How to build a FreeBSD CI/CD environment based on pot containers

Luca Pizzamiglio
pizzamig@FreeBSD.org
FOSDEM 2019
whoami(1)

- Luca Pizzamiglio aka pizzamig@
- FreeBSD enthusiast
- Port committer since August 2017
- Building packages at trivago
Motivations

**CI/CD is a well established best practices in any modern software development process**

**Growing interest on build/test software on FreeBSD**

- Improve portability
- Provide FreeBSD support
- Provide artifacts for FreeBSD

**Lack of FreeBSD support on all major CI systems**

Exception for Cirrus CI

- VM based support to FreeBSD
Ugly workarounds

FROM wezm/port-prebuilt-freebsd11@sha256:43553e2265ec702ec72a63a765df333f56b1858b898e69385749e96d8624e9b9
RUN apt-get update
RUN apt-get install -y --no-install-recommends \
qemu genext2fs xz-utils
RUN apt-get install -y curl ca-certificates gcc
ENTRYPOINT ['"sh"]
ENV PATH=$PATH:/rust/bin 
QEMU=2018-03-15/FreeBSD-11.1-RELEASE-amd64.qcow2.xz 
CAN_CROSS=1 
CAR60_TARGET_X86_64_UNKNOWN_FREEBSD_LINKER=x86_64-unknown-freebsd11-gcc

How to build a FreeBSD CI/CD environment

2019-02-02
My motivation 1/2

Beta

- pizzamig released this just now

- Assets
  - Source code [zip]
  - Source code [tar.gz]

beta beta!
Then, what?

https://github.com/pizzamig/freebsd-ci

A command line tool to locally run a build process on FreeBSD

Written in Rust

Still a lot limitations:

- Supports projects stored in github
- Supports rust projects

freebsd-ci -U username -P project [-T 1.0]
FreeBSD-ci

1. Download a github project and put it in a ZFS dataset
2. It parses the yaml file with instructions
3. From the images catalog, it clones the appropriate image and attach the ZFS dataset
4. It generates the build script
5. Run the build
6. Destroy the image
7. Revert the ZFS dataset and go back to point 3.
How to build a FreeBSD CI/CD environment
The yaml file

```yaml
os: FreeBSD # the operating system
    # FreeBSD is the only one supported
FreeBSD: # The FreeBSD version to use to build
    - '11.2'
    - '12.0'

update: true # Run an update of the image before to build
    # Update packages, toolchain, and so on
language: rust # The project language

rust: # Which rust variant use to build
    - stable
    - nightly

no_deploy: # Which combination shouldn’t be deployed
    rust:
        - nightly
```

How to build a FreeBSD CI/CD environment
The build script template

```bash
#!/bin/sh
export HOME=/root
export PATH=/sbin:/bin:/usr/sbin:/usr/bin
PATH=$PATH:/usr/local/sbin:/usr/local/bin
PATH=/root/.cargo/bin:$PATH

if {{ update }} ; then
    rustup update
    pkg upgrade -y
fi

cd /mnt
if ! cargo clippy --release ; then
    exit 1
fi
if ! cargo build --release ; then
    exit 1
fi
if ! cargo test --release ; then
    exit 1
fi

if {{ upload }} ; then
    cargo install --path . -f
    tgt_dir="{{ os_family }}-{{ os_version }}-{{ project }}"
    tarball="{{ tarball }}"
    mkdir $tgt_dir
    mv $HOME/.cargo/bin/{{ project }} $tgt_dir
    tar zcf ${tarball} $tgt_dir
    if {{ delete_asset }} ; then
        curl -H "Authorization: bearer {{ token }}"
            -X DELETE
            https://api.github.com/repos/{{user}}/
            {{project}}/releases/assets/{{asset_id}}
    fi
    curl -H "Authorization: bearer {{ token }}"
        -H "Content-Type: application/gzip"
        -X POST
        --data-binary @$tarball
        https://uploads.github.com/repos/{{user}}/
        {{project}}/releases/{{release_id}}/assets?name=${tarball}
fi
exit 0
```
Images catalog: the challange

We provide some pot flavors to generate images

#!/bin/sh

[ -w /etc/pkg/FreeBSD.conf ] && sed -i '' 's/quarterly/latest/' /etc/pkg/FreeBSD.conf
ASSUME_ALWAYS_YES=yes pkg bootstrap
touch /etc/rc.conf
sysrc sendmail_enable="NONE"

pkg install -y ca_root_nss curl

fetch -o /root/rustup.sh https://sh.rustup.rs
sh /root/rustup.sh -y --default-toolchain stable
export PATH="$HOME/.cargo/bin:$PATH"
echo setenv PATH $HOME/.cargo/bin:'$PATH' >> $HOME/.cshrc

rustup component add clippy-preview
rustup component add rustfmt

pkg clean -aqy
Next steps

Extend support

More languages
More platforms
Better logging

Remote image catalog

A system to download images from a remote catalog

Adopt an orchestration framework

Nomad is a good candidate
Thanks!

Thanks a lot!

Questions?