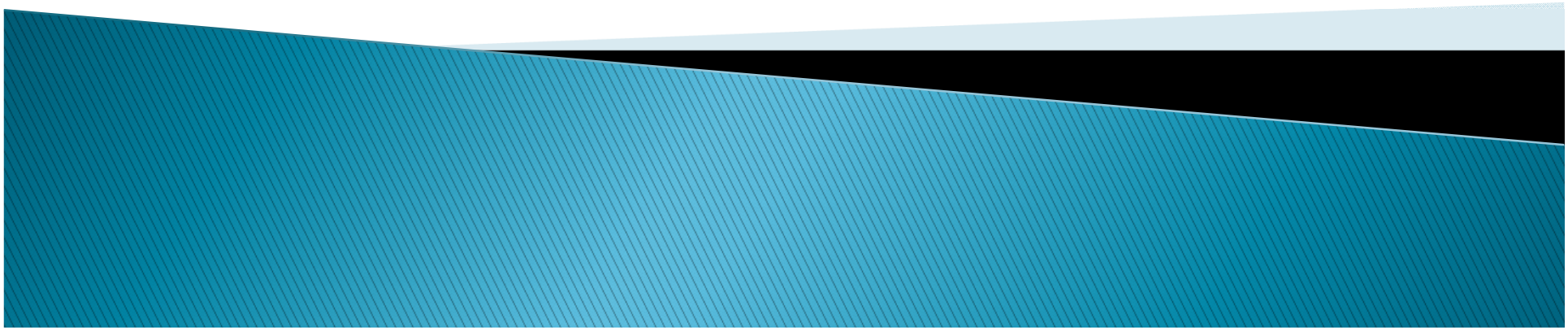
Password Recovery and Forensic Software

Поиск криптографических ключей в RAM

Алексей Чиликов, Евгений Алексеев

Март 2011

Конференция РусКрипто’2011



Коротко о нас

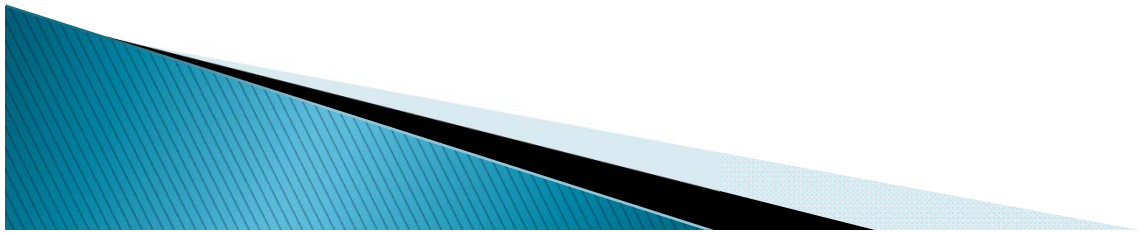
- Passware
  - На рынке с 1998 года
  - Отделения в США и России

- Passware Kit Forensic
  - Восстановление данных для 180+ типов файлов
  - Быстрый поиск защищённых объектов на компьютере
  - Поддержка аппаратного ускорения (Tableau TACC, GPU) для восстановления паролей/ключей
  - Переносимая версия для работы на месте инцидента
  - Анализ данных в памяти

Passware

Краткое содержание

- Кому и зачем это нужно?
- Объекты атак
- Основные задачи
- Алгоритмы и новые результаты
- Пример
- Смежные задачи



Passware

Атаки на RAM – обзор

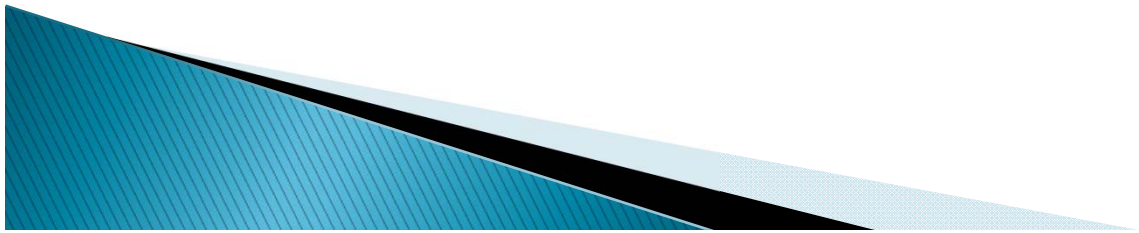
- Проблема:
  - Стойкое шифрование, значительное время работы
- Путь решения:
  - Поиск ключей шифрования в RAM
- Кому это нужно?
  - Криминалисты – сбор цифровых улик
  - Разработчики приложений – анализ защищённости

Типы защищаемых объектов

- Защита файлов
  - Пароли + ключи шифрования
- Сетевые сессии
  - Сеансовые ключи
- Защита дисков
  - Full Disk Encryption (FDE)
    - Software
      - BitLocker
      - PGP
      - TrueCrypt
    - Hardware

Способы доступа к памяти

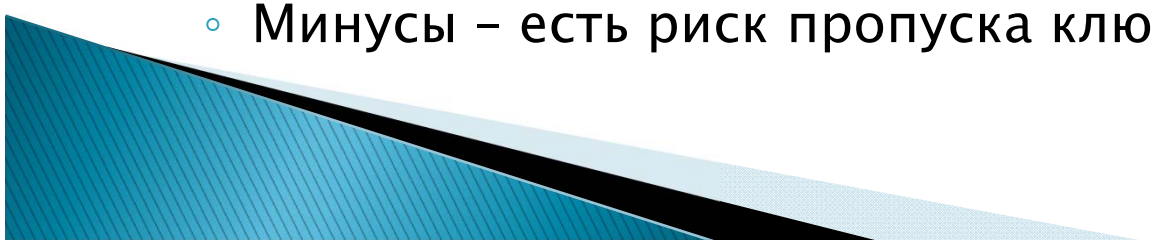
- Предустановленный драйвер
  - Широкий выбор инструментов
- FireWire + DMA
  - Passware FireWire Memory Imager
- Hibernation
  - Passware Kit Forensic
- ColdBoot
  - [D. MacIver, 2006]



Passware

Методы поиска ключей

- Полный перебор
  - Плюсы – универсальность, гарантированный результат
  - Минусы – крайне низкая скорость работы
- Сигнатурный метод
  - Плюсы – высокая скорость работы, гарантированный результат
  - Минусы – привязан к конкретному приложению (версии), требуется глубокий анализ приложения, не всегда есть подходящие сигнатуры
- Энтропийный метод
  - Плюсы – универсальность, относительно высокая скорость работы
  - Минусы – есть риск пропуска ключа

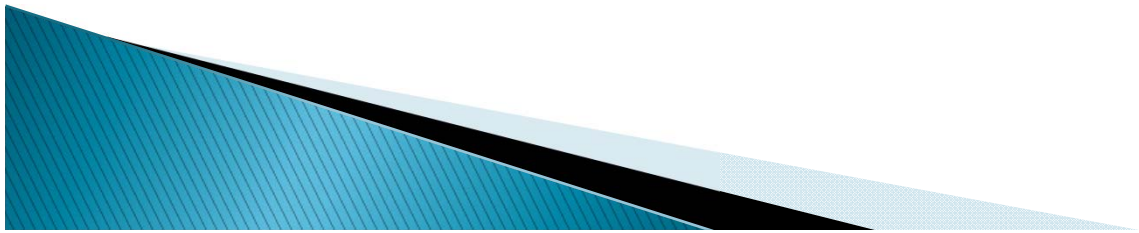


Passware

Методы поиска ключей

- Метод расширенных ключей
  - Плюсы – привязан к алгоритму, но не к приложению, требуется только анализ алгоритма (сложность варьируется), работает относительно быстро
  - Минусы – работает только при кэшировании расширенных ключей

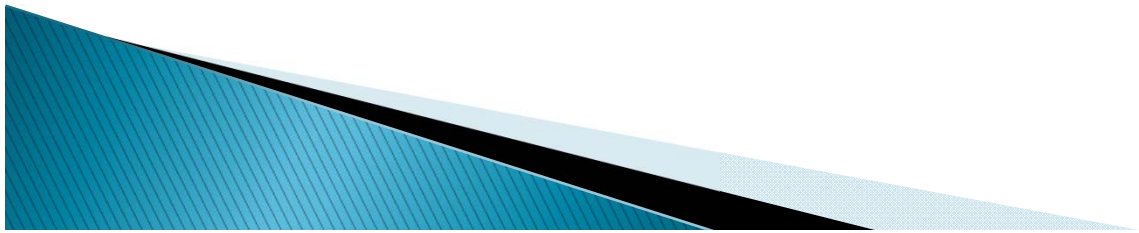
- Гибридные методы
  - Энтропийные тесты + метод расширенных ключей
  - Тонкий момент – расширенные ключи обладают внутренней избыточностью, поэтому требуется адаптировать энтропийные тесты



Целевые алгоритмы

- Простые (AES, Serpent)
  - Оригинальный ключ является началом массива расширенных ключей
  - Проверка массива данных сводится к выработке расширенного ключа из оригинального

- Сложные (Twofish, Blowfish)
  - Оригинальный ключ не присутствует явно
  - Расширенные ключи вырабатываются по сложной схеме
  - Расширенные ключи связаны друг с другом нелинейно



Наши результаты

- Целевые алгоритмы:
  - Twofish
  - Blowfish

- Атака: Быстрое восстановление оригинального ключа из расширенного

- Развитие атаки: Быстрое распознавание расширенных ключей в потоке данных

В ранее опубликованных работах эти алгоритмы атаковались только сигнатурными методами!

Twofish – описание

- 256-битный ключ M = (M0, M1, …, M7)
- Me = (M0,M2,M4,M6), Mo = (M1,M3,M5,M7)
- R = 0x01010101
- A[i] = h(2*i*R, Me)
- B[i] = h((2*i+1)*R, Mo) <<< 8
- K[2*i] = A[i] + B[i]
- K[2*i+1] = (A[i] + 2*B[i]) <<< 9

- Все операции производятся над 32-разрядными словами
- Параметр i принимает значения от 0 до 19 включительно (т.е., всего 40 ключей)
- h – нелинейная функция

Twofish – описание

- Ядро алгоритма – функция h(X, [A,B,C,D])
- Входы: X, A, B, C, D – 32-битные слова
- Выход: Z – 32-битное слово

- Пусть W[i] – i-й байт слова W
- Y – промежуточная переменная (32 бита)
- Y[0] = q[p[p[q[p[X[0]]^ D[0]]^ C[0]]^ B[0]]^ A[0]]
- Y[1] = p[p[q[q[q[X[1]]^ D[1]]^ C[1]]^ B[1]]^ A[1]]
- Y[2] = q[q[p[p[q[X[2]]^ D[2]]^ C[2]]^ B[2]]^ A[2]]
- Y[3] = p[q[q[p[p[X[3]]^ D[3]]^ C[3]]^ B[3]]^ A[3]]
- Z = M*Y
- M – обратимая матрица 4x4,
- q, p – нелинейные обратимые перестановки (байт->байт)

Twofish – идея атаки

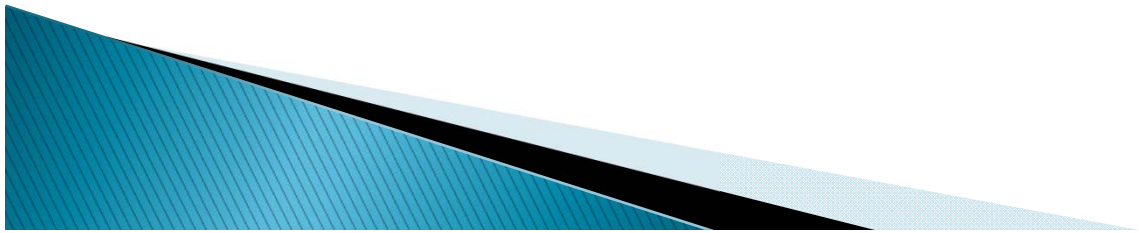
- Итого:
  - Оригинальный ключ отсутствует
  - Зависимости между ключами нелинейны
  - Что же делать?

- Однако:
  - Y[0] = q[p[p[q[p[X[0]]^ D[0]]^ C[0]]^ B[0]]^ A[0]]
  - Используются только младшие байты!
  - => только 2^32 вариантов (A[0],B[0],C[0],D[0])
  - При заданном значении входа X[0] и выхода Y[0] будет ровно 2^24 подходящих (A[0],B[0],C[0],D[0])
  - Более того, можно вычислить A[0] по (B[0],C[0],D[0])
  - Все X[0] заранее известны – используем предвычисления

Passware

Twofish – схема атаки

- Предвычисляем нужные таблицы (достаточно пяти)
- По K[2*i], K[2*i+1] вычисляем h(2*i*R, Me), h((2*i+1)*R, Mo) – тривиально
- По выходам h вычисляем соответствующие Y – тривиально, т.к. матрица M обратима
- Для первых Y и X = 2*i (или 2*i+1) извлекаем таблицы подходящих (A[0],B[0],C[0],D[0]) – по 2^24 элемента (то же – для байтов 1, 2 и 3)
- Ищем пересечения (вероятность 2^{-8} для каждого нового байта)



Passware

Twofish – схема атаки

- Для правильного массива – 1вариант, для неправильного – вероятность случайного совпадения 2^{32-8*n}, где n – число таблиц
- Достаточно 5 таблиц для существенного сокращения перебора
- Так как проверяем сразу 4 байта, не будет ложных срабатываний в массиве длины ~ 4 ГБ
- Сложность предвычислений – n * 2^32

- Пересечение таблиц оптимизируется (детали – в полном тексте)!

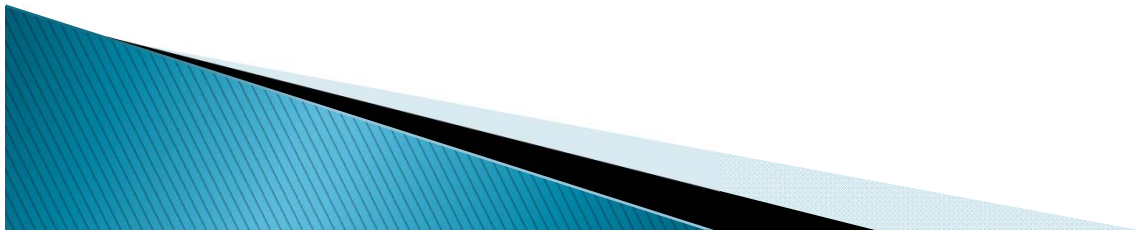
ColdBoot – особенности

- Все методы, не использующие расширенные ключи, требуют перебора вектора ошибок:
  - 1% ошибок, ключ 128 бит (1–2 ошибки) – перебор ~ 2^13 (> 8000)
  - 1% ошибок, ключ 256 бит (2–3 ошибки) – перебор > 2.6 млн
  - 3% ошибок, ключ 128 бит (3–4 ошибки) – перебор > 10 млн
  - 3% ошибок, ключ 256 бит (7–8 ошибок) – перебор > 450 * 10^12
- Энтропийные характеристики могут искажаться по мере накопления ошибок
- Метод расширенных ключей (иногда) может быть адаптирован к ошибкам в потоке данных и без полного перебора!

Противодействие атакам

- Исключение сигнатур
  - Работает против сигнатурного метода
- Маскировка ключей
  - Увеличивает стоимость атаки при неизвестной структуре ключей/масок
- Разреженные ключевые контейнеры
  - Увеличивает стоимость атаки при доступе только к физическим адресам

- НО: Идеальная защита невозможна!!!



Выводы

- Атаки на RAM работают
- Исследования востребованы практикой
- Теория пока отстаёт ☹

- Присоединяйтесь!

Вопросы?

- chilikov@passware.com
- http://www.lostpassword.com