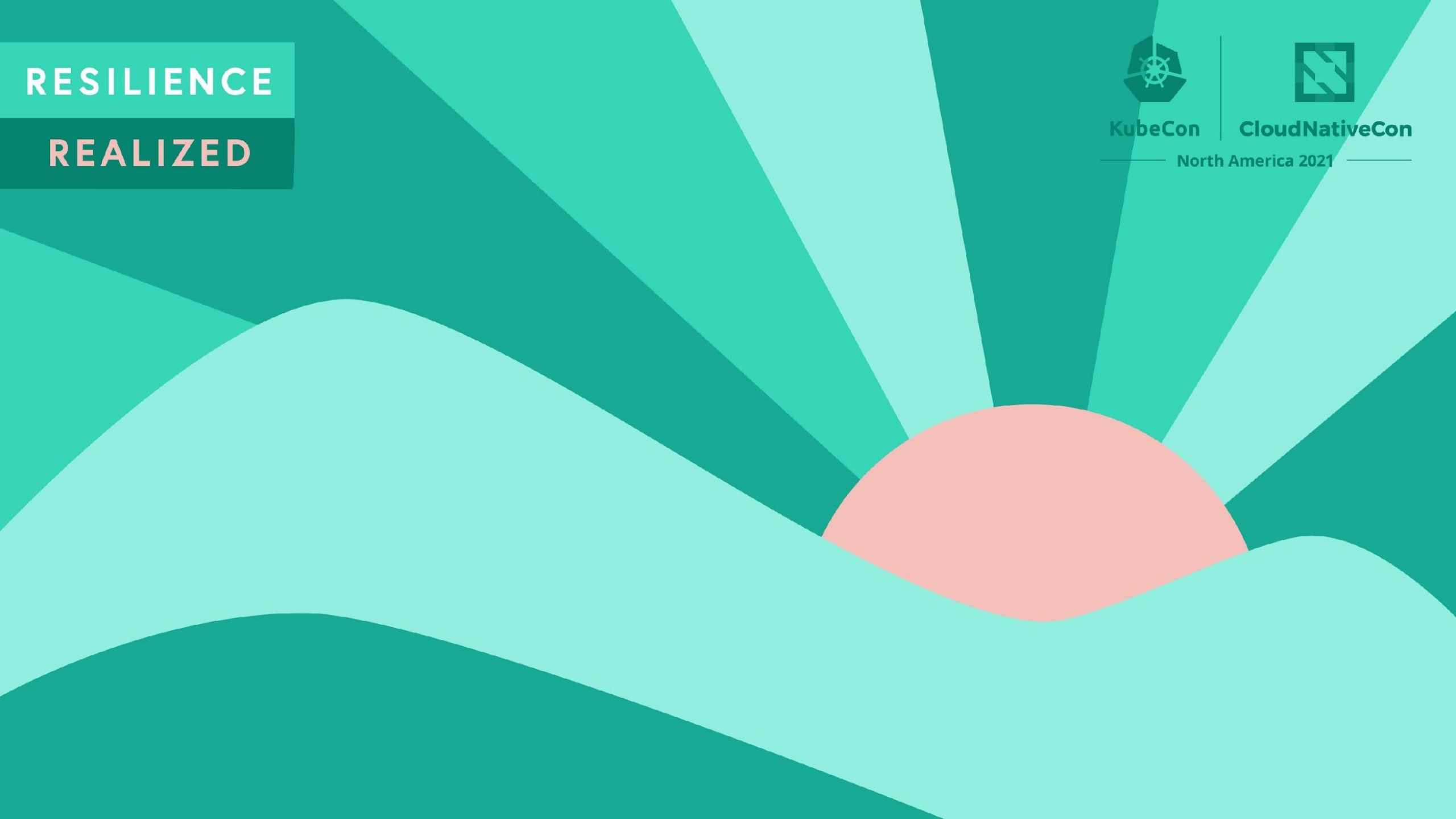
RESILIENCE

REALIZED



KubeCon | CloudNativeCon

North America 2021

RESILIENCE
REALIZED

SIG-NETWORK: Update and Directions

@bowei
@thockin

shortlink: bit.ly/3zt4nZb

SIG-NETWORK charter

Responsible for the Kubernetes network components

- Pod networking within and between nodes (CNI, IPAM, …), ingress and egress
- Service abstractions (service discovery, load-balancing {L4, L7}, …)
- Network policies and access control
- ...and the APIs associated with these functions: Pod, Node, Endpoint/Slice, Service, Ingress, Gateway, NetworkPolicy.

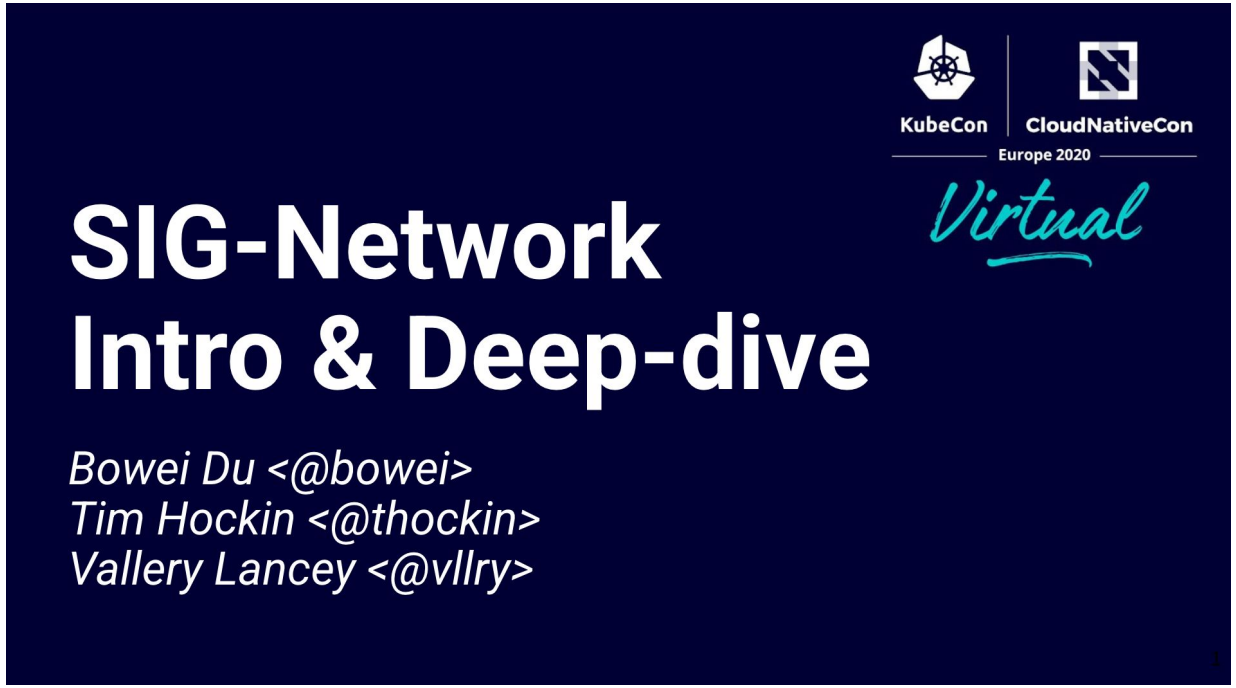
Zoom meeting: Every other Thursday, at 21:00 UTC

Slack: #sig-network (slack.k8s.io)

https://git.k8s.io/community/sig-network

Introductions to K8s Networking

KubeCon CloudNativeCon North America 2021



https://youtu.be/tq9ng_Nz9j8

https://youtu.be/3w-AX4OIdwg

Major themes

Clean up KEY backlog -- one of our focuses has been to get Alpha, Beta KEPs to GA.

Major projects:

- Dual-Stack (now GA in 1.23)
- Gateway API (v1alpha2) for L4/L7
- NetworkPolicy improvments

KEPs, Updates

KEP-1865 : Disable LB NodePorts

Allows user to avoid using up a NodePort when creating a `type=LoadBalancer` service.

GA 1.23

```
apiVersion: v1
kind: Service
metadata:
  name: mixed-protocol
spec:
  type: LoadBalancer
  allocateLoadBalancerNodePorts: false
  ports:
    - name: web
      port: 8080
      protocol: TCP
  selector:
    app: web
```

KEP-2365: Namespace-scoped IngressClass Parameters

Allow `IngressClass` to reference a parameters object that is namespace scoped.

- Addresses common use case for some implementations.

GA 1.23

```
kind: IngressClass
metadata:
  name: external-lb
spec:
  controller: acme.com/ingress
  parameters:
    apiGroup: k8s.example.com
    kind: IngressParameters
    name: external-lb
    namespace: my-params
    scope: Namespace
```

KEP-2433: Topology-Aware Routing

v1.21

- Alpha `topologyKeys` field on `Service` is deprecated.
- Simpler approach: `EndpointSlice` controller allocates endpoints proportionally across zones with hints that proxy implementations can consume
- Each service can opt-in by using an annotation (`service.kubernetes.io/topology-aware-hints`)

v1.22, v1.23

- `TopologyKeys` is now renamed to `DeprecatedTopologyKeys` in `EndpointSlice`
- Topology hints will be Beta in 1.23.

KEP-2595: Expand DNS config

Expand limits on the number of items in the DNS search path as modern resolvers now support > 5.

New limits:

- # of search paths: 32
- Total path length (chars): 2048

Alpha 1.22

```
apiVersion: v1
kind: Pod
  ...
  dnsConfig:
    nameservers:
      - 1.2.3.4
    searches:
      - my.dns.search.suffix
    options:
      - name: ndots
        value: "2"
      - name: edns0
```

KEP-2593: Discontiguous Pod CIDR

Enable the NodeIPAM controller to allocate IPs from multiple, non-contiguous ranges of IPv{4,6} addresses.

- Allow for the cluster admin to add additional IP ranges that can be allocated from dynamically.
- Does not change Node.Spec.PodCIDR behavior -- once allocation is done, this cannot change.

```
kind: ClusterCIDRConfig
me
sp
```

```
kind: ClusterCIDRConfig
metadata:
  name: my-range
spec:
  nodeSelector: ...
  ipv4:
    cidr: 10.0.0.0/20
  ...
```

CVE: Endpoint{Slice} allows cross-NS traffic

CVE-2021-25740: Endpoint & EndpointSlice permissions allow cross-Namespace forwarding

Attack: If a user can create or edit `Endpoint{Slices}` in the Kubernetes API, they can potentially direct a `LoadBalancer` or `Ingress` implementation to expose backend IPs.

Example: If `NetworkPolicy` already trusts the `LoadBalancer/Ingress` source IPs, `NetworkPolicy` can not be used to prevent other namespaces, from bypassing security controls such as `LoadBalancerSourceRanges`.

Mitigation: Remove RBAC allow users access to `Endpoint{Slices}`.


KubeCon | CloudNativeCon

North America 2021

RESILIENCE

REALIZED

Dual Stack IPv{4,6}

Dual Stack (IPv4 + IPv6)

IPv4/IPv6 Dual Stack (and DONE! 🎉)

- Services now support both IPv4 and IPv6
- Cluster migration between `SingleStack` and `DualStack` now possible (within some limits)
- Dual-stack Load Balancing (Services with v4, v6 or both)
- Previous semantics unchanged:
  - Egress IPv4/IPv6
  - A single IPv4 and IPv6 address per pod

KEP-2438: Dual-Stack APIServer

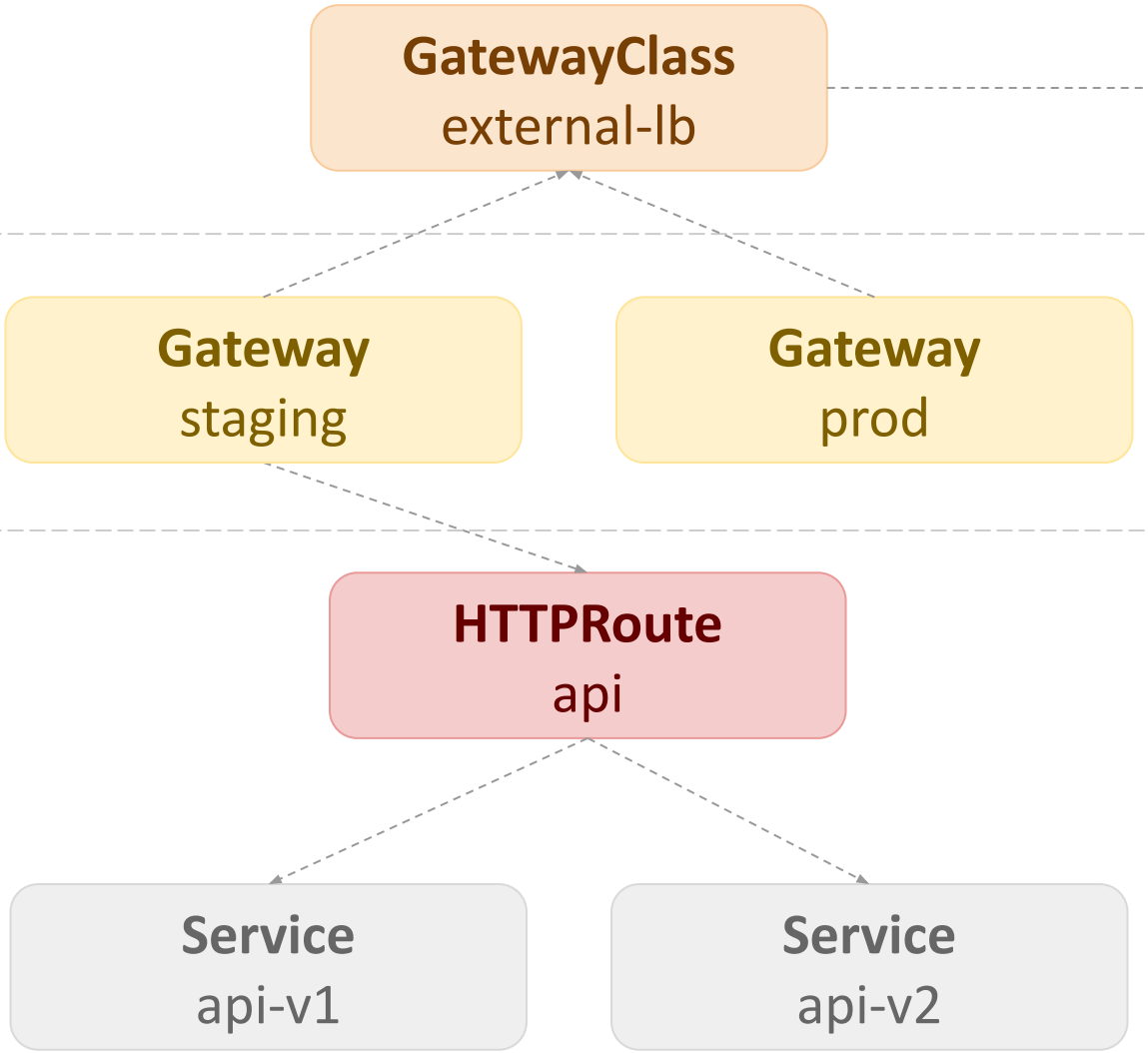
Add dual-stack support to the Kubernetes API Service (Service namespace "default", name "kubernetes")

- kubernetes Service is configured with `ipFamilyPolicy: RequireDualStack`.
- Publish API server endpoints using EndpointSlice
- `rest.InClusterConfig()` will make use of dual-stack when available.
- `KUBERNETES_SERVICE_HOST` env var will remain the same.

RESILIENCE
REALIZED

Gateway API

Gateway API (gateway-api.sigs.k8s.io)



Infrastructure Provider

Provisions cluster and overall environment

Cluster Operator

Cluster-level resources, administration, network management

Application Developer

Application and routing configuration

Gateway API

Steady progress towards v1alpha2!

If v1alpha2 does not hit major issues with the API, it will be lead directly to v1beta1 and GA.

Expect backwards compatibility going forward from v1alpha2.

Release notes

- github.com/kubernetes-sigs/gateway-api/releases/tag/v0.4.0-rc1

Gateway v1alpha2 changes

Moved to the official API group:

- `networking.x-k8s.io` → `gateway.networking.k8s.io`

`Gateway` ↔ `Route` binding

- Gateways select Routes by Kind and namespace, default to same namespace.
- Routes directly reference Gateways they attach to (list of Gateways)

Cross-namespace references managed via `ReferencePolicy`

- New resource: `ReferencePolicy`
- Resources can be referenced from a different Namespace if allowed by the `ReferencePolicy` in the resource's namespace.

Gateway v1alpha2 changes

Design pattern for generic policy attachment and inheritance in the `{Gateway,GatewayClass,Route}` resource graph.

- `BackendPolicy` is now removed as the generic policy attachment is more flexible.

`Route` no longer contains Certificates.

- Rationale: too many edge cases, complexity should be handled by a system outside of Gateway API.
- Major use case replaced by `ReferencePolicy` i.e. Gateway using Certificates from other namespaces.

Many other improvements!
github.com/kubernetes-sigs/gateway-api/releases/tag/v0.4.0-rc1

Gateway v1alpha2 changes

Try out the v1alpha2 API for yourself:

gateway-api.sigs.k8s.io/implementations

RESILIENCE
REALIZED

NetworkPolicy

NetworkPolicy

Working group is hard at work on improvements.

In discussion, not yet in KEP:

- Destinations based on FQDN (vs IP address)
- Source and destination selectors based on Service (vs Pod selectors)
- ClusterNetworkPolicy (but very close)

Beta:

- Port ranges

NetworkPolicy

Added “Port Ranges” to the NetworkPolicy API

Implemented a default namespace label for PLP users writing of network policies

Beta 1.23

```
...
ingress:
  - protocol: TCP
    port: 1000
    endPort: 10000
  - from:
    - ipBlock:
        cidr: 172.17.0.0/16
        except:
        - 172.17.1.0/24
    - namespaceSelector:
        matchLabels:
          kubernetes.io/metadata.name: foo
...
```

NetworkPolicy: ClusterNetworkPolicy

Enable cluster admins the ability to enforce secure by default policies on cluster tenants.

Happy medium of needed functionality without adding too much complexity.

Biggest issue to resolve: Is there inherent complexity, meaning a generic system based on priorities is the best answer:

A. Empower > Deny > Allow > Existing NetworkPolicy rules
B. Empower > Deny > Allow > Existing NetworkPolicy rules
C. Priorities (Override policies based on priority number + priority 0 default)


KubeCon | CloudNativeCon

North America 2021

RESILIENCE

REALIZED

Other activity in the SIG

Other SIG activity

KPNG (kube-proxy NG)

- Looking at moving kube-proxy out-of-tree, clean up, adding functionality

Ingress NGINX

- Reboot of the community meeting, new set of community maintainers

In Progress

KEP project board

github.com/orgs/kubernetes/projects/10

Help wanted!

(Especially to close KEPs in progress from Alpha to GA)

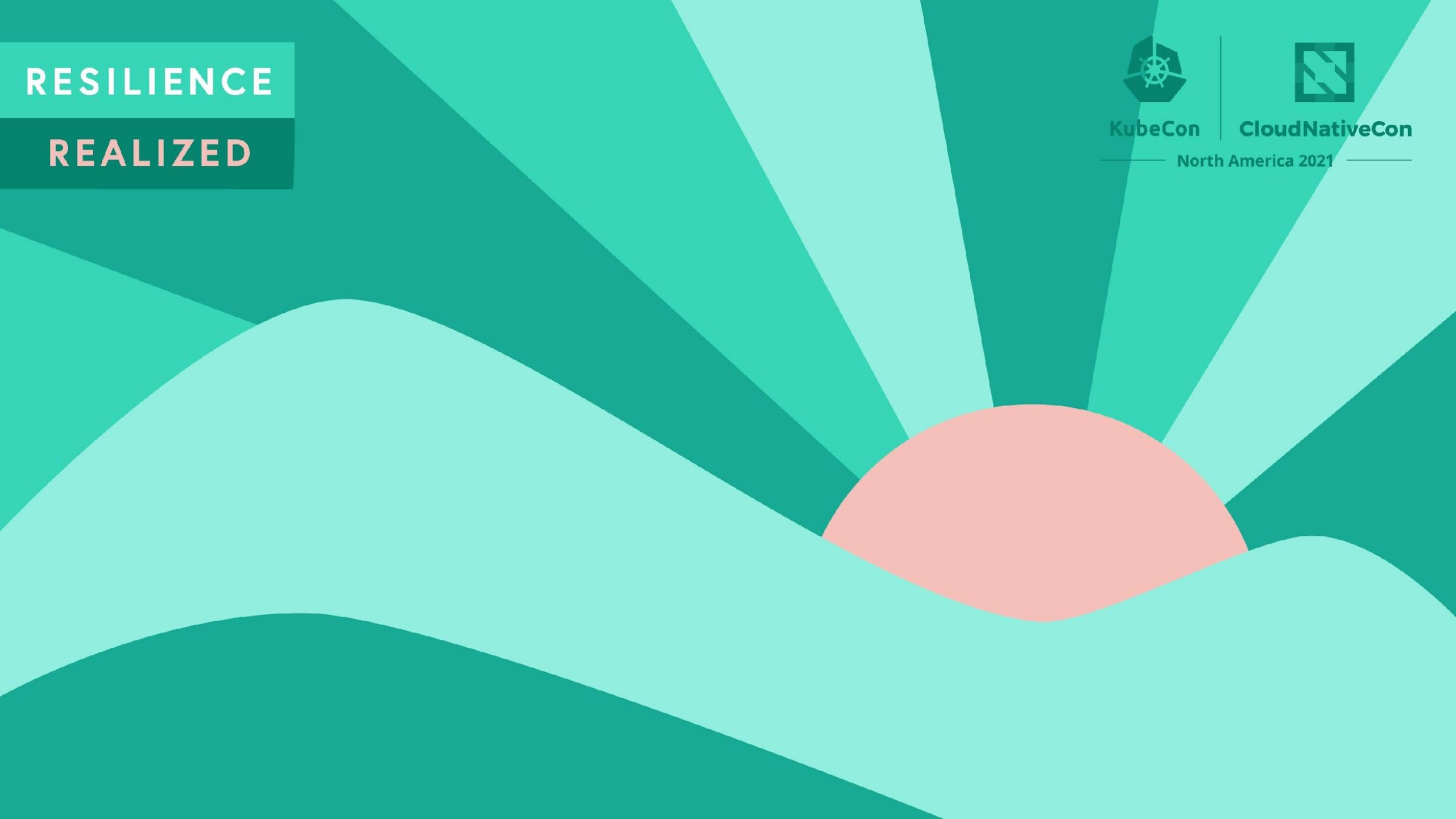
RESILIENCE
REALIZED

Thank you for attending!

Questions and answers

RESILIENCE

REALIZED



KubeCon | CloudNativeCon

North America 2021