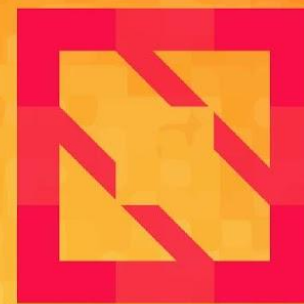




KubeCon



CloudNativeCon

North America 2019





KubeCon



CloudNativeCon

North America 2019

CoreDNS: Deep Dive

Extension Points for Developers

John Belamaric, Google



Three ways to customize CoreDNS

- Rebuilding with external plugins
- Using CoreDNS as a library
- Building your own plugin

Rebuilding with External Plugins

You do not need to know Go to do this!

- “External”
 - Not built into the standard binaries and Docker images
 - Not supported by core team
- No dynamic loading of plugins
 - Plugins are built-in at compile time
 - Controlled by `plugin.cfg`
- Plugin **ordering** is fixed at compile time
- The ones we know about: <https://coredns.io/explugins>

External Plugins

Prerequisites: Docker and a shell

1. Clone CoreDNS
2. Modify `plugin.cfg`
3. Build CoreDNS

External Plugins

1. Clone CoreDNS

```
$ docker run --rm -u $(id -u):$(id -g) -v $PWD:/go golang:1.12 \  
  /bin/bash -c \  
  "git clone https://github.com/coredns/coredns.git && \  
  cd coredns && \  
  git checkout v1.6.5"
```

External Plugins

2. Modify plugin.cfg

```
$ cd coredns  
$ vi plugin.cfg
```

```
...  
dnstap:dnstap  
acl:acl  
firewall:github.com/coredns/policy/plugin/firewall  
...  
whoami:whoami  
on:github.com/mholt/caddy/onevent
```

3. Build CoreDNS

```
$ docker run --rm -v $PWD:/coredns -w /coredns golang:1.12 make
```


CoreDNS as a Library

- Replace the CoreDNS `main.go`
- Allows you to:
 - Reduced the size and memory footprint of the binary
 - Limit the functionality and CLI flags
 - Do extra setup or initialization
- Used, for example, by Node Local DNS in K8s

Example: *dnscached*

- Source is in <https://github.com/coredns/learning-coredns>
- Simple caching DNS server
- Embeds only *bind*, *cache*, *errors*, *forward* and *log* plugins
- CLI args to generate a Corefile internally

Writing a Plugin



KubeCon



CloudNativeCon

North America 2019

- Three categories of plugins
- Best practice: stick to one of these in your plugin
- Backends
 - Source of data
 - *file, forward, hosts, clouddns, template, kubernetes*
- Mutators
 - Modify the inbound request, the outbound response, or both
 - *acl, cache, rewrite, nsid*
- Configurators
 - Modify the internal state or functioning of CoreDNS
 - *bind, log, health, ready*

Four functions



KubeCon



CloudNativeCon

North America 2019

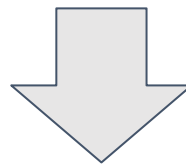
- Name - literally, just returns the name of the plugin
- ServeDNS - request handling
- `init` - register your plugin with Caddy
- `setup` - parse your config

Example: There can be only one!

- onlyone plugin from *Learning CoreDNS*
- Filters out all but one of specific record types

```
onlyone [ZONES...] {  
    types TYPES  
}
```

example.com.	18298	IN A	93.184.216.34
example.com.	18298	IN A	93.184.216.35
example.com.	18298	IN A	93.184.216.36



example.com.	18298	IN A	93.184.216.35
--------------	-------	------	---------------

Functions: Name and init

onlyone.go

```
func (o *onlyone) Name() string { return "onlyone" }
```

setup.go

```
func init() {  
    caddy.RegisterPlugin("onlyone", caddy.Plugin{  
        ServerType: "dns",  
        Action:      setup,  
    })  
}
```


Function: setup



KubeCon



CloudNativeCon

North America 2019

setup.go

```
func setup(c *caddy.Controller) error {
    t, err := parse(c)
    if err != nil {
        return plugin.Error("onlyone", err)
    }

    dnsserver.GetConfig(c).AddPlugin(func(next plugin.Handler) plugin.Handler {
        t.Next = next
        return t
    })

    return nil
}
```

Function: ServeDNS



KubeCon



CloudNativeCon

North America 2019

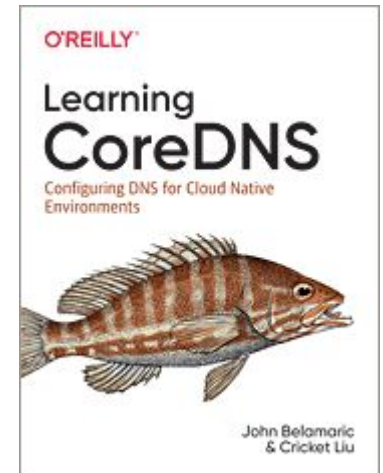
- Let's look at it in [GitHub](#)
- It will be more readable there

Resources

- Plugin how-to: <https://coredns.io/manual/toc/#writing-plugins>
- GitHub: <https://github.com/coredns/coredns>
- [Learning CoreDNS](#), John Belamaric & Cricket Liu, O'Reilly Media
 - Infoblox is giving away 60 copies at their booth tomorrow
 - Thursday at 11:00, Sails Pavilion, Booth G7
 - <https://github.com/coredns/learning-coredns>
- Slack: **#coredns** on <https://slack.cncf.io>

Be sure to come to our session tomorrow:

CoreDNS: Beyond the Basics, Thursday at 3:20pm





KubeCon



CloudNativeCon

North America 2019

Q & A

