CANONICAL

Ubuntu Automated QA
December 8th 2011

Pete Graner

Ubuntu QA Engineering Manager

Contents

- History of Ubuntu QA

- Why Automation?

- Automated Testing – How to Get Started

- Challenges

- More Information

- Questions

CAN●NICAL

History of Ubuntu QA

- From the start of Ubuntu QA was a community effort consisting of:
  - Manual Testing
  - Targeted Testing
  - Bug Triaging and Classification
- The Good
  - Wide hardware coverage on consumer platforms
  - Real End User induced defects (i.e. out of order test operations, typing mistakes etc..)
- The Bad:
  - Lack of consistent results reporting
  - Lack of consistency in testing procedures
  - Difficulty in Interpreting Defects
  - Lack of Escape Analysis

CANONICAL

Why Automation?

- Automation was the next step in Ubuntu's evolution

- The desire to have Ubuntu truly be the "best Open Source Operating System"

- To have consistent and repeatable tests run:

  - Daily ISO Builds

  - On Milestones:

    - Alphas (Alpha1 ...)

    - Beta (Beta1 ...)

    - Release Candidates

  - Stable Release Updates (SRUs)

  - Continuous Integration Testing on Canonical Developed Code

  - Security Testing

  - Regression Testing

  - Cloud Testing – Amazon Ubuntu AMI Images

  - Performance Testing

CANONICAL

Automated Testing – Where to Start?

- Lab

- Hardware

- Network

- Access

- Software

  - Scheduler

  - Test Runner

  - Reporting

  - Defect Reporting

CANONICAL

Automated Testing – Lab

- Things to consider:
  - Power
  - HVAC
  - Racking & Shelving
  - Cabling
- We Determined that for Ubuntu's Needs that Two Physical Labs Were a Better Fit Due To:
  - Server Hardware (Racks & Rackmount hardware)
  - Consumer Hardware (Shelf & Bench hardware)
    - Desktop
    - Note/Netbook
    - Other Yet to be Realized Hardware

CANONICAL

Automated Testing – Server Class Hardware

- Used for:
  - Bare Metal Testing (Kernel, Installation & Unique Hardware i.e. FC adapters)
  - Virtualized Testing
  - Cloud & Virtualization Testing
  - Stress Testing
  - Load Testing
  - Power Consumption Testing
  - Continuous Integration Testing

CANONICAL

Automated Testing – Consumer Grade Hardware

- Used for:
    - Bare Metal Testing (Kernel, Installation)

    - Device Testing & Unique Hardware i.e. Wifi, Display, Multi-monitor

    - Stress Testing

    - Load Testing

    - Power Consumption Testing

CANONICAL

Automated Testing – Network

- Network Considerations

  - Protocol Restrictions i.e. Can't TFTP boot across subnets

  - VLAN

  - VPN Access

  - Wireless

  - Management Lan for critical equipment (PDU, KVM etc.)

CANONICAL

Automated Testing – Access

- Who Needs Access?

- How to Access via Network

  - VPN

  - SSH Tunnel

- Public vs. Private access to:

  - Infrastructure

  - Reports & Testing Results

- Authentication

  - RADIUS

  - LDAP

  - Shared Accounts

  - Equipment Login

CANONICAL

Automated Testing – Software

- Proprietary vs. Open Source Software

- Ubuntu Tries to Use Only Open Source Software for Testing

- Scheduler
  - Jenkins
  - https://www.jenkins-ci.org/

- Test Runner
  - PyUnit
  - http://pyunit.sourceforge.net/

- Reporting
  - Jenkins
  - Custom Reporting Code via JUnit

- Defect Reporting
  - Launchpad Lib
  - https://launchpad.net/launchpadlib

CANONICAL

Automated Testing – Software

- Orchestra

- Is part of Ubuntu and provides the following components:
    - Provisioning Server → Cobbler
    - Monitoring Server → Nagios
    - Management Server → Juju
    - Logging Server → Rsyslog

- This is a one stop tool for allowing the PXE booting, DHCP, and other bits needed to manage a QA testing infrastructure.

- All web driven and extremely easy to use

- http://blog.dustinkirkland.com/2011/08/formal-introduction-to-ubuntu-orchestra.html

CANONICAL

Automated Testing – Challenges

- Need for Private vs. Public access and reporting

  - **Challenge:** Ubuntu is an Open Project, not allowing Community Access to Hardware and Resources is Problematic

  - **Solution:** Facilitate the Community with Manual Testing & provide a mechanism for test results to be integrated into the automated reporting

- Consumer Grade Hardware is Poor for Automated Testing

  - **Challenge:** Retail Note/Netbooks have limited BIOS options for power on at reboot. This limits the ability to use this class of hardware for automated & unattended testing.

  - **Solution:** Open the case and solder power switches so they are "always on"

CANONICAL

Automated Testing – Challenges

- Viewing the Results Multi-Monitor Testing

  - **Challenge:** How to verify that the software is performing correctly when the resolution, orientation & rotation are changed.

  - **Solution:** Multiple

  - Screen Capture & Automated Image Comparison

  - Pan, Tilt, Zoom (PTL) Programmable Video Camera

    → Move to predetermined x,y,z axis and take picture or video

    → Image or Video is then attached to the testing artifacts and reviewed

CAN○NICAL

More Information

- Ubuntu QA Landing Page: http://wiki.ubuntu.com/XXXXXXXX

- QA Strategy: https://wiki.ubuntu.com/QATeam/AutomatedTesting/Strate

- Ubuntu Defect/Bug Info:

  - https://wiki.ubuntu.com/Bugs/Status

  - https://wiki.ubuntu.com/Bugs/HowToTriage

  - https://wiki.ubuntu.com/Bugs/Importance

- Public Results & Reporting: https://jenkins.qa.ubuntu.com/

- Test Code/Cases/Tools:

  - https://code.launchpad.net/~ubuntu-server-iso-testing-dev/ubuntu-server-iso-testing/trunk

CANONICAL

CANONICAL

Thank you

Questions ?