Salty WSGI Documentation

Release 1.0

Andrii Kurinnyi

Sep 05, 2017

Contents

Salty WSGI is a set of Saltstack configuration states for easy deployment of Python WSGI apps (Django, Flask, etc) with Supervisord, Nginx, Virtualenv.

It is oriented towards making a continuous deployment of Django/WSGI apps as simple as possible.

CHAPTER 1

Initial deployment:

- Place the list of your dependencies into `requirements.txt`.
- Create `Procfile` with services that needs to be running (Gunicorn, Celery, etc.)
- Copy `salt/roots/pillar/data.sls.example` -> `salt/roots/pillar/data.sls`, `salt/roots/pillar/top.sls.example` -> `salt/roots/pillar/top.sls`. And fill in your values in `data.sls`. Read more in *Pillar configuration*.
- Provision your virtual machine with Saltstack and a set of Salty WSGI configs
- Add your deployment repo to git remotes

```
$ git remote add deploy {user}@{server}:.repos/{project_name}.git
```

- Push your code to the server

```
$ git push deploy master
```

- Run `syncdb` or similar command for creating your database
- Your website should be up and running.

CHAPTER 2

Updating your website:

- Just push your code to the server

```
$ git push deploy
```

CHAPTER 3

Read more about how to start with Salty WSGI:

Quickstart

The fastest way to start using Salty WSGI is to try it with Vagrant.

First you will need to install the latest version of Vagrant from its Downloads page.

Then install Salty Vagrant by running a command

```
$ vagrant plugin install vagrant-salt
```

Copy `salt/roots/pillar/data.sls.example` -> `salt/roots/pillar/data.sls`, `salt/roots/pillar/top.sls.example` -> `salt/roots/pillar/top.sls`. And fill in your values in `data.sls`. Read more in *Pillar configuration*.

Then run `vagrant up` and you should be good to go.

Pillar configuration

Configuration data is kept in Salt Pillar, in files `data.sls` and `top.sls`. The:file:*top.sls* file is responsible for the fact which server gets which *pillar* data. It matters mostly if you are going to have multi-server setup. For our simple example it just sends *pillar* from `data.sls` to all servers.

```
base:
  '*':
    - data
```

The `data.sls` file is much more interesting. It contains data about things like under which user all our processes are going to run, who are going to have access to deployment, which projects are we going to deploy, etc.

```
main_user:
  companyname
```

```
deployers:
  - github: yourgithubusername

projects:
  - name: projectname
    branch: master
    django_settings: projectname.settings
    requirements: requirements.txt
    port_base: 5000
    nginx:
      fqdn: www.example.com
      default: False
      upstreams:
        - localhost:5000

nginx:
  workers: 4
```

Here `main_user` is a username of Unix user which will own all your web processes. It doesn't have sudo access by default, so even if your web application gets compromised, attacker wouldn't get a control of the whole server.

The second variable `deployers` is a list of GitHub usernames for the people who are going to deploy your project. The *Salty WSGI* will automatically pull their public keys from GitHub, and place them in the `.ssh/authorized_keys` of your `main_user`.

Then there is a list of your `projects`.

`name` *Salty WSGI* will create a bare repo at `.repos/name.git`, working directory at `name` and virtual environment at `.virtualenvs/name-env`

`branch` Name of a git branch, which you are going to use for deployment. By default deployment repository is created by cloning `master` branch from the bare repo.

`django_settings` Is required only for Django projects, and should contain a name of your settings module.

`requirements` Path to your `requrements.txt` file relatively to the project's root.

`port_base` This variable is used to generate Supervisord config from your `Procfile`.

`nginx` Some variables for your Nginx site configuration. Make sure that only one project has *default* value as `True`.

The last variable allows you to set a number of Nginx workers.

CHAPTER 4

Indices and tables

- genindex
- modindex
- search