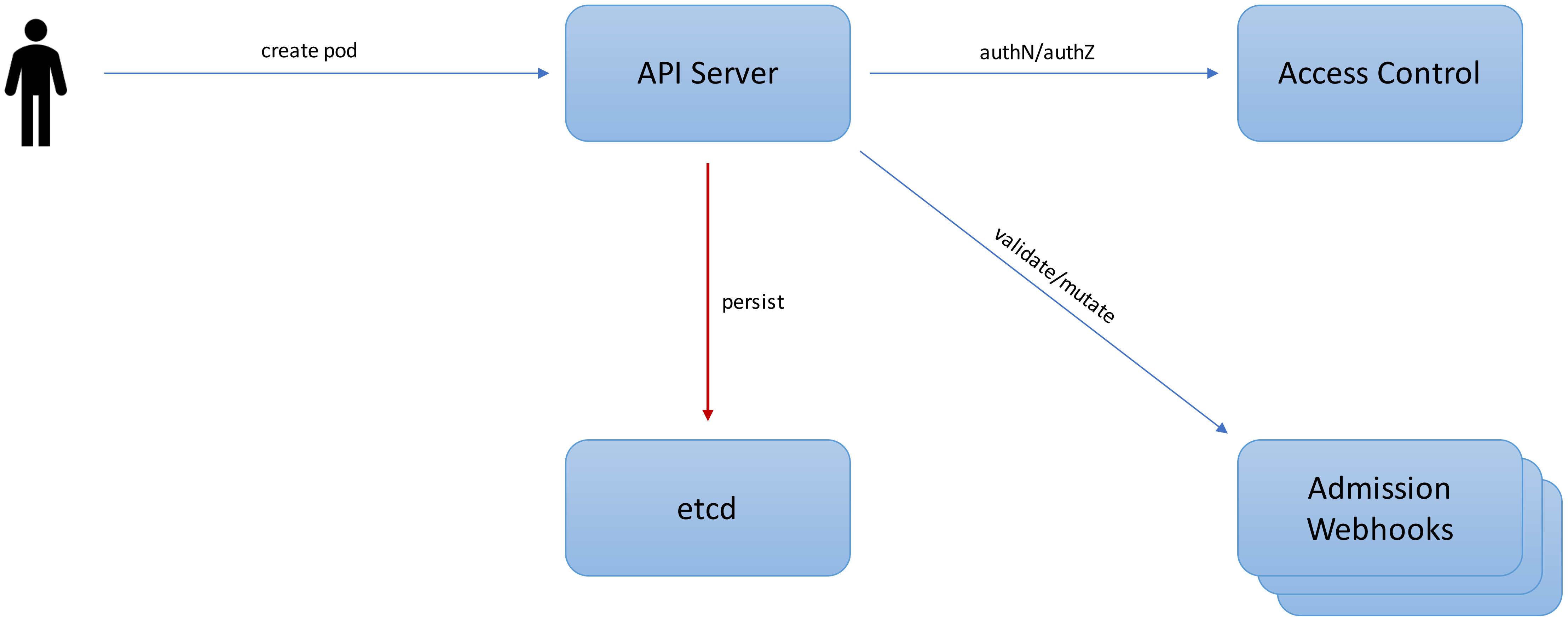
What actually happens when a pod starts?



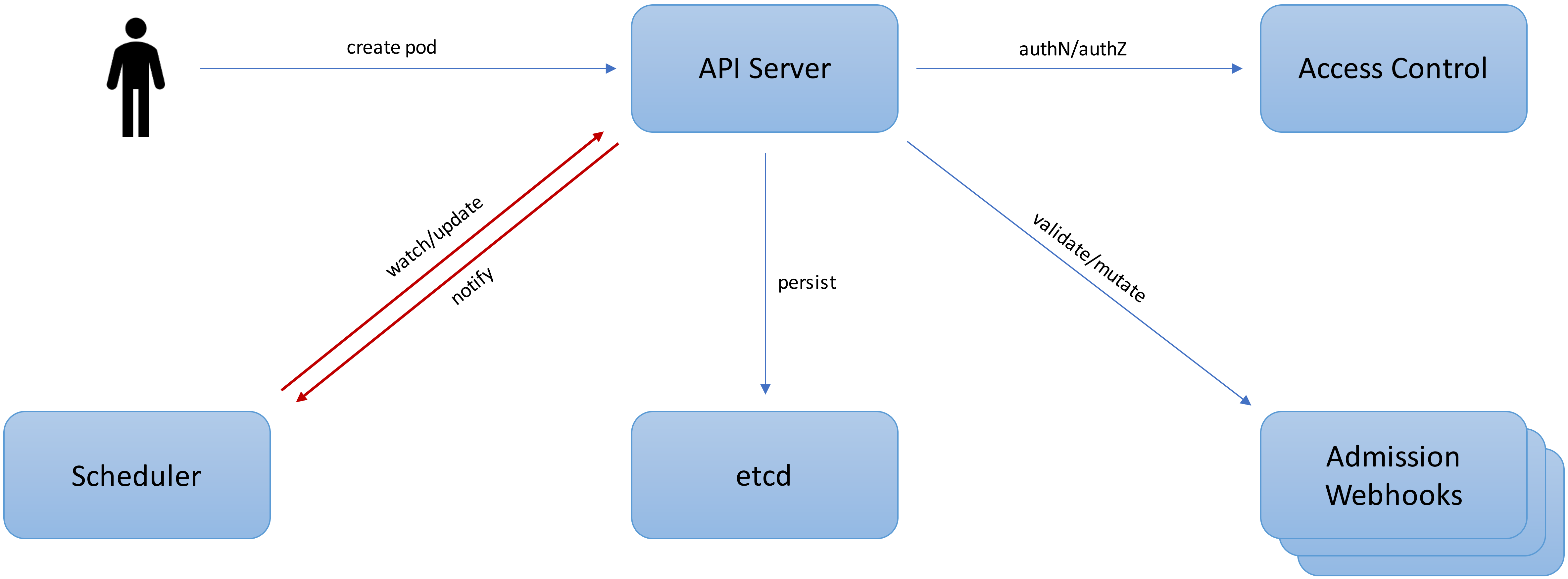
What actually happens when a pod starts?



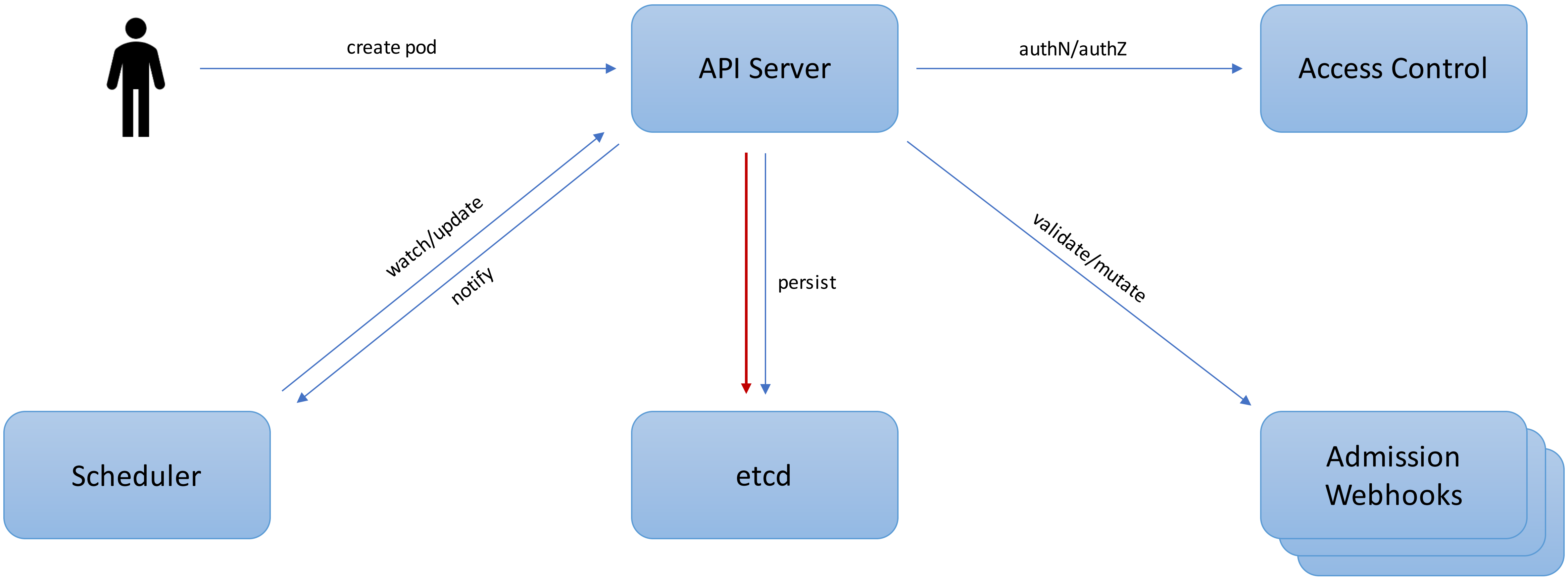
What actually happens when a pod starts?



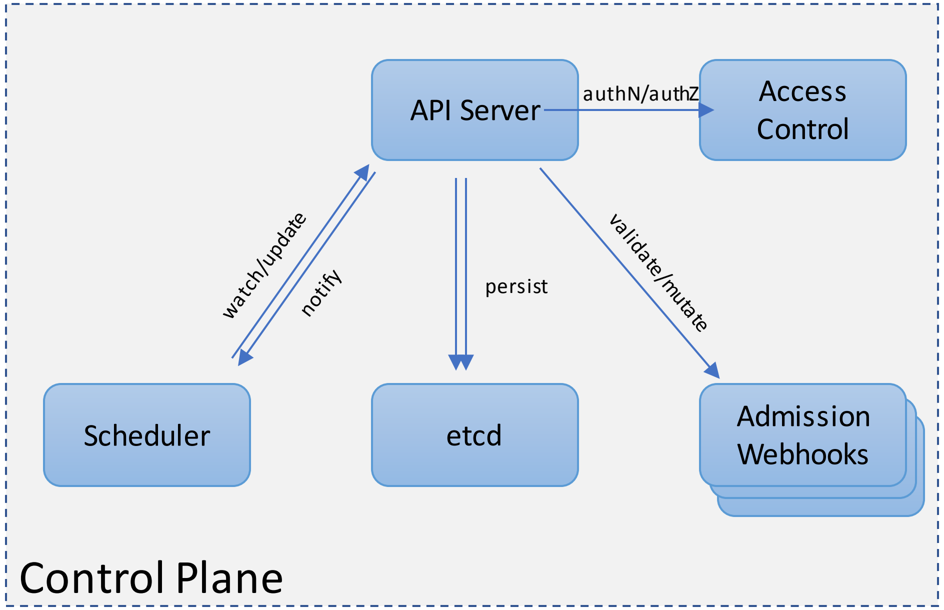
What actually happens when a pod starts?



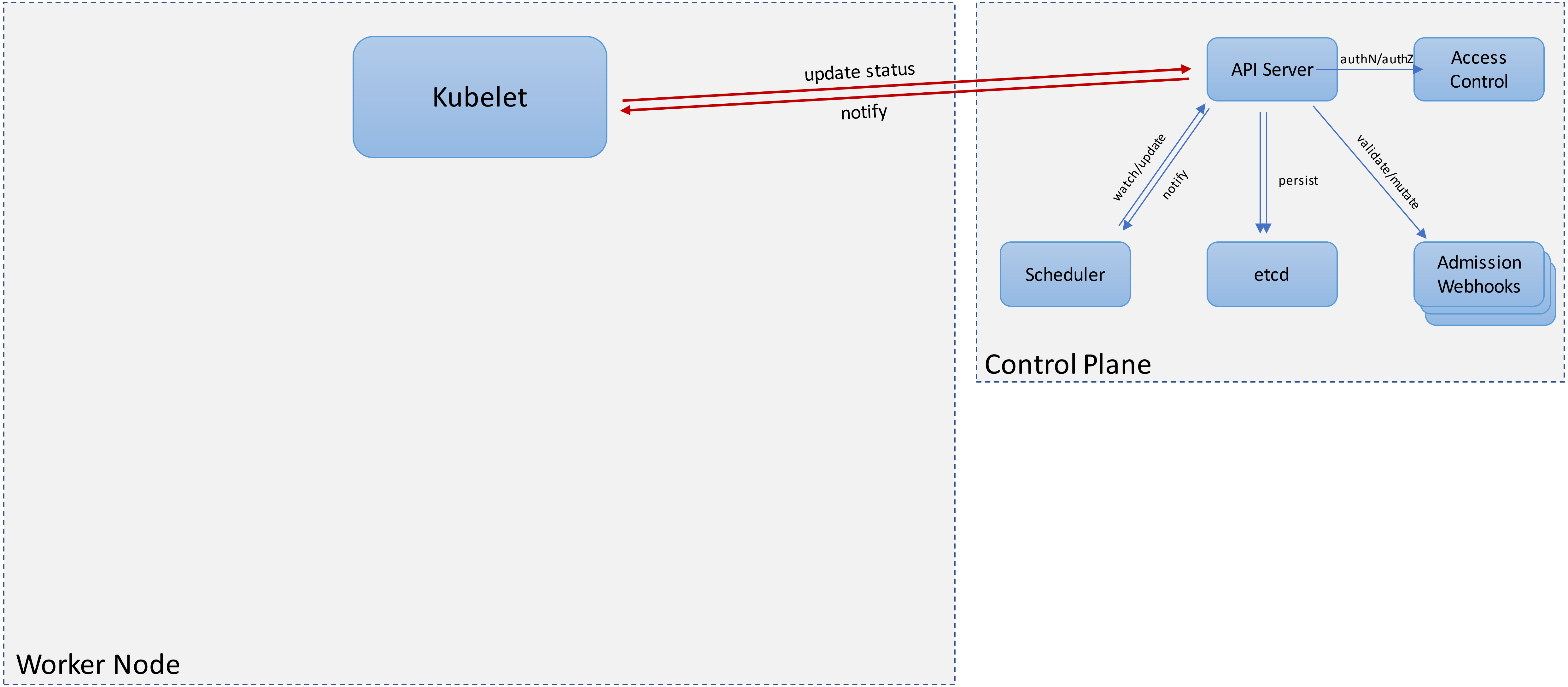
What actually happens when a pod starts?



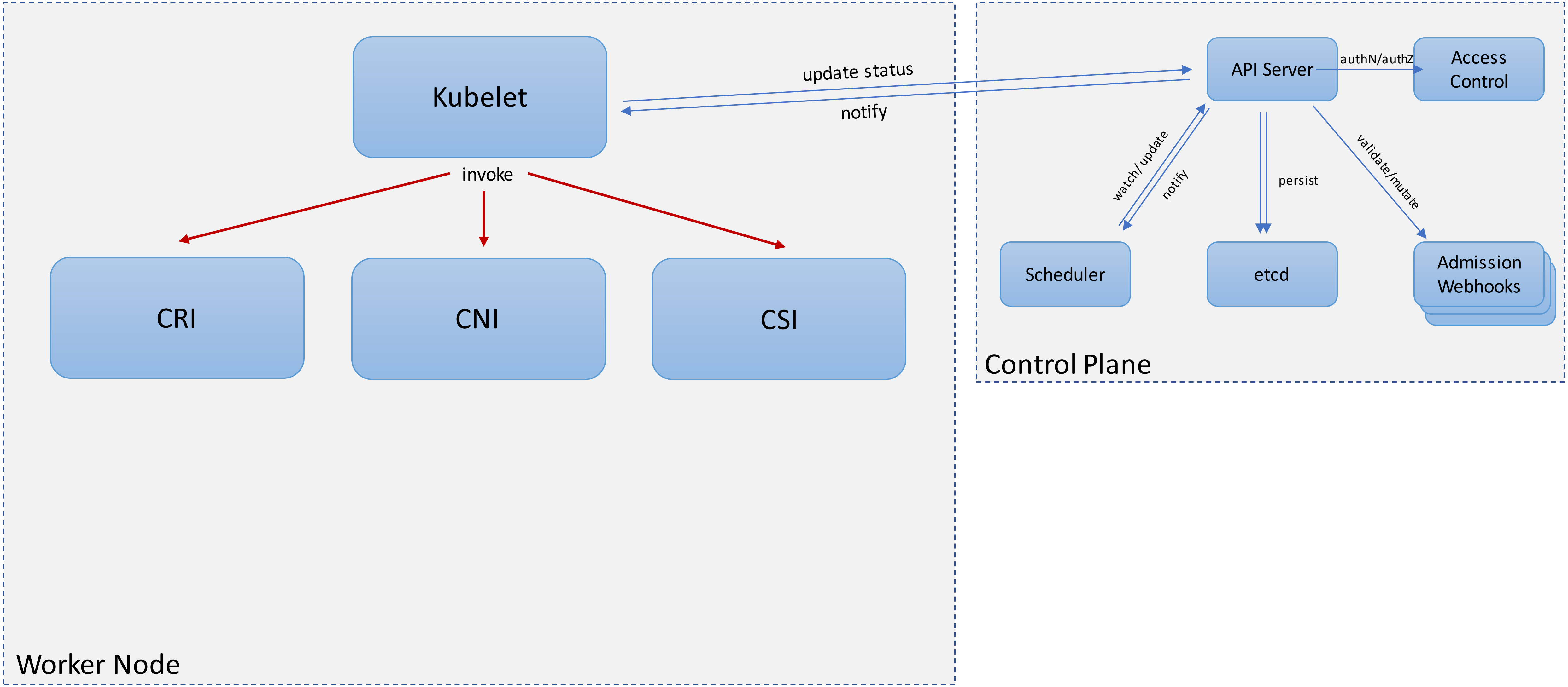
What actually happens when a pod starts?



What actually happens when a pod starts?

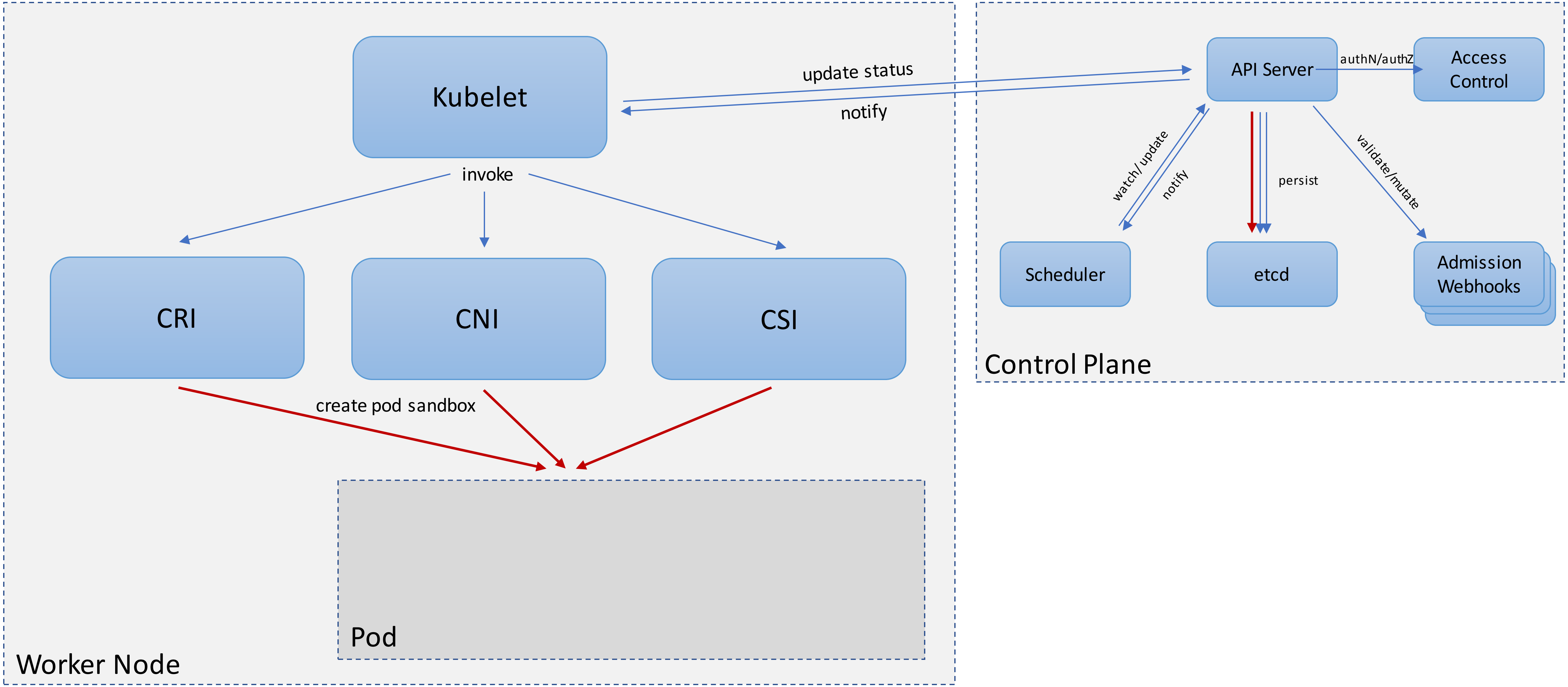


What actually happens when a pod starts?



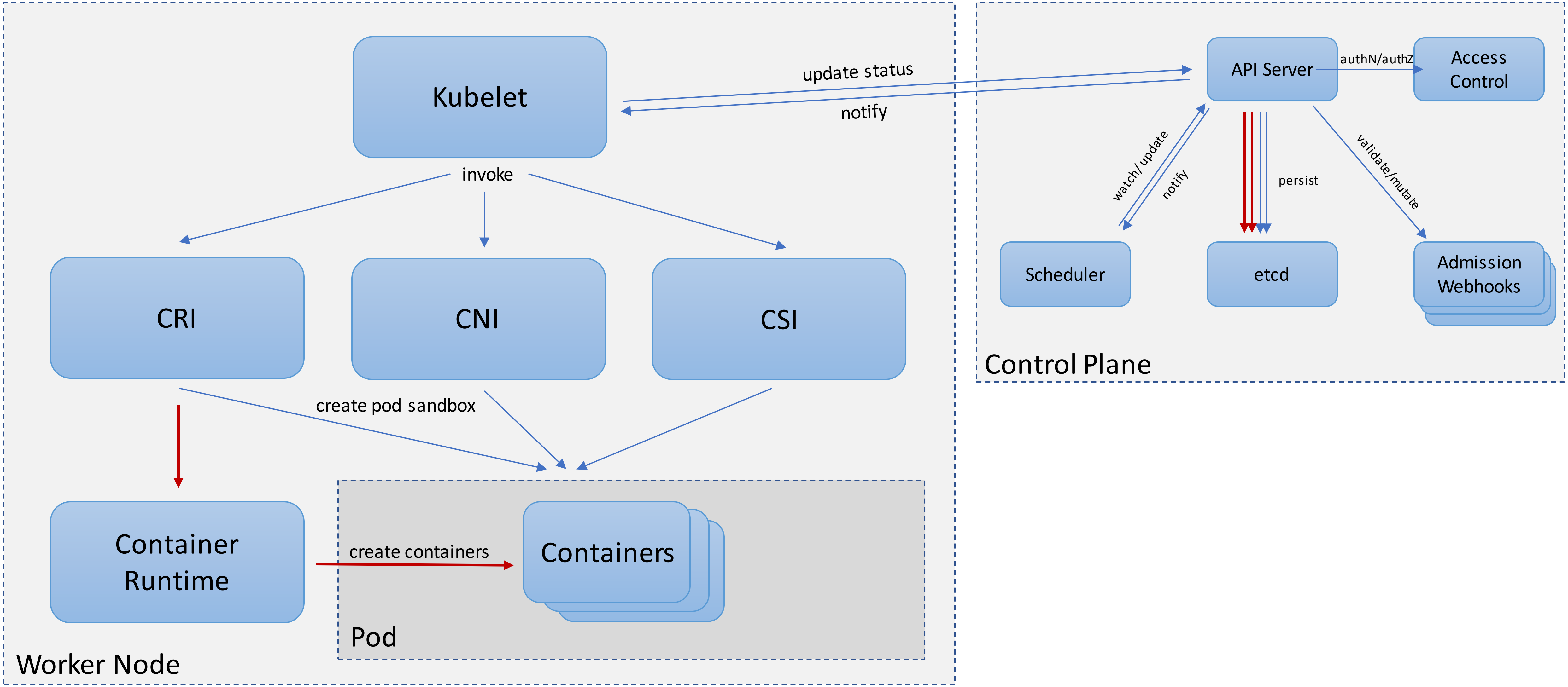
What actually happens when a pod starts?

KubeCon CloudNativeCon North America 2019

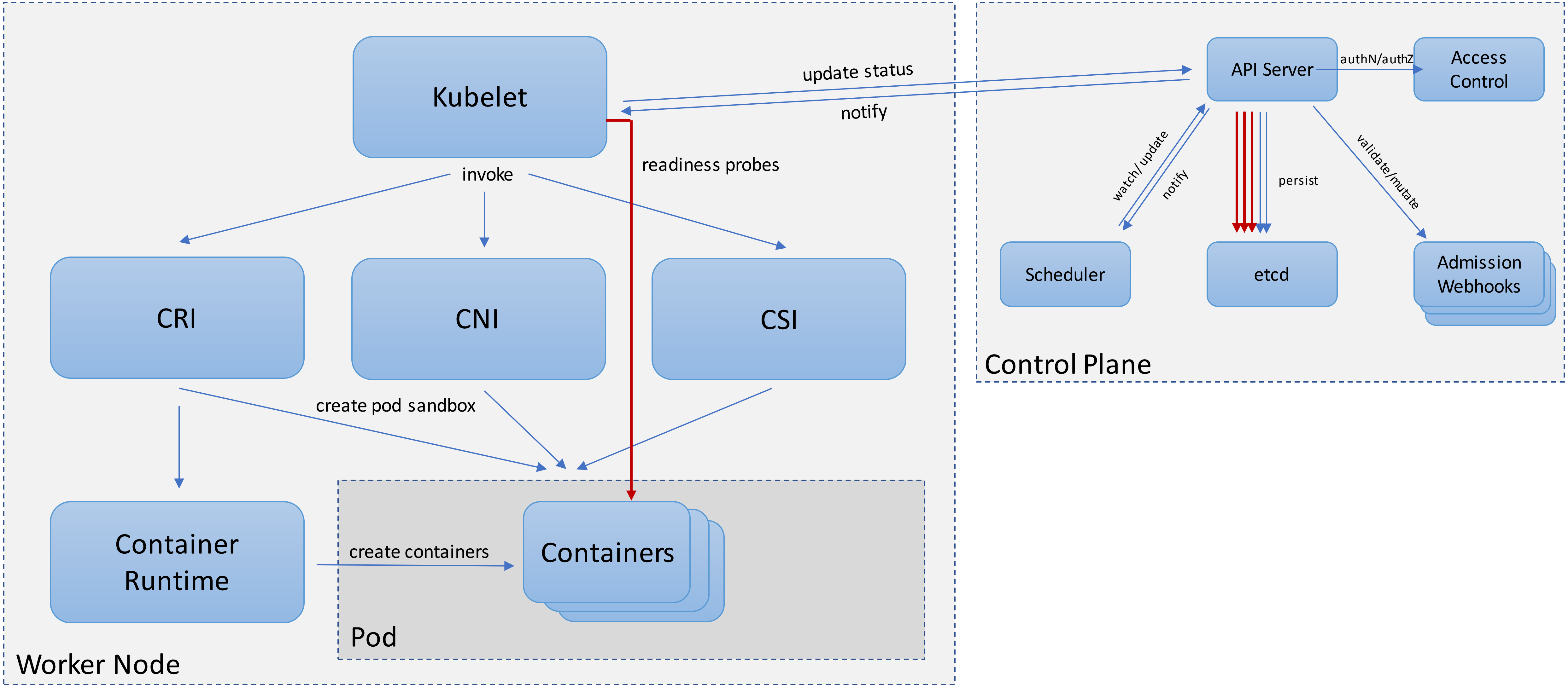


What actually happens when a pod starts?

KubeCon | CloudNativeCon North America 2019



What actually happens when a pod starts?



How much time does it take?

How much time does it take?

It depends

How much time does it take?

Official SLO:
5 seconds

“Startup latency of **schedulable stateless pods**, **excluding time to pull images** and **run init containers**, measured from pod creation timestamp to when all its containers are reported as started and observed via watch, measured as **99th percentile** over last 5 minutes, in **default Kubernetes installation**.”

https://github.com/kubernetes/community/blob/master/sig-scalability/slos/pod_startup_latency.md

How much time does it take?

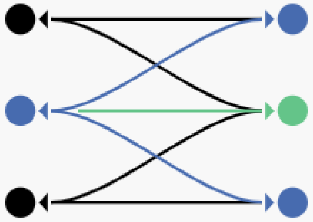
p50:	2.6 seconds
p99:	3.1 seconds

Startup Time with Istio

What is Istio?

Istio

Connect, secure, control and observe microservices

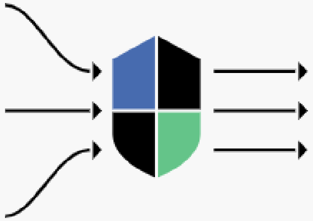


Connect

Intelligently control the flow of traffic and API calls between services

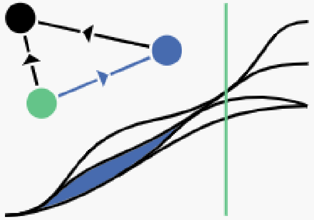
Secure

Secure your services with managed authentication, authorization & encryption



Control

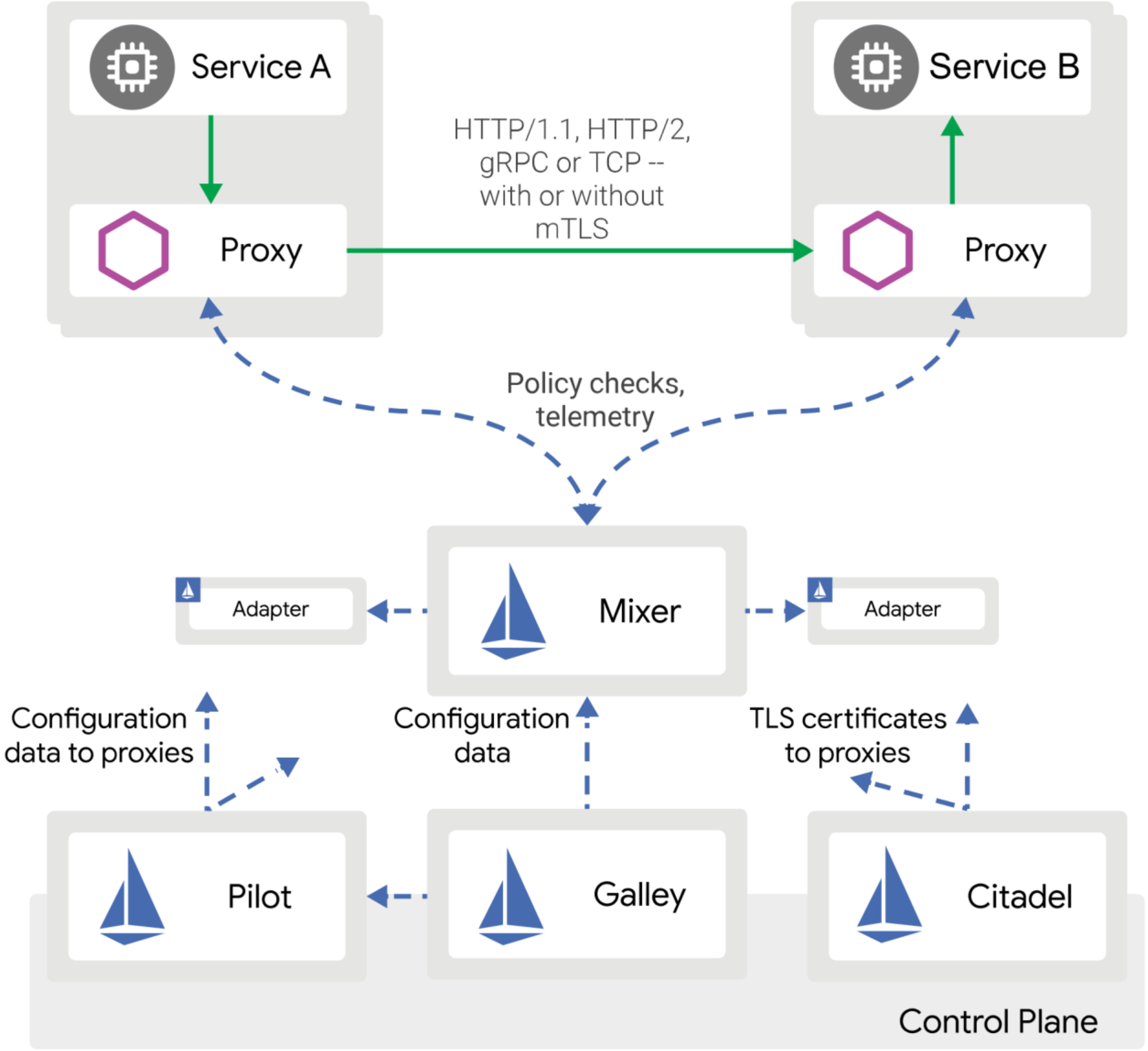
Apply policies and quotas and ensure that they're enforced



Observe

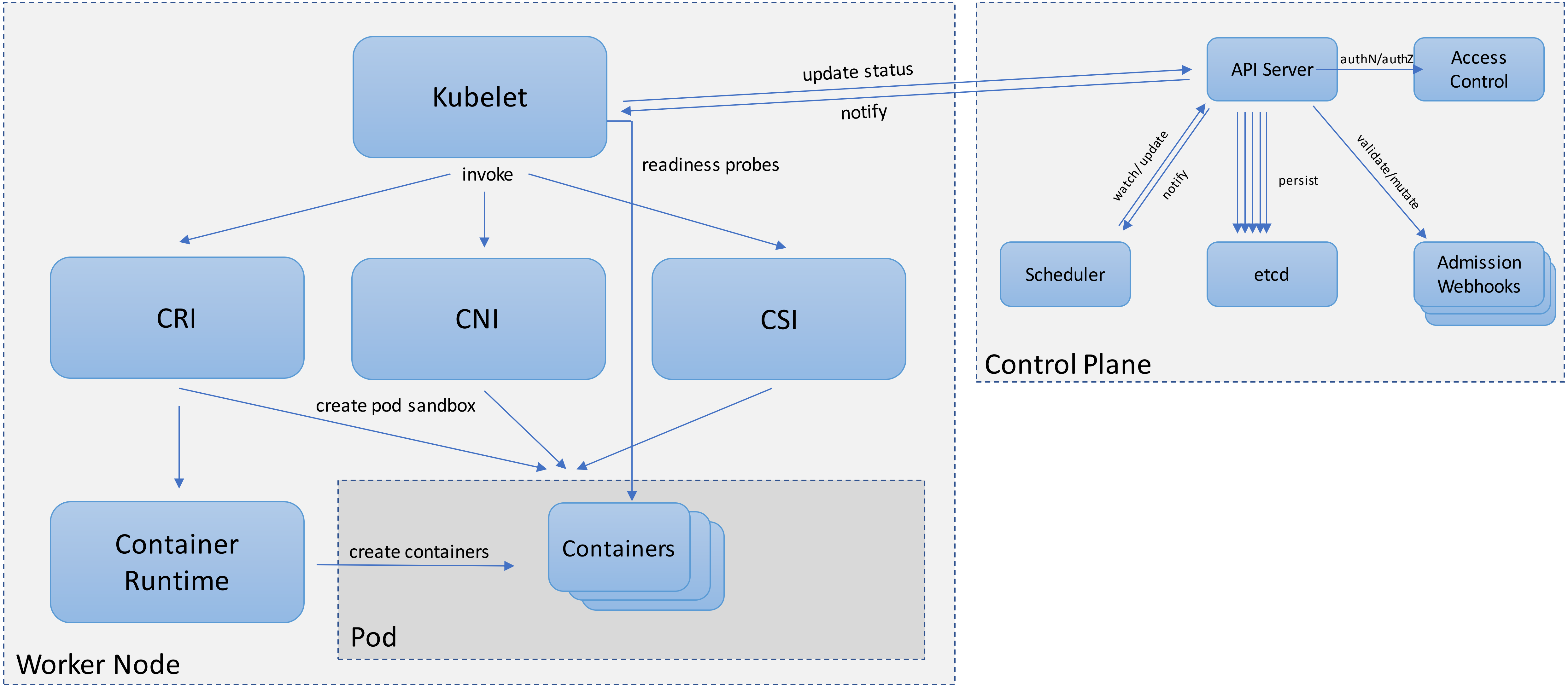
See what's happening with automatic tracing, monitoring, and logging

Istio architecture

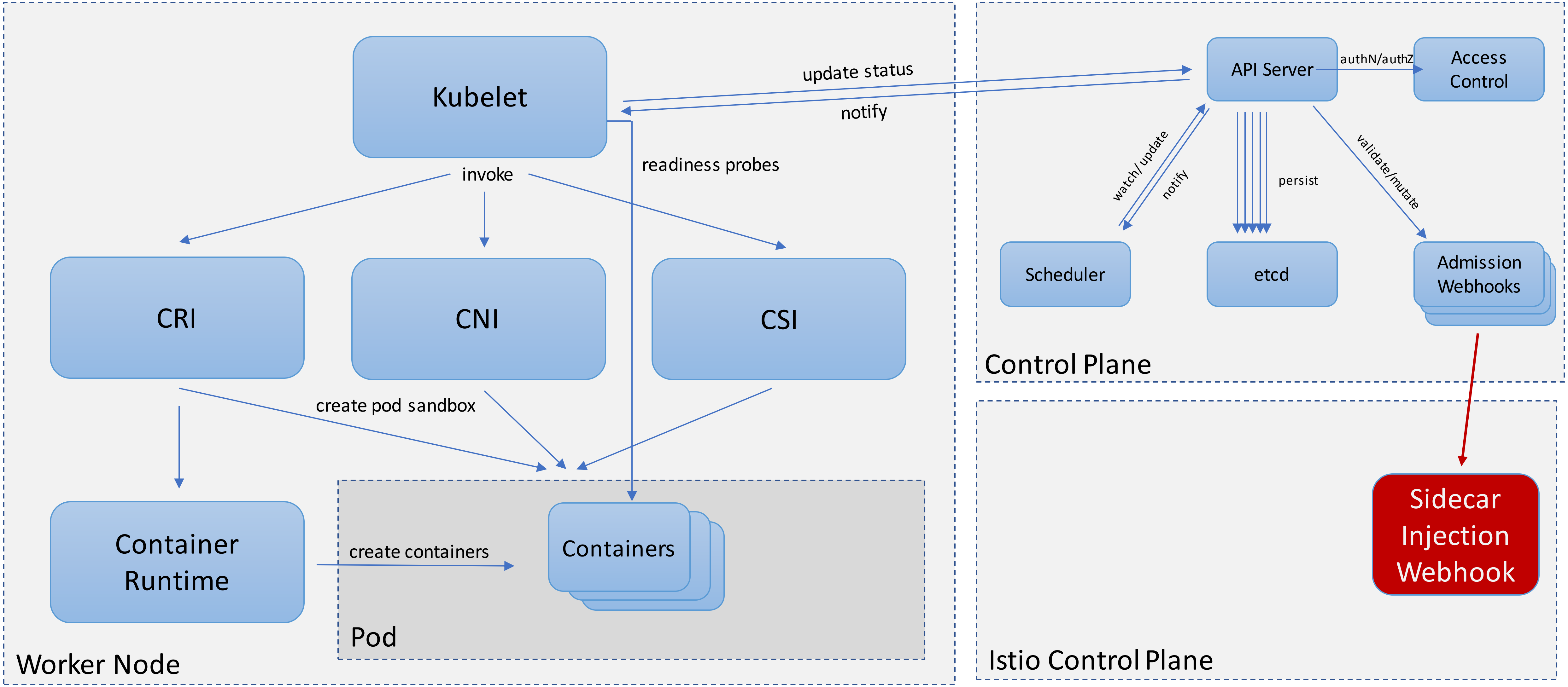


What actually happens when a pod starts?

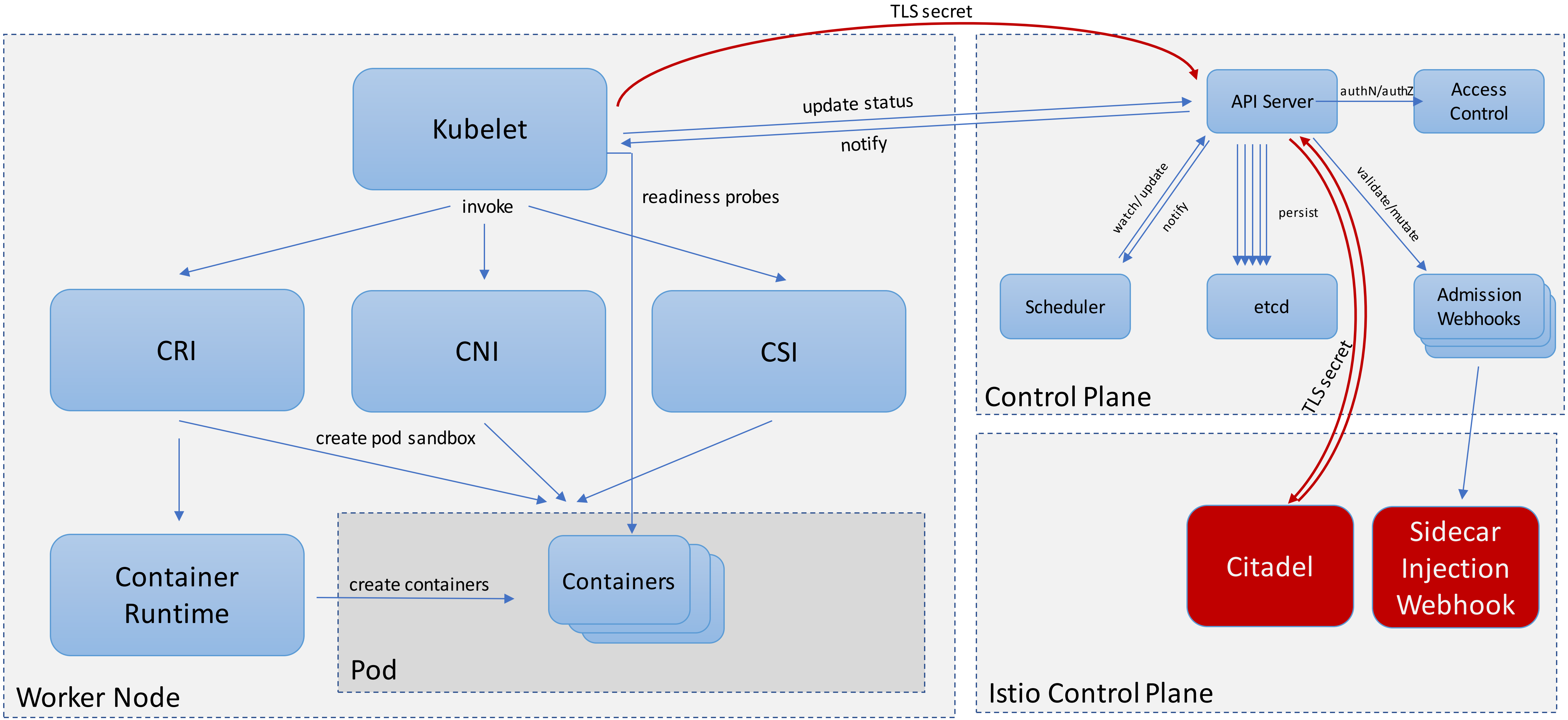
KubeCon CloudNativeCon North America 2019



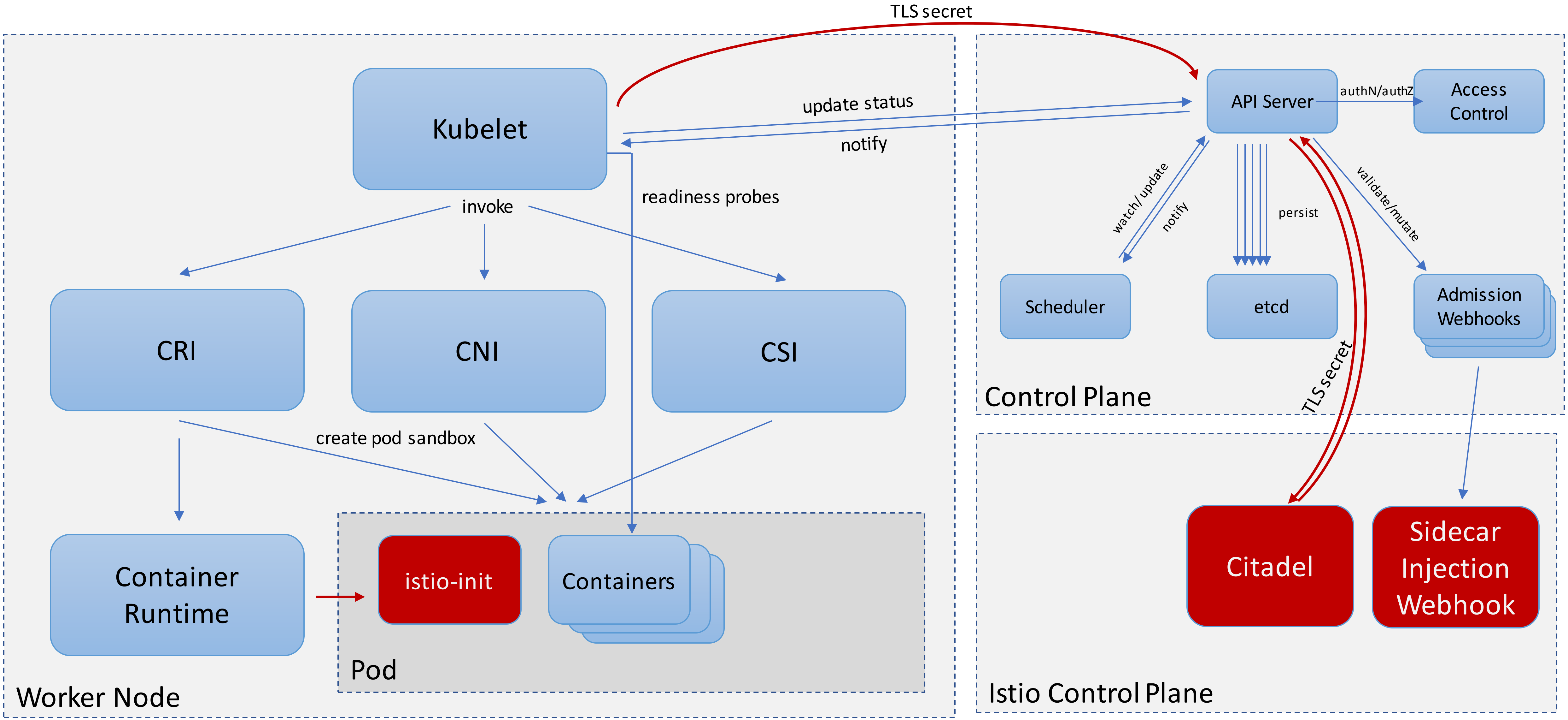
What actually happens when a pod starts?



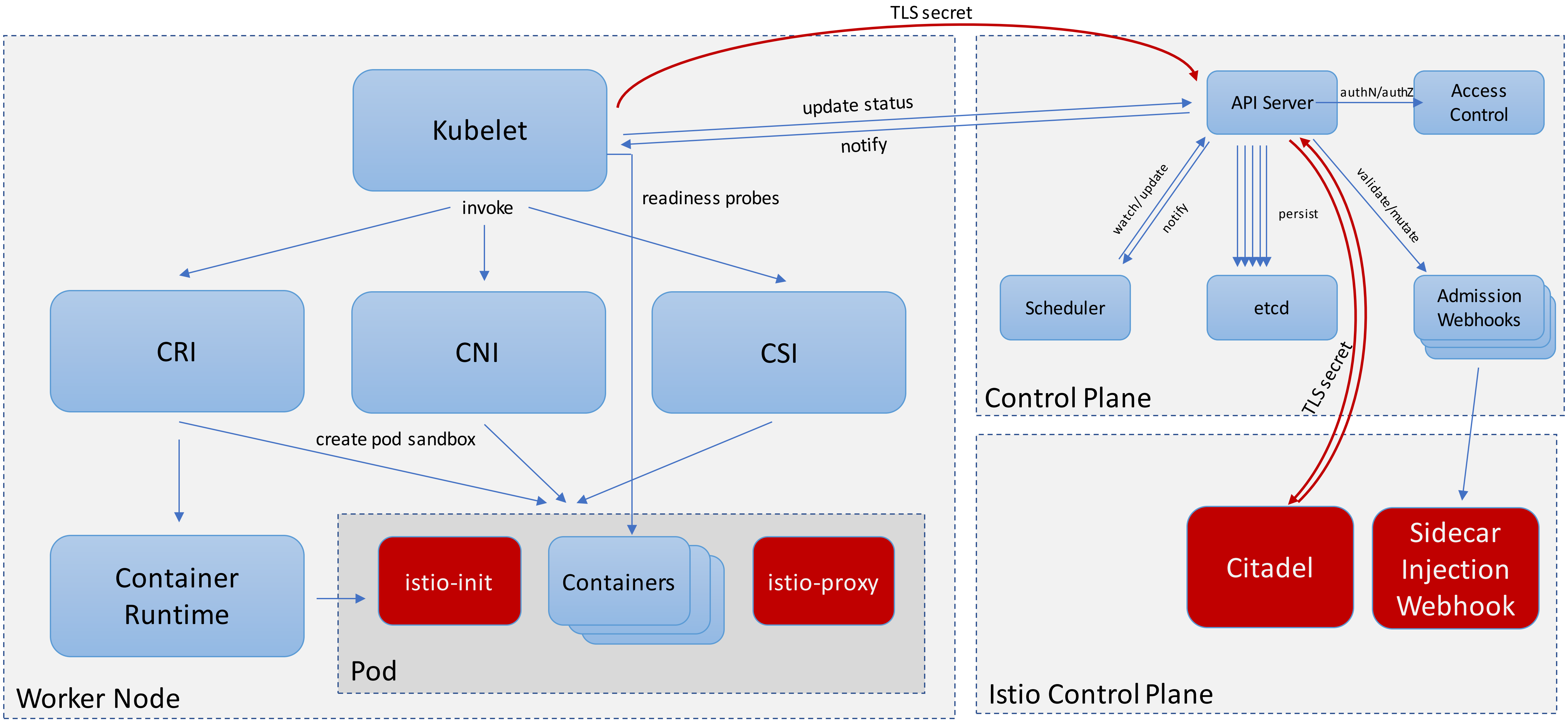
What actually happens when a pod starts?



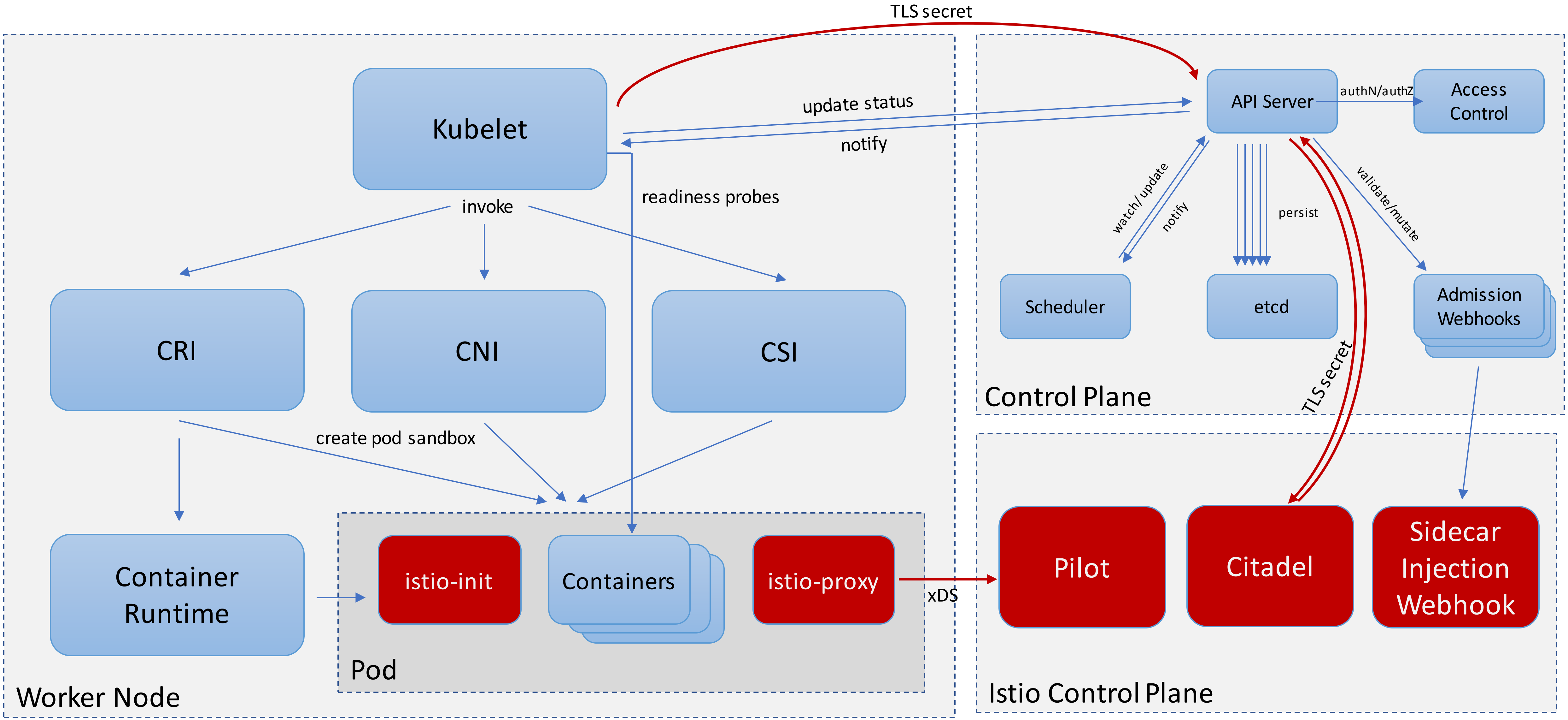
What actually happens when a pod starts?



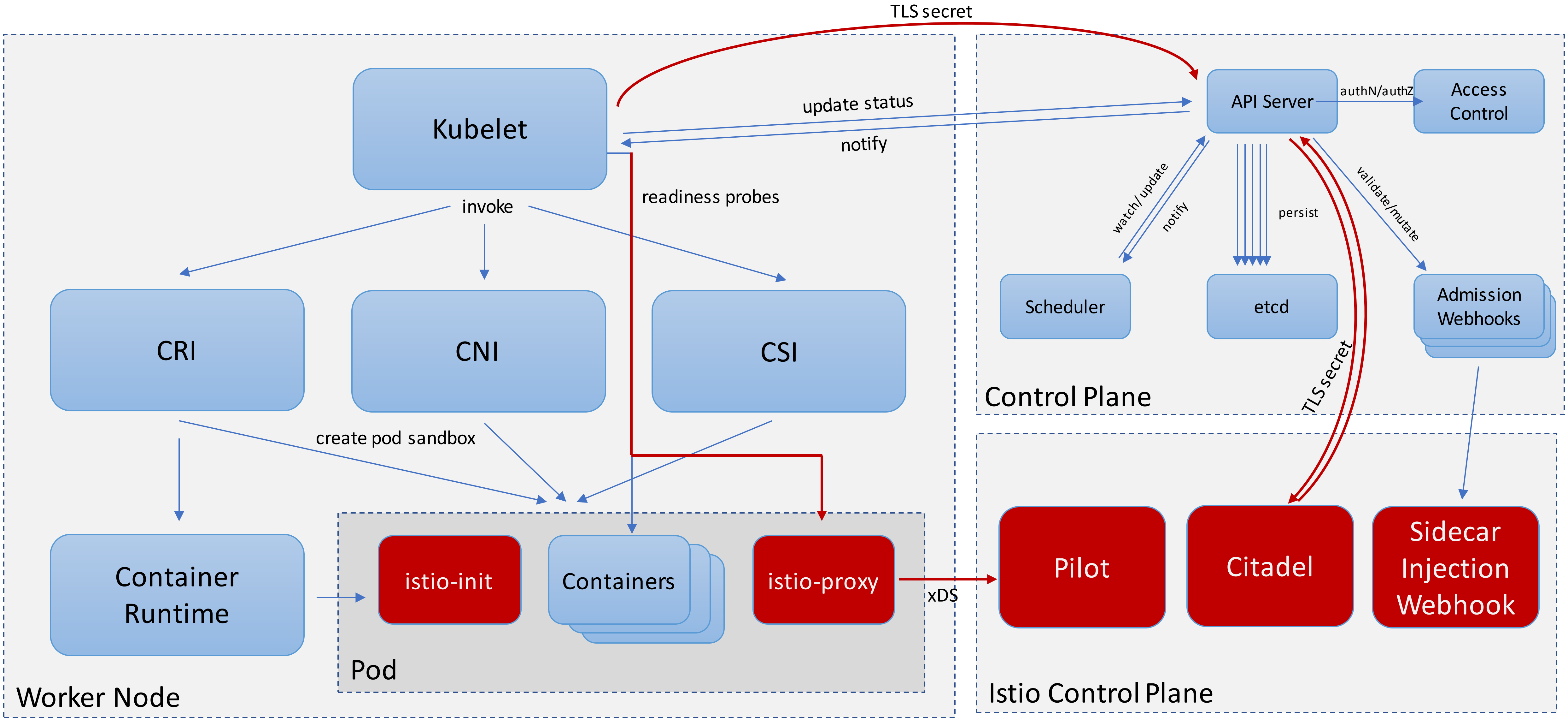
What actually happens when a pod starts?



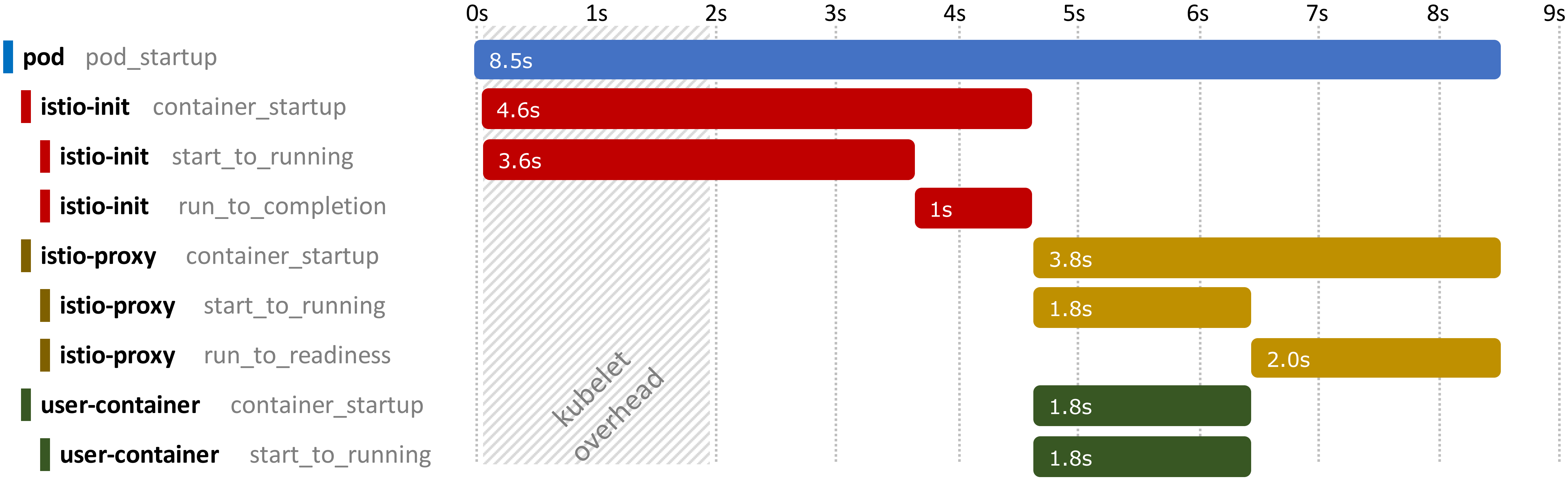
What actually happens when a pod starts?



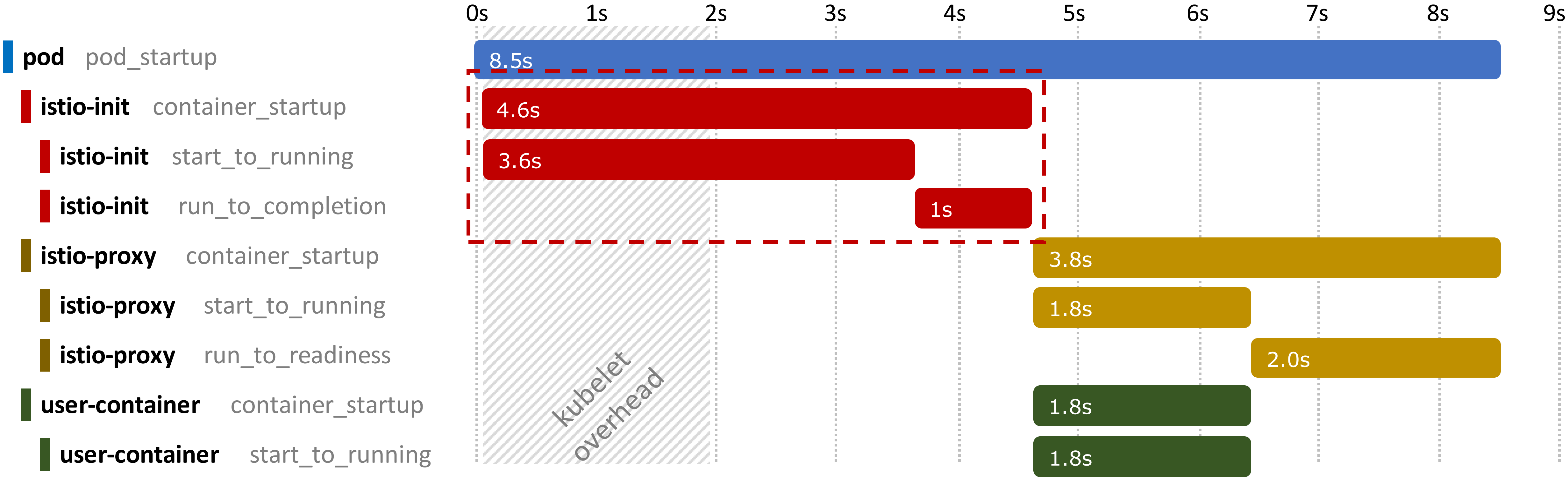
What actually happens when a pod starts?



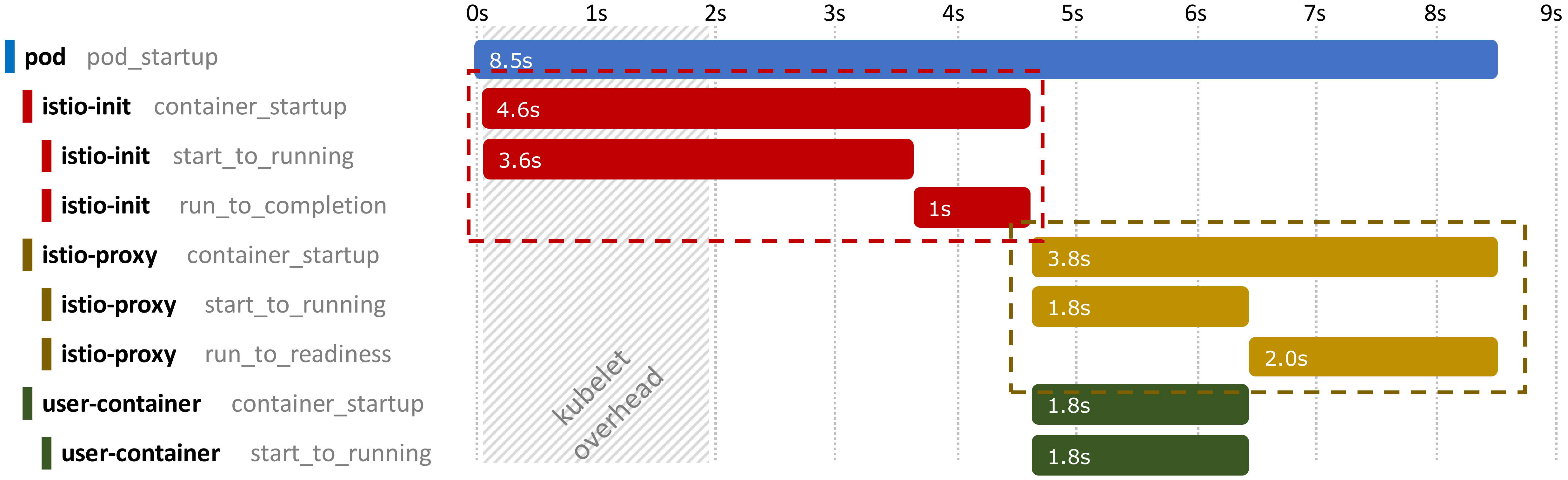
And how much time does it take now?



And how much time does it take now?



And how much time does it take now?

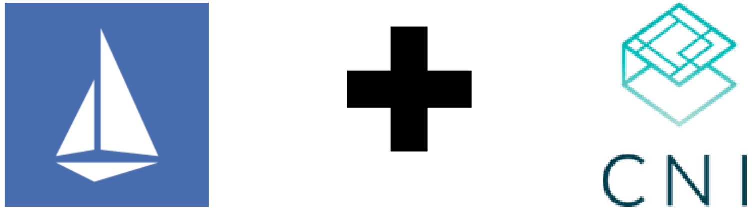


Can we improve?

Improving startup time

- Istio CNI
  - 3s improvement
- Readiness probe tweaks
  - 0.5s-2s improvement
- Static proxy configuration
  - 2s improvement
- Manual sidecar injection
  - Some extra millis for the performance diehard

Istio CNI

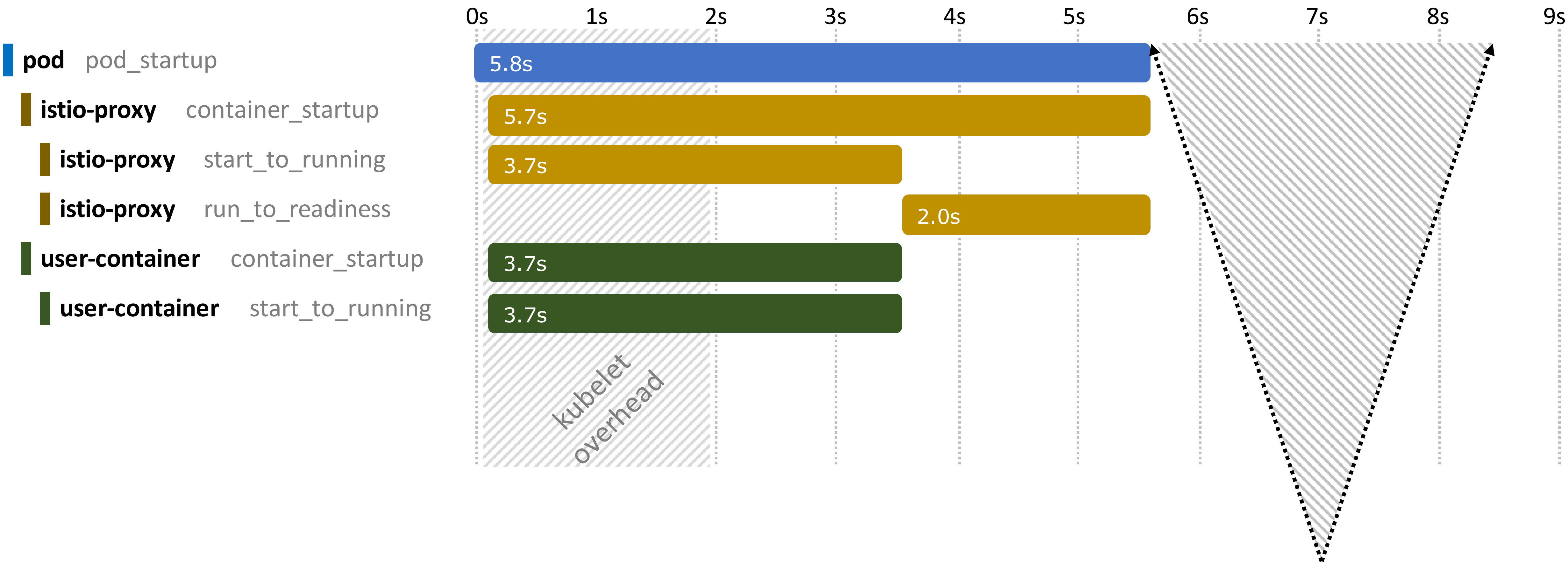


👍 Improved performance (startup time)

👍 Improved security (~~NET_ADMIN privileged pods~~)

👎 No connectivity for init containers

Istio CNI



2.7s improvement!

Readiness probe period

- Istio sidecar container configured with a **readiness probe** with a **2 seconds** period (and a 1 second initial delay).
- Normally, the 2nd probe succeeds.
- Reducing the period to **1 second** can (sometimes) cut down 1 second.
  On average, seems to cut down **400-500ms.**

```
readinessProbe:
    httpGet:
        path: /healthz/ready
        port: 15020
        scheme: HTTP
    initialDelaySeconds: 1
    periodSeconds: 2
    timeoutSeconds: 1
    successThreshold: 1
    failureThreshold: 30
```

Readiness probe period

To configure:

```
$ helm template install/kubernetes/helm/istio
      --name istio --namespace istio-system \
      --set global.proxy.readinessPeriodSeconds=1 | kubectl apply -f -
```

Or, edit the sidecar container template:

```
$ kubectl edit configmap istio-sidecar-injector -n istio-system
```

Or, set the following pod annotation:
readiness.status.sidecar.istio.io/periodSeconds

```
readinessProbe:
  httpGet:
    path: /healthz/ready
    port: 15020
    scheme: HTTP
  initialDelaySeconds: 1
  periodSeconds: 2
  timeoutSeconds: 1
  successThreshold: 1
  failureThreshold: 30
```

Readiness probe of Envoy

Use **Envoy’s** own readiness endpoint instead of **Istio’s pilot-agent**

```
readinessProbe:
  httpGet:
    path: /healthz/ready
    port: 15020
    scheme: HTTP
  initialDelaySeconds: 1
  periodSeconds: 1
  timeoutSeconds: 1
  successThreshold: 1
  failureThreshold: 30
```

```
readinessProbe:
  exec:
   command:
   - curl
   - 127.0.0.1:15000/ready
  initialDelaySeconds: 1
  periodSeconds: 1
  timeoutSeconds: 1
  successThreshold: 1
  failureThreshold: 30
```

- Effect: avoid waiting for xDS config from Pilot
- Can be useful if relying on client retries
- Cuts down some additional 500ms

Readiness probe disabled

Entirely disable the readiness probe:

```
$ helm template install/kubernetes/helm/istio
        --name istio --namespace istio-system \
        --set global.proxy.statusPort=0 | kubectl apply -f -
```

- Not a recommended approach
  - But can demonstrate potential latency saving
- Cuts down some additional 1s
  - 2s compared to out-of-the-box readiness probe

Static proxy configuration

- Replace dynamic configuration (LDS, CDS, EDS, etcDS…) with a static Envoy configuration
  - Fully static/semi-static
- Limited use cases
  - Topology known in advance
  - Knative: route everything via ingress gateway
- For the adventurous-minded
  - Complex to set up (as of now)

Static proxy configuration

Method 1: merge/override default configuration

- Create an Envoy configuration file
  - Name it **custom_bootstrap.json**
- Store in a ConfigMap
  - Same namespace as target pod
- Annotate pod
  - **sidecar.istio.io/bootstrapOverride=<ConfigMap name>**

Static proxy configuration

Method 2: replace configuration file

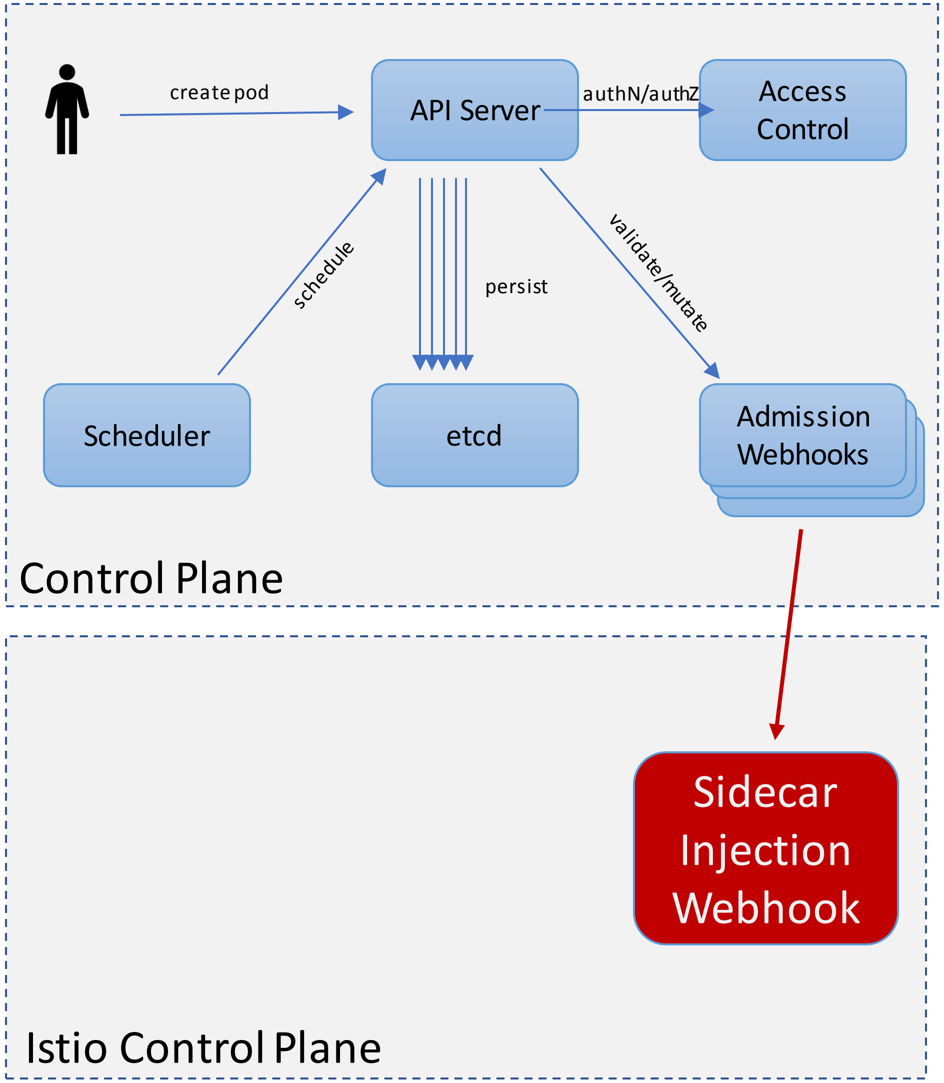
- Create an Envoy configuration file
  - Arbitrary name
- Store in a ConfigMap
  - Same namespace as target pod, but…
- Edit the **istio-system/istio-sidecar-injector** ConfigMap and:
  - Add pod volume pointing to ConfigMap
  - Mount volume to container
  - Add **--customConfigFile** flag to container startup args

Manual sidecar injection

- Istio uses a **mutating admission webhook** to automatically inject the sidecar proxy container into pods
- Applies to any namespace labeled with **‘istio-injection: enabled’**
- Instead, can manually inject pods with **istioctl:**

```
$ istioctl kube-inject -f deployment.yaml | kubectl apply -f -
```

- Can be integrated to CI/CD
- Supports Pods, Deployments, ReplicaSets, DaemonSets, and Jobs.
- Saves additional 15-20ms of startup time.



Thank you!
Questions?