УДК 004.492

# ТАКСОНОМИЯ ОБФУСКАЦИИ HTTP-АТАК ДЛЯ ОБХОДА WAF ФОРМАЛЬНАЯ МОДЕЛЬ

Крылов Илья Дмитриевич, аспирант, nikplay2000@mail.ru,
Россия, Санкт-Петербург, Университет ИТМО

Евглевская Наталья Валерьевна, канд. техн. наук, доцент, n.evglevskaya@gmail.com,
Россия, Санкт-Петербург, Университет ИТМО

**Введение.** Обфускация является одним из базовых инструментов противодействия средствам прикладной защиты веб-приложений, прежде всего Web Application Firewall (WAF). При атаке на уровне HTTP злоумышленник, как правило, не стремится принципиально изменить полезную нагрузку (например, SQL-или XSS-выражение), а модифицирует её представление и/или способ доставки в запросе так, чтобы нарушить корректность распознавания на стороне WAF. На практике это приводит к ситуации, когда две системы - WAF и веб-приложение работают с разными «представлениями» одного и того же запроса. WAF анализирует входной текст как последовательность байтов/строк и применяет сигнатуры и эвристики, тогда как веб-приложение после декодирования, нормализации и парсинга восстанавливает исходный смысл, выполняет опасную операцию [1].

В отраслевых подходах обфускация чаще описывается как набор приёмов (кодирование, смена регистра, вставка пробелов, разбиение строк), без единой формальной рамки и без привязки к тому, какой именно механизм детектирования не корректно работает и на каком этапе обработки запроса. Между тем, для разработки устойчивых методик противодействия требуется систематизация: формальная модель обфускации как преобразований над строками и структурами запроса и таксономия классов обфускации по уровню воздействия [2].

Цель тезисов - предложить таксономию обфускации HTTP-атак, основанную на формальной модели преобразований, и описать подход к эмпирическому картированию, связывающему классы обфускации с типами защитных механизмов. Рассматриваемые классы атак соотносятся с распространёнными веб-рисками (в частности, Injection и SSRF) по OWASP Top 10.

**Основная часть.** Предлагается формальная постановка, которая включает в себя запрос, обфускацию и задачу обнаружения. Пусть $\Sigma$ - алфавит байтов, $\Sigma^*$ - множество строк над $\Sigma$. Любой HTTP-запрос на уровне передачи представим как $b \in \Sigma^*$. На уровне прикладной обработки вводится структурированное представление запроса x (метод, заголовки, параметры, тело). Далее фиксируется два ключевых отображения. $\Phi$ - предобработка (декодирование и нормализация), приводящая поля запроса к каноническому виду. D - детектор (сигнатурный/статистический), работающий на представлении $\Phi(x)$. Обфускация в общем виде задаётся как оператор O, действующий на данные запроса:

$O : \Sigma^* \to \Sigma^*$ или $O : X \to X$.

Злоумышленник стремится построить O так, чтобы интерпретация сервера веб-приложения «сохранила атаку», а средство защиты не распознало её. В практической трактовке означает сохранение атакующей семантики после декодирования и парсинга при изменении внешней формы на уровне анализа WAF.

**Синтаксическая обфускация (уровень кодирования символов).** Класс включает преобразования строк, сохраняющие смысл после декодирования: процентное кодирование

URI (RFC 3986), Base16 / Base64-кодирование (RFC 4648), HTML-сущности и Unicode-escape. Для процентного кодирования RFC 3986 задаёт механизм представления октета в URI как «%», добавляя две шестнадцатеричные цифры и отмечает, что повторное декодирование/кодирование одной и той же строки может приводить к ошибочной интерпретации. Кодирования Base-N стандартизированы RFC 4648, используются как легитимный контейнер двоичных данных, что делает их удобными для сокрытия сигнатурных подстрок. Если анализ выполняется до полной нормализации, сигнатуры, привязанные к конкретным токенам (SELECT, <script>, ../), могут не сработать [2].

**Семантическая обфускация (уровень логически эквивалентных переписываний).**
Семантическая обфускация меняет форму выражения при сохранении логической эквивалентности или вычислимого результата в целевой подсистеме. Для SQL-инъекций создаются преобразования вида $P \Rightarrow P \vee TRUE$, замены операторов и функций (UNION/UNION ALL, LEN/CHAR_LENGTH, COALESCE/ISNULL), перестановка коммутативных фрагментов. Для XSS, использование эквивалентных конструкций JavaScript (setTimeout, Function, косвенные обращения к document.cookie и т. п.). Получаются не «скрытие строки», а преобразование самой логической структуры выражения, из-за чего простые сигнатуры становятся неполными [3].

**Выводы.** Обфускация HTTP-атак целесообразно описывать как семейство операторов, изменяющих представление данных запроса на разных уровнях от кодирования символов до изменений логической эквивалентности. Данный подход позволяет построить таксономию, пригодную для формальной постановки задачи детектирования и проектирования механизмов нормализации. Предложенная таксономия включает два уровня: синтаксический, семантический. Разделение по уровням отражает различие в том, какие допущения нарушаются в сигнатурном и контекстном анализе WAF.

**Список использованных источников**


1. OWASP Foundation. **OWASP Top 10:2021**. 2021. URL: https://owasp.org/Top10/2021/
2. Fielding R., Reschke J., et al. **RFC 9110: HTTP Semantics**. IETF, 2022. URL: https://datatracker.ietf.org/doc/html/rfc9110
3. Iwanicki P. (iSEC/industry blog). Te/Cl, Cl/Te, Te Obfuscation (HTTP desync / request smuggling variants). 2025. URL: https://blog.isec.pl/waf-evasion-techniques/



Крылов И.Д. _____

Евглевская Н.В. _____