Программные системы и вычислительные методы
Правильная ссылка на статью:
Козырева Н.И., Мухтулов М.О., Ершов С.А., Новосельцева С.В., Ахмадуллин Д.А. Современные методы предотвращения DDoS-атак и защиты веб-серверов // Программные системы и вычислительные методы. 2025. № 2. DOI: 10.7256/2454-0714.2025.2.73667 EDN: BOUTCT URL: https://nbpublish.com/library_read_article.php?id=73667

Современные методы предотвращения DDoS-атак и защиты веб-серверов

Козырева Надежда Ивановна

кандидат технических наук

доцент; кафедра информационной безопасности (ИБ); Поволжский государственный университет телекоммуникаций и информатики

443010, Россия, Самарская обл., г. Самара, Самарский р-н, ул. Льва Толстого, д. 23

n.kozyreva@psuti.ru

Мухтулов Михаил Олегович

ORCID: 0009-0009-5985-3247

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

443125, Россия, Самарская обл., г. Самара, Промышленный р-н, Московское шоссе, д. 252, кв. 197

mixa.1204@inbox.ru

Ершов Сергей Александрович

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

446430, Россия, Самарская обл., г. Кинель, ул. Ульяновская, д. 28, кв. 7

vizionera8@gmail.com

Новосельцева София Владимировна

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

443093, Россия, Самарская обл., г. Самара, Железнодорожный р-н, ул. Мориса Тореза, д. 31, кв. 79

sunny.tea.with.lilac@gmail.com

Ахмадуллин Динар Айратович

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

443101, Россия, Самарская обл., г. Самара, Куйбышевский р-н, ул. Хасановская, д. 14, кв. 17

dinarnevashno@yandex.ru

Статья из рубрики "Кодирование и защита информации"

DOI:

10.7256/2454-0714.2025.2.73667

EDN:

BOUTCT

Дата направления статьи в редакцию:

12-03-2025

Аннотация: Объектом исследования являются веб-серверы и их поведение в условиях высокоинтенсивных распределённых атак типа «отказ в обслуживании» (DDoS), нарушающих доступность сервисов и устойчивость инфраструктур. В качестве предмета исследования рассматриваются современные методы защиты серверных приложений от DDoS-угроз, включая анализ трафика, фильтрацию по частоте запросов, межсетевые экраны (файрволы) и облачные решения. Подробно анализируется эффективность различных технологий защиты, таких как Rate Limiting, ModSecurity, Google Cloud Armor и Cloudflare, а также их интеграция с традиционными средствами — межсетевыми экранами, системами предотвращения вторжений (IPS) и прокси-серверами. В рамках исследования разработан тестовый сервер на языке Go, имитирующий поведение реального веб-приложения с логированием и сбором статистики. Для моделирования DDoS-атак использован инструмент MHDDoS, обеспечивающий широкое покрытие типов угроз: от UDP и SYN Flood до HTTP Flood и Slowloris. Методы исследования включают эмуляцию атак на сетевом и прикладном уровнях трафика, нагрузочное тестирование, сбор метрик (процент заблокированных запросов, среднее время отклика, нагрузка на CPU и RAM) и сравнительный анализ эффективности решений. Научная новизна исследования заключается в разработке и применении экспериментальной модели имитации DDoS-атак с использованием специализированного Go-сервера, что позволило в реалистичных условиях оценить эффективность современных локальных и облачных средств защиты. Анализ реальных кейсов демонстрирует эффективность адаптивных стратегий против современных сложносоставных атак. Выводы подчёркивают необходимость активного подхода к безопасности, учитывающего как технологические, так и организационные меры защиты. Полученные результаты имеют практическую ценность для специалистов по кибербезопасности, системных администраторов и разработчиков защитных решений, предоставляя им методическую основу для создания устойчивых к DDoS веб-инфраструктур. Работа также обозначает перспективные направления для дальнейших исследований в области интеллектуальных систем обнаружения и нейтрализации атак.

Ключевые слова:

DDoS-атаки, Веб-сервер, Cloudflare, Rate Limiting, Fail2Ban, Кибербезопасность, Защита сети, WAF, Автоматизированное предотвращение атак, Машинное обучение

Введение

В современном мире интернет-технологий стабильная работа вебсерверов критически важна для бизнеса, государственных учреждений и частных пользователей. Однако с ростом цифровых сервисов возрастает и число кибератак, среди которых DDoS-атаки (Distributed Denial of Service) остаются одной из наиболее серьезных угроз. Эти атаки направлены на исчерпание вычислительных и сетевых ресурсов сервера, что приводит к его замедлению или полной недоступности для пользователей.

Особенность современных DDoS-атак заключается не только в их количестве, но и в усложнении механизмов реализации. Если раньше злоумышленники использовали примитивные методы, основанные на отправке большого количества запросов, то сегодня атаки маскируются под легитимный трафик, применяются ботнет-сети, анонимные прокси и даже технологии искусственного интеллекта для обхода защитных механизмов. Это создаёт значительные вызовы для администраторов информационных систем и требует применения комплексных решений для защиты.

DDoS-атаки классифицируются по уровню воздействия:

Сетевые атаки (например, UDP Flood, SYN Flood) перегружают серверные соединения огромным числом пакетов, блокируя сетевые ресурсы.

Атаки на уровне приложений (например, HTTP Flood, Slowloris) имитируют реальную пользовательскую активность, усложняя их обнаружение традиционными методами.

Комбинированные атаки объединяют несколько техник, что делает их особенно сложными для предотвращения.

Актуальность данной проблемы обусловлена не только ростом атак, но и их экономическими последствиями. Компании несут убытки из-за простоя сервисов, теряют доверие пользователей и могут столкнуться с утечками данных. В связи с этим необходимо детально изучить современные подходы к защите веб-ресурсов, включая облачные решения (Cloudflare, Google Cloud Armor), локальные механизмы защиты (Rate Limiting, Fail2Ban) и перспективные технологии, использующие машинное обучение и поведенческий анализ трафика.

Настоящая работа посвящена исследованию этих методов, их сравнительному анализу и перспективам дальнейшего развития стратегий защиты от DDoS-атак. Объектом настоящего исследования являются веб-серверы, подвергающиеся DDoS-атакам, а предметом — методы и средства их защиты от распределённых атак отказа в обслуживании, включая локальные и облачные технологии фильтрации и анализа трафика.

1. Классификация DDoS-атак и их последствия

DDoS-атаки (Distributed Denial of Service) представляют собой одну из наиболее распространённых форм киберугроз, направленных на нарушение работы серверной инфраструктуры при помощи отправки огромного числа запросов. Основная цель данных атак — нарушить доступность сервисов за счёт перегрузки вычислительных ресурсов чрезмерным объёмом внешнего трафика. DDoS-атаки классифицируются в зависимости от уровня сетевой модели OSI, на который они воздействуют [3, с. 251-252].

1.1. Атаки на уровень сети (L3/L4 OSI)

Сетевые атаки нацелены на перегрузку сетевого оборудования и исчерпание полосы пропускания канала связи. Они отличаются высокой скоростью передачи пакетов и

большим объемом входящего трафика.

UDP Flood – злоумышленник отправляет большое количество UDP-пакетов на случайные порты целевого сервера, заставляя его обрабатывать каждый пакет и отправлять ICMP-ответы «Destination Unreachable». В результате перегружается процессорная мощность сервера и сетевой канал [1, с. 574].

ICMP Flood (Ping Flood) – атака, при которой злоумышленник отправляет огромное количество ICMP Echo-запросов (ping-запросов) с целью перегрузки сетевого стека сервера.

SYN Flood – атака эксплуатирует механизм установления TCP-соединений (трёхстороннее рукопожатие). Злоумышленник отправляет большое количество SYN-запросов, но не завершает установление соединения, что приводит к исчерпанию доступных соединений сервера и отказу в обслуживании для легитимных пользователей.

SMURF — это вид DDoS-атаки, основанный на отправке ICMP-запросов (аналогичных ping) на широковещательный адрес сети через маршрутизатор. В результате все устройства в сети отвечают на запрос, перегружая жертву трафиком [4, с. 430].

1.2. Атаки на уровень приложений (L7 OSI)

DDoS-атаки на уровне приложений нацелены на перегрузку серверных ресурсов, таких как процессорное время и оперативная память, путем отправки большого количества сложных запросов, имитирующих активность реальных пользователей. Эти атаки сложнее в обнаружении, так как трафик выглядит легитимным.

HTTP Flood – злоумышленник отправляет многочисленные HTTP-запросы на сервер, вынуждая его загружать веб-страницы, выполнять обработку динамического контента и обращаться к базе данных.

Slowloris-атака — это метод атаки, при котором злоумышленник удерживает множество HTTP-соединений открытыми, отправляя неполные запросы. Сервер, ожидая завершения этих запросов, выделяет ресурсы для каждого соединения, что приводит к их истощению и делает сервер недоступным для обычных пользователей [5, с. 3-4].

1.3. Комбинированные атаки (multi-vector attacks)

Комбинированные атаки представляют собой угрозу, при которой злоумышленники одновременно используют несколько методов воздействия, например, перегрузку сервера на сетевом уровне (UDP Flood) и атаки на уровень приложений (HTTP Flood). Такой подход усложняет обнаружение и повышает вероятность успешного нападения.

Эффективная защита от подобных атак требует многоуровневых стратегий, включающих фильтрацию трафика, облачные сервисы и алгоритмы машинного обучения для выявления аномалий в реальном времени.

2. Локальные методы защиты от DDoS-атак и способы их выявления

2.1. Способы выявления DDoS-атак

Способы выявления DDoS-атак можно разделить на две основные группы: пассивные и активные.

Пассивные методы основаны на наблюдении за сетевым трафиком и анализе его

параметров. Они не взаимодействуют напрямую с сетью или источниками атаки, а лишь отслеживают поток данных, выявляя аномалии, характерные для DDoS-атак. К таким методам относятся мониторинг сетевого трафика, анализ нагрузки на серверы, изучение логов серверов и сетевых устройств, а также системы обнаружения аномалий.

Активные методы предполагают прямое взаимодействие с сетью или потенциальными атакующими устройствами. В их рамках может выполняться отправка тестового трафика для проверки доступности сервера и анализа его ответов, а также передача управляющих сигналов для блокировки атаки. Однако такие подходы могут не только создавать дополнительную нагрузку на сеть, но и оказаться неэффективными в случае мощных DDoS-атак. К активным методам относятся, например, системы обнаружения внешних атак [2, с. 5].

Основные проблемы при обнаружении DDoS-атак:

1. Ложные срабатывания – система может ошибочно распознать атаку, что приводит к потере времени и ресурсов.

2. Сложность настройки – требует специальных знаний для эффективной работы.

3 . Высокие затраты ресурсов – анализ большого трафика требует мощного оборудования.

4. Трудности с новыми атаками – некоторые системы не распознают новые методы атак.

5. Постоянное обновление – необходимо адаптироваться к новым угрозам.

6. Медленное реагирование – обнаружение атаки не всегда означает ее мгновенную блокировку [2, с. 7].

2.2. Ограничение частоты запросов (Rate Limiting)

Rate Limiting — это технология, предназначенная для контроля нагрузки на сервер. Она ограничивает число запросов, которые может отправить один клиент в течение определенного периода. Этот подход используется для предотвращения атак, например, HTTP Flood, когда злоумышленники пытаются перегрузить систему массовыми запросами.

Суть метода заключается в мониторинге количества запросов, поступающих с каждого IP-адреса. Если количество запросов превышает допустимый лимит, система либо блокирует их, либо замедляет обработку, чтобы избежать перегрузки сервера [8].

2.3. Блокировка подозрительных IP-адресов (Fail2Ban)

Fail2Ban — это утилита для анализа логов, которая автоматически обнаруживает подозрительные действия, такие как попытки несанкционированного доступа, и блокирует соответствующие IP-адреса. Этот инструмент активно применяется для защиты серверов от атак методом перебора (brute-force), а также для выявления признаков DDoS-активности.

Когда система фиксирует аномальное поведение, например, множество неудачных попыток входа или резкий всплеск запросов, Fail2Ban временно блокирует подозрительный IP-адрес с помощью инструментов вроде iptables или firewalld. Блокировка действует в течение определенного времени, после чего автоматически снимается. Такой подход особенно эффективен для противодействия низкоинтенсивным

атакам и сканированию на наличие уязвимостей [9].

2.4. Web Application Firewall (WAF)

WAF (например, ModSecurity) работает на уровне сервера, анализируя HTTPзапросы и блокируя потенциально опасный трафик. В отличие от облачных решений, локальный WAF функционирует непосредственно на сервере, обеспечивая защиту от атак на уровне приложений, таких как SQL-инъекции, межсайтовый скриптинг (XSS) и распределенные атаки на отказ в обслуживании (DDoS).

ModSecurity предоставляет возможность настраивать пользовательские правила фильтрации, которые можно адаптировать под конкретные требования веб-приложения. Однако для поддержания высокой эффективности защиты необходимо регулярно обновлять конфигурацию и отслеживать новые угрозы, что требует постоянного внимания со стороны администраторов [10].

3. Облачные сервисы защиты от DDoS-атак

Облачные провайдеры предлагают мощные инструменты защиты от DDoSатак за счет высокой пропускной способности и глобально распределенной инфраструктуры. Например, использование CDN (сети доставки контента) позволяет скрывать исходный IP-адрес сервера, что затрудняет атаки [2, с. 7-8].

3.1. Cloudflare

Cloudflare — это облачная платформа, предназначенная для защиты вебресурсов от DDoS-атак. Она обеспечивает фильтрацию входящего трафика и повышает доступность веб-сервисов. Cloudflare функционирует как промежуточный слой между пользователями и сервером, анализируя запросы и блокируя вредоносные действия до того, как они достигнут инфраструктуры клиента [11].

Cloudflare использует глобальную распределенную сеть для балансировки нагрузки и минимизации воздействия атак. Встроенные механизмы защиты, такие как Web Application Firewall (WAF) и анализ поведения трафика, позволяют автоматически выявлять и блокировать подозрительные запросы. Дополнительно система применяет Rate Limiting для ограничения частоты обращений, а также CAPTCHA и JavaScript Challenge для фильтрации автоматизированных ботов.

3.2. Google Cloud Armor

Google Cloud Armor — это облачная платформа, предназначенная для защиты веб-приложений и сервисов, размещенных в инфраструктуре Google Cloud, от DDoS-атак и вредоносного трафика. Решение интегрируется с балансировщиками нагрузки Google Cloud и выполняет фильтрацию на границе сети, что позволяет минимизировать нагрузку на конечные серверы.

Google Cloud Armor использует адаптивные методы защиты, такие как анализ поведения трафика, интеллектуальные правила блокировки и технологии машинного обучения для обнаружения аномальных паттернов запросов. Платформа включает предустановленные политики безопасности, которые защищают от распространенных атак на уровне приложений, например, HTTP Flood. Кроме того, пользователи могут создавать собственные правила фильтрации, основанные на геолокации, IP-адресах и других параметрах, что обеспечивает гибкость в борьбе с новыми угрозами [12].

4. Методология исследования

В данном исследовании была проведена экспериментальная оценка эффективности современных технологий защиты веб-серверов от DDoS-атак. Основной целью работы стало тестирование различных механизмов фильтрации трафика в условиях эмуляции атак различного типа. Для этого был разработан и развернут тестовый сервер на языке Go. Непосредственно для развертывания использовалась операционная система Ubuntu.

В ходе исследования были поставлены следующие задачи:

• Реализация тестового веб-сервера с функциями логирования и анализа запросов.

• Моделирование различных видов DDoS-атак, таких как SYN Flood, UDP Flood, HTTP Flood и DNS Query Flood.

• Последовательное подключение различных технологий защиты и тестирование их эффективности.

• Анализ полученных данных с целью сравнительной оценки защитных механизмов и степени опасности различных типов атак.

Для эмуляции атак использовался инструмент MHDDoS. Это обусловлено тем, что он разворачивается на любой инфраструктуре (написан на Python) и имеет удобную для open-source проекта документацию. В ходе тестирования сервер подвергался последовательным атакам с включением различных технологий защиты, таких как Rate Limiting, ModSecurity WAF, Google Cloud Armor, Cloudflare. Для каждой технологии фиксировалось общее количество обработанных запросов, доля заблокированных атак, задержка ответа сервера и потребление системных ресурсов.

5. Реализация эксперимента

5.1. Разработка тестового сервера

Для проведения эксперимента был разработан тестовый сервер на языке Go. Данный язык был выбран благодаря своей высокой производительности, встроенной поддержке параллелизма (через механизм goroutines) и низким требованиям к системным ресурсам. Это делает его особенно подходящим для моделирования сценариев, типичных для DDoS-атак.

Разработанный сервер имитирует поведение и нагрузку, характерные для реальных веб-приложений, и обеспечивает следующие ключевые функции:

• Обработка различных типов HTTP-запросов, включая легкие и ресурсоёмкие (например, запросы к условной «базе данных» и загрузку страницы с внешними ресурсами).

• Логирование входящих запросов, включая регистрацию уникальных IPадресов, путей и признаков потенциальных атак.

• Сбор подробной статистики: общее количество запросов, количество ошибок, среднее время отклика, текущая производительность (RPS), загрузка процессора.

• Предоставление API-интерфейса для доступа к накопленным статистическим данным в формате JSON.

• Обработка UDP-пакетов, что позволяет дополнительно моделировать атаки на сетевом

уровне.

5.2. Подключение технологий защиты и проведение DDoS-атак

Тип атаки	Кол-во запросов / пакетов	Успешных ответов (%)	Ошибки 5xx (%)	Среднее время ответа (мс)	Загрузка CPU (%)
http	100 000	61,5	5.6	180	72
header	100 000	58,1	41,9	195	76
slowloris	45 000 TCPсоединений	21,4	78,6	>6000	89
udp	1 000 000 пакетов	—	—	—	82

Таблица 1. Эффективность DDoS атак на сервер без защиты

Атаки, ориентированные на высокочастотные HTTP-запросы и заголовочные переполнения, вызывают значительное увеличение количества ошибок уровня 5xx (до 37–40%). Среднее время ответа превышает 180 мс, что свидетельствует о деградации качества обслуживания. Нагрузка на CPU достигает 74–79%.

После ослабления параметров HTTP-сервера (удлинённый

ReadHeaderTimeout, увеличенные MaxHeaderBytes), эффективность Slowlorisатаки значительно возросла. Удержание большого количества TCP-соединений привело к резкому снижению доступности — доля ошибок достигла 81%, а среднее время ответа превысило 6 секунд. Загрузка CPU приблизилась к пиковым значениям (91%), а использование памяти выросло более чем в 2 раза по сравнению с HTTP Flood.

Rate Limiting

Тип атаки	Кол-во запросов / пакетов	Заблокировано (%)	Успешных ответов (%)	Ошибки 5xx (%)	Среднее время ответа (мс)	Загрузка CPU (%)
http	100 000	85,3	14,2	0,5	35	26
header	100 000	83,6	15,4	1,0	42	28
slowloris	45000 TCPсоединений	0,0	22,1	77,5	>5000	79
udp	1 000 000 пакетов	0,0	—	—	—	82

Таблица 2. Эффективность DDoS атак на сервер при подключенной защите Nginx Rate Limiting

Результаты повторных экспериментов с использованием механизма ограничения частоты запросов, реализованного на уровне веб-сервера Nginx, демонстрируют существенное повышение устойчивости системы к ряду распространённых DDoS-атак. В частности, при

проведении атак типа HTTP Flood и Header Flood наблюдалась высокая степень фильтрации трафика — 85,3% и 83,6% соответственно. Это позволило существенно разгрузить сервер: среднее время отклика сократилось до 35–42 мс, а доля ответов с ошибками уровня 5xx не превышала 1%. Нагрузка на центральный процессор снизилась до 26–28%.

В то же время защита оказалась малоэффективной при атаке типа Slowloris. Данная атака эксплуатирует особенности обработки TCP-соединений, удерживая их в открытом состоянии максимально долго, при этом не нарушая лимиты частоты запросов. В результате большая часть соединений не была заблокирована, что привело к значительному росту количества внутренних ошибок сервера (77,5%), резкому увеличению среднего времени отклика (свыше 5000 мс) и высокой загрузке системных ресурсов. Загрузка CPU достигла 79%. Эти результаты подтверждают ограниченность применения rate limiting в контексте атак, не характеризующихся высокой частотой запросов, но ориентированных на истощение ресурсов соединений.

Атака типа UDP Flood, как и ранее, оказалась полностью неэффективной в рамках данной защитной конфигурации. Поскольку трафик UDP минует HTTPслой, на котором действует Nginx, фильтрация практически не сработала.

ModSecurity WAF (Web Application Firewall) для Nginx

Тип атаки	**Кол-во запросов / пакетов**	**Заблокировано (%)**	**Успешных ответов (%)**	**Ошибки 5xx (%)**	**Среднее время ответа (мс)**	**Загрузка CPU (%)**
http	100 000	91,2	8,4	0,4	50	34
header	100 000	93,7	5,6	0,7	58	38
slowloris	45000 TCPсоединений	2,1	19,3	78,6	>6000	86
udp	10 000 пакетов	0,0	—	—	—	84

Таблица 3. Эффективность DDoS-атак при включённой защите через ModSecurity WAF

В результате тестирования системы с активным модулем ModSecurity наблюдается выраженное повышение фильтрационной способности в отношении атак, нацеленных на уязвимости прикладного уровня. HTTP Flood был заблокирован в 91,2% случаев, а Header Flood — в 93,7%, что объясняется способностью WAF распознавать и блокировать аномальные заголовки и характерные шаблоны запроса. Однако цена такой фильтрации выражается в росте среднего времени отклика (до 50–58 мс) и увеличении нагрузки на системные ресурсы. Использование CPU при этом достигало 34–38%.

Несмотря на эффективность против HTTP-ориентированных атак, защита посредством WAF оказалась слабой в случае атаки Slowloris. Как результат, только 2% соединений были распознаны как вредоносные, тогда как более 78% завершились ошибками на сервере. Среднее время ответа вновь превысило 6 секунд, а использование ресурсов оказалось близким к пиковым значениям: CPU — 86%.

Таким образом, интеграция WAF-модуля ModSecurity в Nginx позволяет значительно повысить устойчивость к атакам, направленным на уязвимости HTTPинтерфейса, включая

сложные варианты с подменой заголовков и вредоносными шаблонами запроса. Однако эффективность данной защиты ограничивается рамками прикладного уровня.

Cloudflare (WAF + Anycast + CAPTCHA)

Тип атаки	Кол-во запросов / пакетов	Заблокировано (%)	Успешных ответов (%)	Ошибки 5xx (%)	Среднее время ответа (мс)	Загрузка CPU (%)
http	100 000	99,4	0,5	0,1	60	6
header	100 000	99,1	0,7	0,2	64	7
slowloris	45000 TCPсоединений	84,6	11,7	3,7	180	18
udp	1 000 000 пакетов	92,2	—	—	—	22

Таблица 4. Эффективность DDoS-атак при включённой защите через ModSecurity WAF

Подключение Cloudflare было выполнено через делегирование DNS-записи тестового домена на nameservers, предоставленные платформой. После активации режима "I'm Under Attack Mode" система начала применять комплексную фильтрацию, включающую JavaScript-челленджи, CAPTCHA, WAF и георасределённую маршрутизацию Anycast. Уже при проведении атаки HTTP Flood фиксировалась почти полная блокировка вредоносного трафика (99,4%), что сопровождалось минимальной нагрузкой на сервер: загрузка CPU составила не более 6%, а среднее потребление оперативной памяти — около 135 МБ. Аналогичные показатели наблюдались при атаке Header Flood, что свидетельствует о высокой чувствительности защитных механизмов к аномалиям на уровне HTTPзаголовков. Незначительное количество успешных запросов (менее 1%) и минимальный уровень HTTP-ошибок подтверждают корректную отработку фильтрации на стороне прокси-сервиса, не доходящей до исходного сервера.

При атаке типа Slowloris эффективность защиты также оказалась высокой: более 84% соединений были прерваны на ранней стадии за счёт таймаутов и анализа паттернов сетевого поведения. Хотя некоторая часть соединений всё же доходила до сервера, уровень системной нагрузки оставался умеренным — до 18% загрузки CPU и не более 190 МБ использования оперативной памяти. Среднее время ответа увеличилось до 180 мс, что объясняется дополнительной проверкой запросов через облачный WAF.

Интеграция Google Cloud Armor была реализована посредством подключения тестового сервера к облачному балансировщику нагрузки в Google Cloud Platform. Политики безопасности, заданные на уровне L7, включали правила фильтрации по IP-диапазонам, географическому признаку и характеру трафика. При проведении UDP Flood атаки фильтрация происходила ещё до попадания пакетов в периметр внутренней сети, благодаря чему до 92% трафика было отклонено на уровне edgeинфраструктуры.

Таким образом, облачные решения демонстрируют значительно более высокую эффективность по сравнению с локальными средствами защиты. Это обусловлено тем, что обработка и блокировка вредоносного трафика происходит на внешнем периметре, зачастую до установления TCP-соединения. При этом важным аспектом является правильная маршрутизация через защищённые CDN и прокси-сервисы: только при

корректной делегации DNS и активации соответствующих режимов фильтрации возможна реализация полной схемы защиты. Полученные результаты подтверждают, что для современных высокоинтенсивных DDoS-атак применение исключительно локальных механизмов недостаточно. Комбинированный подход с интеграцией облачных сервисов, таких как Cloudflare или Google Cloud Armor, является необходимым условием для устойчивости серверной инфраструктуры.

Научная новизна исследования заключается в разработке и применении экспериментальной модели имитации DDoS-атак с использованием специализированного Go-сервера, что позволило в контролируемых условиях провести сравнительный анализ локальных и облачных защитных механизмов с учётом метрик производительности и эффективности.

Заключение

DDoS-атаки продолжают эволюционировать, становясь всё более масштабными и сложными, что требует от разработчиков применения комплексных и адаптивных подходов к обеспечению устойчивости цифровой инфраструктуры. Традиционные методы фильтрации и базовые брандмауэры уже не обеспечивают необходимого уровня защиты. Эффективное противодействие этим угрозам требует интеграции облачных сервисов, локальных механизмов фильтрации, постоянного мониторинга сетевого трафика и применения алгоритмов машинного обучения.

Для формирования эффективной стратегии защиты от DDoS-атак необходимо проведение всестороннего анализа уровня угроз, выбор соответствующих инструментов мониторинга, а также внедрение механизмов автоматического реагирования. Использование облачных решений, таких как Cloudflare, AWS Shield и аналогичных, позволяет фильтровать вредоносный трафик на ранних этапах его поступления. Дополнение архитектуры локальными механизмами защиты, включая Web Application Firewall (WAF) и алгоритмы ограничения частоты запросов (Rate Limiting), создаёт многослойную систему обороны.

Регулярный аудит журналов событий, проведение нагрузочного тестирования и постоянное совершенствование применяемых технологий способствуют повышению общей устойчивости инфраструктуры. Помимо технологических инвестиций, организации должны выстраивать гибкие стратегии информационной безопасности, ориентированные на быстрое выявление и адаптацию к изменяющимся угрозам. Непрерывное развитие и тестирование защитных механизмов становится ключевым элементом обеспечения стабильной и безопасной работы цифровых систем в условиях растущей интенсивности DDoSатак.

Практическая значимость работы заключается в том, что предложенная методика и полученные результаты могут быть использованы специалистами по кибербезопасности для построения устойчивых веб-инфраструктур, выбора оптимальных защитных решений и их настройки в условиях реальных DDoS-угроз.

В дальнейшем предполагается расширить экспериментальную платформу за счёт интеграции интеллектуальных систем обнаружения атак на основе машинного обучения, а также протестировать эффективность гибридных архитектур с участием различных CDN и IDS-систем.

Библиография

1. Абрамов А. Г. Защита от DDoS-атак своими руками: оперативные разработка и внедрение сервиса в Национальной исследовательской компьютерной сети России //

Программные продукты и системы. 2022. № 4. DOI: 10.15827/0236-235X.140.572-582 EDN: OGJSLQ.
2. Аманжолов О. М. Исследование методов и средств обнаружения DDoS-атак // Молодой ученый. 2023. № 50 (497). С. 5-8. URL: https://moluch.ru/archive/497/109243/ (дата обращения: 18.03.2025). EDN: XIACUA.
3. Орехов А. В., Орехов А. А. Автоматическое обнаружение аномалий сетевого трафика при DDoS-атаках // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2023. Т. 19. Вып. 2. С. 251-263. DOI: 10.21638/11701/spbu10.2023.210 EDN: XYNCXN.
4. Унру П. П., Обухов С. А., Черемухин Д. Н. Меры по защите инфокоммуникационных систем от DDoS-атак с усилением // Молодежь и наука: актуальные проблемы фундаментальных и прикладных исследований. 2022. Ч. 1. С. 1-497. DOI: 10.17084/978-5-7765-1511-8-2022 EDN: OCZGSD.
5. Kangkan Talukdar, Debojit Boro. Slowloris Attack Detection Using Adaptive Timeout-Based Approach // The ISC Int'l Journal of Information Security. 2024. № 1. С. 79-92. URL: https://www.isecure-journal.com/article_183600_e06eaaffd81aef753b956e80b513f82b.pdf (дата обращения: 10.03.2025).
6. Верещагин К. В. Защита корпоративных сетей от DDoS-атак: современные методы и тенденции // Научный лидер. 2023. № 47 (145). С. 12-15. URL: https://scilead.ru/media/journal_pdf145.pdf#page=12 (дата обращения: 10.03.2025). EDN: CSBAGF.
7. Разработка REST-серверов на Go. Часть 1: стандартная библиотека [Электронный ресурс]. URL: https://habr.com/ru/companies/ruvds/articles/559816/.
8. Xu A. Rate Limiting Fundamentals // ByteByteGo Newsletter. 2023 [Электронный ресурс]. URL: https://blog.bytebytego.com/p/rate-limiting-fundamentals (дата обращения: 18.03.2025).
9. What is Fail2Ban with Setup & Configuration? (Detailed Guide) [Электронный ресурс]. URL: https://runcloud.io/blog/what-is-fail2ban (дата обращения: 18.03.2025).
10. What Is a WAF? | Web Application Firewall Explained [Электронный ресурс]. URL: https://www.paloaltonetworks.com/cyberpedia/what-is-a-web-application-firewall (дата обращения: 18.03.2025).
11. Tawde S. What is Cloudflare? // EducbaBlog. 2023 [Электронный ресурс]. URL: https://www.educba.com/what-is-cloudflare/ (дата обращения: 18.03.2025).
12. Dheer P. Understanding Google Cloud Armor: Protect against denial of service and web attacks // TestPrepTraining. 2020 [Электронный ресурс]. URL: https://www.testpreptraining.com/blog/understanding-google-cloud-armor-protect-against-denial-of-service-and-web-attacks/.

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.
Со списком рецензентов издательства можно ознакомиться здесь.

Представленная статья на тему «Современные методы предотвращения DDoS-атак и защиты веб-серверов» соответствует тематике журнала «Программные системы и вычислительные методы» и посвящена актуальному вопросу изучения современных подходов к защите веб-ресурсов, включая облачные решения (Cloudflare, Google Cloud Armor), локальные механизмы защиты (Rate Limiting, Fail2Ban) и перспективные технологии, использующие машинное обучение и поведенческий анализ трафика.
В статье представлен анализ литературных российских и зарубежных и интернет-

источников по теме исследования. Указана теоретико-методологическая основа исследования. Авторами проведено исследование, направленное на анализ методов, их сравнительному анализу и перспективам дальнейшего развития стратегий защиты от DDoS-атак.

Стиль и язык изложения материала является достаточно доступным для широкого круга читателей. Статья по объему соответствует рекомендуемому объему от 12 000 знаков. Статья достаточно структурирована - в наличии введение, заключение, внутреннее членение основной части (1. Классификация DDoS-атак и их последствия, 2. Локальные методы защиты от DDoS-атак и способы их выявления, 3. Облачные сервисы защиты от DDoS-атак, 4. Методология исследования, 5. Реализация эксперимента).

Авторами проведен анализ методов защиты от DDoS-атак, в том числе: облачных решений, которые демонстрируют значительно более высокую эффективность по сравнению с локальными средствами защиты. Полученные результаты подтверждают, что для современных высокоинтенсивных DDoS-атак применение исключительно локальных механизмов недостаточно. Комбинированный подход с интеграцией облачных сервисов, таких как Cloudflare или Google Cloud Armor, является необходимым условием для устойчивости серверной инфраструктуры.

Авторами проведено исследование, в ходе которого они пришли к выводам, что для формирования эффективной стратегии защиты от DDoS-атак необходимо проведение всестороннего анализа уровня угроз, выбор соответствующих инструментов мониторинга, а также внедрение механизмов автоматического реагирования. Использование облачных решений, таких как Cloudflare, AWS Shield и аналогичных, позволяет фильтровать вредоносный трафик на ранних этапах его поступления. Дополнение архитектуры локальными механизмами защиты, включая Web Application Firewall (WAF) и алгоритмы ограничения частоты запросов (Rate Limiting), создаёт многослойную систему обороны.

К недостаткам можно отнести следующие моменты: из содержания статьи не прослеживается научная новизна, не обоснована практическая значимость исследования. Отсутствует четкое выделение предмета, объекта исследования.

Рекомендуется четко обозначить научную новизну исследования, сформулировать предмет, объект, обосновать практическую значимость исследования. Также будет целесообразным добавить о перспективах дальнейшего исследования.

Статья «Современные методы предотвращения DDoS-атак и защиты веб-серверов» требует доработки по указанным выше замечаниям. После внесения поправок рекомендуется к повторному рассмотрению редакцией рецензируемого научного журнала.

Результаты процедуры повторного рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться здесь.

В рецензируемой статье поднимает важный в условиях повсеместного использования цифровых технологий вопрос защиты удаленных серверов от несанкционированного воздействия. Возможные экономические и репутационные последствия, необходимость анализа новых способов выполнения атак и сопоставление эффективности подходов защиты определяют актуальность выполненного исследования. Авторы формулируют предмет исследования, задачи, приводят классификацию DDoS атак с анализом основных механизмов воздействия и потенциальных способов защиты. Вместе с тем авторы отмечают, что существует проблема ложного срабатывания, высокие требования

к аппаратному обеспечению, а также необходимость постоянного мониторинга новых методов атак, а значит и внесение коррективов в методы защиты, среди которых отдельное внимание уделяется облачным технологиям.

Исследование авторов включало моделирование атак и анализ трафика для их обнаружения и сравнение эффективности различных механизмов защиты. Авторами был разработан тестовый сервер, имитирующий нагрузку веб-приложений с учетом входящих запросов, регистрацию IP-адресов, подключение внешних ресурсов. В качестве критериев оценки результатов выбрано среднее время ответа, загрузка CPU, доля успешных ответов. Приводятся анализ полученных результатов в первоначальных условиях и с дополнительным фактором (ограничение частоты запроса), оценкой влияния на нагрузку сервера.

Результаты работы показывают высокую результативность исследуемых механизмов на обнаружение атак, однако это сопровождается значимым увеличением времени отклика и нагрузки на CPU. Устойчивость к атакам также ограничивается прикладным уровнем.

Структура статьи и стиль изложения отвечают требованиям. имеется экспериментальная часть, приведены количественные оценки.

Библиография содержит 12 источников, ссылки по тексту имеются.

Замечания.

Статья перегружена сокращениями и англоязычными терминами, что несмотря на принятую в данной предметной области терминологию затрудняет общее восприятие.

Высокие показатели при некоторых анализируемых атаках вызывают сомнение в целесообразности их рассмотрения, вероятно данный факт не нуждается в подтверждении или был прогнозируем на стадии планирования эксперимента.

Для использованных оценок было бы желательно привести некоторое пороговое значение, начиная с которого эффект становится критичным. Если ощутима загрузка CPU, например, 25%, то для результатов в табл. 4 все рассчитанные значения удовлетворяют подобным условиям. Если пороговое значение ниже, то возможно отдельный тип атак становится критичным.

Не вполне ясно каким образом рассчитаны количественные оценки. Если это некоторые средние значения, то каким образом получены составляющие их слагаемые.

Желательно увеличить список источников при составлении обзора, отдав предпочтение публикациям, содержащим исследование. Рекомендуется расположить источники или в алфавитном порядке, или в порядке упоминания в тексте.

Имеются отдельные ошибки (орфографические, стилистические).

Статья может быть полезна специалистам, чья профессиональная деятельность связана с сетевыми технологиями, и опубликована после внесения технических правок.