

# Анализ уязвимостей драйверов

Никита Тараканов

Positive Technologies

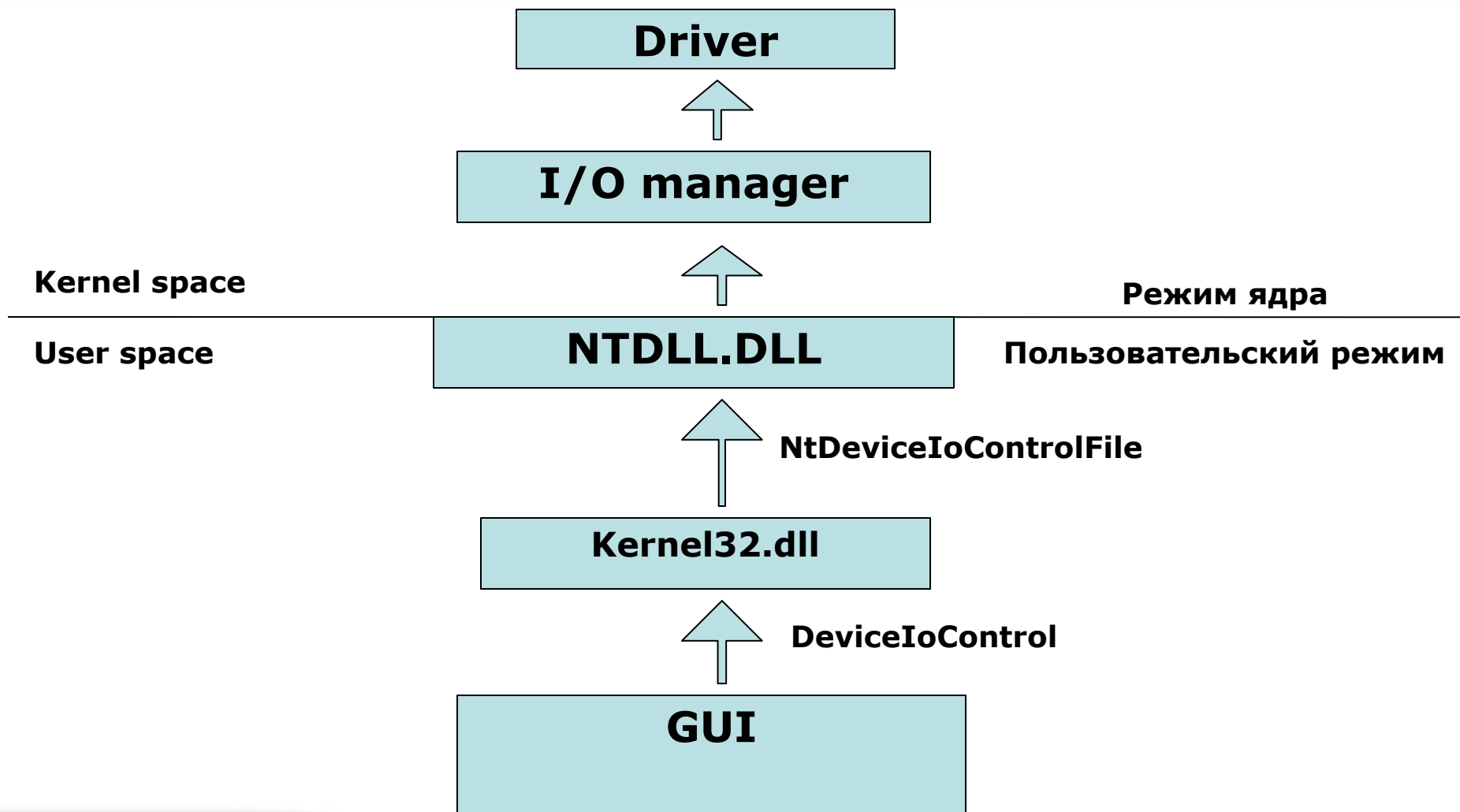


## Драйверы часто встречаются в

- **Антивирусах** – доступ к файловой системе, перехват системных функций
- **Межсетевых экранах** – низкоуровневый доступ к сетевому интерфейсу, перехват пакетов
- **ПО для виртуализации** – низкоуровневый доступ к аппаратному обеспечению
- **Эмуляторах** – низкоуровневый доступ к аппаратному обеспечению
- **Криптосистемах** – низкоуровневый доступ к аппаратному обеспечению



# Взаимодействие



# Функция DeviceIoControl

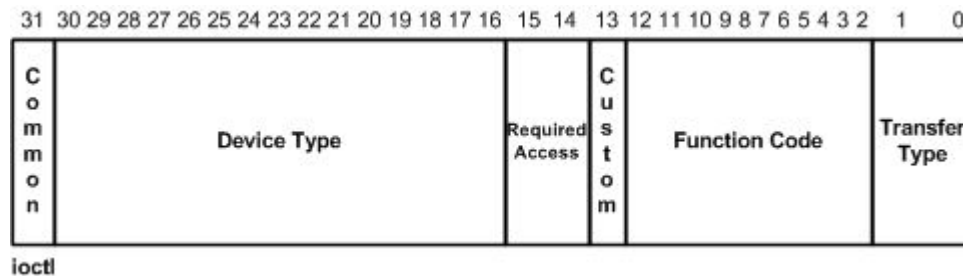
- **Параметры**

- *hDevice* – описатель устройства
- *dwIoControlCode* - управляющий код ввода-вывода
- *lpInBuffer* – указатель на буфер с входными данными
- *nInBufferSize* - размер входного буфера в байтах
- *lpOutBuffer* - указатель на буфер с выходными данными
- *nOutBufferSize* - размер выходного буфера в байтах
- *lpBytesReturned* - количество байт, скопированных в выходной буфер
- *lpOverlapped* - указатель на структуру OVERLAPPED



# Информация закодированная в *dwIoControlCode*

- **Метод передачи данных** – способ передачи входных данных
- **Идентификатор функции** – идентификатор функции будущей запущенной драйвером
- **Доступ к устройству** – права доступа к устройству
- **Тип устройства** – тип устройства



## Методы передачи входных данных

- **METHOD\_BUFFERED**
- **METHOD\_IN\_DIRECT**
- **METHOD\_OUT\_DIRECT**
- **METHOD\_NEITHER**



## Метод: METHOD\_BUFFERED

- Проверка входных параметров функциями ProbeForRead, ProbeForWrite
- I/O manager выделяет память в пространстве ядра
- Объём – максимальный из длин входных и выходных данных
- Копирует данные входного буфера в выделенную память



## Метод: METHOD\_IN\_DIRECT, METHOD\_OUT\_DIRECT

- Проверка входных параметров функциями ProbeForWrite, ProbeForRead
- I/O manager выделяет память в пространстве ядра
- Объём – длина входного буфера
- Копирует данные входного буфера в выделенную память
- I/O manager создаёт MDL для доступа к входному/выходному буферу





## Метод: METHOD\_NEITHER

- **I/O manager передаёт указатели на данные без какой-либо проверки**
- **Отсутствуют проверки ProbeForRead, ProbeForWrite**
- **Все проверки достоверности данных должны выполняться в коде драйвера**
- **Потенциально небезопасный метод**



# Функции ProbeForRead, ProbeForWrite

- **Параметры:**
  - **Address** – указатель на буфер памяти пользовательского режима
  - **Length** – длина в байтах
  - **Alignment** – выравнивание буфера в байтах
- **Проверяют диапазон (Address+Length) на нахождение в пользовательском адресном пространстве**
- **Проверяют адрес на соответствие выравниванию**
- **Проверяют атрибут чтения/записи**



## Обход функций ProbeForRead, ProbeForWrite

- При длине равной нулю, не вызывает исключения, и не выполняет никаких проверок
- Обход всех проверок при длине равной нулю описан только в блоге Microsoft Security Research and Defense, ms08-025 уязвимости win32k.sys  
<http://blogs.technet.com/srd/archive/2008/04/09/ms08-025-win32k-vulnerabilities.aspx>
- Отсутствие данной информации в msdn!!!



# Типичные уязвимости

- **METHOD\_NEITHER**
  - Arbitrary kernel memory write
  - NULL pointer dereference
- **METHOD\_BUFFERED**
  - Buffer overflows
  - NULL pointer dereference
- **METHOD\_IN\_DIRECT, METHOD\_OUT\_DIRECT**
  - Buffer overflows
  - NULL pointer dereference



# Методология поиска уязвимостей

- **Fuzzing**

тестирование без предварительной информации о коде драйвера

- **Fuzzing+мониторинг**

информация о коде драйвера восстанавливается в процессе тестирования

- **Ручной анализ**

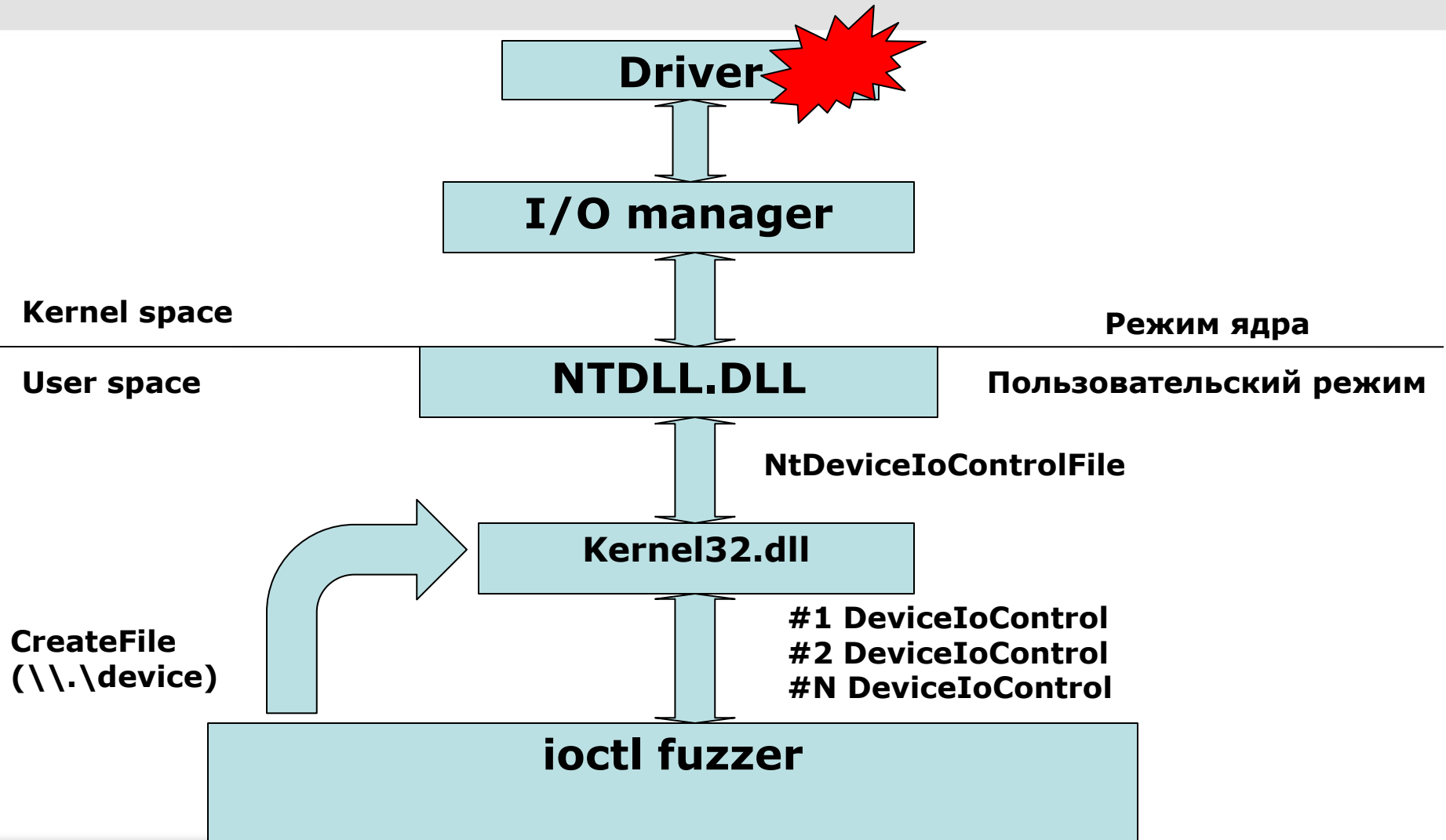
тестирование на основе предварительного изучения кода драйвера (дизассемблирование, отладка)



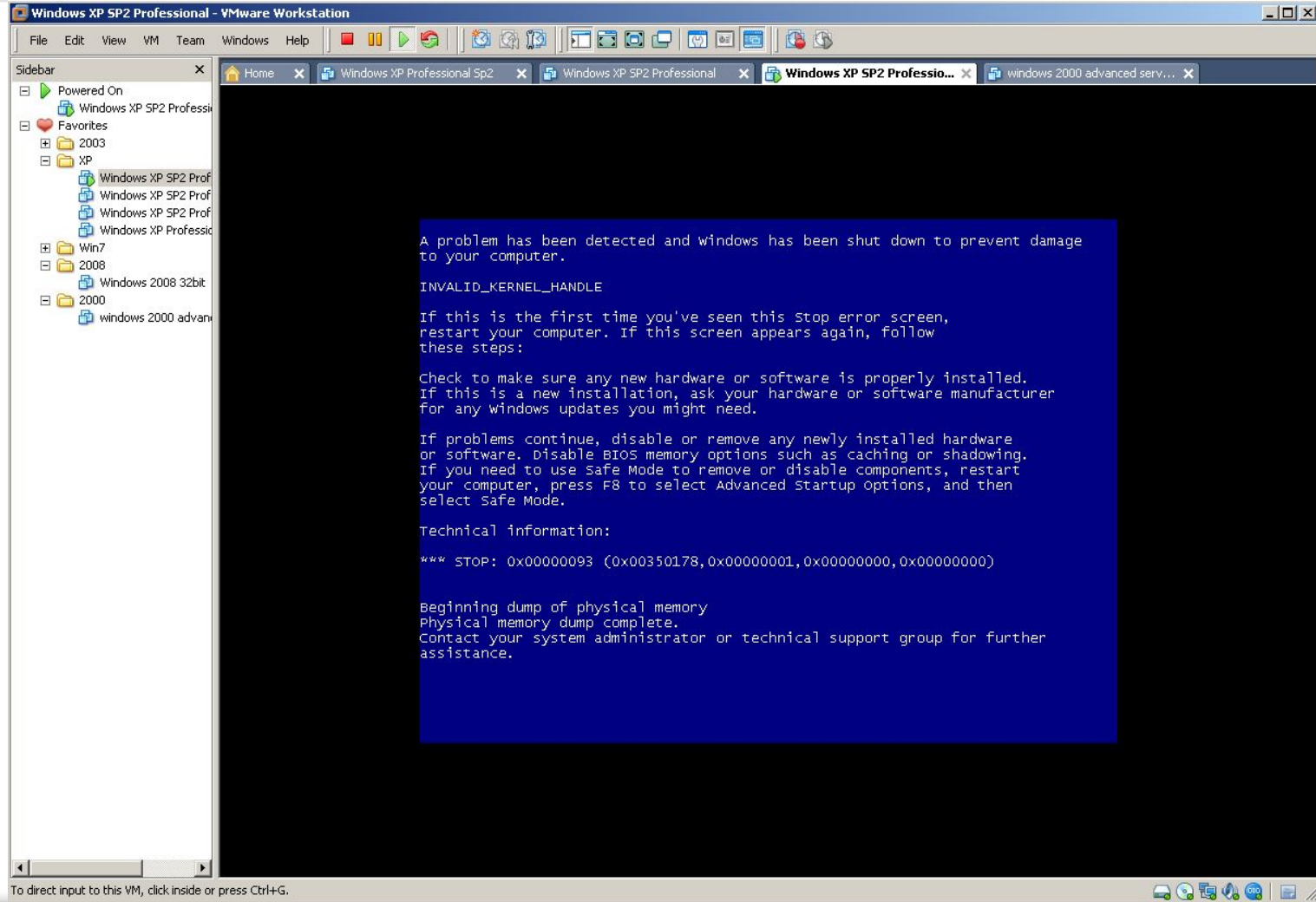
- **Перебор входных параметров DeviceIoControl:**
  - *dwIoControlCode* - DWORD
  - *lpInBuffer* – PVOID
  - *nInBufferSize* - DWORD
  - *lpOutBuffer* – PVOID
  - *nOutBufferSize* - DWORD



# Fuzzing



# 5 минут работы ioctl fuzzer'a





## Fuzzing: Тестовые наборы

- ***dwIoControlCode*** - **полный перебор**
- ***lpInBuffer*** – **NULL, invalid kernel space address**
- ***nInBufferSize*** – **0, 0 – 0x1000, 0x80000000**
- ***lpOutBuffer*** – **NULL, invalid kernel space address**
- ***nOutBufferSize*** - **0, 0 – 0x1000, 0x80000000**



# Fuzzing: Плюсы и минусы

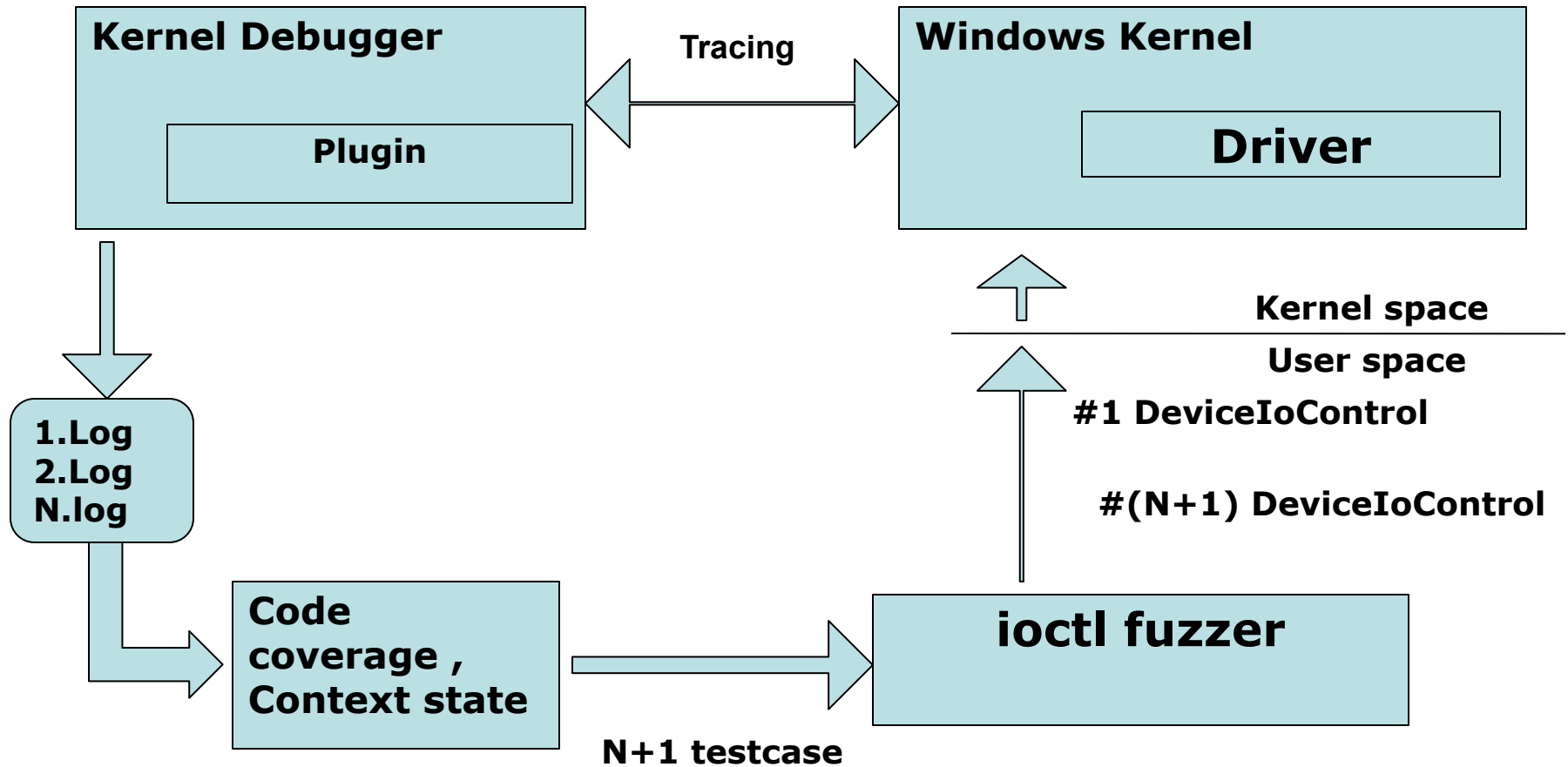
- **Плюсы**
- *Лёгкий в реализации(40 строк C кода)*
- *Неплохие результаты*
- **Минусы**
- *Пропуск уязвимостей, зависящих от определённых значений в буфере или от определённой длины*



- **Отслеживание прохода для определённого тестового набора данных(Code coverage)**
- **Протоколирование состояний контекста(Data flow)**
- **Протоколирование “падений”**
- **Генерация N+1 тестового набора данных**



# Fuzzing + мониторинг



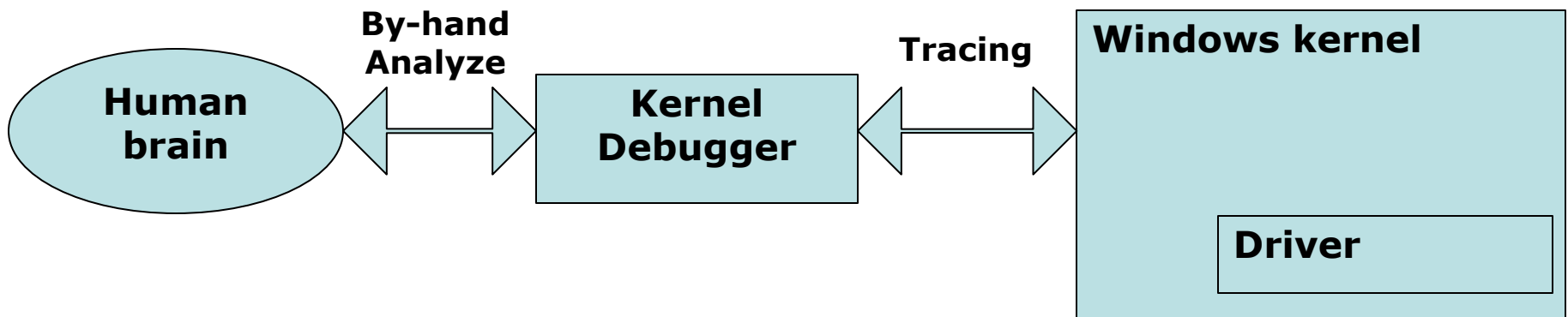
# Fuzzing+мониторинг: Плюсы и минусы

- **Плюсы**
- *Покрытие кода для конкретного тестового набора*
- *Протоколирование найденной уязвимости*
- **Минусы**
- *Сложность реализации*



# Ручной анализ

- Локализация IOCTL обработчика
- Анализ обрабатываемых IOCTL значений
- Формирование тестовых наборов
- Анализ обработки входных значений



# Ручной анализ: Плюсы и минусы

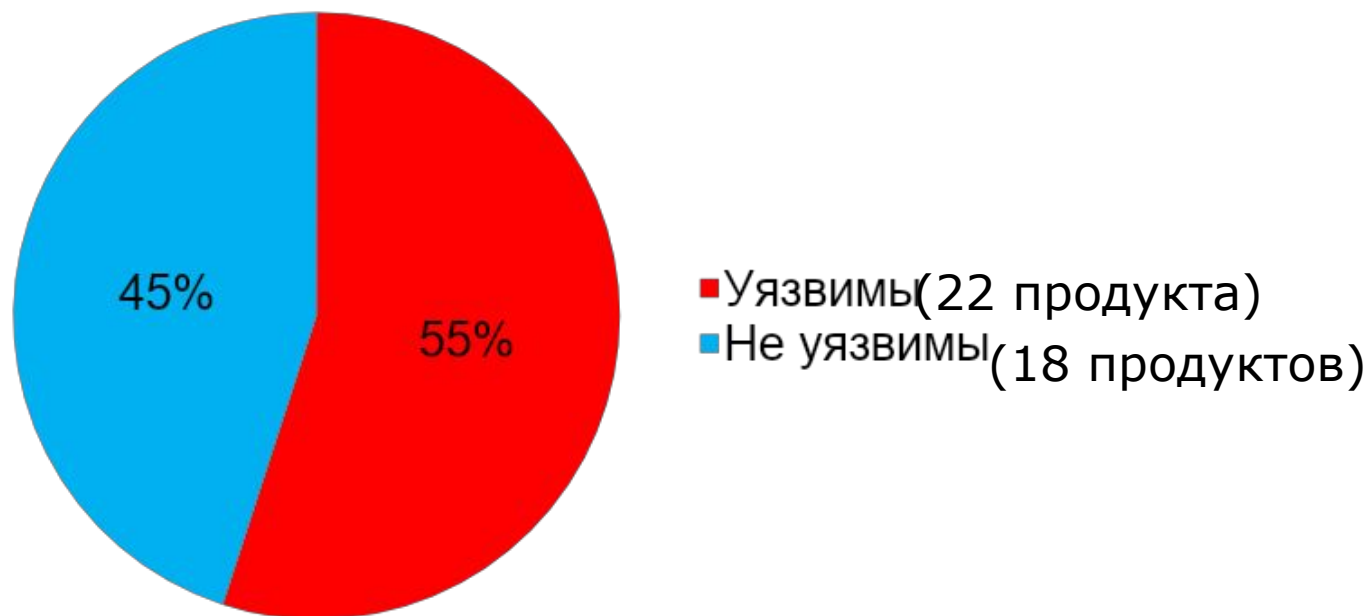
- **Плюсы**
- *Нахождение наибольшего количества уязвимостей*
- *Лучший в комбинации с fuzzing подходом*
- **Минусы**
- *Трудозатратный, требует больших временных ресурсов*



# Статистика протестированных продуктов

Всего в исследовании участвовали 40 продуктов: антивирусы, межсетевые экраны, виртуальные машины, эмуляторы, криптосистемы.

## Результаты тестирования

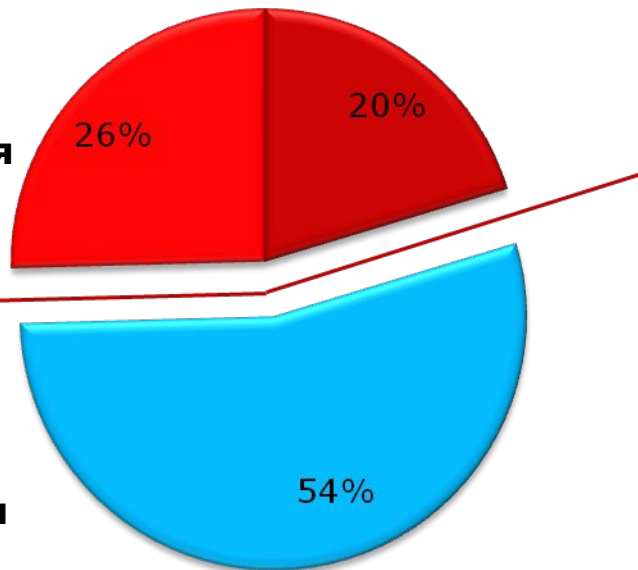




# Количество обнаруженных уязвимостей

Arbitrary memory write	Null pointer dereference	Kernel Pool overflow
12	32	15

**Полная  
компрометация  
системы**



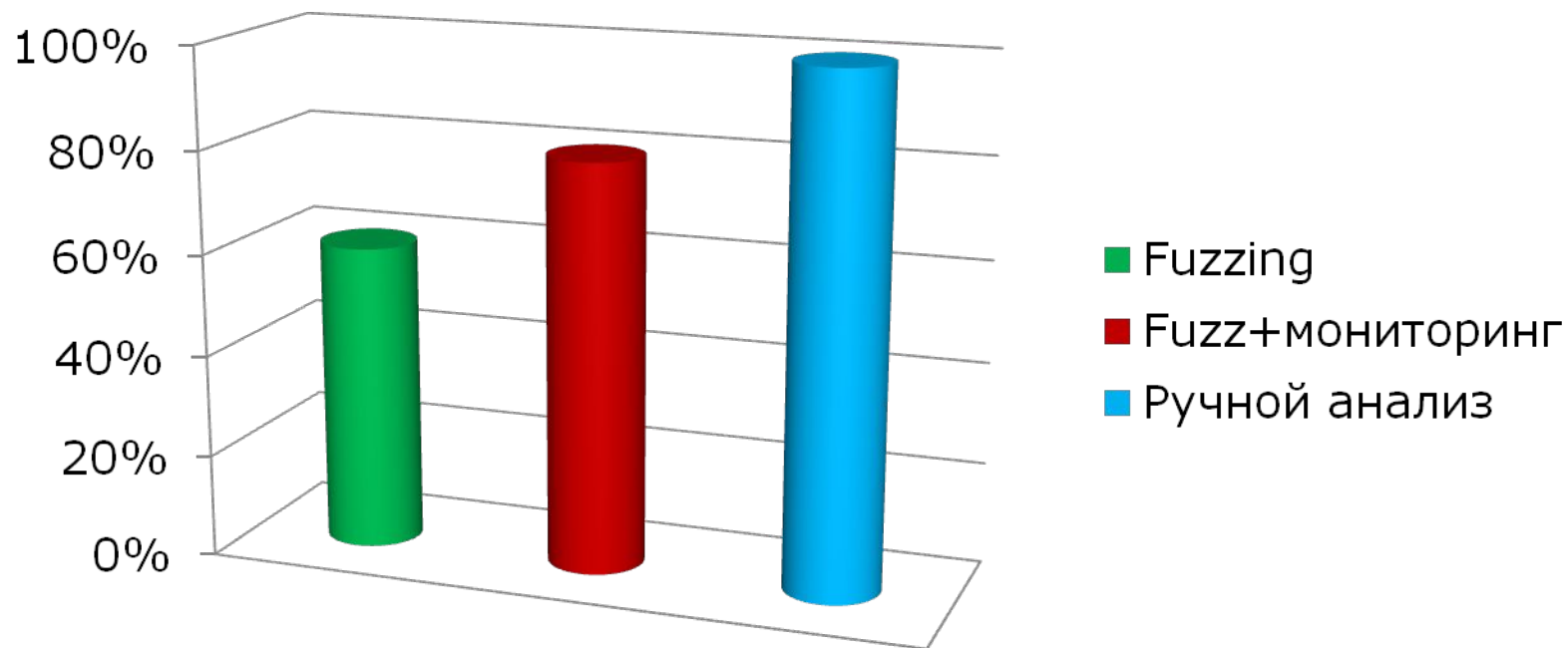
**Отказ в  
обслуживании**

- Arbitrary memory write
- Null pointer dereference
- Kernel pool overflow



# Сравнение методов поиска уязвимостей

## Эффективность



## Пути решения

- **IoCreateDeviceSecure** – безопасное создание устройства (Microsoft, Dr. Web)
- **PsGetCurrentProcessId** – проверка идентификатора процесса (Eset)
- **KeGetPreviousMode** – проверка из какого адресного пространства пришёл вызов (Kaspersky)
- **Hook NtCreateFile, NtDeviceIoControlFile** – перехват функций взаимодействия (Tall Emu)



## Выводы

- **Тестирование драйверов в большинстве компаний не проводится**
- **Большое количество уязвимостей, которые можно быстро обнаружить и исправить**
- **Исправление такого типа уязвимостей в некоторых случаях занимает более 150(!!!) дней**



# Welcome!

- <http://www.securitylab.ru/lab/>

## PT-2008-08 - Microsoft

**Статус:** Исправление отсутствует

**Рейтинг опасности:** Средний (4.7)

AV:L/AC:M/Au:N/C:NI/N/A:C

**Статус уведомления:**

19.11.2008 - Производитель уведомлен

21.11.2008 - Получен ответ от производителя

**Вектор:** Локальный

**Дней с момента уведомления производителя:**



**Уязвимость обнаружена:** Никита Тараканов, Positive Technologies Research Team

## PT-2008-06 - VMWare

**Статус:** Исправление отсутствует

**Рейтинг опасности:** Средний (6.9)

AV:L/AC:M/Au:N/C:SI/CI:A:C

**Статус уведомления:**

14.10.2008 - Производитель уведомлен

16.10.2008 - Получен ответ от производителя

16.10.2008 - Отправлена детальная информация

**Вектор:** Локальный

**Дней с момента уведомления производителя:**



**Уязвимость обнаружена:** Никита Тараканов, Positive Technologies Research Team



## Будущие исследования

- Полные результаты исследования и детальная информация по методам поиска уязвимостей будут опубликованы на сайте компании Positive Technologies в разделе "Аналитика" [www.ptsecurity.ru/analytics.asp](http://www.ptsecurity.ru/analytics.asp)
- Race condition уязвимости : скрытые опасности METHOD\_NEITHER, METHOD\_BUFFERED
- Доработка методов автоматического поиска уязвимостей, публикация исходных кодов ioctl fuzzer'a



# Вопросы?

**Никита Тараканов**  
**[ntarakanov@ptsecurity.com](mailto:ntarakanov@ptsecurity.com)**



- **Раздел "Лаборатория" [www.securitylab.ru/lab](http://www.securitylab.ru/lab) на портале SecurityLab публикует уязвимости, обнаруженные в программном и аппаратном обеспечении. Реализацию этой задачи осуществляют эксперты компании Positive Technologies, входящие в состав Positive Technologies Research Team**





## Ссылки

- <http://msdn.microsoft.com/en-us/library/cc264618.aspx>
- <http://www.uninformed.org/?v=4&a=4&t=txt>
- [http://www.reversemode.com/index.php?option=com\\_remository&Itemid=2&func=fileinfo&id=51](http://www.reversemode.com/index.php?option=com_remository&Itemid=2&func=fileinfo&id=51)

