Robotics In A Hostile World
Security, Automated Maintenance and Upgrades

Eric Jensen
What you’re about to hear

Security Landscape Today

Why Ubuntu Core

Advantages of Snaps
every company needs to prepare themselves for thingbot attacks, every business and government entity with IoT devices deployed should be securing them.”

— F5 Labs
Dec 10: EU takes further steps to strengthens Cybersecurity Act, ENISA

Bruce Schneier calls for governments to regulate IoT security
F5 labs reports that IoT now the top attack target

The Serious Security Problem Looming Over Robotics
Wired, Aug 2018
“Security is best built in from day one, rather than bolting it on.”

— Mark Nunnikhoven, VP cloud research Trend Micro
Why your next robot should run Ubuntu Core
Setting up a new project... not fun!
Linux Software Delivery

Hidden Challenges

- Requires knowledge of Linux internals
- Complex packaging code
- Not transactional
- Unbounded
- Risky

Working in this loosely-defined environment is unpredictable, time-consuming, and costly
The nightmare begins!

Maintenance
Ubuntu Core benefits

- A minimal, transactional Ubuntu for appliances
- Safe, reliable updates with tests and rollback
- App containment and isolation with managed access to resources
- Amazing ‘devops for IoT’ experience from device to cloud
- Familiar development environment for Linux developers
- Easily create app stores for all your devices
Automatically **confines** applications

Snaps are confined and isolated
Build Environment

Automation

You receive a pull request on GitHub

Test with Travis or other CI system

The code lands on your GitHub master

Snapcraft builds a new snap version

Auto-released to the snap store for testing

You promote to beta / candidate / stable
Update and Deployment Strategies

Channels and Tracks Offer Flexible Deployment Options

- Stable
- Candidate
- Beta
- Edge
- snap

Support multiple simultaneous dev efforts

snap

With different needs
Transactional updates: Apps, OS and kernel

- **Original data**
  - Writable area
  - Original snap

- **Modified data during upgrade**
  - Writable area
  - Updated snap

- **Original data**
  - Writable area
  - Original snap

- **Rollback on failure**

  - Original data is kept on device

  - Original snap
Ubuntu Core benefits

A minimal, transactional Ubuntu for appliances

Safe, reliable updates with tests and rollback

App containment and isolation with managed access to resources

Amazing ‘devops for IoT’ experience from device to cloud
Familiar development environment for Linux developers

Easily create app stores for all your devices
Simple, Declarative App Definitions

name: ros-talker-listener
version: 0.1
summary: ROS Talker/Listener Example
description: |
    This example requires roscore as well as a talker and listener.

grade: devel
confinement: devmode

parts:
ros-tutorials:
    source: https://github.com/ros/ros_tutorials.git
    source-branch: kinetic-devel
    plugin: catkin
    rosdistro: kinetic
    source-space: roscpp_tutorials/
    catkin-packages: [roscpp_tutorials]

apps:
run:
    command: roslaunch roscpp_tutorials talker_listener.launch
Backed by Canonical

It is our mission to make open source software available to people everywhere. We believe the best way to fuel innovation is to give the innovators the technology they need.
Over 3 million developers use Ubuntu.
Additional Advantages of Snaps
Snap Package Architecture

- **MAC** to other resources (Paths (/home), Devices /dev, etc) mediated with **interfaces**

**Snap code & assets**
(squashfs, RO bind-mounted in /snap/<snap_name>/<version>)

$SNAP

- **Common root writable area**
  $SNAP_COMMON

- **Versioned root writable area**
  $SNAP_DATA

- **Common User writable area**
  $SNAP_USER_COMMON

- **Versioned User writable area**
  $SNAP_USER_DATA

- /tmp

- **CLI**

- **GUI**
Security Alerts, Audits and Compliance

- Automatic Security Warnings on your snaps
  - Daily checks on your software components with alerts

- Audit your systems through package verification
  - For OS and apps
  - Confirm that exactly what you deployed is still running on the device, no more or less

- Compliance management
  - Archive source code used to build your snaps
  - Aggregate licenses
Snap Store

Reach millions of devices with the Snap Store.

- Public store hosted by Canonical
- Ubuntu systems point to the snap store by default
- Access the store and manage your snaps from the CLI or the web
Brand Store

- snap store as a service
- Full control over the store content
- Review queue for third parties
- Endless possibilities
Maintenance

And relax
Visit snapcraft.io
Visit ubuntu.com/core
Updates published instantly
● Cryptographically signed and tamper-proof
● Risk-based update channels for creating software testing programmes
Automatic over-the-air updates

- No additional infrastructure or code needed
- End user systems automatically apply updates, four times per day
- Eliminate the long tail of releases to support and lower your costs
Out of the box discovery by tens of millions

- All Ubuntu versions since 14.04 and other major Linux distributions gives broad coverage
- All software welcome, regardless of license or cost
Developer dashboards

Weekly active devices

Past 30 days

By version

Weekly active devices

32,824

Territories

2018-09-08

1.1.23.353.a15c... 23,594
1.1.22.442.a28c... 6,985
1.1.21.7a5.eff2... 934
1.0.42.117.b43... 857
1.0.89.311.g44... 324
6 other 130

52,827
Over 4100 snaps available for download
Enterprise Proxy

- **Enterprise Proxy** enables fine-grained control over which snaps and revisions of snaps are available for *consumption* by their fleet of devices
  - Also provides a means to access the Snap Store for devices with restricted network access