

Odin

# Тестирование ПО, основанного на сторонних компонентах, на примере дистрибутива ОС Linux

Денис Силаков  
Sr. Software Architect



# Дистрибутив Linux

- ~**200** млн строк кода
- Оценочная стоимость разработки: **\$10 млрд**  
(оценка *The Linux Foundation* для *Fedora 9*)
- Доля собственных разработок каждого вендора – **0-5%**
  - Собственное уникальное ПО
  - Доработка сторонних компонентов

# Virtuozzo (ex-Parallels Cloud Server)

- Виртуализационное решение
  - Модифицированное ядро Linux
  - Инструменты управления контейнерами и VM
  - ...
- «Под капотом» готового продукта – дистрибутив Linux

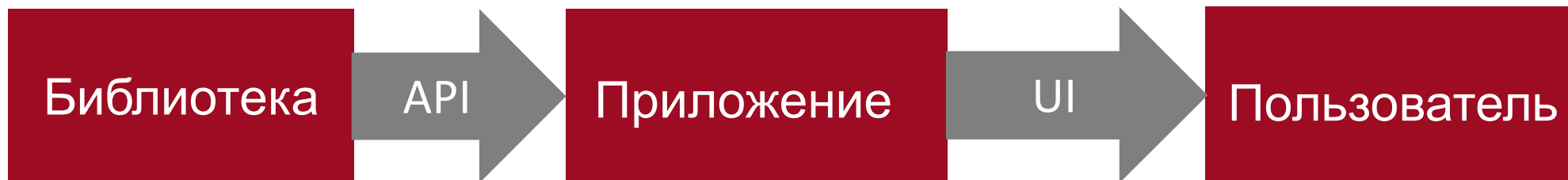
**Odin** | Virtuozzo

# Природа ошибок в Linux-based продукте

- Ошибки в собственном коде
  - Ошибки upstream
  - Интеграционные проблемы
- ⇒ Размер тестируемого кода превосходит размер разрабатываемого
- ⇒ Нужна автоматизация и переиспользование существующих решений

## Модульные тесты Для библиотек и приложений

# Библиотеки vs приложения



- Использование API – обычно формализуется
- Использование интерактивного UI – обычно нет

# Тестирование API

- Проработанный научный базис
- Хорошая инструментальная поддержка
- Относительно высокая степень автоматизации

# Ресурсоемкость

Производительность человека при разработке тестов для Linux Standard Base (~ 40.000 функций):

- Базовые (“**shallow**”) тесты – **10-50** тестов в день
  - Обычные (“**normal**”) тесты – **2-3** функции в день
  - Продвинутые (“**deep**”) – **0.5-1** функция в день
- ⇒ разрабатывать тесты для сторонних компонентов – накладно



# Существующие решения

- **Self-tests**, поставляемые с кодом приложения
  - CentOS 7.1: тесты есть в **~750** из **2500** пакетов
- Открытые тестовые наборы
  - Тесты **Linux Standard Base**
  - **Linux Test Project**
  - **The Fuzzing Project**
  - **TIS Interpreter** – expected in 2016

# Проблемы сторонних тестов

- Не всегда рассчитаны на запуск на сборочных фермах (ВМ без доступа в сеть и программным таймером)
- Отставание от развития тестируемого кода
- У «продвинутых» тестов – ложные срабатывания
- Большое суммарное время работы (> 24 часов)

## Тесты для приложений с графическим интерфейсом

# Особенности тестирования GUI-приложений

- Технологические сложности воспроизведения действий человека
- Интересны не атомарные воздействия, а сложные сценарии работы
- Оценка корректности работы не всегда формализуема

# Воздействие на GUI

- AT-SPI – технологии для поддержки людей с ограниченными возможностями
- Средства X11
  - Xvfb – запуск графических приложений без X-сервера
- Средства виртуальных машин

# AT-SPI

- Предоставляет доступ к внутренней структуре приложения (a-la Form Designer)
- Возможность применения зависит от используемых графических библиотек:
  - **OK:** Qt, Gtk, Java, Mono
  - **Missing:** Motif, FLTK, WxWidgets

# X11

## Расширения для эмуляции действий пользователя

- Не зависят от библиотек, только к X11
  - ? Wayland, Mir ?
- Позволяют работать только с низкоуровневой структурой окон

# Средства виртуальных машин (QEMU)

Эмуляция пользовательских действий внутри окна VM

- Нет привязки к ПО, используемому внутри VM
- Ничего не знает об окнах, графических стеках и прочем



# Проверка результатов GUI-тестов

- Снимок экрана
  - Сравнение с эталоном
  - Поиск заданного текста на картинке через OCR
- AT-SPI
  - Анализ доступного меню, текстовых полей, ...
- X11
  - Анализ доступных окон и их заголовков

## Интеграционные тесты

# Runtime тесты

- Модульные тесты приложения == интеграционные тесты для внешних компонентов, которые оно использует
- ⇒ Установить все приложения и протестировать
- ⇒ Осталось написать тесты для всех приложений

# Статические тесты

- Конфликты по файлам
- Замкнутость репозиториев по зависимостям и бинарным символам
- Права доступа к файлам приложения
- ...

*См.: SECR-2013 – Автоматизация поддержки репозиториев ПО для Linux*

# ROSA FBA









<http://fba.rosalinux.ru>

## FBA

Home Dependency Tests ▾ RPMLint ▾ Platform Analysis ▾ ABF Autotest

### Welcome to FBA - Frontend Brother of Abf

This is a central place where we are going to collect statistics about ROSA distributions, that is needed to maintain package base.

Platform	Arch	Repository	Reposlosure	File Conflicts	Repomanage	Obsoleted	ELF Closure	Alternatives	Rpmlint
ROSA Enterprise Desktop X1	i586	<a href="#">main</a> 	<a href="#">86</a>	<a href="#">0</a>	<a href="#">738</a>	<a href="#">21</a>	<a href="#">34</a>	<a href="#">838</a>	<a href="#">447 / 0</a>
		<a href="#">contrib</a> 	<a href="#">772</a>	<a href="#">0</a>	<a href="#">100</a>	<a href="#">28</a>	<a href="#">207</a>	<a href="#">1701</a>	<a href="#">645 / 0</a>
		<a href="#">non-free</a> 	<a href="#">4</a>	<a href="#">0</a>	<a href="#">0</a>		<a href="#">42</a>	<a href="#">10</a>	<a href="#">466 / 0</a>
		<a href="#">restricted</a> 	<a href="#">6</a>	<a href="#">0</a>	<a href="#">0</a>		<a href="#">47</a>	<a href="#">230</a>	<a href="#">456 / 0</a>
	x86_64	<a href="#">main</a> 	<a href="#">86</a>	<a href="#">0</a>	<a href="#">738</a>	<a href="#">20</a>	<a href="#">34</a>	<a href="#">838</a>	<a href="#">487 / 0</a>
		<a href="#">contrib</a> 	<a href="#">715</a>	<a href="#">0</a>	<a href="#">98</a>	<a href="#">33</a>	<a href="#">188</a>	<a href="#">1871</a>	<a href="#">913 / 0</a>
		<a href="#">non-free</a> 	<a href="#">8</a>	<a href="#">0</a>	<a href="#">0</a>		<a href="#">39</a>	<a href="#">9</a>	<a href="#">503 / 0</a>
		<a href="#">restricted</a> 	<a href="#">0</a>	<a href="#">0</a>	<a href="#">0</a>		<a href="#">45</a>	<a href="#">215</a>	<a href="#">496 / 0</a>

# Однотипные тесты для множества приложений

- Возможность запустить сервис (при наличии service-файла)
- Возможность запустить программу (при наличии desktop-файла с параметрами запуска)

## Тестирование на соответствие требованиям стороннего заказчика

Например, сертифицирующей лаборатории

# ROSA Test Suite

Пункт ПМИ ⇔ тест

- Разумное количество тестов
- Разумное время работы
- Возможно использование тестового набора в процессе приемки



## Итоги

Процесс тестирования реального Linux-based продукта

# Запуск тестов

- При сборке пакета
  - Self-tests, возможность установки, RpmLint, ...
- Ежедневно
  - Сборка ISO-образа и тестирование установки
  - ROSA Test Suite, LTP, LSB (частично), избранные приложения
- Еженедельно
  - LSB, тесты service- и desktop-файлов

# Характеристики тестов

(для ~2.500 официально поддерживаемых пакетов)

Тестовый набор	Число тестируемых пакетов	Качество
Self-tests	~750	Normal
LSB	~50	From Shallow to Deep
LTP	~10	From Normal to Deep
Rosa Test Suite	~100	From Shallow to Normal
Desktop files	~500	Shallow
Service files	~100	Shallow

Odin

Thank you