УДК 004.056.53

Агентная модель DDoS-атаки на объекты критической информационной инфраструктуры: исследование устойчивости сервера от параметров реализации кибератаки

Agent-based model of DDoS attack on critical information infrastructure objects: study of server resilience from the parameters of cyberattack realization

Булгакова / Bulgakova E.
Елена Валерьевна
(koordinator-proekta@mail.ru)
кандидат юридических наук, доцент.
ФГОБУ ВО «Финансовый университет при Правительстве Российской Федерации» (Финуниверситет),
доцент кафедры информационной безопасности.
г. Москва

Петров / Petrov I.
Илья Васильевич
(llya_petrov202@mail.ru)
Финуниверситет,
магистр кафедры информационной безопасности.
г. Москва

Хананашвили / Khananashvili M.
Марк Давидович
(xananashvilim@mail.ru)
ФГБОУ ВО «Московский технический университет связи и информатики»,
аспирант кафедры безопасности телекоммуникаций.
г. Москва

Кубанков / Kubankov A.
Александр Николаевич
(kan9991@gmail.com)
доктор военных наук, профессор.
ФГБУ «Российский институт стандартизации,
главный научный сотрудник.
г. Москва

Ключевые слова: имитационное моделирование – simulation modeling; критическая информационная инфраструктура (КИИ) – critical information infrastructure; информационная безопасность – information security; DDoS-атаки – DDoS attacks.

В статье рассматривается возможность применения методов имитационного моделирования, в частности агентного моделирования, для тестирования устойчивости серверов к атакам типа DDoS. Описаны основные подходы для построения имитационных моделей. Рассмотрены задачи обеспечения безопасности объектов КИИ, которые решаются с применением этих методов. Разработана модель процесса проведения DDoS-атаки на кластеры серверов.

The article considers the possibility of using simulation modeling methods, in particular agent-based modeling, to test the resistance of servers to DDoS attacks. The main approaches for building simulation models are described. The tasks of ensuring the safety of critical information infrastructure facilities, which are solved using these methods, are considered. A model of the process of DDoS attack on a cluster of servers is developed.

Введение

На сегодняшний день объекты критической информационной инфраструктуры (КИИ) являются важными элементами, обеспечивающими стабильное функционирование экономики, государственных органов и жизненно важных отраслей. Угроза информационной безопасности для таких объектов неизменно растет из-за постоянного усложнения кибератак и появления новых угроз безопасности от злоумышленников. Одной из передовых практик обеспечения информационной безопасности объектов КИИ является применение методов имитационного моделирования, которые позволяют анализировать и прогнозировать потенциальные уязвимости и последствия кибератак на стадии проектирования систем. В данной статье рассматриваются основные методы имитационного моделирования, а также их возможности в решении задач, связанных с повышением устойчивости КИИ к кибератакам, в частности атакам типа DDoS. Основная цель работы – продемонстрировать эффективность применения агентного подхода на примере разработанной модели имитации DDoS-атаки типа HTTP-flood на кластеры серверов объекта КИИ с дальнейшим исследованием зависимости устойчивости кластера от изменений параметров модели.

Необходимость купирования ситуаций, связанных с отказом в обслуживании, соответствует задачам обеспечения безопасности объектов критической информационной инфраструктуры в соответствии с Федеральным законом «О безопасности критической информационной инфраструктуры Российской Федерации» от 26 июля 2017 №187-ФЗ [1].

Цели применения методов имитационного моделирования в обеспечении безопасности значимых объектов критической информационной инфраструктуры (КИИ) заключаются в предварительном апробировании проектируемой системы для выявления уязвимостей и соответствии реализуемого проекта специфическим требования к защите объектов КИИ, сформулированным в законодательной базе Российской Федерации [2]. Методы имитационного моделирования позволяют решить следующие задачи в рамках достижения обозначенных целей: задачи по аудиту, задачи по мониторингу, задачи по тестированию, задачи по оценке рисков. Но прежде необходимо обозначить, что такое имитационное моделирование, какие в нем используются методы и средства.

Имитационное моделирование является методом исследования, который основан на имитации функционала системы (либо ее отдельных компонентов) с применением средств электронно-вычислительной техники и специализированного программного обеспечения, размещенного на ней [3–5]. Основная концепция метода заключается в построении алгоритмов и логических цепочек, воспроизводящих поведение и изменение системы с течением времени. В процессе имитации цифровой двойник системы сохраняет те свои ключевые свойства и характеристики, которые соответствуют исследуемым параметрам и необходимым условиям, заданным при формировании задачи проводимого исследования.

Главная задача имитационного моделирования – воспроизвести исследуемую систему в среде виртуализации, при этом сохранив логическую последовательность и структуру происходящих в результате ее функционирования процессов. Авторы статьи [6] утверждают: «при имитационном моделировании реализующий модель алгоритм воспроизводит процесс функционирования системы во времени, причем имитируются явления, составляющие процесс, с сохранением их логической структуры и последовательности протекания во времени». Таким образом, применение этого подхода позволяет получать информацию о поведении системы, ее аппаратно-технических возможностях и ограничениях, степени надежности и отказоустойчивости еще до ее физической реализации на практике.

В настоящий момент применение имитационного моделирования можно назвать передовым методом проведения исследований и анализа тех систем, которые отличаются повышенной сложностью и количеством требующих учета параметров [7–9]. При работе с такого рода системами имитационное моделирование порой становится единственной возможностью для проведения корректного анализа. Такие модели способны учитывать как дискретные, так и непрерывные элементы системы. Кроме того, они могут отражать нелинейные свойства компонентов, которые входят в систему. Еще одной чертой имитационных моделей является способность, в отличие от иных видов моделирования, учитывать различного рода факторы случайности, которые влияют на чистоту получаемых результатов исследования. В особенности это касается тех факторов, которые формируются во внешней среде.

Методы и средства имитационного моделирования

Существуют следующие методы имитационного моделирования [3–6]:

1. **Дискретно-событийное моделирование.** Метод, основанный на представлении моделируемой системы в виде последовательности дискретных событий, каждое из которых оказывает эффект, приводящий к изменению состояния системы.

2. **Системная динамика.** Данный тип моделирования применяется для систем с непрерывным временем. Этот метод позволяет анализировать изменения системы на макроуровне.

3. **Агентное моделирование.** Сущность данного подхода в моделировании поведения каждого отдельно взятого агента, взаимодействующего как с другими агентами, так и с окружающей их средой.

4. **Метод Монте-Карло.** Метод имитации, который применяет случайные величины для моделирования процессов, которым присущи элементы случайности и неопределенности.

Применение методов имитационного моделирования обладает рядом преимуществ, которые стали основной причиной роста актуальности применения данного подхода при анализе сложных систем [10]:

1. Методы имитационного моделирования позволяют описывать системы в тех случаях, когда классические методы, такие как составление аналитической модели, в значительной степени теряют свою точность, становясь лишь отдаленно похожими на реальную ситуацию.

2. Благодаря реализации в среде виртуализации на базе электронно-вычислительной техники методы имитационного моделирования позволяют производить множество альтернативных версий системы, тестировать каждую из них и выбирать оптимальную в соответствии с поставленной задачей.

3. Несмотря на ресурсозатратность развертывания качественной имитационной модели, применение этого подхода является экономически более выгодным, чем создание тестовой инфраструктуры на физических машинах.

4. Благодаря имитационному моделированию можно произвести исследование поведения системы на длительных периодах за короткий промежуток реального времени.

5. Динамическое имитационное моделирование позволяет получить большое количество выходных данных, основанных на множественных итерациях эксперимента с разными параметрами.

Результаты, полученные в процессе моделирования, могут быть обработаны методами интеллектуального анализа данных, в частности нейронной сетью [11]. Скорость обработки информации нейросетью значительно превосходит человеческие возможности. При этом применение систем искусственного интеллекта позволит исключить человеческий фактор из точности расчета.

Касательно средств имитационного моделирования на данный момент существуют следующие наиболее распространенные программные реализации продуктов для построения подобных моделей:

1. *AnyLogic*. Представляет собой многофункциональную платформу высокой производительности, которая поддерживает дискретно-событийное моделирование, системную динамку и агентное моделирование.

2. *Simulink*. Это инструмент для моделирования, симуляции и проведения анализа динамических систем.

3. *Arena*. Программный продукт для дискретно-событийного моделирования, ориентированный на производственные системы, логистику и анализ бизнес-процессов.

4. *NetLogo*. Представляет собой агентно-ориентированный язык программирования и интегрированную среду разработки.

5. *FlexSim*. Программный инструмент для имитационного моделирования в 3D-окружении, применяемый для анализа производственных и логистических систем.

Задачи и возможности применения имитационного моделирования в разработке объектов КИИ

Применение методов имитационного моделирования при разработке стратегии обеспечения безопасности значимых объектов критической информационной инфраструктуры (КИИ) включает в себя ряд задач, связанных с обеспечением безопасности. Теперь необходимо подробнее остановиться на каждой из этих задач и роли имитационного моделирования в их реализации (таблица 1).

Стоит также отметить, что модели, созданные с применением методов имитационного моделирования, могут быть эффективно использованы для понимания

Таблица 1

Решение задач методами имитационного моделирования

№	Задача	Роль имитационного моделирования в решении задачи
1.	Аудит	Анализ соответствия нормативным требованиям. Проверка конфигураций безопасности. Анализ соответствия политик безопасности.
2.	Мониторинг	Динамическое мониторирование сетевых взаимодействий. Мониторинг работы системы в стрессовых условиях. Проверка процессов реагирования на инциденты.
3.	Тестирование	Тестирование систем на устойчивость к кибератакам. Тестирование резервных механизмов восстановления.
4.	Оценка рисков	Оценка вероятности возникновения угроз. Оценка последствий инцидентов безопасности. Анализ сценариев катастрофических отказов.
5.	Оптимизация	Оптимизация архитектуры безопасности. Оптимизация расходов на защитные меры. Оптимизация процессов управления доступом.

принципов, по которым реализуются атаки [12–14]. Этот факт делает имитационные модели пригодными для обучения как персонала, работающего на защищаемом объекте, так и специалистов по информационной безопасности в высших учебных заведениях.

Реализация и применение имитационной модели для тестирования устойчивости сервера к DDoS-атакам

В рамках исследования необходимо доказать эффективность применения агентного моделирования при разработке стратегии обеспечения безопасности объектов критической информационной инфраструктуры и решения одной из приведенных выше задач. Для этого в среде имитационного моделирования AnyLogic [15] была реализована модель процесса осуществления DDoS-атаки типа HTTP-flood на сеть серверных кластеров в режиме горячего резервирования active-active для исследования взаимосвязи его устойчивости от конфигурации системы и параметров реализации атаки. В дальнейшем эта информация может применяться для, например, корректировки правил фильтрации, оказывающих значительное влияние на производительность кластера. В частности, данная модель способствует решению задачи «тестирование».

В соответствии со сценарием исследования модель представляет собой 2 кластера веб-серверов по 4 сервера в каждом кластере, связанных между собой балансировщиком нагрузки. На веб-серверы поступают HTTP-запросы от пользователей, находящихся в открытой сети Интернет. Серверы подключены к сети через маршрутизирующее устройство с функцией межсетевого экранирования.

Каждый запрос от пользователя обрабатывается по времени от 1 до 500 мс. В частности, 1–10 мс для запроса на статический контент с веб-страницы и 10–500 мс для динамического контента. В секунду от легитимных пользователей поступает 150–500 HTTP-запросов разного размера. Таким образом, на каждый сервер приходится порядка 1.5–5 запросов в мс, каждый из которых занимает 1–500 мс на обработку.

DDoS-атака продолжалась в течение 30 минут. В ходе исследования устойчивость системы определялась по следующим критериям:

1. В случаях, когда система успешно обрабатывала поступающие запросы от легитимных пользователей за время активной атаки, считалось, что система устойчива к DDoS-атаке.

2. В случаях, когда у системы возникал отказ в обслуживании всех серверов обоих кластеров, система считалась неустойчивой.

В процессе исследования для каждой конфигурации системы производилось по 8 итераций экспериментов с возрастающей интенсивностью входящего от ботнет-сети трафика. В каждой итерации интенсивность трафика изменялась с течением времени. Начиная с 0 до 25 минут моделирования интенсивность трафика росла, а с 25 до 30 минут снижалась до изначального значения. Каждая итерация определялась количеством запросов в мс на один сервер и задавалась таблицей 2.

В качестве моделируемых серверов использовались Apache HTTP Server 2.4.41 на базе операционной системы CentOS 7.4 64bit. Сервера были представлены в трех конфигурациях с разным количеством одновременно обрабатываемых запросов:

1. Ядро – 1, ОЗУ – 512 Мб, SSD – 25 Гб, 60 одновременных запросов

2. Ядро – 2, ОЗУ – 1 Гб, SSD – 50 Гб, 120 одновременных запросов

3. Ядро – 3, ОЗУ – 2 Гб, SSD – 100 Гб, 240 одновременных запросов

Единицей дискретизации модели является минута. Все агенты в модели представляются в виде HTTP-запросов, поступающих на веб-серверы. У агентов есть два источника, представляющих собой элементы source «от_пользователей» и «от_ботов». Каждый источник помечает свои агенты тегами «user» и «attacker» соответственно.

Для моделирования процесса обработки запросов были применены элементы типа «service», которые

Таблица 2

Распределение интенсивности нагрузки по уровням

№	Интенсивность (з/мс)
1	87,5–267,5
2	175–535
3	262,5–802,5
4	350–1070
5	437,5–1337,5
6	525–1605
7	612,5–1872,5
8	700–2140

имитировали сервера: время обработки каждого запроса задавалось треугольной функцией распределения triangular(min, max, mod), где:

min – минимальное время, затрачиваемое на обработку входящего запроса, в рамках данной модели не ниже 1 мс;

max – максимальное время обработки запроса, поделенное на количество одновременно обрабатываемых сервером запросов;

mod – модальное значение, обозначающее наиболее частые значения из заданного диапазона.

В зависимости от конфигурации серверов данные значения равнялись: triangular(1, 8.3, 4.65), triangular (1, 4.2, 2.6), triangular(1, 2, 1.5)

Все треугольные функции в данной работе симметричные, дабы избежать излишнего количества крайних значений. При реализации подобной модели на реальном предприятии необходимо формировать распределение на основе данных, полученных за предшествующее построению модели время работы системы.

В случаях, когда загрузка сервера превышает 90 %, сервер выдает отказ в обслуживании и перестает принимать новые запросы, пока сервер не обработает уже поступившие и загрузка не снизится. Если запросы не могут попасть на сервер для обработки, они перенаправляются на другой свободный сервер. При отсутствии таковых запросы отбрасываются.

Балансировщик равномерно распределяет нагрузку между кластерами, постоянно производя мониторинг состояния серверов. Когда все сервера одного из кластеров выдают отказ в обслуживании, весь трафик перенаправляется на другой кластер.

Схема балансировки и перераспределения нагрузки на сервера изображена на рис. 1.

Перед поступлением запросов на балансировщик они проходят через межсетевой экран. В рамках данного исследования было выдвинуто 3 сценария, в которых межсетевой экран фильтровал 50 %, 70 % и 90 % запросов от злоумышленника. Межсетевой экран реализован через элемент *selectOutput* с условием *agent.sourceType.equals*("*user*") ‖ (!*agent.sourceType.equals*("*user*") && *uniform*() < 0.5). Схема модели представлена на рис. 2.

Для моделирования поведения DDoS трафика использовались константное значение интенсивности от 1 до 8 в переменной *volume*.

Константа задавала нагрузку в пересчете на один сервер в соответствии с таблицей 2. Переменная с

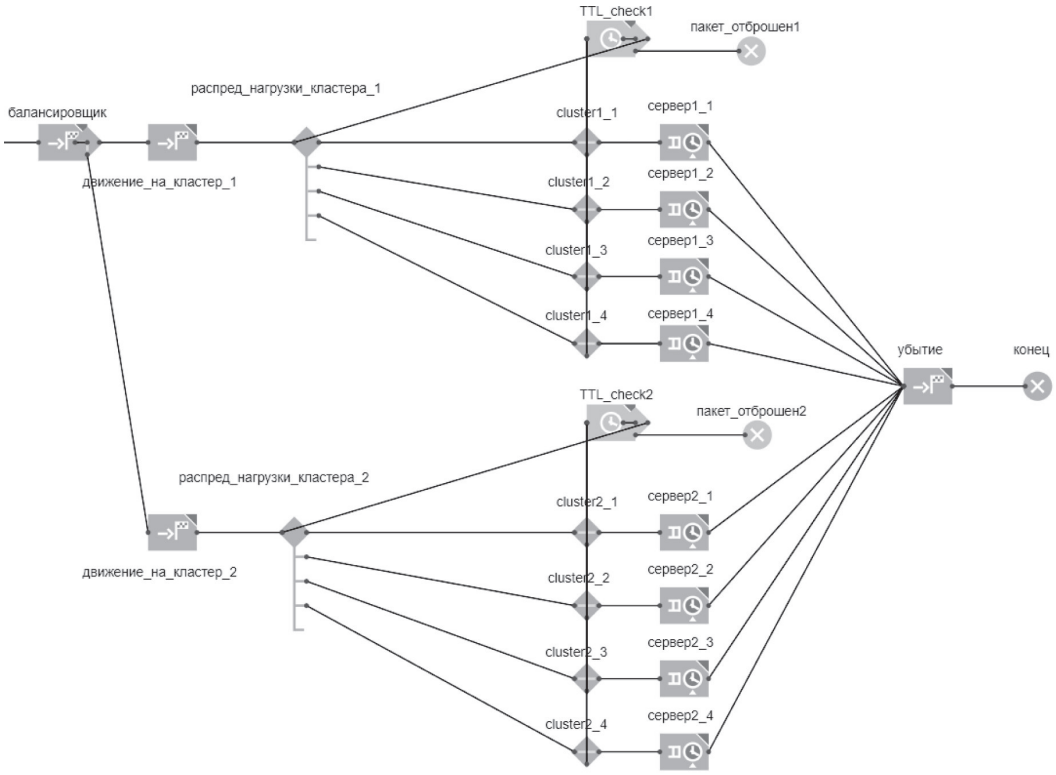

Рис. 1. Схема модели кластера

начальным значением 70 000 (700 з/мс) изменялась каждую минуту, поднимаясь до 214 000 к 25 минуте и снова опускаясь до 70 000 к 30 минуте. Реализовано через элемент событие *up_time*, каждую минуту исполняющее скрипт, повышающий, а затем понижающий значение *volume* (рис. 3).

Для моделирования легитимного трафика использовалась треугольная функция распределения *triangular* (150, 500, 325), в соответствии с которой каждую секунду поступал трафик, равный 150–500 запросам в пересчете на каждый сервер.

В процессе исследования отслеживалась зависимость устойчивости системы от изменения следующих параметров модели:

1. Производительность серверов.
2. Интенсивность DDoS трафика.
3. Эффективность фильтрации межсетевого экрана.

Также рассчитывался средний процент загруженности серверов обоих кластеров (Avg C1 и Avg C2) и суммарное время работы в ходе реализации атаки. Для отслеживания нагрузки применялись события *load_sum*1 и *load_sum*2, которые раз в минуту высчитывали средние показатели загруженности всех серверов кластеров. С каждым новым расчетом средних показателей в переменным *avgCluster*1 и *avgCluster*2 присваивались значения, соответствующие средним показателям нагрузки кластеров с начала моделирования. По завершении моделирования в этих переменных хранились значения средней загрузки кластера на всем протяжении работы модели. Скрипт, ответственный за расчет, приведен на рис. 4.

Все полученные данные отображались на временном графике, способствующем отслеживанию динамики нагрузки серверов (рис. 5).

В рамках модели параметры производительности серверов и эффективности межсетевого экрана представлены в 3 вариантах каждый. Было проведено 9 экспериментов для отслеживания эффективности различных комбинаций этих параметров между собой. Для определения предела устойчивости каждой комбинации они проверялись в условиях нарастающей интенсивности DDoS трафика, заданного 8 значениями. Каждый эксперимент с каждым из значений интенсивности проводился 100 раз, после чего из полученных результатов выбиралось медианное значение и заносилось в таблицу для дальнейшего анализа.

В соответствии с полученными результатами:

1. При конфигурации серверов, соответствующей обработке 60 запросов одновременно, система считается устойчивой при параметрах фильтрации 90 % (таблица 3).

2. При конфигурации серверов, соответствующей обработке 120 запросов одновременно, система считается устойчивой при параметрах фильтрации 90 % (таблица 4).

3. При конфигурации серверов, соответствующей обработке 240 запросов одновременно, система считается устойчивой при параметрах фильтрации 70 % (таблица 5).

4. При конфигурации серверов, соответствующей 240 одновременно обрабатываемым запросам и параметрах фильтрации 90 %, система считается устойчивой (таблица 6).

Все остальные эксперименты, включающие в себя параметры фильтрации 50 % и 70 % при производственных мощностях, равных 60 и 120 одновременно обрабатываемым запросам, а также параметрам фильтрации 50 % при мощностях в 240 одновременно обрабатываемых запросов, показали себя как не способные обеспечить устойчивость системы, выдавая отказ системы на различных уровнях интенсивности трафика.

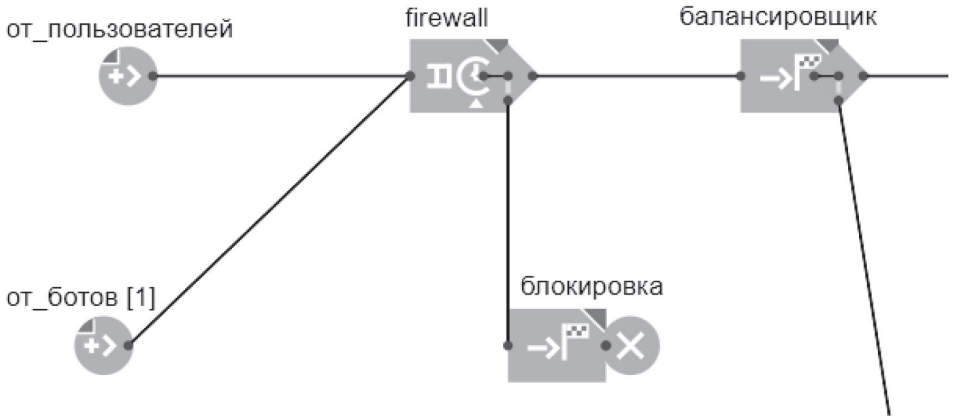

Рис. 2. Схема модели межсетевого экрана

```
if (время_работы <= 25) {
    volume = 70000 + (214000 - 70000) * (время_работы / 25.0);
} else if (время_работы <= 30) {
    volume = 214000 - (214000 - 70000) * ((время_работы - 25) / 5.0);
} else {
    volume = 70000;
}

traceln("Время: " + время_работы + " мин → Volume: " + volume);
```

Рис. 3. Код скрипта контроля переменной volume

```
double load1 = загрузкаСервера2_1;
double load2 = загрузкаСервера2_2;
double load3 = загрузкаСервера2_3;
double load4 = загрузкаСервера2_4;

double average = (load1 + load2 + load3 + load4) / 4.0;

sumClusterLoad2 += average * 100;
countMeasurements2++;
avgCluster2 = sumClusterLoad2 / countMeasurements2
```

Рис. 4. Код скрипта расчета средней нагрузки кластеров

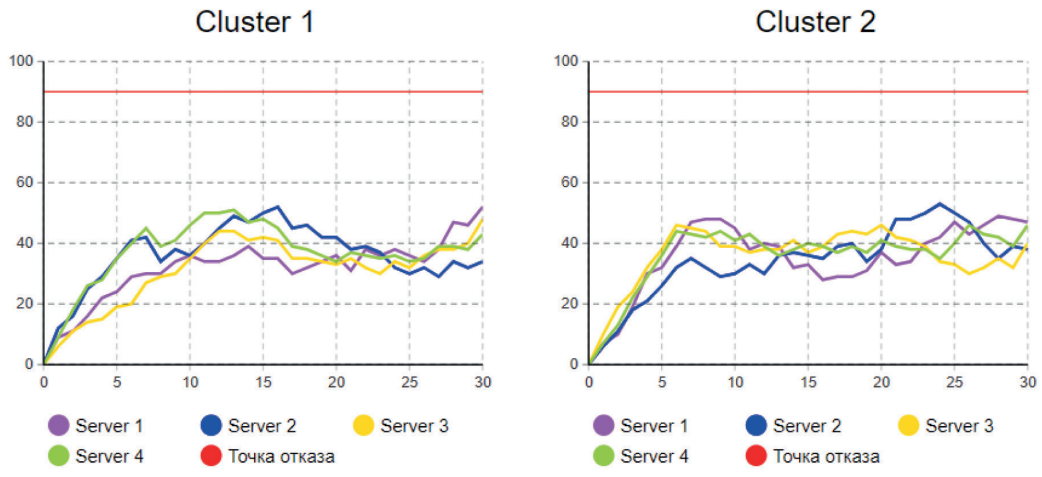

Рис. 5. График нагрузки кластеров во время эксперимента 3.8

Таблица 3

Результаты эксперимента 3

№	Интенсивность (з/мс)	Обработано легитимного трафика (з)	%	Обработано DDoS трафика (з)	%	Avg C1	Avg C2	Время (с)
0	нет	4 716 157	100	-	-	-	-	1800
1	87,5 – 267,5	4 654 557	29	11 619 724	71	6.894	6.01	1800
2	175 – 535	4 676 388	18	21 631 843	82	11.462	9.731	1800
3	262,5 – 802,5	4 726 039	13	30 733 219	87	17.433	13.567	1800
4	350 – 1070	4 753 920	10	40 392 110	90	23.288	19.394	1800
5	437,5 – 1337,5	4 678 393	8	53 128 476	92	28.375	22.779	1800
6	525 – 1605	4 652 834	7	61 602 345	93	32.365	26.76	1800
7	612,5 – 1872,5	4 643 517	6	71 258 909	94	36.481	30.683	1800
8	700 – 2140	4 660 844	5	82 461 102	95	42.163	34.49	1800

Таблица 4

Результаты эксперимента 6

№	Интенсивность (з/мс)	Обработано легитимного трафика (з)	%	Обработано DDoS трафика (з)	%	Avg C1	Avg C2	Время (с)
0	нет	4 667 242	100	-	-	-	-	1800
1	87,5 – 267,5	4 668 307	28	11 969 038	72	4.25	3.356	1800
2	175 – 535	4 708 708	17	23 238 616	83	6.798	6.279	1800
3	262,5 – 802,5	4 655 673	12	33 739 982	88	10.048	8.981	1800
4	350 – 1070	4 701 882	9	45 781 450	91	13.01	11.337	1800
5	437,5 – 1337,5	4 689 048	8	56 702 867	92	16.904	12.346	1800
6	525 – 1605	4 730 662	6	70 212 790	94	20.837	15.327	1800
7	612,5 – 1872,5	4 645 670	6	78 611 983	94	22.365	17.644	1800
8	700 – 2140	4 703 394	5	90 931 204	95	27.923	23.058	1800

Таблица 5

Результаты эксперимента 8

№	Интенсивность (з/мс)	Обработано легитимного трафика (з)	%	Обработано DDoS трафика (з)	%	Avg C1	Avg C2	Время (с)
0	нет	4 680 234	100	-	-	-	-	1800
1	87,5 – 267,5	4 711 206	11	37 312 589	89	6.663	5.067	1800
2	175 – 535	4 645 253	6	70 842 196	94	10.5	9.125	1800
3	262,5 – 802,5	4 657 134	4	104 798 741	96	16.404	12.183	1800
4	350 – 1070	4 623 893	3	142 897 376	97	20.413	16.962	1800
5	437,5 – 1337,5	4 640 406	3	170 401 920	97	25.885	20.423	1800
6	525 – 1605	4 682 133	2	208 003 138	98	30.173	24.077	1800
7	612,5 – 1872,5	4 697 177	2	254 598 431	98	37.933	30.317	1800
8	700 – 2140	4 666 948	2	288 001 866	98	39.471	32.298	1800

Сравнительный анализ полученных результатов показал, что среди конфигураций системы, продемонстрировавших устойчивость к моделируемой DDoS-атаке, есть большие отличия в показателях загрузки. С точки зрения поиска конфигурации, удовлетворяющей как финансовым потребностям, так и требованиям к безопасности, оптимальным решением будет модернизировать правила фильтрации на межсетевом экране, при которых недорогих серверов Apache HTTP Server 2.4.41 на базе операционной системы CentOS 7.4 64bit с 1 ядром, 512 Мб ОЗУ и SSD с размером 25 Гб будет достаточно. Такой подход позволит при минимальных затратах ресурсов добиться реализации системы, устойчивой к DDoS-атаке типа HTTP-flood.

Заключение

В процессе исследования отслеживалось поведение модели во времени, оценивалась зависимость отказа в обслуживании кластера от изменения параметров модели: интенсивности трафика, резкий рост которого является характерной чертой DDoS-атак, изменения производственных мощностей, входящих в кластер серверов, а также изменения эффективности правил фильтрации. Разработанная система характеризуется многоканальностью и сложностью внутренних взаимодействий, как например балансировки нагрузки. Стандартные аналитические методы, такие как теория массового обслуживания, не применимы в условиях DDoS-атак, так как поток запросов перестает быть пуассоновским. Результаты моделирования продемонстрировали зависимость устойчивости кластера в условиях DDoS-атаки от производственных мощностей входящих в него серверов, интенсивности трафика, эффективности межсетевого экрана. Внесение коррективов в изменяемые параметры отражалось на устойчивости системы.

В результате исследования была продемонстрирована эффективность применения агентного подхода при моделировании кибератак с целью исследования их влияния на защищаемые системы. Особенностью применения методов имитационного моделирования является возможность произвести испытания проектируемой системы безопасности на устойчивость сразу нескольким сценариям, что было продемонстрировано в рамках исследования путем проведения девяти экспериментов с различными параметрами элементов системы. Итогом проведенного исследования стало выявление зависимости устойчивости серверного кластера от параметров реализации атаки, а также создание модели, позволяющей оценить степень устойчивости кластера, пригодной для последующей выработки и применения мер по устранению потенциальных уязвимостей.

Литература

1. О безопасности критической информационной инфраструктуры Российской Федерации : федеральный закон от 26 июля 2017 года № 187-ФЗ // КонсультантПлюс : [сайт]. – URL: https://www.consultant.ru/document/cons_doc_LAW_220885/ (дата обращения: 09.09.2024).

2. Приказ ФСТЭК «Об утверждении требований к обеспечению защиты информации в автоматизированных системах управления производственными и технологическими процессами на критически важных объектах, потенциально опасных объектах, а также объектах, представляющих повышенную опасность для жизни и здоровья людей и для окружающей природной среды» от 14.03.2014 № 31 (с изм. и допол. в ред. от 15.03.2021) // Федеральная служба по техническому и экспортному контролю.

Таблица 6

Результаты эксперимента 9

№	Интенсивность (з/мс)	Обработано легитимного трафика (з)	%	Обработано DDoS трафика (з)	%	Avg C1	Avg C2	Время (с)
0	нет	4 715 311	100	-	-	-	-	1800
1	87,5 – 267,5	4 670 911	28	12 181 672	72	2.596	2.346	1800
2	175 – 535	4 717 994	17	23 311 448	83	3.673	3.442	1800
3	262,5 – 802,5	4 639 612	12	34 018 557	88	4.962	4.587	1800
4	350 – 1070	4 757 769	9	44 732 925	91	5.808	5.971	1800
5	437,5 – 1337,5	4 684 312	8	57 328 614	92	7.462	7.019	1800
6	525 – 1605	4 721 508	6	71 468 390	94	8.779	9.202	1800
7	612,5 – 1872,5	4 678 299	5	82 668 775	95	10.317	10.038	1800
8	700 – 2140	4 650 796	5	92 819 043	95	12.058	10.913	1800

3. Боев, В. Д. Имитационное моделирование систем / В.Д. Боев. – Москва : Издательство Юрайт, 2020. – 253 с.

4. Замятина, О. М. Моделирование систем : учебное пособие / О.М. Замятина. – Томск : Изд-во ТПУ, 2009. – 204 с.

5. Кобелев, Н. Б. Имитационное моделирование / Н.Б. Кобелев, В.А. Половников, В.В. Девятков. – Москва : КУРС, 2020. – 352 с.

6. Белов, А. Г. Методы имитационного моделирования / А.Г. Белов, С.А. Моисеев, А.В. Григорьев // Труды международного симпозиума «Надежность и качество». – 2014. – Т. 1. – С. 277–279.

7. Долгова, О. И. Имитационное моделирование бизнес-процессов промышленных компаний в условиях Индустрии 4.0 / О.И. Долгова, А.Ю. Никитаева // П-Economy. – 2023. – Vol. 16, Iss. 4. – P. 26–40.

8. De Paula Ferreira, W. Simulation in industry 4.0: A state-of-the-art review / W. de Paula Ferreira, F. Armellini, L.A. Santa-Eulalia // Computers & Industrial Engineering. – 2020. – Vol. 149. – P. 106868.

9. Черезов, Н. С. Имитационное моделирование производственных процессов / Н.С. Черезов, А.В. Кириллов, Я.Ю. Григорьев // Молодежь и наука: актуальные проблемы фундаментальных и прикладных исследований : материалы V Всероссийской национальной научной конференции молодых учёных (Комсомольск-на-Амуре, 11–15 апреля 2022). – Комсомольск-на-Амуре: Комсомольский-на-Амуре государственный университет, 2022. – Т. 2. – С. 411–413.

10. Law, A. M. How to build valid and credible simulation models / A.M. Law // 2022 Winter Simulation Conference (WSC). – 2022. – P. 1283–1295.

11. Hybrid simulation modelling in operational research: A state-of-the-art review / S. Brailsford, T. Eldabi, M. Kunc [et al.] // European Journal of Operational Research. – 2019. – Vol. 278, No. 3. – P. 721–737.

12. Simulation for cybersecurity: state of the art and future directions / H. Kavak, J.J. Padilla, D. Vernon-Bido [et al.] // Journal of Cybersecurity. – 2021. – Vol. 7, Iss. 1. – P. 1–13.

13. Cyber security threat modeling based on the MITRE Enterprise ATT&CK Matrix / W. Xiong, E. Legrand, O. Eberg, R. Lagerstrцm // Software and Systems Modeling. – 2022. – Vol. 21. – P. 157–177.

14. Ткаченко, А. Л. Имитационное моделирование распространения кибератак на промышленные предприятия / А.Л. Ткаченко, А.Ю. Гордеева, А.В. Шавренко // Инновационные технологии, экономика и менеджмент в промышленности : сборник научных статей по итогам IV международной научной конференции (Волгоград, 22–23 апреля 2021). – Москва : Конверт, 2021. – Т. 1. – С. 238–240.

15. Ефромеева, Е. В. Имитационное моделирование: основы практического применения в среде AnyLogic : учебное пособие / Е.В. Ефромеева, Н.М. Ефромеев. – Саратов : Вузовское образование, 2020. – 120 с.