Информационная безопасность
iBatyr

Web Application Firewall нового поколения

Особенности процессов обеспечения информационной безопасности веб-приложений и требования к WAF

В связи с этим, интерфейс управления WAF должен требовать минимум ресурсов на решение задач по мониторингу и расследованию инцидентов, а также, по настройке, что в свою очередь, подразумевает обязательное выполнение следующих требований:

- централизованное управление всеми узлами анализа, независимо от того, какие приложения они обслуживают, возможность создания правил для отдельных приложений, групп приложений или всех приложений. Выполнение данного требования позволит существенно упростить эксплуатацию решения, т.к. обычно имеется большое количество приложений разных типов.
- группировка событий по различным критериям и скоринговая система для оценки критичности событий.
- подавление ложных срабатываний (в том числе – автоматическое, но с возможностью корректировки результатов).
- автоматическое сокрытие в интерфейсе заведомо неважной информации, в частности запросов к статическим ресурсам (особенно актуально для Заказчиков, у которых запросов в трафике очень значительное количество).
- базовая настройка WAF должна требовать минимальных ресурсов.
- тонкая настройка WAF под имеющееся приложение не должна требовать программирования или доработки решения.
- возможность автоматического обучения WAF, в том числе – инкрементного (дополнительно к уже имеющимся настройкам) и ручной корректировки результатов обучения.
- возможность автоматического обучения WAF на уровне бизнес-логики - определение логических действи, параметров действий и моделей этих параметров. Поскольку описание логики сложного внутрикорпоративного приложения, особенно в условиях постоянных изменений – ресурсоемкая задача.

Также необходимо отметить, что как правило у специалистов по информационной безопасности на данный момент недостаточно знаний и опыта в области безопасности веб-приложений (application security) как с точки зрения возможных атакующих воздействий (offensive security), так и с точки зрения мер защиты (defensive security), в том числе, по настройке и обслуживанию WAF.

При этом необходимо отметить, что у специалистов в области информационной безопасности в любом случае также будет ограниченный уровень знаний в области веб-разработки, поэтому в некоторых случаях им необходимо будет обращаться к разработчикам приложений (например, при настройке WAF или расследовании инцидентов).

- WAF должен давать возможность для специалиста по информационной безопасности понять суть происходящего в приложении без глубоких знаний в области веб-разработки и с минимальным привлечением разработчиков.

Возможности iBatyr WAF

Требование	iBatyr WAF
Защита от атак на HTTP-протокол - протокольная валидация (переполнение буфера, HTTP Parameter Pollution, HTTP Verb Tampering и др.).	Да
Защита от синтаксических атак класса Injection (SQL Injection, OS Command Injection, Diectory Traversal, Code Injection и т.п.)	Да
Защита от переборных атак и атак типа «умный DoS»	Да (уникальная возможность установки независимых параметров модуля защиты от переборных атак для отдельных действий в приложении)
Защита сессий пользователей (контроль связности параметров сессии).	Да
Защита от раскрытия чувствительных данных (данных об архитектуре или конфигурации приложения и его компонентов)	Да
Защита от 0-day и 1-day угроз	Да. За счет использования позитивной и негативной (сигнатуры) моделей.
Глубокий разбор передаваемых данных с различными уровнями вложенности	Да – можно реализовать разбор как стандартных, так и кастомных форматов на произвольном уровне вложенности.
Защита от логических атак на механизмы авторизации	Да
Защита от модификации параметров запроса без нарушения синтаксиса	Да
Контроль нарушения обычной последовательности запросов (действий)	Да
Возможность идентификации отдельных логических действий и их параметров в запросах пользователей.	Да
Возможность установки моделей параметров действий (диапазоны длин, перечисления, регулярные выражения, статистические модели и т.п.) и реагирования в случае отклонения.	Да
Возможность контроля авторизации в приложении на уровне отдельных логических действий, реагирование в случае нарушения.	Да
Возможность установки на WAF для приложения модели авторизации, отличной от модели самого приложения.	Да
Возможность передачи данных в формате бизнес-логики во внешнюю аналитическую систему (например, SIEM или BI) для целей расследования действий работников в приложении, выявления нарушений процессов, предотвращения злоупотреблений и т.п.	Да
WAF должен включаться в инфраструктуру доставки приложений по схеме «в разрыв» с возможностью блокировки подозрительной активности.	Да
Компоненты WAF должны поддерживать виртуализацию в контейнерах Docker	Да
Разные компоненты WAF должны иметь возможность размещаться в разных контейнерах/на разных серверах.	Да
WAF должен иметь возможность работать в качестве выделенного модуля для Nginx (в той конфигурации, что уже используется в инфраструктуре доставки приложений).	Да

WAF должен обеспечивать высокую доступность защищаемых ресурсов в вышеописанных конфигурациях.	Да. Доступность защищаемых сервисов обеспечивается режимом «программный bypass» при котором, при отсутствии ответа от компонентов WAF, пользовательский трафик направляется на приложение без проверки. Аппаратный bypass – только для аппаратного решения.
WAF должен обеспечивать хорошую горизонтальную масштабируемость (в случае повышения нагрузки или появления новых приложений).	Да. Масштабируется добавлением новых узлов анализа к имеющейся системе управления (БД)
Централизованное управление всеми узлами анализа, независимо от того, какие приложения они обслуживают, возможность создания правил для отдельных приложений, групп приложений или всех приложений.	Да
Группировка событий по различным критериям и скоринговая система для оценки критичности событий.	Да. Группировка событий выполняется. Скоринговая система относительно простая – на основе рейтинга выявленной уязвимости и настроек правила.
Подавление ложных срабатываний (в том числе – автоматическое, но с возможностью корректировки результатов).	Да. Имеется отдельный настраиваемый механизм подавления ложных срабатываний. Используется безопасный метод обнаружения ложных срабатываний на основе машинного обучения.
Автоматическое сокрытие в интерфейсе заведомо неважной информации, в частности запросов к статическим ресурсам	Да. Статические ресурсы и однотипные запросы (роботы) выявляются автоматически с помощью машинного обучения.
Базовая настройка WAF должна требовать минимальных ресурсов.	Да. Работает «Из коробки»
Тонкая настройка WAF под имеющееся приложение должна требовать минимум ресурсов.	Да. Графический интерфейс для настройки любых позитивных моделей. Программирование не требуется даже для сложных приложений и сценариев использования.
Возможность автоматического обучения WAF, в том числе – инкрементного (дополнительно к уже имеющимся настройкам) и ручной корректировки результатов обучения.	Да (определение логических действий, параметров действий и моделей этих параметров)
Возможность автоматического обучения WAF на уровне бизнес-логики - определение логических действии, параметров действий и моделей этих параметров.	Да
WAF должен давать возможность для специалиста по информационной безопасности понять суть происходящего в приложении без глубоких знаний в области веб-разработки и с минимальным привлечением разработчиков.	Да. Интерфейс WAF обеспечивает несколько форматов представления одной и той же информации одновременно. Администратор может выбрать какой ему больше подходит: «сырой запрос», графическое «дерево разбора» запроса или уровень бизнес-логики.