Пара 15-16

**Теоретическая часть:**

**Основные понятия**

**1. Вредоносное ПО (Malware)** — это программы, созданные для нанесения ущерба данным, устройствам или пользователям.

**2. Принципы анализа вредоносного ПО**

**a) Статический анализ (без запуска)**

**Что это:** Исследование кода/структуры файла без его выполнения.

**Методы:**

**Дизассемблирование** (перевод бинарного кода в ассемблер).

**Декомпиляция**

**Анализ строк и сигнатур** (поиск подозрительных строк, API-вызовов).

**b) Динамический анализ (с запуском)**

**Что это:** Наблюдение за поведением программы в изолированной среде.

**3. Основные правила безопасности при исследовании ВПО**

1. **Изолированная среда**

Использование виртуальных машин без доступа к основной системе.

Отключение общих папок и сетевых мостов.

1. **Отключение интернета**

Вредоносное ПО может заразить другие системы или передать данные злоумышленнику.

1. **Использование инструментов с защитой**

**Режим "только чтение"** для анализа файлов.

**Безопасные дампы памяти** (Volatility).

1. **Фиксация изменений**

Снимки (snapshots) виртуальной машины до и после запуска.

Логирование всех действий.

1. **Запрет на анализ на "рабочей" системе**

Даже "простой" троян может заразить реальные данные.

1. **Использование "песочниц"**

Cuckoo Sandbox, Any.Run — автоматический анализ без риска.

1. **Юридические аспекты**

Исследование только собственных или легально полученных образцов.

**Задача 2: Анализ EICAR-образца**

**1. Статический анализ в Ghidra**

В окне **"Defined Strings"** находим строку:

EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H\*

Ghidra покажет, что файл состоит в основном из этой строки и не содержит сложного кода. Файл не выполняет вредоносных действий, только выводит строку.

**2. Динамический анализ с FakeNet**

**Сетевые запросы:** отсутствуют (FakeNet не зафиксирует активность).

**Логи FakeNet:** нет запросов, так как файл не взаимодействует с сетью.

**Задача 3: Анализ поведения EICAR**

**Ожидание:** Нет.

**Результат:** В Process Monitor не будет записей о CreateFile.

**Изменение реестра:**

**Ожидание:** Нет.

**Результат:** нет событий.

**Запуск процессов:**

**Ожидание:** Нет.

**3. Пример безопасного поведения (гипотетический)**

Если бы файл был модифицирован для теста, он мог бы:

1. **Создать файл:**
2. **Изменить реестр:**
3. **Завершиться без вреда.**

**Анализ поведения:**

**Ожидаемые результаты для оригинального EICAR:**

**Создание файлов (CreateFile)** - **Нет**.

EICAR не создает файлов на диске.

**Изменение реестра (RegSetValue)** - **Нет**.

EICAR не вносит изменений в реестр Windows.

**Запуск процессов (Process Create)** - **Нет**.

EICAR не запускает дочерние процессы.

Задание 4 Исследование трояна

|  |  |  |  |
| --- | --- | --- | --- |
| Образец | Создает файлы | Меняет реестр | Сетевые запросы |
| Eicar.com | Нет | Нет | Нет |
| Game.exe | Нет | Нет | Да(при наличии secret.txt) |

**Вопросы**

**1. Ключевые признаки вредоносного поведения**

|  |  |  |
| --- | --- | --- |
|  | Game.exe | EICAR |
| Чтение файлов | Secret.txt | нет |
| Сетевые запросы | Example.com | нет |
| Изменение реестра | нет | нет |
| Создание файла | нет | нет |

**2. Почему EICAR безопасен?**

Не выполняет реальных действий (не читает файлы, не меняет реестр).

**3. Как FakeNet помогает анализировать сеть?**

**Эмуляция серверов**: подменяет C&C-серверы, чтобы вредонос думал, что подключился.

**Логирование запросов**: фиксирует URL, данные, методы.

**4. Вредоносное ПО (ВПО): типы**

**Вирусы** — заражают файлы.

**Трояны** — маскируются под легитимные программы.

**Шпионское ПО** — крадёт данные (ключи, логины).

**5. Цели анализа ВПО**

Понимание функционала (кража данных, DDoS и т.д.).

Разработка сигнатур для антивирусов.

Защита инфраструктуры (патчи, правила IDS/IPS).

**6. Статический vs. динамический анализ**

|  |  |  |
| --- | --- | --- |
|  | Статический | Динамический |
| Запуск кода | Нет | Да |
| Методы | Декомпиляция, дизассемблирование | Мониторинг процессов, сети и реестра |
| Инструменты | Ghidra | Process monitor |

**7. Инструменты статического анализа**

**Ghidra/IDA Pro** — дизассемблирование, декомпиляция.

**PEiD** — определение упаковщиков.

**Strings** — поиск подозрительных строк в бинарнике.

**8. Обфускация кода**

Запутывание кода для усложнения анализа методами шифрования строк и динамической генерации кода

**9. Признаки вредоносного кода**

Вызовы опасных API, попытки отключить антивирус,

сетевые подключения к подозрительным IP.

**10. Песочница (sandbox)**

Автоматически запускает код в изолированной среде и предоставляет отчет о совершаемых действиях

**11. Обход антивирусов**

**Полиморфизм**: Изменение кода при каждом запуске.

**Анти-эмуляция**: Проверка, работает ли код в песочнице.

**Инъекция**: Внедрение в подлинные процессы (например, explorer.exe).

**12. Инъекция кода**

Внедрение вредоносного кода в процесс жертвы.

**13. Этапы динамического анализа**

1. Запуск в песочнице.
2. Мониторинг:
3. Анализ дампов памяти.

**14. Данные статического анализа**

Импортируемые библиотеки.

Строки.

**15. Польза изучения поведения**

Помогает создавать сигнатуры для антивирусов и правила для обнаружения угроз

**16. Сценарии использования ВПО**

Кража данных (банковские трояны), cаботаж (удаление файлов, DDoS), майнинг

**17. Обратная разработка (Reverse Engineering)**

**Анализ кода без исходников с целью поиска уязвимостей и выявления принципов работы**

**18. Риски анализа в реальной среде**

Заражение системы.

Утечка данных (если вредонос активен).

**19. Меры предосторожности**

Изолированная среда (VM).

Отключение интернета.

Снимки состояния (snapshots) до/после анализа.

**20. Польза в расследованиях**

Определение источника угрозы и методов его работы

**21. Этические аспекты**

Анализировать только легально полученные образцы.

Не использовать знания для создания вирусов.

Сообщать об уязвистях ответственно.