ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ТЕЛЕКОММУНИКАЦИИ
INFORMATION TECHNOLOGY AND TELECOMMUNICATIONS

УДК 004.896

DOI: 10.21822/2073-6185-2025-52-2-139-149 Оригинальная статья/ Original article

Механизм защиты от DDoS–атаки, направленной на блокирование учётных записей

П.В. Разумов, Л.В. Черкесова, Е.А. Ревякина, А.А. Клевцов
Донской государственный технический университет,
344000, г. Ростов-на-Дону, пл. Гагарина, 1, Россия

Резюме. Цель. Целью исследования является программный анализ кибератаки, осуществляемой с целью блокировки учётных записей и реализация защиты механизма защиты. **Метод.** Исследование основано на применении языка программирования PHP в среде IDE PhpStorm от компании JetBrains. Помимо данного языка использовались языки верстки CSS и JS, язык разметки HTML. **Результат.** Подробным образом рассматривается DDoS-атака типа блокирования учётных записей, ее возможные последствия и существующие механизмы защиты. С целью противодействия осуществлению данной атаки предложена модификация алгоритма, которая позволяет значительно снизить вероятность её успешного проведения. Результатом данной работы является разработанные программное средство и проведенный сравнительный анализ с аналогичными продуктами на рынке. **Вывод.** Проанализированы готовые программные решения и разработан механизм защиты, являющийся современным решением, обеспечивающим и надежную защиту учётных записей пользователей, и защиту от атак, направленных на них.

Ключевые слова: DDOS атака, блокировка, учётные записи, механизм реализации атаки, противодействие атаке, алгоритм нейтрализации атаки

Для цитирования: П.В. Разумов, Л.В. Черкесова, Е.А. Ревякина, А.А. Клевцов. Механизм защиты от DDoS–атаки, направленной на блокирование учётных записей. Вестник Дагестанского государственного технического университета. Технические науки. 2025; 52(2):139-149. DOI:10.21822/2073-6185-2025-52-2-139-149

Protection mechanism against ddos attack intended to block account

P.V. Razumov, L.V. Cherkesova, E.A. Revyakina, A.A. Klevtsov
Don State Technical University,
1 Gagarin Square, Rostov-on-Don 344000, Russia

Abstract. Objective. The aim of the study is a software analysis of a cyber attack carried out to block accounts and implement protection of the protection mechanism. **Method.** The study is based on the use of the PHP programming language in the PhpStorm IDE environment from JetBrains. In addition to this language, the CSS and JS layout languages, the HTML markup language were used. **Result.** A DDoS attack of the account blocking type, its possible consequences and existing protection mechanisms are considered in detail. Also, in order to counteract the implementation of this attack, a modification of the algorithm is proposed, which can significantly reduce the likelihood of its successful implementation. The result of this work is the developed software tool and a comparative analysis with similar products on the market. **Conclusion.** Ready-made software solutions are analyzed and a protection mechanism is developed, which is a modern solution that provides reliable protection of user accounts and protected from attacks directed at them.

Keywords: DDOS attack, blocking, accounts, attack implementation mechanism, attack countermeasures, attack neutralization algorithm

For citation: P.V. Razumov, L.V. Cherkesova, E.A. Revyakina, A.A. Klevtsov. Protection mechanism against ddos attack intended to block account. Herald of Daghestan State Technical University. Technical Sciences. 2025;52(2):139-149. (In Russ)DOI:10.21822/2073-6185-2025-52-2-139-149

Введение. На сегодняшний день имеется огромное количество пользовательских платформ, использующих для авторизации учётные записи пользователя. С каждым днем их становится все больше, они внедряются повсеместно и хранят в себе конфиденциальные данные, которые должны защищаться от атак злоумышленника.

Так, в последнее время наблюдается значительное количество DDoS-атак на ключевые российские компании топливно-энергетического сектора, финансовой отрасли, производственной сферы и телеком-индустрии. Атакам подверглось большое количество крупных организаций. И непосредственно для обеспечения безопасности и возможности своевременного доступа пользователя к своей учётной записи разрабатываются новые и совершенствуются уже имеющиеся механизмы защиты [1].

Кроме того, на сегодняшний день совершенствование подходящей административной структуры в Российской Федерации укрепляет наиболее распространенный способ вывода такого рода инноваций на передовой рынок. Наиболее перспективными подходами в области данных являются использование криптографических методов для защиты данных для различных целей, включая использование государственными органами и бизнес-ассоциациями, средствами электронной коммерции, использование методов шифрования и методов защиты веб-ресурсов от DoS-атак.

В текущей работе подробным образом рассматривается DDoS-атака типа блокирования учётных записей, ее возможные последствия и существующие механизмы защиты. Также, с целью противодействия осуществлению данной атаки предложена модификация алгоритма, которая позволяет значительно снизить вероятность её успешного проведения. Результатом данной работы является разработанное программное средство и проведенный сравнительный анализ с аналогичными продуктами на рынке.

Постановка задачи. Целью исследования является программный анализ кибератаки, осуществляемой с целью блокировки учётных записей и реализация механизма защиты.

Для достижения цели необходимо решить следующие задачи: проанализировать существующие программные решения, применяемые для защиты от различных типов DDoS атак и в частности от атаки, направленной на блокирование учётных записей; исследовать современные методы детектирования сетевых DDoS–атак; проанализировать готовые программные решения и методы борьбы с атаками на учётные записи; на основе проведенного анализа составить алгоритм и схему работы web – сервиса; выполнить програмную реализацию и тестирование разработанного программного средства.

Методы исследования. Самой распространённой атакой на учетные записи являются, так называемые, атаки грубой силы (brute-force attack). Данная атака направлена на перебор пароля, способ можно считать эффективным, когда в системе используются довольно простые пароли. Длина и сложность, а также использование специальных символов сильно затрудняют атаку. Также достаточно часто используется обратная атака грубой силы, когда один распространенный пароль проверяется сразу на многих логинах пользователей.

Атака по словарю, dictionary attack. Данный метод использует часто используемые повседневные слова и фразы. Такие списки злоумышленники формируют на этапе сбора информации, либо используют уже готовые словари.

Гибридная атака представляет собой объединение атаки по словарю и грубой силы. Таким образом, такая атака, вместо того, чтобы перебирать каждый пароль, использует

незначительные модификации слов, отдельных символов по всему словарю, что иногда приводит к реализации атаки.

Атака по радужным таблицам: данный метод отличается от перечисленных выше атак тем, что взламывает не пароли, а хеш-функцию паролей. Такими способом злоумышленники предварительно вычисляют таблицу данных, содержащую кэш хеш-алгоритмов и производит взлом пароля автономно.

Так как успешное выполнения атаки account lockout приводит к отказу в обслуживание системы или сервера, рассмотрим атаки этого типа. Атака блокировки учётных записей является атакой типа отказа в обслуживании или распределенной при атаке с разных устройств не ограниченного количества. Так как при блокировании большого количества учётных записей на сервер поступает огромное количество запросов, не каждая система сможет стабильно обрабатывать такой поток информации в связи с чем рядовой пользователь не сможет получить доступ к Web–сайту. Особенно актуально защищаться от данного типа атак в данное время, в расцвет кибервойн и кибертерроризма. Таким образом данная атака относится к атакам прикладного уровня и реализуется посредством HTTP – POST, HTTP – GET запросов, ботов и серверов, реализующих атаки на учётные записи.

Существующие методы защиты постоянно совершенствуются, но зачастую защищают владельца аккаунта от атак, направленных на перебор или подбор пароля, но при этом используя как один из методов борьбы с злоумышленником – блокировку учётной записи. Учётная запись – это хранимая на сервере или на компьютере пользователя совокупность данных, необходимых для его авторизации и предоставления доступа к его конфиденциальным, личным данным. Это рабочий и распространённый способ борьбы с угрозами данного типа, но не стоит забывать какие последствия может повлечь за собой данный способ, а именно невозможность пользователя своевременно получить доступ к своей учётной записи, а для сервиса блокирования учётных записей в огромном количестве может привести только к одному исходу – отказу в обслуживании, что влечет для компании финансовые потери. На практике сталкиваясь с такими атаками многие сервисы чтобы избежать отказа в обслуживании сервера, перестают блокировать учётные записи, что позволяет злоумышленнику провести свою атаку и в последствии заполучить пользовательские данные.

Рассмотрим уже существующие механизмы защиты в онлайн сервисах. Стоит отметить, что количество таких программ постоянно и все они имеют свои особенности, и более подробные методы приведены ниже в табл. 1.

Таблица 1. Способы защиты от DDOS – атак
Table 1. Methods of protection against DDOS attacks

Механизмы защиты от DDOS – атак DDOS attack protection mechanisms						
Фильтрация Filtration		По ресурсам By resources			Безопасность подключения Connection security	
Пакеты Packages	Потоки Streams	Изменение количества ресурсов Changing the amount of resources	Перенос ресурсов Transfer of resources	Разграничение ресурсов Resource delineation	SSL / TLS	Шифрование запросов Encrypting requests

Рассматривая тип атаки на account lockout в первую очередь, нужно правильно защищаться от всевозможных атак, реализуемых посредством HTTP – запросов и ботами, и различными программные средства по перебору паролей. При этом самый простой методы защиты от HTTP – атак это переход на безопасное SSL/TLS соединение посредством протокола HTTPS и использование внутреннего шифрования запросов с использованием симметричных шифров. Способы защиты от ботов реализовать сложнее, так как поведение бота нужно отличать от поведения реального пользователя для этого реализуют различные фильтры и смарт-сканирование, но только этим проблема не решается ведь как и методы защиты боты также усовершенствуются. Наиболее популярным способом защиты от ботов является полностью автоматизированный публичный тест Тьюринга для различения компьютеров и людей.

Основная идея предложить задачу, легко решаемую человек и крайне сложно решаемую компьютером. Имеет огромное количество реализаций, но существует один недостаток – не всегда можно разобрать текст, иногда приходится вводить по несколько раз. Хотя это достаточно простой способ затрудняющий перебор пароля, его используют далеко не все онлайн сервисы. К тому же если у злоумышленника нет цели похитить данные, ведь для этого нужно иметь идентификатор пользователя номер телефона или логин, а просто заблокировать учётную запись, многие сервисы не защищены от атак такого типа. Исходя из данных представленной таблицы, можно заметить, что все сервисы имеют свой механизм защиты частично отличающиеся друг от друга. Где не используется *Captcha* после определенного количества некорректного ввода данных авторизации идет блокировка входа в учётную запись. Многие большие компании, на которые направленная данная атака, просто не блокируют записи пользователя чтобы не блокировать сервис целиком, что дает возможность злоумышленнику попробовать перебрать пароль. Чтобы добиться максимальной защиты от блокировки учётных записей, так и от атак, нацеленных на перебор и подбор пароля пользователя, стоит не только использовать один из перечисленных методов, но и их комбинацию. Методы подробно перечислены в табл. 2.

Таблица 2. Методы защиты от атак на учётные записи
Table 2. Methods of protection against attacks on accounts

<table>
<tr><th>Способы осуществления Methods of implementation</th><th colspan="4">Методы защиты Methods of protection</th></tr>
<tr><td>HTTP</td><td colspan="2">SSL/TLS (HTTPS – соединение) compound</td><td colspan="2">Шифрование POST и GET запросов Encryption of POST and GET requests</td></tr>
<tr><td>Боты Bots</td><td>Анализ трафика Traffic analysis</td><td>Межсетевые экраны Firewalls</td><td>Компьютерные тесты Computer tests</td><td>Блокирование опасного трафика Blocking dangerous traffic</td></tr>
<tr><td>Перебор пароля Password Brute Force</td><td colspan="2">Современные требования к паролю Modern password requirements</td><td colspan="2">Хеширования данных авторизации в БД Hashing authorization data in the database</td></tr>
</table>

Рассматривая методы борьбы с онлайн атаками, можно заметить, что одним из самых распространенных способов предотвращения похищения данных в следствии перебора пароля является блокировка учётной записи пользователя. Но это ограничит своевременный доступ пользователя к его личному кабинет и доступ к полноценному функционалу нужного интернет–ресурса.

Для того чтобы обезопаситься от возможности данной атаки, рассмотрим следующие подходы:

- Переход от стандартного протокола HTTP на HTTPS. Https – расширение протокола передачи данных (HTTP) который обеспечивает шифрование соединения, что в свою очередь защитит от прослушивания сетевого соединения и затруднит или не допустит выполнение атаки реализующиеся с помощью POST и GET - запросов.
- Использование внутреннего шифрования POST и GET запросов, посредством симметричного алгоритма шифрования, чтобы обезопасить передачу данных.
- Хранение учётных данных в хешированном виде. В случае перехвата данных, хакер не сможет ими воспользоваться без предварительной расшифровки.
- Реализация анализа трафика и блокирование потенциально опасных подключений, для защиты от ботов и атак с распределённых устройств.
- Использовании программного средства по типу *Captcha*, при превышении попыток ввода неверных данных авторизации, что замедлит выполнение атаки.
- Добавления скрытого поля, при заполнении которого данные на сервер отправляться не будут. Это дополнительная защита от ботов.

– Рандомизация времени блокировки учётных записей пользователей для затруднения выполнения атаки, и возможность пользователя разблокировать свою учётную запись с помощью своей электронной почты.
– Использование современных требований к паролю при регистрации для усложнения перебора пароля злоумышленником, что при современных требованиях безопасности может увеличить затраты времени на сотни лет.

Используя данные методы, получим схему работы web–сервиса, (рис. 1). Использование данных методов и способов защиты позволит не только защитить аккаунт пользователя от входа злоумышленника и от атак различного вида, таких как атаки посредством HTTP – запросов, перебор пароля, в том числе и в автономном режиме, атаки с использованием ботов для подбора пользовательских данных либо блокирование учётных записей.

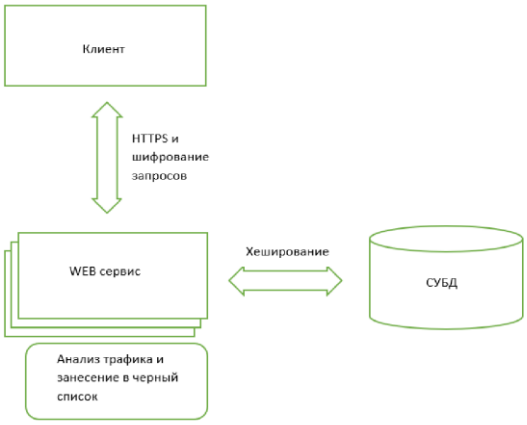

Рис. 1 - Схема работы web – сервиса
Fig. 1 - Web service scheme

Обсуждение результатов. Для разработки программного средства был выбран язык PHP, так как этот язык зарекомендовал себя как один из самых популярных и широко используемых языков в веб-разработке. Вместе с языком программирования была выбрана IDE PhpStorm от компании JetBrains, которая выпускает продукты практически для всех популярных на данный момент языков программирования, так же ее продукты зарекомендовали себя как многофункциональные и удобные IDE для разработки и ведения контроля версий. Помимо данного языка использовались языки верстки CSS и JS, язык разметки HTML.

PHP — скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических Web–сайтов [3 – 9]. Является сравнимо быстрым языком, сравнивая с более популярным Python и более молодым Golang, является удобным и сравнимо быстрым, хоть и использует интерпретатор.

Ajax — асинхронный JavaScript и XML, подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером [10 – 16].

В качестве базы данных было использованное современное СУБД MySQL на базе программного пакета MAMP. Разработанное программное средство можно использовать как встраиваемый механизм защиты любых страниц авторизации или регистрации.

Программное средство соответствует следующим требования к надежности:

1. Требования к надежности пароля по современным стандартам;
2. Использование HTTPS протокола для защищенного соединения;
3. Шифрование запросов с помощью симметричного алгоритма шифрования, в данном случае AES 256;
4. Хеширования пользовательских данных авторизации (login, password) и хранение на сервере в базе данных;

5. Наличие компьютерного теста *Captcha*;
6. Автоматическое блокирование потенциально опасных соединений.

Написанный алгоритм шифрования AES 256 хранится в отдельном файле aes.php. На рис. 2 приведена структура данного алгоритма. При создании программного средства использовался ранее локальный Web–сайт со страницей регистрации и входа с использованием AJAX–технологии.

```php
class Aes
{
    // основная ф-я шифрования
    public static function cipher($input, $w)
    {
        $Nb = 4; //
        $Nr = count($w) / $Nb - 1; // количество раундов: 10/12/14 для ключей 128/192/256-бит

        $state = [];  // 4xNb state
        for ($i = 0; $i < 4 * $Nb; $i++) {
            $state[$i % 4][floor( num: $i / 4)] = $input[$i];
        }

        $state = self::addRoundKey($state, $w,  rnd: 0, $Nb);

        // применить Nr раундов
        for ($round = 1; $round < $Nr; $round++) {
            $state = self::subBytes($state, $Nb);
            $state = self::shiftRows($state, $Nb);
            $state = self::mixColumns($state, $Nb);
            $state = self::addRoundKey($state, $w, $round, $Nb);
        }

        $state = self::subBytes($state, $Nb);
        $state = self::shiftRows($state, $Nb);
        $state = self::addRoundKey($state, $w, $Nr, $Nb);

        $output = [4 * $Nb];
```

Рис. 2 - Алгоритм AES
Fig. 2 - AES algorithm

Взаимодействие с программой начинается с попадания на страницу входа. При регистрации задается требование надежности логина и пароля, требования к паролю задавались, опираясь на документацию компании Касперский, которая сформировала современные рекомендации, ставшие общепринятыми: обязательно требуется использовать заглавные и строчные буквы, цифры, пробелы и специальные символы, набранные на разных регистрах; пароль должен содержать не менее 8 символов; данные для авторизации не должны совпадать; пароль не должен содержать фразы, которые легко подобрать. Например: login, parol, qwerty, qazwsxedc или abcdefg, 0123456789.

Схема реализации шифрования POST и GET запросов представлена на рис. 3:

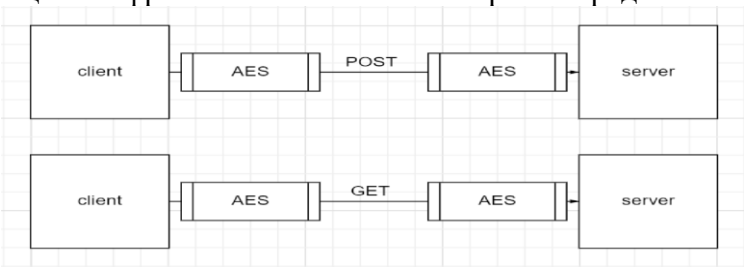

Рис. 3 - Схема шифрования запросов
Fig. 3 - Request ciphering scheme

Структура локального Web–сайта состоит из шести блоков. Первый блок подкаталог Ajax в котором происходит обработка авторизации и регистрации с использованием данной технологии. Второй блок подкаталог blocks в котором содержатся в котором содержатся не изменяемые части страницы, размещенные в отдельные файлы для удобства подключения. Третий блок css, содержит в себе файлы данного языка, использовавшиеся для верстки. Четвертый блок fonts, это используемые шрифты. Также подкаталог img хранящий в себе

используемые на Web–сайте изображения. И корневой каталог, содержащий в себе используемые файлы и страницы. На рис. 4 представлена директория локального Web–сайта.

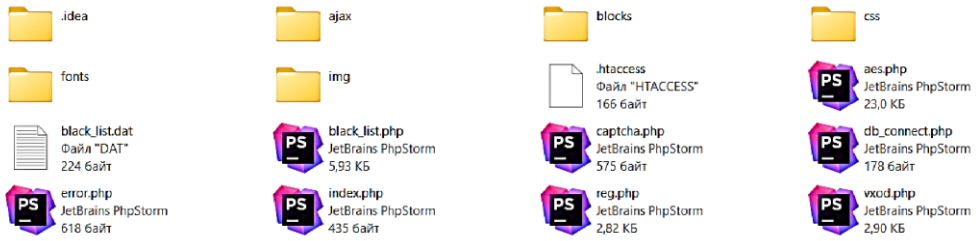

Рис. 4 - Директория Web–сайта
Fig. 4 - Website folder

На рис. 5 представлена структура локального Web–сайта, со всеми файлами во вложенных директориях.

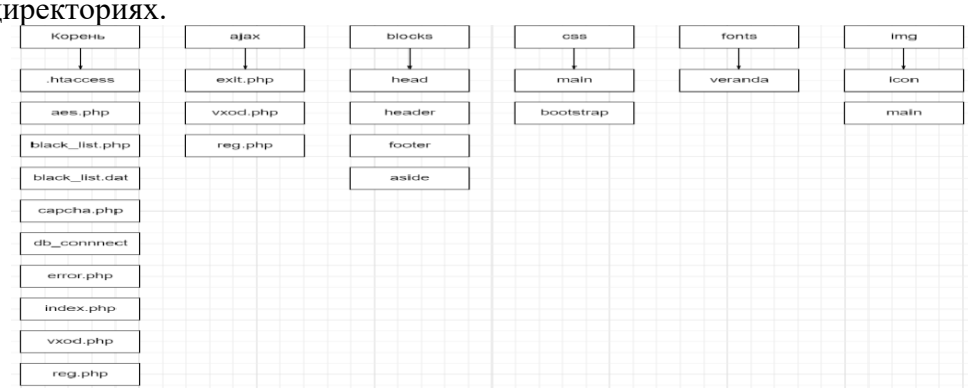

Рис. 5 - Структура Web–сайта
Fig. 5 - Website scturcure

Файл .htaccess обрабатывает ошибки подключения к страницам Web–сайта и хранит в себе заблокированные IP адреса, т.е. подключения с которых был потенциально опасный трафик, а так же IP адреса с которых имеется возможность заниматься администрированием Web–сайта. Для обработки был написан так называемый межсетевой экран, который представлен в файле black_list.php и фильтрует поступающий трафик, т.е. все подозрительные подключения будут записаны в black_list.dat и заблокированы. Таким образом, нагрузка на сервер упадет.

Хранение пароля в базе данных в шифрованном виде реализовано с помощью хеш-функции sha256, что видно на рис. 6. Использование данной хеш-функции является важным аспектом безопасности, так как она является современным и криптостойким решением на сегодняшний день. Сравнивая ее с sha512, по криптостойкости они будут примерно равный, но при этом другие, достаточно известные решения, такие, как md5 и sha1 использовать уже не целесообразно, так как они уже были ранее взломаны.

```
if (strlen($email)<=8)
  $error = 'Введите имейл';

if (strlen($pass) < 8)
  $error = "Password too short!";

if (!preg_match( pattern: "#[0-9]+#", $pass))
  $error = "Password must include at least one number!";

if (!preg_match( pattern: "#[a-zA-Z]+#", $pass))
  $error = "Password must include at least one letter!";

$pass = hash( algo: 'sha256', $pass);
```

Рис. 6 - Обработка пароля
Fig. 6 - Password processing

При начале работы с программой заходим на первичную страницу регистрации пользователя, показанной на рис. 7, где обязательно происходит проверка на надежность пароля и занесение пользовательских данных в хешированном виде непосредственно в БД с помощью SQL запросов.

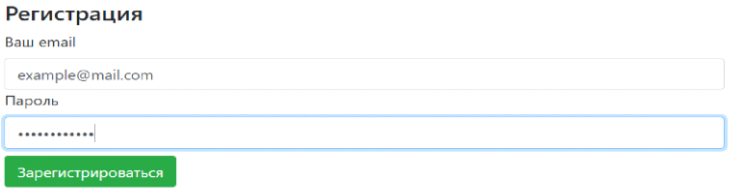

Рис. 7 - Окно регистрации
Fig. 7 - Registration window

После первичной регистрации при выходе из учётной записи появится окно авторизации, где и можно проверить работоспособность механизма защиты. При вводе одного и того же пароля выводит сообщение «неверный логин или пароль», что показано на рис. 8, при этом все POST и GET запросы перед на стадии отправкой клиентом и расшифровывается непосредственно на сервере, посредством симметричного алгоритма шифрования AES.

Рис. 8 - Ошибка авторизации
Fig. 8 - Registration failure

Если попытаться вводить не корректные данные авторизации будет выводиться сообщение и появляться *Captcha*, показано на рис. 9. Такие действия затруднят перебор или подбор пароля, так как некоторые боты не поддерживают ввод капчи либо увеличат время, затрачиваемое на атаку. Блокироваться данная учётная запись будет при 10 ошибках авторизации, при это блокироваться будет на непродолжительное и случайное время.

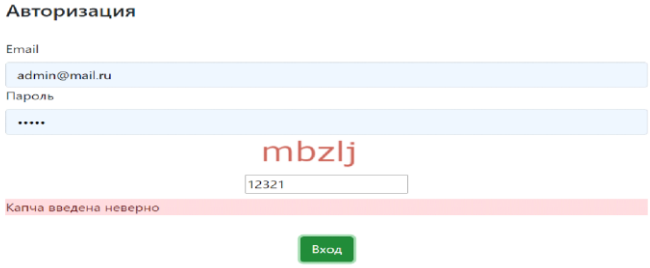

Рис. 9 - Работа Captcha
Fig. 9 - Captcha login prevention

Следующий шаг авторизации пользователя при попытке входа в учётную запись уже с верным паролем. Появится диалоговое окно аутентификации по коду их письма по электронной почте, что является одной из вариации защиты от атаки фишинга. При некорректном вводе, учётная запись блокироваться не будет, но, чтобы защитить конфиденциальные данные пользователя от злоумышленника, после пятого неправильного ввода секретного пароля, появится *Captcha*, которая сильно затруднит хакеру его манипуляции с перебором или подбором символов.

Данный механизм имеет ряд преимуществ по сравнению с аналогами:

– Возможность использовать программное средства бесплатно.

- Имеет перспективу в модификации. Как важный аспект данный механизм можно дорабатывать, улучшать скрипты и добавлять свои методы защиты, чтобы улучшить защиту от различный кибератак.
- Не нагружает систему. Так как все используемые средства встроены в локальный Web–сайт и работают на достаточно быстром языке программирования.
- Не занимает много места, так как весь механизм легко разработчиками может быть встроен в любой Web–сайт.
- Может работать совместно с другими программами, что позволит защищаться от компьютерных атак различного типа.

Вывод. Можно сделать вывод, что механизм защиты использует современные требования к обеспечению безопасности пользовательских данных, успешно справляется с защитой от атак нацеленных на блокировку учётных записей, а также несколько других DOS–атак.

Существует ряд рекомендаций, которым пользователю следует придерживаться, чтобы защитить себя от DOS–атак [11–16]:

1. Уменьшение зон, доступных для атаки. Этого можно добиться, ограничив доступ к портам, протоколам или приложениям, взаимодействие с которыми не предусмотрено, в том числе ограничить прямой интернет-трафик к серверам БД.
2. План масштабирования. В данном пункте необходимо выделить два элемента: пропускная способность и производительность сервера. Пропускная способность (транзитивный потенциал) – необходимо размещать ресурсы в непосредственной близости с конечными пользователями и крупными узлами межсетевого обмена трафиком. Производительность сервера – чем лучше производительность сетевых интерфейсов и сетевая конфигурация, тем лучше будет происходить обработка больших объемов трафика.
3. Сведения о типичном и нетипичном трафике. Сбор сведений о нетипичном и трафике, его анализ и блокировка потенциально опасного.
4. Развертывание брандмауэров для отражения сложных атак уровня приложений. Использование межсетевых экранов позволит противодействовать попыткам внедрения SQL – кода или подделки межсетевых запросов, а так же фильтрация и ограничение трафика и работу сторонних приложений.
5. Использование безопасного подключения HTTPS, как способ защиты от HTTP флуда, а также сторонних механизмов защиты от различных спам и флуд атак, реализующиеся в том числе и ботами.

В результате исследования все поставленные цели были достигнуты: проведено исследование компьютерных атак; произведен анализ существующих механизмов защиты; разработана программная реализация механизма защиты; проведен сравнительный анализ реализованного механизма защиты с аналогами и модификациями.

Тестовая среда, используемая для атаки, не смогла обмануть программное средство. Тестирование прошло успешно, что доказывает о правильности работы расширения и возможности в перспективе модификации её. Проведенные исследования еще раз доказывают, что каждый пользователь должен себя обезопасить от любых несанкционированных угроз. Поэтому были анализированы наиболее распространенные типов кибератак приводящие к блокировке учётных записей, а также были представлены общие рекомендации по защите от кибератаки account lockout. Механизм защиты является современным решением, обеспечивающим и надежную защиту учётных записей пользователей и защищенных от атак на них направленных.

Библиографический список:

1. Razumov P.V., Safaryan O.A., Cherckesova L.V., et all. “Developing of Algorithm of HTTP Flood DDoS Protection”, IEEE 3rd International Conference on Computer Applications & Information Security, IEEE ICCAIS’20. Saudi Arabia, Er-Riyadh, 2020. p.6.

2. Burnett M., Foster J. “Hacking the Code”, ASP.NET Web Application Security, 2004.
3. Бабенко, Л.К. Современные алгоритмы блочного шифрования и методы их анализа: учебное пособие для студентов вузов / Л.К. Бабенко, Е.А. Ищукова. — Москва: Гелиос АРВ, 2006. 376 с.
4. Стариков А.А., Лысенко А.В., Клевцов А.А. Разработка и анализ скорости работы блочного симметричного алгоритма шифрования AES с использованием различных языков программирования, Молодой исследователь Дона, 2022. № 4 (37)
5. Баранова Е.К., Гырнец К.В. Моделирование и анализ безопасности и риска в сложных системах Труды Международной научной школы МАБР. 2016. СПб. ГУАП, 2016. С. 155–161.
6. Панасенко С.П. Алгоритмы шифрования. Спец. справочник. СПб. БХВ–Петербург, 2009. 576 с.
7. Habr. Как устроен AES/habr.com: [сайт]. URL:https://habr.com/ru/post/112733/ (дата обр: 02.05.20 22).
8. Al-Odat Z., Abbas A., Khan S. Randomness Analyses of the Secure Hash Algorithms, SHA-1, SHA-2 and modified SHA, 2019 International Conference on Frontiers of Information Technology (FIT), 2019. Pp. 3160-3165. doi: 10.1109/FIT47737.2019.00066
9. Karthiga S., Velmurugan T. Security based Approach of SHA-384 AND SHA–512. Algorithms in Cloud Environment. J. of Computer Science, 2019. Vol. 16(10) Pp. 1439–1450. DOI:10.3844/jcssp.2020.1439.1450
10. Dong X., Sun S., Shi D. Quantum Collision Attacks on AES-like Hashing with Low Quantum Random Access Memories, Advances in Cryptology. ASIACRYPT–2020, South Korea, Daejeon, Springer International Publishing, Vol. 12492. Pp.727–757 DOI: 10.1007/978-3-030-64834-3
11. Razumov P., Safaryan O., Cherckesova L., et all. IAS’2019 Cryptographic Protocol Allowing to Protect the Key in the Open Communication Channel, 3rd International Conference on Computer Applications & Information Security, 2020. p. 1–5. DOI:10.1109 "http://dx.doi.org/10.1109/ICCAIS48893.2020.9096729"
12. Короченцев Д.А., Черкесова Л.В., Ревякина Е.А. и др. Импортозамещающие технологии обеспечения информационной безопасности и защиты данных: Уч. пособие. Ростов-на-Дону: ДГТУ, 2021 – 335 с.
13. Shui Yu. Distributed Denial of Service Attack and Defense”, Springer Briefs in Computer Science, 2014
14. Özçelik İ., Brooks R. Distributed Denial of Service Attacks, Chapman and Hall / CRC, 2020
15. Ibrahim S.A., Mohammad S., Khader S. Proc. Computer Science. Chapman and Hall/CRC, 2016, Pp. 7–15
16. PHP. Справочная информация / php.net: [сайт]. URL: https://www.php.net/ (дата обр. 01.05.2022)

References:

1. Razumov P.V., Safaryan O.A., Cherckesova L.V., et all. “Developing of Algorithm of HTTP Flood DDoS Protection”, IEEE 3rd International Conference on Computer Applications & Information Security, IEEE ICCAIS’20. Saudi Arabia, Er-Riyadh, 2020. p.6.
2. Burnett M., Foster J. “Hacking the Code”, ASP.NET Web Application Security, 2004.
3. Babenko, L.K. Modern block cipher algorithms and methods of their analysis: a tutorial for university students / L.K. Babenko, E.A. Ishchukova. - Moscow: Helios ARV, 2006. 376 p.
4. Starikov A.A., Lysenko A.V., Klevtsov A.A. Development and analysis of the performance of the block symmetric encryption algorithm AES using various programming languages, Young researcher of the Don, 2022;4 (37)
5. Baranova E.K., Gyrnets K.V. Modeling and analysis of security and risk in complex systems Proceedings of the International Scientific School of the International Academy of Sciences of the Russian Academy of Sciences. 2016. St. Petersburg. GUAP, 2016. Pp. 155–161.
6. Panasenko S.P. Encryption algorithms. Special reference book. St. Petersburg. BHV-Petersburg, 2009:576
7. Habr. How AES works/habr.com: [site]. URL: https://habr.com/ru/post/112733/ (date of receipt: 02.05.20 22).
8. Al-Odat Z., Abbas A., Khan S. Randomness Analyzes of the Secure Hash Algorithms, SHA-1, SHA-2 and modified SHA, 2019 International Conference on Frontiers of Information Technology (FIT), 2019. Pp. 3160-3165. doi: 10.1109/FIT47737.2019.00066
9. Karthiga S., Velmurugan T. Security based Approach of SHA-384 AND SHA–512. Algorithms in Cloud Environment. J. of Computer Science, 2019;16(10):1439–1450. DOI:10.3844/jcssp.2020.1439.1450
10. Dong X., Sun S., Shi D. Quantum Collision Attacks on AES-like Hashing with Low Quantum Random Access Memories, Advances in Cryptology. ASIACRYPT–2020, South Korea, Daejeon, Springer International Publishing, Vol. 12492. Pp.727–757 DOI: 10.1007/978-3-030-64834-3
11. Razumov P., Safaryan O., Cherckesova L., et all. IAS’2019 Cryptographic Protocol Allowing to Protect the Key in the Open Communication Channel, 3rd International Conference on Computer Applications & Information Security, 2020;1–5.DOI:10.1109 HYPERLINK http://dx.doi.org/10.1109/ICCAIS48893.2020.9096729"
12. Korochentsev D.A., Cherkesova L.V., Revyakina E.A. et al. Import-substituting technologies for ensuring information security and data protection: Tutorial. Rostov-on-Don: DSTU, 2021 – 335 p.
13. Shui Yu. Distributed Denial of Service Attack and Defense”, Springer Briefs in Computer Science, 2014
14. Özçelik İ., Brooks R. Distributed Denial of Service Attacks, Chapman and Hall / CRC, 2020
15. Ibrahim S.A., Mohammad S., Khader S. Proc. Computer Science. Chapman and Hall/CRC, 2016, Pp. 7–15
16. PHP. Reference / php.net: [site]. URL: https://www.php.net/ (date obr. 01.05.2022

Сведения об авторах:

Разумов Павел Владимирович аспирант 3 курса, кафедра «Кибербезопасность информационных систем»; razumov1996@inbox.ru; ORCID0000-0003-2454-3600

Черкесова Лариса Владимировна, доктор физико-математических наук, профессор, профессор, кафедра «Кибербезопасность информационных систем»; chia2002@inbox.ru

Ревякина Елена Александровна, кандидат технических наук, доцент, доцент, кафедра «Кибербезопасность информационных систем»; Revyelena@yandex.ru

Клевцов Алексей Алексеевич, студент, кафедра «Кибербезопасность информационных систем»; prazumov96@gmail.com

Information about authors:

Razumov Pavel Vladimirovich, 3rd year Postgraduate student, Department of Cybersecurity of Information Systems; razumov1996@inbox.ru; ORCID0000-0003-2454-3600

Larisa V. Cherkesova, Dr. Sci. (Physics and Mathematics), Prof., Prof., Department «Cybersecurity of information Systems»; chia2002@inbox.ru

Elena A. Revyakina, Cand. Sci. (Physics and Mathematics), Assoc. Prof., Assoc. Prof., Department «Cybersecurity of information systems»; Revye-lena@yandex.ru

Aleksey A. Klevtsov, Student, Department of Cybersecurity of Information Systems; razumov96@gmail.com

Конфликт интересов/Conflict of interest.

Авторы заявляют об отсутствии конфликта интересов/The authors declare no conflict of interest.

Поступила в редакцию/ Received 15.12.2024.

Одобрена после рецензирования/ Reviced 30.01.2025.

Принята в печать/ Accepted for publication 29.04.2025.