

OPENedX[®] 2019



Configuration Clarified

Removing roadblocks to getting started with OpenEdX

Presented by:
Feanil Patel
Fred Smith

Feanil

- feanil@edx.org
- @feanil just about everywhere
- 5.5 years at edX Inc.

Fred

- fred@edx.org
- @derf in slack
- 5 years at edX Inc.

Outline

- Intro
- What is Configuration?
- How Settings work today
- What's Changing
- How does it impact you
- Recap
- Questions

What is Configuration?



Ansible Code that builds OpenEdX

- How Edx.org configures OpenEdX
- Tooling that builds, installs, and configures services
- Takes 45 minutes to build edx-platform from base Ubuntu.
- Needs to be run to change any settings

How does it work Today?



How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`

How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`
 - Create a default variable named `EDXAPP_MY_NEW_SETTING`

How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`
 - Create a default variable named `EDXAPP_MY_NEW_SETTING`
 - Assign it to an app variable named `MY_NEW_SETTING`

How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`
 - Create a default variable named `EDXAPP_MY_NEW_SETTING`
 - Assign it to an app variable named `MY_NEW_SETTING`
 - Nest your setting under `lms_env_config`

How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`
 - Create a default variable named `EDXAPP_MY_NEW_SETTING`
 - Assign it to an app variable named `MY_NEW_SETTING`
 - Nest your setting under ~~`lms_env_config`~~

How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`
 - Create a default variable named `EDXAPP_MY_NEW_SETTING`
 - Assign it to an app variable named `MY_NEW_SETTING`
 - Nest your setting under ~~`lms_env_config`~~
`generic_env_config`

How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`
 - Create a default variable named `EDXAPP_MY_NEW_SETTING`
 - Assign it to an app variable named `MY_NEW_SETTING`
 - Nest your setting under ~~`lms_env_config`~~
~~`generic_env_config`~~

How a setting gets created

`github.com/edx/configuration`

- `playbooks/roles/edxapp/defaults/main.yml`
 - Create a default variable named `EDXAPP_MY_NEW_SETTING`
 - Assign it to an app variable named `MY_NEW_SETTING`
 - Nest your setting under ~~`lms_env_config`~~
~~`generic_env_config`~~
`generic_auth_config`

So how does the app
know about it?

edx-platform

- `aws.py`
 - load `common.py`
 - load two json files that you wrote with ansible
 - arbitrarily choose which file overrides which setting
 - ``JWT_AUTH``
 - `env.json`
 - ``SESSION_INACTIVITY_TIMEOUT_IN_SECONDS``
 - `auth.json`

This is *really* annoying.

Footprint of a setting

- Levels of Indirection: 4
- Number of Repos Touch: 3
- Number of defaults: 3

Things need to change.

What's Done?



Move edx-platform config to YAML

- Ansible generates a single YAML file per service
 - lms.yml
 - studio.yml
- Edx-platform reads yaml with production.py
- Now edx-platform matches other services

What's Next?



Move App defaults into Django

- One place for defaults.
 - In the application repo
- Everything is overridable with your YAML file
- Your YAML override values directly correspond to entries in the django settings dictionary

Impact



New Settings File

- aws.py => production.py
- Plugins
 - See DEPR-14 for migration path
 - We've already migrated all default plugins

Move from JSON to YAML

- In Juniper
 - Ansible will stop generating your app config file
 - We will provide migration documentation

App is Built Using Make Targets

- Single workflow for
 - Ansible
 - Dockerfiles
 - manual installs

Recap

One Story

The apps read config from a yaml file and override any defaults set in your app code.

App Configuration

- No ansible in your app config.
- Edx-platform and IDAs load settings the same way

Enable better choices up and down the entire ecosystem

Ansible, Dockerfile, Salt, Garfl, etc.

Thank You
Questions?

How can you help?

INCR-25