МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ, КОМПЛЕКСОВ И КОМПЬЮТЕРНЫХ СЕТЕЙ / MATHEMATICAL SOFTWARE FOR COMPUTERS, COMPLEXES AND COMPUTER NETWORKS

DOI: https://doi.org/10.23670/IRJ.2024.141.31

МОНИТОРИНГ ДОСТУПНОСТИ ВЕБ-СЕРВИСА В РАСПРЕДЕЛЕННЫХ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМАХ

Обзор

Малыгин Д.С.[1, *]
[1] ООО «Т1 Диджитал», Санкт-Петербург, Российская Федерация

* Корреспондирующий автор (dmitrima10[at]gmail.com)

Аннотация

Масштабы и сложность DDoS-атак в Интернете растут в геометрической прогрессии. Как свидетельствуют последние события, тактика атак начинается с массированного сетевого взлома, а затем переходит к более сложным многовекторным атакам на уровне приложений, которые трудно обнаружить и нейтрализовать. С учетом вышеизложенного, статья посвящена рассмотрению вопросов, связанных с доступностью веб-сервиса в распределенных инфокоммуникационных системах. Новизной проведенного исследования является авторский механизм, позволяющий защитить веб-серверы от ресурсоемких DdoS-атак, который базируется на двух методологиях: управление доступом на основе белых списков и обнаружение потоков атак на основе анализа служебной информации. Практическая значимость авторских предложений заключается в том, что механизм обнаружения потоков атак позволит более эффективно выявлять вложенные потоки на основе симптомов или нагрузки на сервер, поскольку становится все труднее идентифицировать вредоносные потоки только на основе моделей входящего трафика.

Ключевые слова: веб-сервис, атака, защита, обнаружение, приложение.

WEB SERVICE AVAILABILITY MONITORING IN DISTRIBUTED INFO-COMMUNICATION SYSTEMS

Review article

Maligin D.S.[1, *]
[1] T1 Digital LLC, Saint-Petersburg, Russian Federation

* Corresponding author (dmitrima10[at]gmail.com)

Abstract

The scale and sophistication of DDoS attacks on the Internet is growing exponentially. As evidenced by recent events, the attack tactics start with massive network hacking and then move to more complex multi-vector application-level attacks that are difficult to detect and neutralize. In view of the above, the article is dedicated to the consideration of issues related to web service availability in distributed infocommunication systems. The novelty of the research is the author's mechanism that allows to protect web servers from resource-intensive DdoS attacks, which is based on two methodologies: access control based on whitelists and detection of attack flows based on the analysis of service information. The practical significance of the authors' proposals is that the attack flow detection mechanism will enable more efficient detection of nested flows based on symptoms or server load, as it is becoming increasingly difficult to identify malicious flows based on incoming traffic patterns alone.

Keywords: web service, attack, defence, detection, application.

Введение

Web-сервисы – это новая разновидность Web-приложений, призванная создать следующее поколение распределенных прикладных программ. Другими словами – это программируемая логика приложения, доступная по стандартным протоколам Интернета, представляющая собой набор функций, которые упаковываются как единое целое и публикуются в сети для использования другими программами [1]. Доступ к Web-сервисам может осуществляться на любом языке, с применением любой компонентной модели, под управлением любой операционной системы. В качестве базового транспорта выступает протокол передачи гипертекста (HTTP), что позволяет запросам функций проходить через межсетевые экраны. Для форматирования входных и выходных параметров запроса используется расширяемый язык разметки (XML), поэтому запрос не привязан к какой-либо конкретной компонентной технологии или соглашению о вызове объекта [2]. Web-сервисы являются строительными блоками для создания открытых распределенных систем и дают возможность компаниям и частным лицам быстро и дешево сделать свои цифровые активы доступными по всему миру. Web-сервисы выполняют функции, которые могут быть самыми разными – от простых запросов до сложных бизнес-процессов. В то же время, учитывая важность Web-сервисов для современного мира особую актуальность приобретает вопрос их безопасности.

Существует множество сложностей, свойственных Web-сервисам, которые усложняют задачу их защиты от действий злоумышленников. Многочисленные угрозы могут нарушить конфиденциальность, целостность или доступность Web-сервиса, а также внутренних систем, к которым он может подключаться [3]. Некоторые из этих угроз характерны для обычных систем Web-приложений, другие – для Web-сервисов. Поэтому очень важно четко идентифицировать эти угрозы, знать, как их предупредить или каким образом на них реагировать. Для этого

необходимы определенные системы мониторинга состояния каждого элемента, способные анализировать возможные проблемы, которые могут возникать во время их работы, и запускать средства защиты в самом сервисе или уведомлять людей о возможной угрозе.

Такие системы должны в очень короткое время отслеживать все изменения, включая как различные инъекции, так и атаки, при которых сервис перестает нормально функционировать. Одним из возможных вариантов развития анализатора может быть самодиагностика сервиса, когда анализатор сам проверяет подключенные системы на дыры в системе защиты, что позволяет сообщать владельцам о проблеме. Это, в свою очередь, дает возможность оперативно решить ее, пока она не нанесла не только финансовый ущерб компании, но и не привела к возможным потерям конфиденциальных данных пользователей.

Таким образом, обозначенные обстоятельства обуславливают необходимость дальнейшего исследования проблематики, связанной с усовершенствованием техники защиты Web-сервисов, что и предопределило выбор темы данной статьи.

Анализ публикаций по теме исследования. Над разработкой комплексного адаптивного подхода к обеспечению безопасности Web-сервисов, который способен гарантировать защиту от векторов угроз по мере развития и эволюции приложений трудятся такие авторы как: Раимов М. Е., Мукашева А. К., Исаева Г. Б., Исхаков А. Ю., Богачева Д. Н., Молотов А. А., Stollberg B., Zipf A., Ware J. M.

Усовершенствованию инструментов, позволяющих выявлять уязвимости, связанные с веб-службами, посвятили свои работы Беликов Г. В., Крылов И. Д., Селищев В. А., Жуков С. В., Леджиев Д. Ю., Desmet L., Jacobs B., Piessens F., Joosen W.

Особенности реализации защиты от Ddos атак на основе концепции периода занятости, определяемого для каждой пары IP-адресов клиента и сервера, описаны в публикациях Халилаевой Э. И., Масловой М. А., Герасимова В. М., Chumash T., Yao D.

Ferrari E., Bertino E., Karabulut Y., Миркович и др. классифицировали DoS-атаки на две категории. К первой авторы отнесли атаки типа flooding, которые направлены на переполнение ресурсов серверов путем отправки им достаточно большого объема трафика. Второй тип – атаки на уязвимости, использующие уязвимость ресурсов.

В последнее время было зарегистрировано несколько типов низкоскоростных DoS-атак. Одним из примеров является атака Shrew на TCP. Злоумышленник посылает всплески пакетов для создания потерь пакетов в канале связи и увеличивает таймаут повторной передачи для определенных TCP-потоков. Всплески посылаются только в моменты истечения времени действия этих потоков, чтобы снизить общую пропускную способность. Другой пример – низкоскоростные DoS-атаки на серверы приложений.

Что касается защиты от этих низкоскоростных DoS-атак, то в работе Chadwick D., Preneel B. сообщается, что модель трафика ON/OFF атаки Shrew может быть обнаружена с помощью автокорреляции сигнала скорости трафика и динамической временной деформации.

Нерешенные части общей проблемы. Однако, несмотря на имеющиеся труды и наработки, возрастающая с каждым годом глобальная киберпреступность выдвигает на первый план задачи усовершенствования методов, которые позволят обеспечить автоматизированное, точное обнаружение и подавление атак на Web-сервисы с минимальным количеством ложных срабатываний.

Итак, *цель статьи* заключается в исследовании методов мониторинга доступности веб приложений в распределенных инфокоммуникационных системах на основе анализа угроз, которые могут влиять на систему, и нахождения способов решения проблем защиты от Ddos атак.

Задачи исследования:

1) рассмотреть способы обнаружения DdoS-атак;
2) разработать авторский механизм защиты от распределенного отказа в обслуживании;
3) предложить подход составления «белого» списка;
4) формализовать последовательность переноса логов в базу данных.

Основные результаты

При DdoS-атаке множество устройств атакуют один сервер или сеть. Цель атаки – перегрузить целевой сервер или сеть многочисленными поддельными запросами, мешающими регулярному трафику. Это приводит к перегрузке сетевых ресурсов и, как следствие, к нарушению обслуживания легитимного трафика. Такие атаки осуществляются с помощью сетей подключенных к Интернету устройств, в том числе ПК и других устройств (например, IoT-устройств), зараженных вредоносным программным обеспечением и, таким образом, способных к удаленному манипулированию [4].

Существует два основных способа обнаружения DdoS-атак: поточная проверка всех пакетов и внеполосное обнаружение с помощью анализа записей потоков трафика. Любой из этих подходов может быть развернут как в локальной сети, так и с помощью облачных сервисов. Базовые возможности сетевых устройств, таких как балансировщики нагрузки, межсетевые экраны или системы предотвращения вторжений по обнаружению DdoS-атак в режиме реального времени, возможно, и обеспечивали приемлемый уровень их идентификации, когда DdoS-атаки были менее масштабными, но при больших объемах атак эти устройства могут быть перегружены, поскольку в них используются методы проверки состояния, занимающие много памяти.

На сегодняшний день широкое распространение получила платформа HADEC, которая представляет собой структуру для обнаружения высокоскоростной DdoS-атаки, происходящей на сетевых и прикладных уровнях, таких как TCP-SYN, HTTP GET, UDP и ICMP. Платформа состоит из двух основных компонентов: сервера обнаружения и сервера захвата. Обнаружение DdoS в реальном времени начинается с сервера захвата, который отвечает за перехват живого сетевого трафика и передачу журнала на сервер для обработки. Далее вычисляется входящий пакет для UDP, ICMP и HTTP, чтоб выявить атаку, если количество соединений источника превышает заданный порог [5]. Несмотря

на высокую результативность метода, его недостатком является высокая затратность как сетевых, так и финансовых ресурсов.

Для обнаружения четырех типов трафика: законного пользователя, низкоскоростного, высокоскоростного и быстрого трафика используется система D-FACE. Процесс выявления атаки базируется на расчете разницы энтропии, которая содержит нормальный поток трафика, тогда как значение энтропии IP-источника является матрицей обнаружения для учета атаки. Выявление незаконных действий начинается с извлечения связанного заголовка, который классифицирует сеть в уникальный сетевой поток. Разделение трафика событий низкой скорости и мгновенного события основано на сравнении текущей скорости входящего трафика в каждом временном окне и анализе информационного трафика.

Также хорошие результаты дает MLP-GA алгоритм. Этот алгоритм позволяет обнаружить DdoS атаку используя четыре параметра для генерации обнаружения на прикладных уровнях. Метод обнаружения подсчитывает количество HTTP GET-запросов, полученных веб-сервером, и вычисляет количество IP-адресов, направленных на сервер в течение 20 секунд. Предлагаемое обнаружение также проверяет номер порта, используемого HTTP DdoS.

Обсуждение. На основе анализа типов угроз, которые чаще всего влияют на работу Web-сервисов предлагаем новый механизм защиты от распределенного отказа в обслуживании, базирующийся на двух методологиях: управление доступом на основе белых списков и обнаружение потоков атак на основе анализа служебной информации.

Схема предлагаемого двухступенчатого механизма защиты от DDoS атак представлена на рисунке 1. Согласно схеме, при поступлении пакета, если он предназначен для узла-жертвы, подвергшегося DDoS-атаке, к нему применяется политика управления допуском на основе «белого списка». Политика заключается в том, чтобы просто принять пакет, если IP-адрес источника уже зарегистрирован в белом списке, и отбросить его, если IP-адрес источника не зарегистрирован в белом списке [5]. Таким образом, данная идея представляет собой первый этап двухступенчатого механизма защиты Web-сервисов. После этого, на втором этапе, к пакету применяется алгоритм анализа служебной информации.

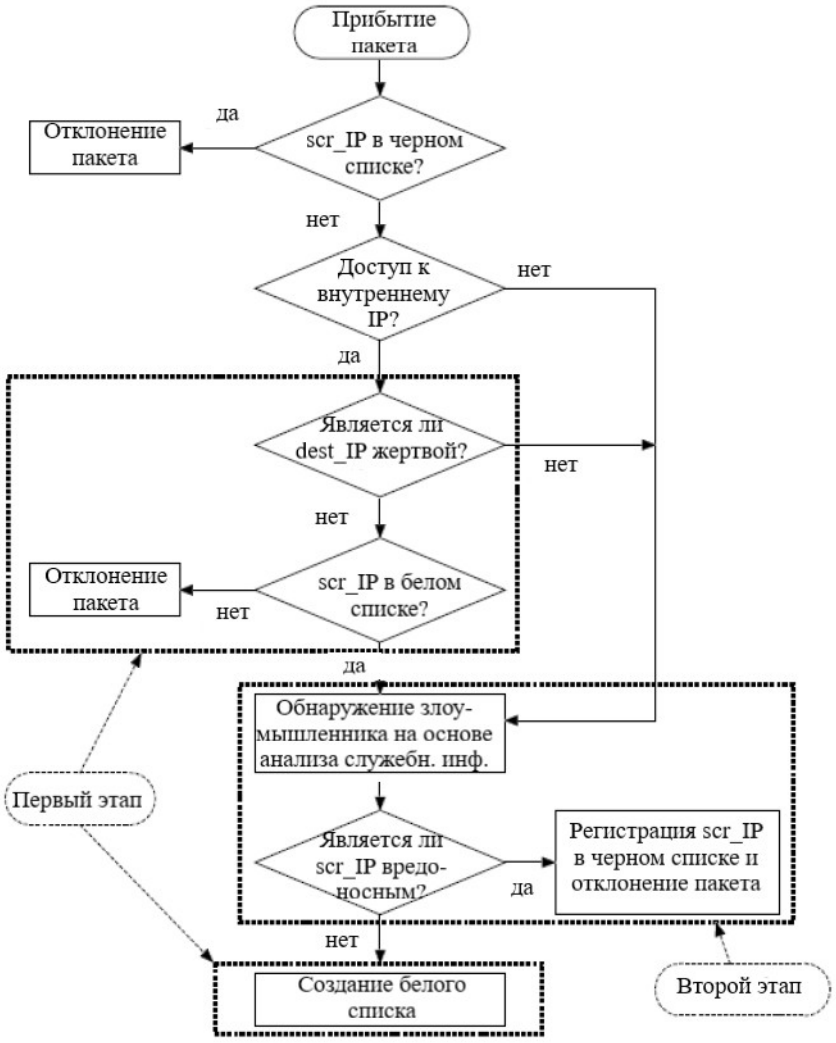

Рисунок 1 - Схема двухступенчатого механизма защиты Web-сервиса от DDoS атаки

DOI: https://doi.org/10.23670/IRJ.2024.141.31.1

Когда система защиты отслеживает пакет, направляющийся на защищаемые серверы или исходящий от них, она сначала проверяет, не идет ли он с IP-адресов, внесенных в черный список. В этом случае пакет оперативно отбрасывается. Если не отброшенный пакет предназначен для «жертвы» внутри защищенной подсети, то проверяется, не пришел ли этот пакет с IP-адресов из «белого списка». Все не отброшенные пакеты проверяются с помощью алгоритма анализа служебной информации. Если IP-адрес источника расценивается как злонамеренный, то он заносится в черный список, и пакет отбрасывается. Если пакет проходит этот этап, то отслеживается нагрузка на каждый внутренний IP-адрес [6]. Если количество узлов, обращающихся к определенному серверу, превышает некоторый заранее заданный порог, то для этого сервера формируется белый список. Если нагрузка на сервер превышает другой порог, то сервер объявляется «жертвой», и доступ к нему разрешается только тем IP-адресам, которые находятся в белом списке.

Следует отметить, что схема управления допуском на основе белых списков активизируется только при обнаружении DdoS-атаки, основанной на нагрузке на сервер. Таким образом, в обычной ситуации фильтрация пакетов по белому списку не используется. С другой стороны, схема обнаружения потока атак на основе анализа служебной информации работает всегда и защищает сервер с черным списком.

Как видно из рисунка 1, первый этап, т.е. этап управления допуском на основе белого списка, состоит из двух фаз.

Первая – фильтрация пакетов на основе белого списка, особенно для «серверов-жертв».

Второй – этап построения «белого списка» для узлов потенциальных жертв.

В рамках проводимого исследования более подробно рассмотрим вторую фазу первого этапа.

Белый список обычно довольно короткий, и для его построения не требуется аппаратная помощь, а фильтрация должна осуществляться на IP-уровне. Возможна фильтрация и на прикладном уровне. Ускорить фильтрацию могут межсетевые экраны с изменяемым состоянием. Существуют особые подходы к построению белого списка, такие как Gold Image и Digital Certificates [7]. При использовании, например, Gold Image для статических систем список создается путем предварительного хэширования стандартного образа рабочей станции [8]. Однако после использования образа для создания первого белого списка поддержание его в актуальном состоянии является самой сложной задачей.

На рисунке 2 представлен алгоритм, согласно которому можно построить белый список. Этап построения белого списка состоит из двух подэтапов. На первом подэтапе происходит выявление потенциальных «жертв». Если количество внешних машин, обращающихся к внутреннему IP-адресу, больше или равно порогу *N*, то этот внутренний IP-адрес рассматривается как потенциальная «жертва».

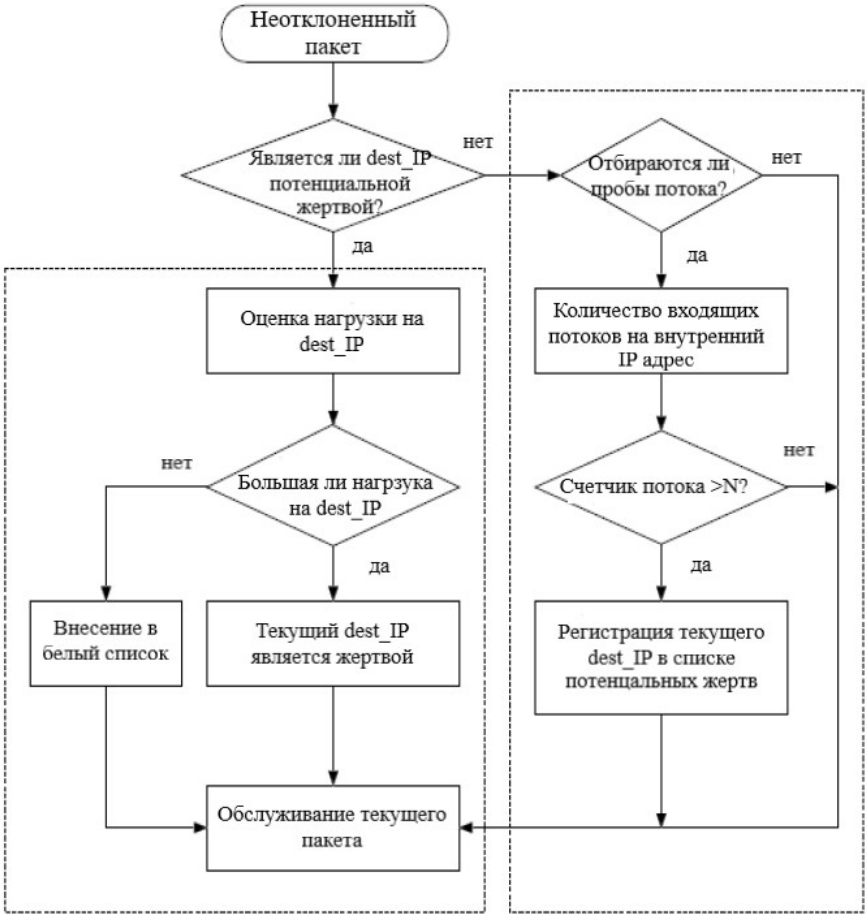

Рисунок 2 - Блок-схема, согласно которой создается белый список

DOI: https://doi.org/10.23670/IRJ.2024.141.31.2

Для уменьшения объема памяти и накладных расходов на обработку пакетов производится выборка пакетов, и на этом этапе проверяются только выбранные пакеты. IP-адреса потенциальных жертв хранятся в таблице состояния соединений.

Поскольку большинство систем защиты функционируют на уровне ядра сети и часть входящих запросов перенаправляется в Black Hole, единственным возможным вариантом повышения эффективности защиты системы является логирование всех запросов на этапе прохождения этапа фаервола входящими пакетами. Тогда, распарсив данные лога, можно узнать всю нужную информацию [9]. Поскольку постоянное считывание из файла довольно затратно в плане ресурсов, представляется целесообразным переносить логи в базу данных. Это позволит ускорить поиск и фильтрацию «нужных» логов. Наиболее идеальным вариантом было бы использование так называемой in-memory базы данных. Такие базы данных держат данные в операционной памяти, что позволяет быстро выполнять операции записи и поиска [10].

На рисунке 3 приведены упрощенные шаги реализации записи/считывания лога в базу данных.

Рисунок 3 - Упрощенные шаги переноса логов в базу данных
DOI: https://doi.org/10.23670/IRJ.2024.141.31.3

Вторым элементом двухступенчатого механизма защиты Web-сервиса от DDoS атаки является алгоритм, который позволяет выделить пользователя среди других запросов, атакующих сервис. Для этого в базу данных отправляется запрос, который находит все обращения к веб-серверу в промежутке времени от 24 часов до запроса и до самого момента этого же запроса. После чего для ускорения поиска сортируется список всех запросов по IP-адресу. Далее в отдельном модуле ищется максимальное количество запросов для каждого отдельного IP-адреса и полученное число возвращается в главный метод, после чего среди всех запросов, совершенных в течение 24 часов, находится IP-адрес, с которого и было отправлено наибольшее количество запросов. Для того чтобы проверить, не отличается ли максимальное количество запросов от количества запросов других пользователей, нужно найти медиану количества запросов среди всех других пользователей. В дальнейшем проводится сравнение, будет ли максимальное количество запросов больше, чем медиана, умноженная на установленный в ходе экспертной оценки коэффициент.

Заключение

Таким образом, подводя итоги проведенного исследования, можно сделать следующие выводы.

В статье предложен двухступенчатый механизм, позволяющий защитить веб-серверы от ресурсоемких DdoS-атак. В основе предлагаемого механизма лежат две ключевые идеи. Первая – схема управления допуском, основанная на белых списках, которая защищает серверы от внезапного всплеска потоков атак. Вторая – анализ служебной информации, базирующийся на оценке длительности сессии, согласно лог файлов. Этот механизм позволит повысить эффективность системы мониторинга безопасности веб-приложений в распределенных инфокоммуникационных системах.

Прогнозируется, что предложенная система защиты будет способна эффективно противостоять DDoS-атакам даже при большом количестве потоков запросов. Возможность ее развертывания на linux-машине не будет вызывать сложностей, это позволит обрабатывать пакеты, поступающих со скоростью, близкой к скорости канала связи. В отличие от существующих подходов, авторский алгоритм на основе белого списка защищает сервер на начальном этапе DDoS-атаки, а механизм обнаружения потоков атак на основе периода занятости отличает потоки атак от обычных потоков и эффективно фильтрует IP-адреса злоумышленников из белого списка.

Конфликт интересов

Не указан.

Рецензия

Все статьи проходят рецензирование. Но рецензент или автор статьи предпочли не публиковать рецензию к этой статье в открытом доступе. Рецензия может быть предоставлена компетентным органам по запросу.

Conflict of Interest

None declared.

Review

All articles are peer-reviewed. But the reviewer or the author of the article chose not to publish a review of this article in the public domain. The review can be provided to the competent authorities upon request.

Список литературы / References

1. Аллакин В. В. Общий подход к построению перспективных систем мониторинга распределенных информационно-телекоммуникационных сетей / В. В. Аллакин, Н. П. Будко, В. Н. Васильев // Системы управления, связи и безопасности. — 2021. — №4. — С. 25–227.

2. Бабичева М. В. Тестирование web-приложений на устойчивость к низкоинтенсивным атакам отказа в обслуживании / М. В. Бабичева, Е. О. Цвелев // Вестник Донецкого национального университета. Серия Г: Технические науки. — 2020. — № 2. — С. 16–24.

3. Будко Н. П. Общие принципы функционирования и требования к построению структур перспективных систем мониторинга распределенных информационно-телекоммуникационных сетей / Н. П. Будко // Техника средств связи. — 2021. — №2(154). — С. 38–58.

4. Иванов И. И. Модель функционирования распределенных информационных систем при использовании маскированных каналов связи / И. И. Иванов // Системы управления, связи и безопасности. — 2020. — №1. — С. 198–234.

5. Козьминых С. Разработка системы защиты веб-приложений от компьютерных атак на производственных объектах / С. Козьминых, Р. Кулиев // Информационные ресурсы России. — 2022. — № 3(187). — С. 16–25.

6. Смирнов С. И. Техника атаки на веб-приложения на основе внедрения заголовков электронной почты / С. И. Смирнов, Г. М. Кумуржи // Цифровая наука. — 2022. — № 1. — С. 25–33.

7. Idhom M. Implementation of Web Server Security Against Denial of Service (DoS) Attacks / M. Idhom // IOP Conference Series. Materials Science and Engineering. — 2021. — Vol. 1125. — Is. 1. — P. 123–129.

8. Iswandi W. S. Analysis of Denial of Service Attack on Web Security Systems / W. S. Iswandi, M. Zarlis // Journal of Physics. Conference Series. — 2021. — Vol. 1811. — Is. 1. — P. 89–94.

9. Eassa A. M. NoSQL Injection Attack Detection in Web Applications Using RESTful Service / A. M. Eassa // Programming and Computer Software. — 2019. — Vol. 44. — P. 435–444.

10. Shandilya S. K. Cyber Attack Evaluation Dataset for Deep Packet Inspection and Analysis / S. K. Shandilya // Data in Brief. — 2023. — Vol. 46. — P. 199–203.

Список литературы на английском языке / References in English

1. Allakin V. V. Obshchij podhod k postroeniyu perspektivnyh sistem monitoringa raspredelennyh informacionno-telekommunikacionnyh setej [General Approach to the Construction of Advanced Monitoring Systems for Distributed Information and Telecommunication Networks] / V. V. Allakin, N. P. Budko, V. N. Vasil'ev // Sistemy upravleniya, svyazi i bezopasnosti [Control, Communication and Security Systems]. — 2021. — No. 4. — P. 25–227. [in Russian]

2. Babicheva M. V. Testirovanie web-prilozhenij na ustojchivost' k nizkointensivnym atakam otkaza v obsluzhivanii [Testing Web Applications for Resistance to Low-Intensity Denial of Dervice Attacks] / M. V. Babicheva, E. O. Cvelev // Vestnik Doneckogo nacional'nogo universiteta. Seriya G: Tekhnicheskie nauki [Bulletin of the Donetsk National University. Series G: Technical Sciences]. — 2020. — No. 2. — P. 16–24. [in Russian]

3. Budko N. P. Obshchie principy funkcionirovaniya i trebovaniya k postroeniyu struktur perspektivnyh sistem monitoringa raspredelennyh informacionno-telekommunikacionnyh setej [General Principles of Functioning and Requirements for the Construction of Structures of Promising Systems for Monitoring Distributed Information and Telecommunication Networks] / N. P. Budko // Tekhnika sredstv svyazi [Communication Technology]. — 2021. — No. 2(154). — P.38–58. [in Russian]

4. Ivanov I. I. Model' funkcionirovaniya raspredelennyh informacionnyh sistem pri ispol'zovanii maskirovannyh kanalov svyazi [Model of the Functioning of Distributed Information Systems When Using Masked Communication Channels] / I. I. Ivanov // Sistemy upravleniya, svyazi i bezopasnosti [Control, Communication and Security Systems]. — 2020. — No. 1. — P. 198–234. [in Russian]

5. Kozminykh S. Razrabotka sistemy zashchity veb-prilozhenij ot komp'yuternyh atak na proizvodstvennyh ob"ektah [Development of a System for Protecting Web Applications from Computer Attacks at Production Facilities] / S. Koz'minyh, R. Kuliev // Informacionnye resursy Rossii [Information Resources of Russia]. — 2022. — No. 3(187). — P. 16–25. [in Russian]

6. Smirnov S. I. Tekhnika ataki na veb-prilozheniya na osnove vnedreniya zagolovkov elektronnoj pochty [Techniques for Attacking Web Applications Based on Email Header Injection] / S. I. Smirnov, G. M. Kumurzhi // Cifrovaya nauka [Digital Science]. — 2022. — No. 1. — P. 25–33. [in Russian]

7. Idhom M. Implementation of Web Server Security Against Denial of Service (DoS) Attacks / M. Idhom // IOP Conference Series. Materials Science and Engineering. — 2021. — Vol. 1125. — Is. 1. — P. 123–129.

8. Iswandi W. S. Analysis of Denial of Service Attack on Web Security Systems / W. S. Iswandi, M. Zarlis // Journal of Physics. Conference Series. — 2021. — Vol. 1811. — Is. 1. — P. 89–94.

9. Eassa A. M. NoSQL Injection Attack Detection in Web Applications Using RESTful Service / A. M. Eassa // Programming and Computer Software. — 2019. — Vol. 44. — P. 435–444.

10. Shandilya S. K. Cyber Attack Evaluation Dataset for Deep Packet Inspection and Analysis / S. K. Shandilya // Data in Brief. — 2023. — Vol. 46. — P. 199–203.