

Программа факультатива / 2013

I Анализ бинарного кода

Манипуляции с объектным (бинарным) кодом с точки зрения разработчика программ. Утилиты пакета GNU binutils: ar, strings, strip, nm, size, readelf, objdump. Позиционно независимый код. Динамические и разделяемые библиотеки, особенности организации и загрузки.

Статический и динамический анализ бинарного кода. Классический подход: дизассемблирование, восстановление меток, интерактивный дизассемблер IDA Pro. Представление результатов анализа: граф вызовов функций, граф потока управления, граф потока данных.

Вспомогательные данные, используемые в анализе бинарного кода. Символы системных функций и данных. Распознавание в машинном коде конструкций языка Си.

II Архитектура Intel64

Краткий обзор архитектуры Intel64, ее отличия от IA32. Соглашение о связях, используемое в 64-разрядном коде.

Регистры общего назначения. Примеры команд. Регистры XMM. Примеры скалярных команд. Примеры векторных команд.

Режимы работы процессора: 64-разрядный режим, режим совместимости, 32-разрядный режим. Различия между ними. стек в 64-разрядном режиме.

Соглашение о связях System V AMD64. Соглашение о связях Microsoft.

Шины передачи данных, используемые в современных настольных компьютерах, серверах, встраиваемых системах.

III Анализ кода ОС на примере MS Windows

Использование аппаратуры процессоров семейства x86 для организации многозадачного режима в ОС Windows. Адресные пространства, процессы и потоки. Пространство имен диспетчера объектов. Взаимосвязь основных объектов ядра. Прохождение запроса в/в между user и kernel mode. Основы построения драйверов ядра.

Средства анализа кода в ОС Windows. Примеры проведения динамического анализа с использованием трассировщика.

IV Архитектура ARM

Обзор архитектуры ARM. Соглашение о связях. Регистры общего назначения. Примеры команд. Кодирование команд в виде ARM и Thumb. Векторная архитектура NEON: векторные регистры, примеры команд.

Оптимизации компилятора, полезные для архитектуры ARM. Рассмотрение практических задач: оптимизация программ на языке Си для платформы ARM, автонастройка компилятора для данной программы, оптимизация энергосбережения.

V Виртуальные машины

Программная эмуляция компьютерной системы. Бинарная трансляция машинного кода. Виртуальная машина Qemu.