KubeCon

North America 2017

CoreDNS Salon

John Belamaric, Infoblox
john@coredns.io

Agenda

- CoreDNS Basics & Roadmap
  - CoreDNS Architecture
  - Available plugins and how to use them
- CoreDNS in Kubernetes
  - Using CoreDNS for your cluster DNS in Kubernetes
  - Status of plans for CoreDNS to replace Kube-DNS as the default cluster DNS
  - The “autopath” plugin
- Advanced Stuff
  - Writing external plugins
  - External *policy* plugin
- General Q&A

Get your swag!

- Hoodies and T-shirts to give away
- **Production users**
  - Add your company to ADOPTERS.md
  - https://github.com/coredns/coredns/blob/master/ADOPTERS.md
  - You can edit it write in GitHub…
  - First come, first served
- Other users
  - Stickers and anything left...

Basics and Roadmap

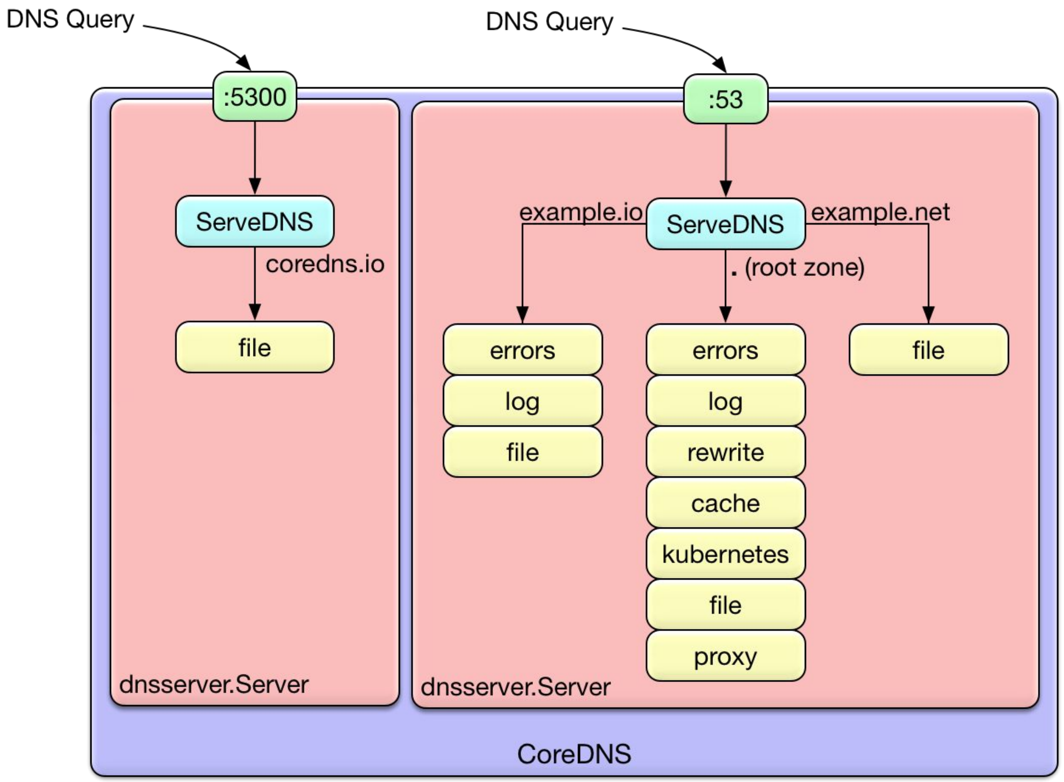
What is CoreDNS?

- Cloud native, authoritative DNS server written in Go
  - *Not* a recursive DNS server (yet...?)
- Successor to SkyDNS2 for dynamic DNS-based service discovery
- Flexible, plugin-based, extensible request pipeline
- Started and led by Miek Gieben, SRE at Google
- Supported by Infoblox and soon to be used in its SaaS offerings
- Hosted as an inception project at CNCF
  - Going for incubation now

https://github.com/coredns/coredns

Architecture

- Features are contained in independent plugins
  - Logging
  - Caching
  - Metrics
  - Many more...
- Queries routed based on zone
- Different plugin chains for different zones



https://github.com/coredns/coredns

Plugins!

- 28 In-Tree, Standard Plugins
  - Built into the standard release images
  - Hosted in the coredns/coredns repository
  - Widely applicable
- External Plugins
  - Live outside the main repo - anywhere you can `go get`
  - Most need a custom build

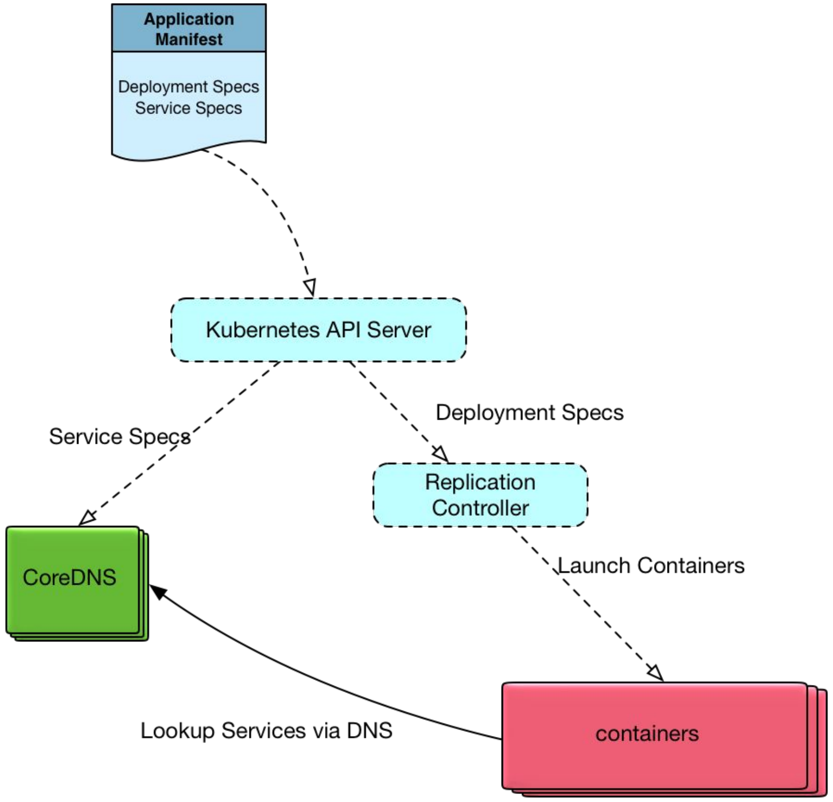
Roadmap

- Kubernetes-driven features
- Community-driven features
- Policy integration (starting as external)
- Telemetry (starting as external)
- Automate DNSSEC

CoreDNS in Kubernetes

CoreDNS Kubernetes Integration

- Single container in the DNS pod
- Kubernetes plugin talks to the API
- Same design pattern as other components
  - CoreDNS listens for changes on the API server
  - New services or endpoints result in new DNS records
  - CoreDNS serves up special names for services, and proxies external requests out
- Of course you can also use other plugins at the same time



https://github.com/coredns/coredns

Features for Kubernetes

- Implements the K8s DNS Spec plus more
- Filter records by namespace - selectively expose namespaces
- Filter records by label selector - selectively expose services
- `endpoint_pod_names` uses Pod names for headless service pods (kubernetes#47992)
- autopath - improves latency of queries
- `pods verified` mode - verify pod exists pod queries
  - Query A record `1-2-3-4.namespace.pod.cluster.local`
  - **kube-dns** always returns 1.2.3.4
  - **CoreDNS** ONLY returns 1.2.3.4 if there is a pod in that namespace with that IP

CoreDNS as Default Cluster DNS

- Planned Schedule
  - Kubernetes 1.9 - Alpha
  - Kubernetes 1.10 - Beta
  - Kubernetes 1.11 - GA
- Links
  - Feature Issue
  - Community Proposal

Autopath - the problem

- Kubernetes has a long DNS search path and ndots value
  - `<namespace>.svc.cluster.local`
  - `svc.cluster.local`
  - `cluster.local`
  - plus the nodes search path
- Enables flexible use of names, but leads to extra queries

```
dnstools# host -v google.com
Trying "google.com.default.svc.cluster.local"
Trying "google.com.svc.cluster.local"
Trying "google.com.cluster.local"
Trying "google.com"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 62752
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4,
ADDITIONAL: 4
...
```

Autopath - the solution

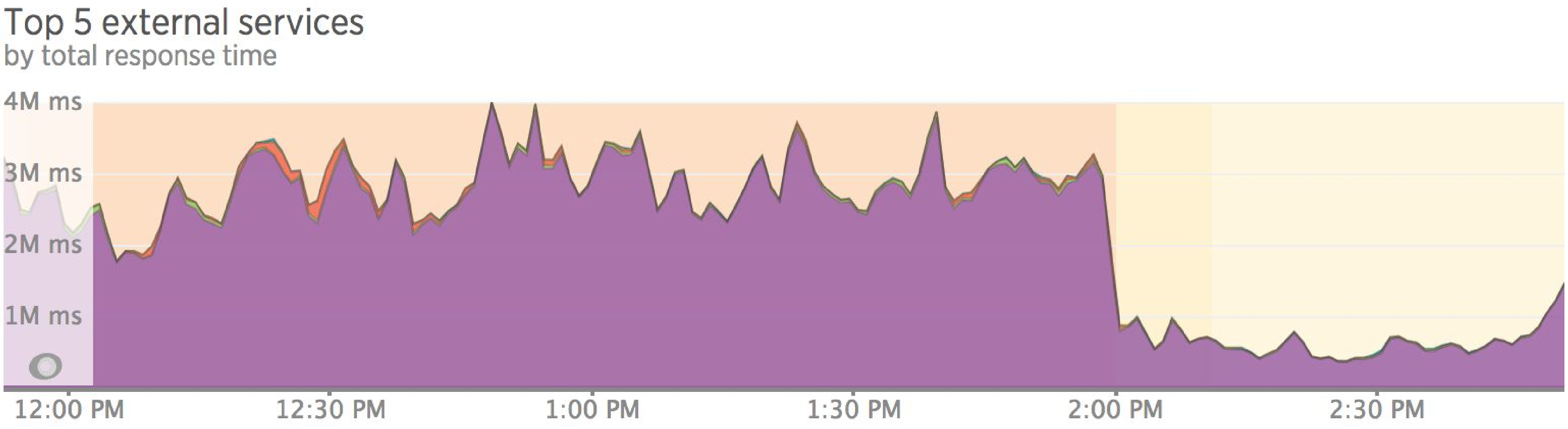
- kubernetes pods verified + autopath
- Since CoreDNS knows the namespace of the source pod IP, it knows the search path
- Execute the search path server-side

```
dnstools# host -v google.com
Trying "google.com.default.svc.cluster.local"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38177
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;google.com.default.svc.cluster.local. IN A

;; ANSWER SECTION:
google.com.default.svc.cluster.local. 13 IN CNAME google.com.
google.com.      13   IN   A    172.217.9.142
...
```

Autopath results….



https://github.com/coredns/coredns

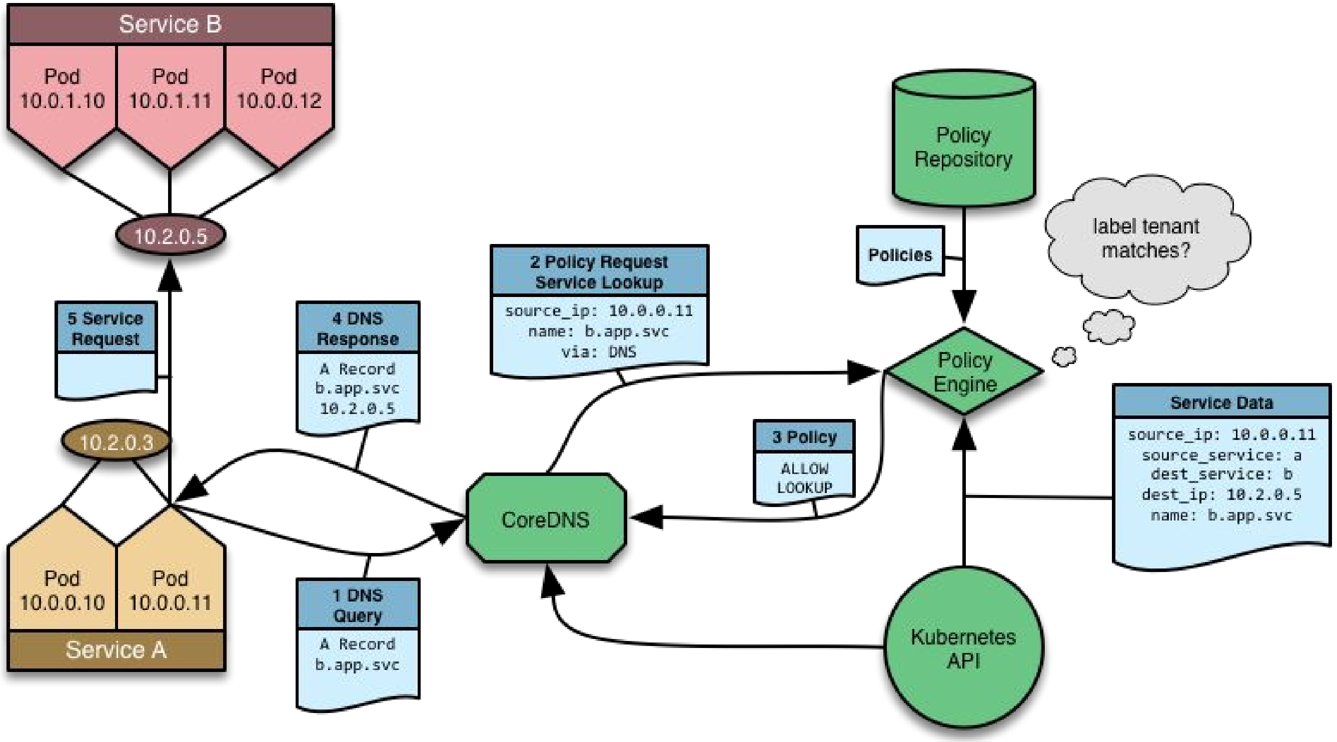
Advanced Stuff

Writing External Plugins

- Check out the tutorial
- Your plugin must:
  - Register itself
  - Parse its setup config
  - Implement `Name()` and `ServeDNS()`
- You must:
  - Modify `plugin.cfg` to point to your plugin
  - Configure its use in your `Corefile`
- Let’s see one!

Policy in K8s Service Discovery

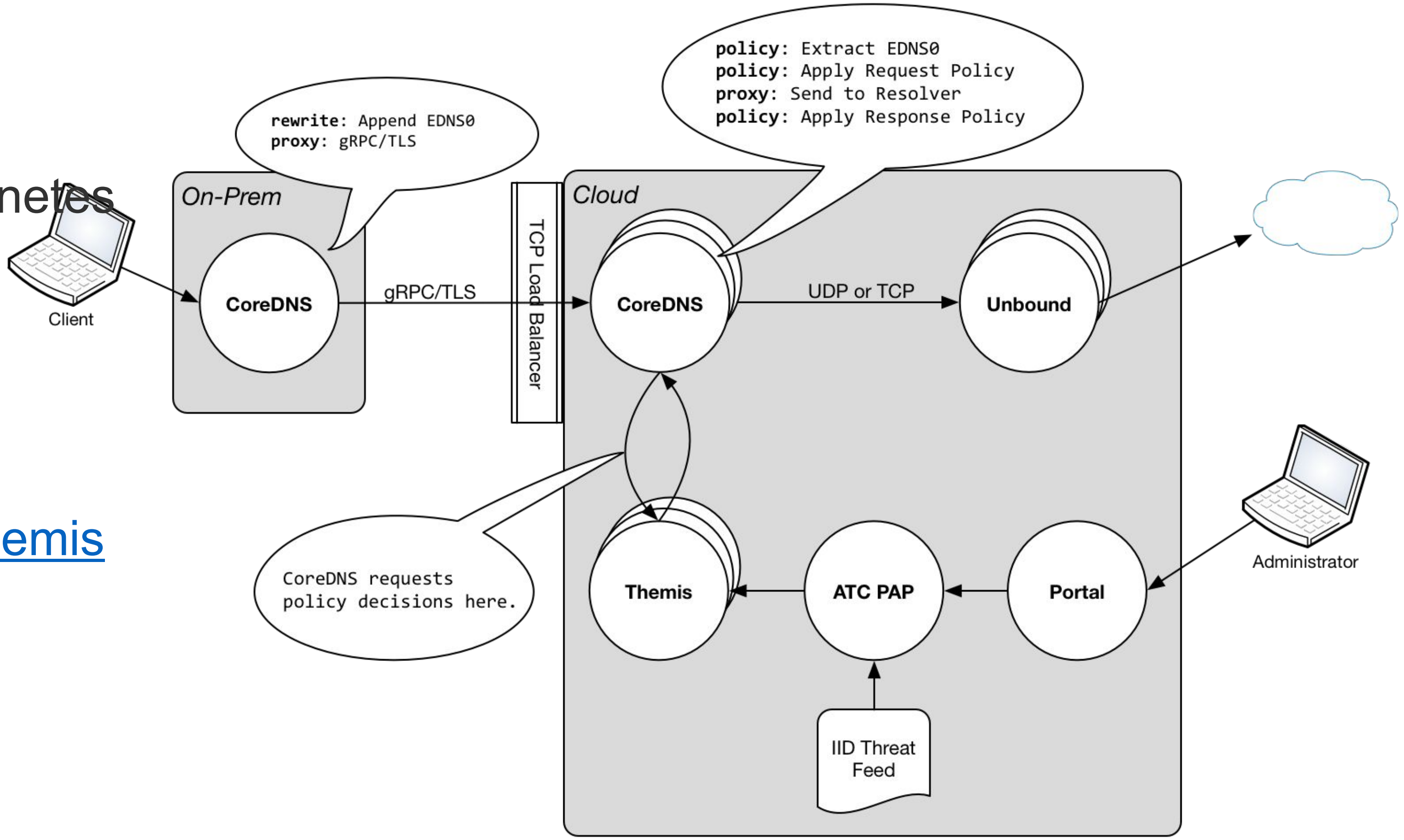
- Insert decision in service discovery
- Example Use Cases
  - Enforce K8s RBAC in DNS
  - Multi-tenant DNS
  - Block if tenants of requesting and responding service differ
  - Block if environment (dev/test/prod) differ
  - Resolve same DNS name to IP of matching tenant and/or environment
  - Redirect to closest instance of requested service
  - Redirect to instance with lowest load
  - Transparently insert intermediaries in service chain
  - Rolling upgrade and version constraints



https://github.com/coredns/coredns

Policy Plugin - Active Trust Cloud

- CoreDNS is more than just Kubernetes service discovery
- Integrates with Infoblox high-performance policy engine Themis to provide ATC features
- https://github.com/infobloxopen/themis



Q & A

Join the CoreDNS community!

Web	https://coredns.io
GitHub	https://github.com/coredns
Slack	slack.cncf.io #coredns
Mailing List	coredns-discuss
Twitter	@corednsio
Docker Hub	coredns/coredns