## Приложение № 4. Дополнительные задания

Дополнительные задания предусматривают возможность получения дополнительных баллов. Выдаются данные работы по усмотрению преподавателя и по просьбе студента.

Дополнительное задание может заменить основное задание по усмотрению преподавателя

**Дополнительное задание № 1.**

Тестирование безопасности посредством проверки файлов с популярных сайтов на наличие персональных данных.

Начнём с простого. Каждая сделанная фотография и выложенная в интернет имеет свои метаданные. “ Метаданные — информация о другой информации, или данные, относящиеся к дополнительной информации о содержимом или объекте. Метаданные раскрывают сведения о признаках и свойствах, характеризующих какие-либо сущности, позволяющие автоматически искать и управлять ими в больших информационных потоках.”. Что же это значит? А означает это что при должном умение злоумышленник может узнать о вас почти всё, ведь метаданные существуют не только в фотографиях. Например, посмотрев свойства текстового документа злоумышленник может узнать, начиная от модели устройства, на которое, было сделано и заканчивая вашим адресом и персональным IMEI кодом. Благодаря этим данным можно узнать ваш логин и пароль. Благодаря найденной информации мы можем составить интернет слепок жертвы. Знать его привычки и любимого питомца. Но не эти ли данные мы вводим как секретный вопрос?

Пинтесты имитируют известные способы сетевых атак. Успешность теста на проникновения во многом зависит от полноты и качества составления профиля жертвы. Какими сервисами и программами пользуется жертва? На каких протоколах и портах у нее есть открытые подключения? С кем, как, и когда она общается? Почти все это можно получить из открытых источников.

Нашей учебной целью является сайт. Для получения начальной информации воспользуемся приложением theHarvester. В Kali LINUX эта программа уже есть, после прогона 500 запросов по всем известным поисковикам, можно найти информацию о email. Если у них доменная система, а почтовиком назначен сервер Xchange, то есть вероятность что одна из этих найденных электронных почт будет доменной учетной записью. Теперь обратимся к метаданным. Из них мы можем узнать версию ПО, название

учетной записи, название папок или сетевых дисков. С помощью найденных данных мы можем подобрать эксплойты, из них составить список потенциальных логинов взяв из графы автор или определиться с актуальными темами для фишинговой рассылки, так как в некоторых документах указана личная почта и т.д.

Для данных действий нужно собрать как можно больше метаданных, конечно, есть возможность сделать все это вручную, но займёт это очень много времени. Существует много программ для данных операций, но использовать мы будем только ту, которая распространяется открытым способом и не является вредоносным ПО.

Далее используем программу FOCA. Потенциал у этой программы намного больше, но мы будем её использовать ради сканирование определенного домена в поиске популярных форматах с помощью трёх поисковиков и с последующим извлечением метаданных

Так как скачивание и обработка занимает тоже какое-то количество времени мной был остановлен поиск, так как это мы делаем ради примера, а не для вредоносных атак. Из данной кучи документов я смог узнать:

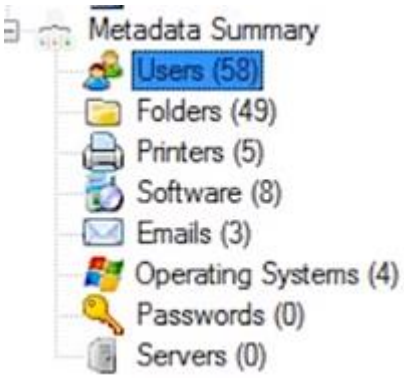


*Рисунок 1. Найденная информация через приложение FOCA*


*Рисунок 2. Найденная информация через приложение FOCA.*

1) 58 юзеров, которые чаще всего ставят имя своего же профиля как логин к какой-нибудь информации. Скажем для авторизации в базе данных пентагона.

2) Так же нашлось 49 папок, некоторые из них сетевые жесткие диски.

3) 5 сетевых принтеров

4) 8 программ, точно установленных на компьютерах. Из них можно выявить самые слабые, подверженные той или иной уязвимости, чтобы эксплуатировать при атаках.

5) 3 персональные (не публичные) электронные почты.

6) И 4 компа, с операционной системой. Организации редко проводят апгрэйды, поэтому велика вероятность наличие системы с критическими уязвимостями.

Но это была лишь одна минута тестирования публичной программой, которая нигде не запрещена и это только один из способов получить данные.

Данное задание разделяется на три этапа и выполняется поэтапно. Язык программирования может быть любым. Выполняется без команд

1 этап.

1. Выбрать популярный веб сайт
2. Ручным методом проверить сайт на наличие различных файлов, таких как документы, изображения и т.д.
3. Выгрузить найденных файлы к себе на компьютер
4. Проверить данные файлы посредством встроенного функционала ОС.
5. Сделать сводную статистику с приложенными скриншотами, о том какой атрибут чаще всего встречался в файлах

2 этап.

1. По выбранному раннему сайту провести визуальный анализ
2. Написать программу, которая будет сканировать сайт (парсилка) и выгружать все выбранные вами файлы
3. Проанализировать найденные файлы, посредством проверки их в автоматическом режиме
4. Сделать сводную статистику по найденным файлам, посредствам работы вашей программы.

3 этап.

1. По выбранному сайту провести дополнительный визуальный анализ и проверить не заблокировали вам доступ к сайту при массивной выгрузке файлов.
2. Выбрать в интернете готовую программу, которая выгружает и анализирует данные с сайта. Пример FOCA

3. Сделать сводную статистику по найденным файлам, посредствам работы готовой программы

После прохождения трех этапов вам следует сверить полученную статистику. Сделать вывод о безопасности персональных данных. Привести кейс, в котором вы могли бы использовать найденную информацию.

**Дополнительное задание № 2.**

В рамках дополнительного задания необходимо:

1. Выбрать сайт для тестирования
2. Если сайт достаточно большой ограничить тестирования путем разработки ТЗ
3. Провести SEO-тестирование, юзабилити-тестирование и проверка на безопасность с помощью нескольких доступных различных интернет-ресурсов.
4. В отчете указать описание на выбранный сайт для тестирования, описания на выбранные интернет-ресурсы, описание пройденных тестов, сравнительные характеристики, выводы

Критерии оценки работ строятся на использовании нескольких платформ для тестирования и сравнения функционала и полученных характеристик после проведения тестирования.

Информативность, полнота и понятийная составляющая должна присутствовать в отчете.

**Дополнительное задание № 3.**

Используя наработанный опыт по предыдущим практическим заданиям и используя изученный материал по дисциплине, необходимо:

1. Разбиться на команды.
2. Первое команда составляет ТЗ на программный продукт необходимый для разработки. Очень важно не прописывать досконально все пункты необходимого программного обеспечения (работа от противного). В дальнейшем должно произойти взаимодействие между командами в рамках валидации системы.
3. Вторая команда выступает в качестве разработчиков программного продукта. Очень важно чтобы в продукт были внесены ошибки\баги. Критерием оценивания работ будет являться трудоемкость при поиске

созданных ошибок и их нетривиальность. Также будет учитываться нахождение не задокументированных багов при тестировании системы.

4. Третья команда выступает в качестве тестировщиков разработанной системы. Используя полученный опыт при выполнении предыдущих работ, необходимо протестировать и найти ошибки в переданном программном продукте.
5. Важно использовать и отобразить в конечном отчете все взаимодействия, которые происходят между командами (если данный пункт нарушает информативность отчета, взаимодействия привести в приложениях)
6. Все команды должны пройти через каждый из этапов. В конечный отчет свести разработанное ТЗ, разработанный согласно нему программный продукт, найденный в ходе тестирования ошибки, ошибки, внесенные в программный продукт согласно заданию, отчеты о валидации системы и изменениях, внесенных после взаимодействия между командами. Выводы.