# ОБНАРУЖЕНИЕ ВРЕДОНОСНЫХ JPEG-ФАЙЛОВ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

**Сафиуллин Т. Т.**

*Белорусский государственный университет, Минск, Беларусь,*
*e-mail: tuleubay.safiullin@mail.ru*

В эпоху развития информационных технологий, интернета, создания множества приложений широко используются файлы формата JPEG. Данный вид графических файлов не стал исключением для внедрения вредоносного вложения. В связи с этим актуальной задачей является обнаружение вредоносных JPEG-файлов.

Изображение JPEG – это двоичный файл, состоящий из последовательности сегментов, каждый из которых начинается с маркера [1]. Маркеры указывают, где именно в файле хранится определённая информация. Чаще всего маркеры размещаются в соответствии со значением длины конкретного сегмента. Самый первый важный маркер – 0xFFD8, который определяет начало изображения. Не менее важен и маркер 0xFFD9, который определяет конец файла. После каждого маркера, за исключением диапазона 0xFFD0 – 0xFFD9 и 0xFF01, сразу идёт значение длины сегмента этого маркера. Маркеры начала и конца файла всегда длиной два байта каждый. Внедрение вредоносного вложения в маркер увеличивает его размер и, следовательно, размер файла.

Изображения JPEG в основном используют два класса сегментов: сегменты маркера и сегменты с энтропийным кодированием. Маркерные сегменты содержат общую информацию (метаданные), такую как: таблицы (таблицы квантования, таблицы энтропийного кодирования и т. д.) необходимые для интерпретации и декодирования сжатых данных изображения, информация о заголовках. Сегменты с энтропийным кодом содержат энтропийно-закодированные данные.

Как отмечается в [2], 99.21% легитимных JPEG-файлов имеют от 0 до 9 DQT-маркеров (таблиц квантования), в то время как только 13.74% вредоносных изображений имеют такое же количество маркеров, а 38.49% из них имеют свыше DQT-10200 маркеров. Некоторые вредоносные JPEG-файлы содержат данные (обычно код) после маркера конца файла (EOI).

В связи с этим для классификации легитимных и вредоносных файлов использовались следующие признаки: размер файла, максимальный размер маркеров, количество маркеров, количество байтов после конца файла. Для извлечения всех этих признаков была написана программа на языке программирования Python.

Для проведения экспериментов была сформирована обучающая выборка JPEG-файлов, состоящая из 9112 изображений: 9000 (98,7%) легитимных и 112 (1,2%) вредоносных. Таким образом, объемы классов обучающей выборки были существенно несбалансированными. Легитимные изображения собраны из социальных сетей (Instagram, ВКонтакте и т. д.) и они были проверены с помощью онлайн-антивируса VirusTotal. В качестве вредоносных отбирались JPEG-файлы, которые были отмечены как вредоносные VirusTotal не менее 5 из 69 антивирусных программ.

Для классификации легитимных и вредоносных JPEG-файлов были использованы следующие методы машинного обучения: деревья решений и ансамбли деревьев решений: случайный лес и стохастический градиентный бустинг [3]. Как отмечается в [3],

эти классификаторы являются устойчивыми к несбалансированности объемов классов обучающей выборки.

Для оценки классификаторов проводилась 3-кратная кросс-валидация и использовались следующие метрики [2]: AUC – площадь под ROC-кривой; TPR (True Positive Rate) – доля правильно классифицированных легитимных файлов; TNR (True Negative Rate) – доля правильно классифицированных вредоносных файлов.

Для исследования эффективности классификации легитимных и вредоносных JPEG-файлов с использованием методов машинного обучения проведены три эксперимента.

Первый эксперимент проводился для сравнения точности классификации легитимных и вредоносных файлов с использованием различных наборов признаков. В качестве наборов признаков использовались признаки, описанные выше, а также построенные с использованием методов MinHash и гистограмм [2].

Метод MinHash генерирует сигнатуру размера N для JPEG-файла на основе N простых хэш-функций. В качестве признаков классификации используются хэш-функции [2].

Метод гистограмм создает гистограмму фиксированного размера, построенную в соответствии с содержимым файла и нормированные значения гистограмм можно использовать в качестве функций для алгоритмов машинного обучения.

Наибольшая точность классификации с использованием в качестве признаков хеш-функций была получена при использовании 200 хеш-функций и классификатора случайный лес. В методе гистограмм наибольшая точность достигнута при размере окна 1024 и интервалом 256 и также с использованием случайного леса.

В таблице 1 приведены наибольшие оценки точности классификации с использованием различных методов построения признаков и методов машинного обучения.

*Табл. 1. Наибольшие оценки точности классификации*

| Методы извлечения признаков | Классификатор | TPR | FPR | AUC |
|---|---|---|---|---|
| MinHash | Случайный лес | 0.805 | 0.050 | 0.893 |
| Гистограмма | Случайный лес | 0.805 | 0.054 | 0.895 |
| Предложенный метод | Стохастический градиентный бустинг | 0.951 | 0.004 | 0.997 |

Отсюда следует что, лучшие по точности результаты получены с использованием предложенного метода построения признаков и стохастического градиентного бустинга.

Второй эксперимент проводился для оценки времени извлечения признаков. Для каждого метода произведены замеры времени, которое необходимо методу для извлечения признаков из одного файла. Все JPEG-файлы были разбиты по размерам на 4 группы. В таблице 2 приведено время извлечения признаков из одного JPEG-файла различными методами построения признаков.

*Табл. 2. Время извлечения из одного JPEG-файла для каждого метода построения признаков в миллисекундах*

| Размер файла (кб) | Предложенный метод | Байтовая гистограмма | Символьная гистограмма | Метод MinHash |
|---|---|---|---|---|
| 0 - 100 | 6 | 6 | 6 | 191 |
| 400 - 500 | 44 | 44 | 47 | 483 |
| 900 - 1000 | 85 | 93 | 105 | 756 |
| 9500 - 9600 | 814 | 910 | 1015 | 4934 |

Результаты показывают, что MinHash самый медленный метод извлечения признаков. Методы написанной программы работают с такой же скоростью, как байтовая и символьная гистограммы.

В третьем эксперименте проведено были сравнены показатели TPR лучших антивирусных программ с показателем TPR классификатора, который обеспечил лучший показатель AUC в первом эксперименте. Был использован тот же онлайн-антивирус VirusTotal, чтобы получить анализ точности обнаружения вредоносной коллекции JPEG-файлов, которая содержит 112 изображений и их классификацию, основанную на 69 антивирусных программах VirusTotal. Был проанализирован отчет, подготовленный VirusTotal, и вычислены показатели TPR для каждой программы, которые представлены в таблице 3.

*Табл. 3. Показатели TPR антивирусных программ ПО*

| Антивирусные программы | Показатель TPR |
|---|---|
| Fortinet | 0.821 |
| Avira | 0.799 |
| Cyren | 0.797 |
| NANO – Antivirus | 0.791 |
| GData | 0.740 |
| Ad – Aware | 0.725 |
| Avast | 0.719 |
| BitDefender | 0.706 |
| Arcabit | 0.702 |
| Qihoo – 360 | 0.698 |

Наибольшее значение показателя TPR, полученное антивирусом Fortinet, на 15% меньше, чем значение показателя TPR, полученное с использованием стохастического градиентного бустинга и предложенным набором признаков. Необходимо отметить, что среднее значение показателя TPR у 10 лучших антивирусных движков (0.73) меньше по сравнению со средним значение показателя TPR у классификаторов, которые были использованы в первом и втором экспериментах (0.929).

Полученные результаты показывают, что методы машинного обучения могут эффективно обнаруживать вредоносные JPEG-изображения.

**Литература**

1. Understanding the Most Popular Image File Types and Formats [Электронный ресурс]. 2016 –Режим доступа: https://1stwebdesigner.com/image-file-types. Дата доступа: 10.12.2020

2. Machine Learning Based Solution for the Detection of Malicious JPEG Images [Электронный ресурс]. 2020. – Режим доступа: https://ieeexplore.ieee.org/document/8967109/metrics#metrics. Дата доступа: 11.12.2020

3. Harrington, P. Machine Learning in Action / P. Harrington. – New York: Manning, 2012. – 382 с.