Актуальные проблемы Системного программирования

Научный руководитель ИСП РАН
академик Иванников В.П.
ivan@ispras.ru

Технологии разработки ПО(1)
Рост размеров кода ОС

Система	**Год**	**Размер (10^6 LOC)**	**Размер команды**
Windows NT 3.5	1994	7-8	300
Linux Kernel 2.0.0	1996	0.78	
Windows NT 4.0	1996	11-12	800
Windows 2000	1999	29	1400
Windows XP	2001	45	1800
Linux Kernel 2.6.0	2003	5.93	
Debian 4.0	2007	283	
Linux Kernel 2.6.32	2009	12.6	
Linux Kernel 3.6	2012	15.9	
Debian 7.0	2012	419	

Технологии разработки ПО(2)
Характеристики современного ПО

- Эскалация размеров и сложности
- Увеличение функциональных возможностей
- Рост объемов перерабатываемых данных
- Расширение использования параллелизма и распределенности
- Рост требований к переносимости и совместимости

Технологии разработки ПО(3)
Верификация ПО

- Экспертиза
  - Задействуются опыт и знания людей
- Статический анализ
  - Поиск возможных дефектов в коде по некоторым шаблонам
- Динамический анализ
  - Тестирование, мониторинг, профилирование, фаззинг
- Формальные методы
  - Дедуктивная верификация, проверка моделей (model checking)
- Смешанные техники
  - Статический/динамический анализ с использованием (неполных) формальных моделей

Технологии разработки ПО(4)
Дедуктивная верификация (ДВ)

- Формальная модель требований к системе (что она должна делать) – R
- Формальная модель функционирования системы (как она работает, обычно строится на базе кода) – I
- Формальная модель окружения (предположения о среде, компиляторе и взаимодействии с пользователями и др. программами) – E
- Нужно формально доказать выводимость (E & I) $\Rightarrow$ R

➢ Из-за сложности нужны специализированные языки формальных моделей и инструменты поддержки

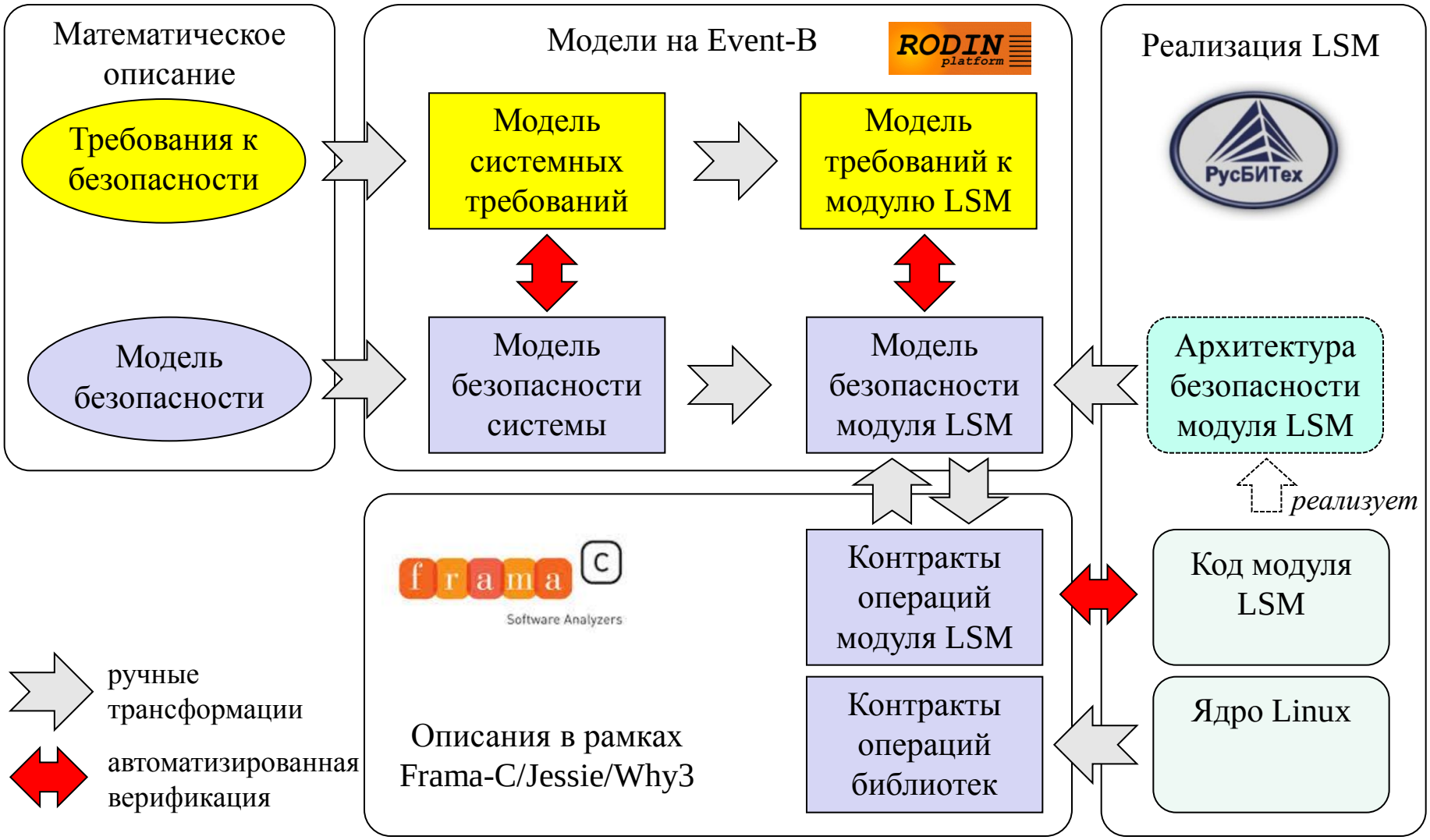
➢ Сейчас средний размер подходящих систем ~$\leq 10^4$ строк
Проект iFACTS (UK, контроль полетов) ~ $25 \cdot 10^4$ строк

Технологии разработки ПО(5)
Инструменты и применения ДВ

- Возникновение метода 1969
  - R. Floyd, C.A.R. Hoare
- Появление инструментов ~1993-2005
  - SunRise, ESC/Java, Frama-C, LOOP, Boogie/VCC, Rodin
- Промышленные применения ~2003-2014
  - Системы управления АЭС (Франция, UK)
  - Автономные транспортные системы (Франция)
  - Авионика (UK, Airbus, NASA)
  - Ядра операционных систем и гипервизоров seL4, PikeOS, Hyper-V
  - Компоненты микропроцессоров (проектная модель) Intel, Verisoft

seL4: ~$1.4 \cdot 10^4$ строк кода
~$3.3 \cdot 10^4$ строк модели
~22.5 человек·лет

Технологии разработки ПО(6)
Верификация модели безопасности AstraLinux



Управление данными (1)
Предпосылки

- Google
  - 1998 год*
    - 26 миллионов страниц
  - 2008 год*
    - 1 триллион (10^{12}) страниц
  - 2015 год**
    - 30 триллионов страниц
    - Индекс – 100 миллионов гигабайт (10^{17})
    - 2 000 000 запросов в минуту
- Количество данных удваивается каждые 1.5 года

* https://googleblog.blogspot.ca/2008/07/we-knew-web-was-big.html
** http://www.google.com/insidesearch/howsearchworks/thestory/

Управление данными (2)
Аппаратное обеспечение

- Суперкомпьютеры или кластеры из дешевых потребительских компьютеров?
  - Google выбрал второй подход
  - Причины: **стоимость** и **масштабируемость**
- Проблема: устойчивость к сбоям
  - Один диск ломается раз в 5 лет (1826 дней)
  - В кластере с 2000 дисками в среднем более одной поломки каждый день
  - Пользователь (программист) не должен этого замечать

Управление данными (3)
Программное обеспечение

- Распределенная файловая система GFS (Google 2002 год)
  - Сбои скорее правило, чем исключение
    - Обнаружение ошибок
    - Автоматическое восстановление
  - Специфика работы с файлами
    - Большие файлы (десятки гигабайт)
    - Отсутствуют записи в произвольное место (только дописывание в конец файла)
    - Чтение файла происходит намного чаще, чем изменение
- Подход MapReduce (2004 год)
  - Программирование в терминах ключ-значение.
  - **Map** получает на вход ключ и значение и возвращает ноль или более новых пар ключ – значение. При этом, программа перемещается к данным.
  - R**educe** получает на вход все значения с одним ключом и возвращает ноль или более новых пар ключ – значение.

Управление данными (4)
Программное обеспечение

- Yahoo
  - **HDFS** + Hadoop MapReduce – разработка с 2005
  - **Apache Hadoop** – с 2009 свободное ПО
  - Сейчас Apache Hadoop – де-факто промышленный стандарт
- Facebook
  - Более миллиарда пользователей
  - 100+ PB на одном кластере Hadoop
  - Разработали **Apache Cassandra** (СУБД в стеке Apache Hadoop)
- Twitter
  - 347000 сообщений каждую минуту
  - **Apache Storm** – система обработки потоковых данных (свободное ПО)

Управление данными (5)
Современное состояние ПО

- Сейчас вокруг Apache Hadoop собраны десятки свободно распространяемых систем для обработки больших объемов данных
- Анализ проблем и тенденций в UC Berkeley привел к созданию Berkeley Data Analytics Stack
  - **Apache Spark** – Map Reduce в оперативной памяти (до 100 раз быстрее, чем Hadoop MapReduce)
  - **Spark SQL** – запросы к данным на языке SQL
  - **Spark Streaming** – обработка потоковых данных
  - **Spark Mllib** – машинное обучение
  - **GraphX** – обработка графов
  - …

Управление данными (6)
Некоторые современные задачи

- Банковская аналитика
  - 1 миллиард счетов,
  - более 50000 транзакций в секунду
  - используются технологии **I**n-**M**emory **D**ata **G**rid
- Вычислительная биология
  - анализ генома (~3 ГБ один геном),
  - анализ белковых взаимодействий,
  - анализ нервных связей (коннектом)
- Анализ социальных взаимодействий в сети Интернет
- Анализ поведения пользователей для показа персонализированной рекламы

Управление данными (7)
Проблемы

- Термин «Большие данные» - применяется для обозначения очень широкого круга проблем и технологий
  - Технологии зависят от предметной области
  - Не хватает опытных технических специалистов, обладающих пониманием как задач предметной области, так и технологий
- Пример: использование Internet, как одного из основных источников больших данных
  - Быстрое снижение актуальности данных после их появления
  - Ограничения возможности сбора данных
  - Многообразие источников информации
  - Противоречивость информации
  - Высокое содержание информационного шума
  - Многообразие форматов и способов представления данных

Проблемы компьютерной безопасности (1)

Обеспечение компьютерной безопасности – одна из наиболее актуальных задач системного программирования.

Рынок ПО для обеспечения защиты информации в последние годы растет гораздо быстрее, чем весь рынок ИТ.

Причины:

1. Глобализация информационного пространства (широкое распространение сетевых технологий, интенсивное развитие средств телекоммуникации);

2. Внедрение электронных средств хранения и обработки информации во все сферы экономической, политической и военной деятельности (документооборот, связь, базы данных, системы поддержки принятия решений и др.);

3. Ускорение и удешевление разработки ПО в ущерб его качеству в условиях острой конкуренции на рынке ИТ;

4. Бытовая компьютеризация (средства мобильной связи, беспроводные сети и пр.)

Проблемы компьютерной безопасности (2)
Угрозы

1. Выведение из строя ПО (распространение вредоносных программ, компьютерных «вирусов», троянских программ, подмена мобильного кода).
2. Выведение из строя компьютерных сетей (создание перегрузок в сети, сетевые вторжения)
3. Создание и распространение ПО со скрытыми дефектами (внедрение в ПО закладок, встроенных уязвимостей и пр.).
4. Нарушение конфиденциальности и целостности информации (несанкционированный доступ к базам данных и электронным архивам, распространение программ-шпионов, взлом криптосистем).
5. Компьютерное пиратство и нарушение авторских прав на ПО (нелегальное копирование и распространение ПО).

Проблемы компьютерной безопасности (3)
Средства обеспечения

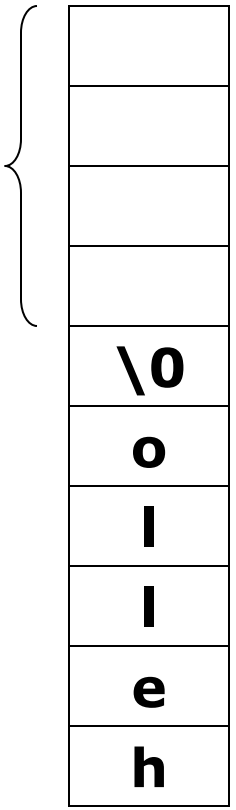
1. Антивирусные программы, интеллектуальные системы мониторинга, системы защиты от утечки данных
2. Системы обнаружения и предотвращения сетевых вторжений (сетевой мониторинг, межсетевые экраны, шлюзы, брандмауэры).
3. Использование ПО с открытым кодом (Linux vs Microsoft).
4. Сертификация и аудит ПО, системы верификации и поиска уязвимостей в программах.
5. Политики безопасности в информационных системах (разграничение доступа, аутентификация и авторизация).
6. Криптографические средства защиты информации (системы шифрования, электронной подписи, криптографические протоколы, обфускация программ и пр.).
7. Отслеживание распространения ПО (водяные знаки, отпечатки пальцев).

Проблемы компьютерной безопасности (4)

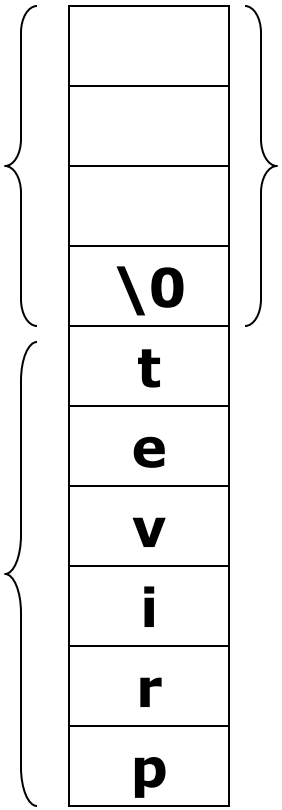
Простейший пример

```
f(char * p)
{     char s[6];
      strcpy(s,p);
}
main1 ()
{       f("hello");
}
main2 ()
{       f("privet");
}
```

Стек после выполнения функции f, вызванной из main1



Стек после выполнения функции f, вызванной из main2



В случае main2 адрес возврата перезапишется и управление будет передано не на main2, а на другой участок кода

Проблемы компьютерной безопасности (5)

- Статический и динамический анализ исходного кода
- Статический и динамический анализ бинарного кода