

# **Building And Utilizing Purpose-Built GNU/Linux Distribution Images Using Mkosi**

#### **Akashdeep Dhar**

Red Hat Community Platform Engineering Elected Representative, Fedora Council







## Motivation



#### **First Interaction With Minimalism**



Setting up Arch Linux distribution manually

**USING PACSTRAP** 



Picking packages and services that you need and

**NOTHING MORE** 



Enabling distribution support in wide spectrum of

**PHYSICAL HARDWARE** 



Creating bespoke configuration settings catered to

**CUSTOM PURPOSES** 

#### **Tools From Other Distributions**



Debian Linux

#### **Debootstrap**

- Installs the base system into a subdirectory
- Needs access to the Debian Linux repositories
- Usable on another operating systems
- Supports creating images for other architectures
- Alternatives with Chroot support are available



Fedora Linux

#### Supermin

- Creates minified purpose-built appliances
- Part of Libguestfs suite for virtual machines
- Usable on another operating systems
- Can build QEMU-runnable virtual machines
- Previously was called Febootstrap until v3.x

### **Exploring Distro Agnostic Tools**



Supports various cloud-centric distributions

#### **Cloud-Init**

- Requires a custom template image to start with
- Used in Ubuntu Linux, CentOS Stream etc.
- Runs on first boot for initializing installations
- Configures users, partitions, networking etc.
- Uses a YAML-styled configuration format



Supports various container-optimized distributions

#### Ignition

- Requires a custom template image to start with
- Used by Fedora CoreOS, Red Hat CoreOS etc.
- Runs on first boot for initializing installations
- Configures users, partitions, networking etc.
- Uses a JSON-based configuration format

## **Something More With Mkosi**



Minimal images are usable across various scenarios

**FROM SCRATCH** 



Settings are human readable and extensively customizable

**INI-STYLED CONFIGURATION** 



Generated images are predictable across environments

**CONSISTENTLY REPRODUCIBLE** 



As containers and virtual machines or on physical hardware

**DIVERSELY DEPLOYABLE** 

## **Something More With Mkosi**



Fine grained control is ensured on the system's behaviour **SYSTEMD INTEGRATED** 



Images can be generated to facilitate building and testing

**ENVIRONMENT ISOLATION** 



Wraps most well-known package managers for building

**PACKAGE MANAGEMENT** 



Injecting custom scripts per build event is possible

**CUSTOMIZATION HOOKS** 

## **Configuration Sections**



Localizing customization on multiple images

**MATCH** 



Localizing customization on single image

CONFIG



Selecting GNU/Linux base for image

**DISTRIBUTION** 



Customizing image format attributes

**OUTPUT** 



Tailoring images and injecting scripts

**CONTENT** 



Customizing secure boot attributes

**VALIDATION** 



Executing as container or virtual machine

**HOST** 



## **Examples**



### **Configuration Example**

mkosi.conf (Page 1 of 2)

#### [Distribution]

Distribution=centos

Release=9

Architecture=x86-64

Mirror=https://centos-stream.excellmedia.net/

RepositoryKeyCheck=no

#### [Output]

Format=disk

Output=centos-stream

ManifestFormat=json, changelog

ImageVersion=0.0.1

ImageId=centos-stream

#### SECTION NAME

Distribution name
Release version
System architecture
Mirror location
Verify repository

#### **SECTION NAME**

Image format
Image name
Metadata types
Image version
Image identifier



### **Configuration Example**

mkosi.conf (Page 2 of 2)

#### [Content]

Bootable=yes

**Bootloader**=systemd-boot

Packages=kernel, systemd, systemd-boot, udev, dnf, iproute, ncurses, basesystem

WithRecommends=no

WithNetwork=yes

CleanPackageMetadata=yes

Locale=en\_US.UTF-8

Timezone=Asia/Kolkata

Hostname=centos-stream

RootPassword=centos-stream

RootShell=/usr/bin/bash

Autologin=no

#### SECTION NAME

Bootable

Bootloader name

Packages installable

Optional dependencies

Enable network

Clear repodata

Default locale

Default timezone

Default hostname

Password for ROOT

Shell for ROOT

Allow autologin

### **Usecase Example**

```
mkosi.postinst.chroot (Page 1 of 1)

useradd -c centosuser -G wheel -m centosuser
echo "centosuser:password" | chpasswd

echo 'nameserver 8.8.8.8' >> /etc/resolv.conf
dnf install unzip python3-pip --releasever=9-stream --assumeyes
python3 -m pip install poetry
mkdir /home/centosuser/test/

curl https://example.net/main.zip -o /home/centosuser/test/main.zip
unzip /home/centosuser/test/main.zip -d /home/centosuser/test/
Extr
cd /home/centosuser/test/main
python3 -m poetry check
Crea
```

Creating user
Modifying auth

Setting nameserver Installing packages Installing modules Creating directory

Downloading artifact
Extracting artifact
Changing directory
Checking config





## **Build**

## **Application Examples**



Immutable environments for consistency

INFRASTRUCTURE



Environment for customized purposes

**EMBEDDED** 



Including the initialization services

**CONTAINERIZATION** 



Isolated environments for checks

**TESTING** 



Reproducible environments for learning

**EDUCATION** 



Specific yet disposable environment

**DEVELOPMENT** 



Manage software development lifecycle

CI/CD PIPELINES



## **Demonstration**



## **Usage Environments**



Flashing generated image on a drive to boot from

#### **PHYSICAL HARDWARE**



Isolating generated image in a shared namespace

#### **CHROOT JAIL**



Isolating generated image in a separate namespace

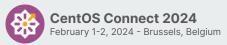
#### **SYSTEMD NSPAWN**



Utilizing generated image as a virtual machine

**QEMU/KVM/LIBVIRT** 

## Burn



## **Chroot**

## **Boot**



## **Qemu**



## Contribute



## Thank you!

- in https://www.linkedin.com/in/gridhead
- https://twitter.com/gridheader
- https://t.me/gridhead
- https://matrix.to/#/@t0xic0der:fedora.in
- https://github.com/gridhead
- https://gitlab.com/gridhead