

# Building IoT projects with Ubuntu Core

James Henstridge <james@jamesh.id.au>  
Perth Linux Users Group, March 2019

# A simple IoT project

Buy a single board computer like a Raspberry Pi

Flash a Linux distro to the card

Make some configuration changes

Install your application

Deploy

# Keeping the device up to date

Is the kernel up to date and secure?

Is the base operating system up to date? (libc, systemd, OpenSSL, etc)

Is your application up to date?

How reliable is the update strategy?

# Ubuntu Core

A stripped down Ubuntu base.

Transactional atomic updates

Applications are separated from the base system and confined from each other

System wide automatic updates

# Packaging the app

An Ubuntu Core system is comprised of a number of snap packages

Snaps can also be installed on traditional Linux systems where snapd is installed

This means some testing can be done from a traditional Linux system

# Setting up a Snapcraft project

Install Snapcraft build tool:

```
$ sudo snap install --classic snapcraft
```

```
$ sudo snap install --classic --edge multipass
```

Create a project:

```
$ snapcraft init
```

Edit the snap/snapcraft.yaml template

# Simple web app snap (1)

name: test-jamesh-webapp

version: '0.1'

summary: A simple test web application

description: |

This is a simple web application written in Go, for use in demonstrating Ubuntu Core.

license: MIT

base: core18

grade: stable

confinement: strict



# Simple web app snap (2)

apps:

webapp:

command: bin/test-jamesh-webapp

daemon: simple

plugs:

- network
- network-bind

# Simple web app snap (3)

parts:

webapp:

plugin: go

go-importpath: [github.com/jhenstridge/test-jamesh-webapp](https://github.com/jhenstridge/test-jamesh-webapp)

source: .

source-type: git

Build-packages:

- gcc

# Building and testing the snap

Build:

```
$ snapcraft  
... build output ...
```

Install locally:

```
$ snap install --dangerous test-jamesh-webapp_0.1_amd64.snap
```

Automatic builds

# Automatic Builds from Git

Host code on Github

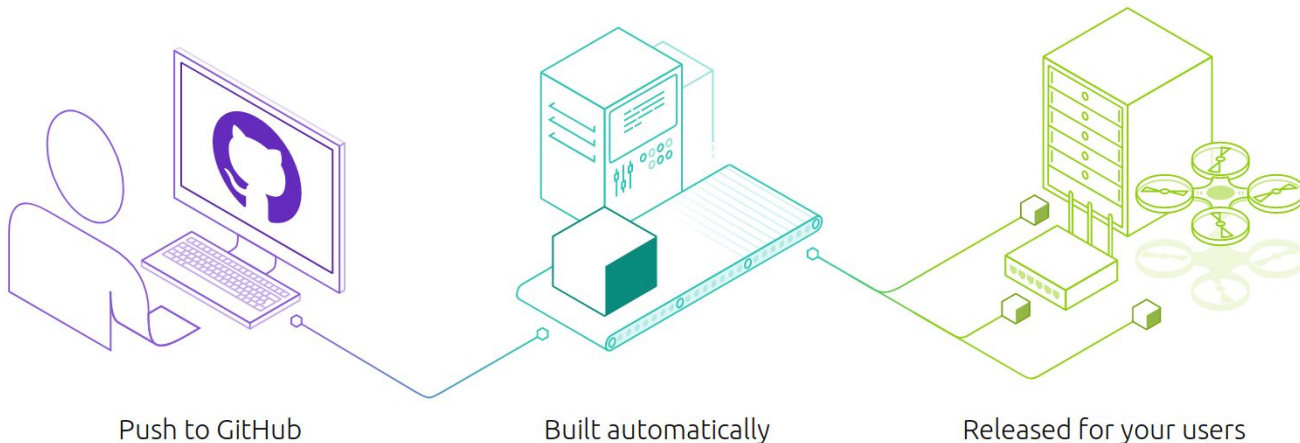
Triggers on commit to master branch

Snap is built on multiple architectures

Successful builds published to the “edge” channel

<https://build.snapcraft.io>

# Auto-build and publish software for any Linux system or device



Set up in minutes 

<authorise access to GitHub account>















# James Henstridge

jhenstridge


## Repos to build

[Add repos...](#)

Name	Configured	Registered for release	Latest build
jhenstridge/gtk2-demo-snap		  gtk2-demo	 Built and released 5 months ago 
jhenstridge/portal-test-snap-pkg		  portal-test	 Failed to build 2 months ago 

Back to top 



We use cookies to improve your experience. By your continued use of this site you accept such use. To change your settings please [see our policy](#). 



# James Henstridge

jhenstridge

## Add repos



- jhenstridge/snapd-glib
- jhenstridge/snaplint
- jhenstridge/test-jamesh-webapp
- jhenstridge/upnp-present
- jhenstridge/voctomix
- jhenstridge/www.jamesh.id.au
- jhenstridge/xdg-desktop-portal
- jhenstridge/yaru

1 selected of 41 repos ([Any missing?](#))

Cancel

Add



# James Henstridge

jhenstridge

## Repos to build

[Add repos...](#)

Name	Configured	Registered for release	Latest build
jhenstridge/gtk2-demo-snap		gtk2-demo	Built and released 5 months ago
jhenstridge/portal-test-snap-pkg		portal-test	Failed to build 2 months ago
jhenstridge/test-jamesh-webapp		Not registered	Never built

To release in the snap store, this repo needs a registered name. You need to sign in to Ubuntu One to register a name.

[Cancel](#)[Sign in...](#)

<authorise access to Ubuntu One account>



# James Henstridge

jhenstridge

## Repos to build

[Add repos...](#)

Name	Configured	Registered for release	Latest build
jhenstridge/gtk2-demo-snap		gtk2-demo	Built and released 5 months ago
jhenstridge/portal-test-snap-pkg		portal-test	Failed to build 2 months ago
jhenstridge/test-jamesh-webapp		<input type="text" value="test-jamesh-webapp"/>	Never built

To release in the snap store, this repo needs a registered name.

Lower-case letters, numbers, and hyphens only.

Cancel







Register name

[My repos](#) >

# jhenstridge/test-jamesh-webapp

snap **built and released** 

## Latest builds

Number	Build trigger	Architecture	Duration	Result	
<a href="#">#485076</a>	Commit  <a href="#">34bd385</a>	s390x	3 minutes	Built and released	<a href="#">a day ago</a>
<a href="#">#485075</a>	Commit  <a href="#">34bd385</a>	ppc64el	4 minutes	Built and released	<a href="#">a day ago</a>
<a href="#">#485074</a>	Commit  <a href="#">34bd385</a>	arm64	7 minutes	Built and released	<a href="#">a day ago</a>
<a href="#">#485073</a>	Commit  <a href="#">34bd385</a>	armhf	9 minutes	Built and released	<a href="#">a day ago</a>
<a href="#">#485072</a>	Commit  <a href="#">34bd385</a>	amd64	4 minutes	Built and released	<a href="#">a day ago</a>
<a href="#">#485071</a>	Commit  <a href="#">34bd385</a>	i386	3 minutes	Built and released	<a href="#">a day ago</a>

[Build manually now](#)

# Manage published snaps

Snap packages can be managed on <https://snapcraft.io>

Edit description, icons, screenshots for the store page.

Promote builds from edge ⇒ beta ⇒ candidate ⇒ stable

View usage metrics

Add other developers as collaborators

[◀ My snaps](#)

# test-jamesh-webapp

Listing

Releases

Metrics

Publicise

Settings

Updates to this information will appear immediately on the [snap listing page](#).

Preview

Revert

Save



Title:

Snap name:

test-jamesh-webapp

Publisher:

James Henstridge (jamesh)

Category:

Include a category and your snap will be better ranked within the Snap Store. It will also allow users to discover your snap via browsing the available categories.

Summary:



Ubuntu Core system

# Ubuntu Core Manifest

An Ubuntu Core system is described by a model assertion, describing the snaps on the system, including:

- The “base” snap providing the root file system
- The “kernel” snap holding the kernel and modules
- The “gadget” snap describing how the system is constructed
- Any required application snaps

# Base Snap

A read-only Linux root file system.

Used both for infrastructure like systemd, and for applications

One base used to boot the system. Multiple bases can be installed to support applications (e.g. to support some apps built on top of Ubuntu 16.04 and some built for 18.04).

All devices of the same architecture use the same base

# Kernel Snap

Contains matched kernel, modules, and downloadable firmware

May be specific to a family of devices

# Gadget snap

Describes how Ubuntu Core is configured on a particular device

- Partition layout
- Bootloader configuration (grub, uboot, etc)
- Device specific features (GPIO, I2C, SPI, etc)

Usually specific to a single device

# Canonical Model Assertions

Canonical publishes standard model assertions for a number of devices.

They can be downloaded with the snap tool:

```
$ snap known --remote model series=16 brand-id=canonical \  
  model=ubuntu-core-18-amd64 > ubuntu-core-18-amd64.model
```

Other models include i386, various Raspberry Pi models, and the Dragonboard

# Building an Ubuntu Core Image

Ubuntu core images can be built with the `ubuntu-image` command:

```
$ snap install --classic ubuntu-image
```

We can build an image with extra snaps side-loaded:

```
$ sudo ubuntu-image snap --extra-snaps=test-jamesh-webapp \  
    Ubuntu-core-18-amd64.model -o image
```

Write image file to device and boot

Demo



# Further Customisation

Create your own model assertion

- Bake in required application snaps
- Set up default user accounts

Custom gadget snap

- Custom device preparation script
- Automatically grant non-default permissions to snaps

Questions?