Введение

Современные веб-технологии способствуют развитию новых методов незаконного использования веб-сайтов и различного контента, предназначенного для пользователей. Речь идет о применении специальных ботов – программ, которые на практике выступают как подмена реальных пользователей веб-ресурсов, осуществляющих определенный перечень действий, целью который может выступать:

- синтаксический анализ сайтов и страниц (парсинг) для последующего копирования их содержимого;
- агрессивное внешнее воздействие на серверы и рабочие станции с целью отказа системы в обслуживании пользователем, т.е. недоступности ресурсов (DDoS-атаки);
- анализ слабых мест системы для ее взлома и других целей.

Но кроме негативного воздействия боты также могут приносить и «пользу». Существенным отличием от других выступает отсутствие маскировки, а данные о них могут быть переданы в клиентские приложения, применяющие определенный сетевой поток. Среди них можно выделить ботов поисковых машин или сервисов мониторинга доступности ресурса.

На сегодняшний день весьма актуально выявление ботов, наносящий урон серверам. Настоящее исследование направлено на понимание разницы между нормальным трафиком и паразитирующим, возможностей последнего, выявление способов его распознавания и защиты от него.

1. Выявление бот-трафика

Запросы ботов в общем трафике можно разделить на следующие виды:

- по принципу работы – они могут быть статическими, динамическими и смежными;
- по объектам анализа – разделяют по единичным запросам, сессиям и группам сессий;
- по параметрам запроса.

Проблема выявления ботов заключается в его взаимодействии с пользователями и последующим обучением поведенческим факторам последнего. Важно отметить, что маскировка под пользователя может быть весьма реалистична, что крайне сложно отследить.

Создание запросов ботнетом с зараженного компьютера можно разделить по следующим параметрам: [1]

- доступность маршрутов: существующие, случайные и смешанные;
- состав отправляемого трафика: статический или динамический;
- промежутки между отправками запросов: статические или динамические.

Сложность отличия пользовательского трафика от ботнета состоит в поведении последнего. Если он имеет набор статических запросов, то его легко обнаружит система защиты и мониторинга. Предполагается, что в этом случае бот имеет заранее заданный перечень команд управления, что позволяет ему выполнять одни и те же действия, делая его уязвимым перед антиботами и прочими средствами безопасности. Однако при наличии динамических запросов работы, его деятельность практически не отличается от реального пользователя.

Учитывая активное развитие ботнета, проверка реальности пользователя претерпела существенные изменения. В связи с тем, что большинство из них способны распознать и ввести символы с предлагаемого изображения системой безопасности ресурса, была осуществлена модификация и усложнение процедуры подтверждения для большинства ботов. Например, активно применяются простые математические действия, не

представляющие сложности для человека, но создающие непреодолимую преграду вредоносной программе.

Однако, побочным эффектом данной защиты является задержка пользователя при прохождении идентификации перед непосредственным использованием ресурса. Это заставляет многих покидать его, тем самым нанося серьезный урон бизнесу.

2. Цель работы

Цель работы – понимание разницы между нормальным трафиком и паразитирующим, возможностей последнего, выявление способов его распознавания.

Задачи:

- поиск информации для последующего анализа;
- анализ распознавания ботов системами безопасности;
- оценка способов распознания ботов с учетом выбранных показателей.

3. Подготовка данных

С целью анализа объема бот-трафика в настоящем исследовании применяется выгрузка логов веб-сервера Nginx 1.10.2 за фиксированный промежуток времени с последующей обработкой. Данный ресурс имеет стандартный формат ведения лога событий.

Полученные данные прошли предварительную обработку, в которой под внимание попадают открытые сессии с фиксированным интервалом, т.к. ботнеты обладают постоянным промежутком временем захода на веб-сервер. Также на основе анализа из выгрузки сессий исключались те, где было меньше 5 запросов, как не целевые. В последствии к ним вычисляются следующие переменные:

- число запросов в сессии;
- длительность сессии (сек.);
- число запросов к страницам;
- длительность запроса к странице (сек.);
- объем ответа;
- интервал между запросами в сессии (фиксированный, плавающий);
- интервал между запросами к страницам (фиксированный, плавающий).

Всего используемых переменных: 7.

Всего объектов в выборке: 11435, из них помечены как боты: 299.

4. Выбор метрики качества

Измерением качества классификации в настоящем исследовании был выбран показатель F-score [1]. С целью вычисления F-score используется термин «точность» (precision) и «полнота» (recall).

$$\text{precision}=\text{TP}/(\text{TP}+FP) \quad (1)$$

$$\text{recall}=TP/(TP+FN) \quad (2)$$

$$F=2*\text{precision} * \text{recallprecision} + \text{recal} \quad (3)$$

где TP – истинно-положительное решение;
TN – истинно-отрицательное решение;
FP – ложно-положительное решение;
FN – ложно-отрицательное решение.

Показатель F-score – это среднее гармоническое между точностью и полнотой,

подходящее для применяемой в настоящем исследовании измерения качества классификации.

5. Сравнение методов машинного обучения

Согласно произведенному анализу различных публикаций по рассматриваемой теме, было выявлено некоторое количество удовлетворяющих условий для расчета:

- логистическая регрессия;
- SVM;
- нейронные сети;
- «случайный лес» (random forest);
- градиентный бустинг над решающими деревьями (AdaBoost, xgboost и т.д.).

Представленные выше алгоритмы использовались в рамках библиотеке skleam [2] для языка программирования python. С целью единовременного анализа и оценки работы классификаторов применялась кросс-валидация [3], которая позволяет с высокой точностью понять его качество. В последствии каждый классификатор сопровождался запуском поиска оптимального набора параметров настройки, сравнение между которыми представлено в таблице ниже.

Таблица - Сравнение качества алгоритмов классификации

Алгоритм	**Значение F-score**
Логистическая регрессия	0.90
SVM	0.89
XGBoost	0.95
Многослойный персептрон (2 слоя)	0.84
Случайный лес	0.93

Самым точным оказался xgboost, который и был взят за основу настоящего исследования.

6. Повышение точности классификации

С целью увеличения точности классификации ботов использовался ряд методов:

- сэмплинг (sampling) [4] – уменьшает дисбаланс классов по причине большего числа объектов, помеченных как «бот», нежели объектов класса «человек»;
- удаление «мусора»;
- алгоритм RFE (Recursive Feature Elimination) – отбирает наиболее значимые признаки для классификации.

Полученный итог показателя F-score: 0,981.

Заключение

Цель настоящей статьи заключалась в исследовании возможностей поведения паразитирующего трафика и его распознавании в общем потоке. В процессе работы производилась выборка информации, ее классификация и сравнительный анализ. В результате проделанной работы наилучшим показателем среди классификаторов стал xgboost, который был подвергнут улучшениям, что подтверждается достигнутым показателем F-score = 0.981 при кросс-валидации. Итог является положительным, а цель настоящего исследования – достигнутой.

Список литературы

1. Большев, А.К. Алгоритмы преобразования и классификации трафика для

обнаружения вторжений в компьютерные сети: дис. канд. техн. наук: 05.13.11. – СПб., 2011. – 142 с.

2. Scikit-learn: Machine Learning in Python / Fabian Pedregosa [и др.] // Journal of Machine Learning Research. – 2011. – №12. – С. 2825-2830.

3. A survey of cross-validation procedures for model selection / Sylvain Arlot [и др.] // Statistics Surveys. – 2010. – №4. – С. 40-79.

4. SMOTE : Synthetic Minority Over-sampling Technique / Nitesh V. Chawla [и др.] // Journal of Artificial Intelligence Research. – 2002. – №16. – С. 321-357.