Краткое содержание

Спам

Для блокировки спама в PHP по списку запрещенных слов (spam_list.txt), используйте функцию file() для чтения файла и stripos() для проверки наличия запрещенных фраз в данных формы. Скрипт проверяет входящие $_POST данные и прекращает выполнение при обнаружении спама. Реализация блокировки: Создайте файл spam_list.txt с запрещенными словами/фразами, по одной на строку.

spam_list.txt

spam_list.txt

```
dead
fuck
```

block_spam.php

block_spam.php

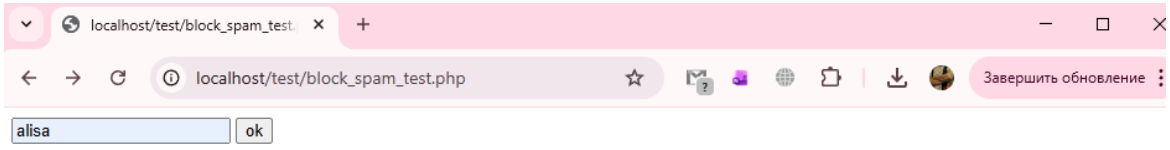
```php
<?php
// Файл со списком спама (каждое слово/фраза с новой строки)
$spam_file = 'spam_list.txt';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Читаем спам-лист в массив
    if (file_exists($spam_file)) {
        $spam_words = file($spam_file, FILE_IGNORE_NEW_LINES |
FILE_SKIP_EMPTY_LINES);
    } else {
        $spam_words = []; // Если файла нет, считаем, что спам-листа нет
    }

    // Собираем данные формы (например, комментарий)
    $comment = $_POST['username'] ?? '';

    // Проверяем данные на наличие спама
    foreach ($spam_words as $word) {
        if (!empty($word) && stripos($comment, $word) !== false) {
            // Если нашли спам, блокируем отправку
            die("Сообщение заблокировано спам-фильтром.");
        }
```

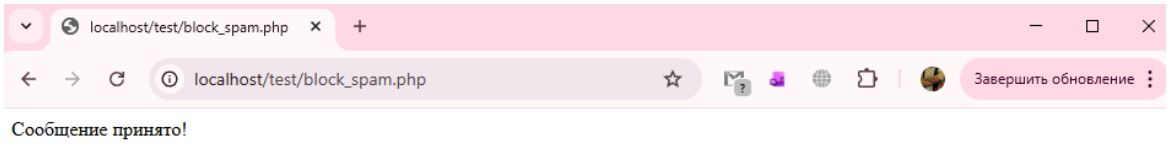
```
    }

    // Если спама нет, обрабатываем форму дальше
    echo "Сообщение принято!";
}
?>
```

block_spam_test.php

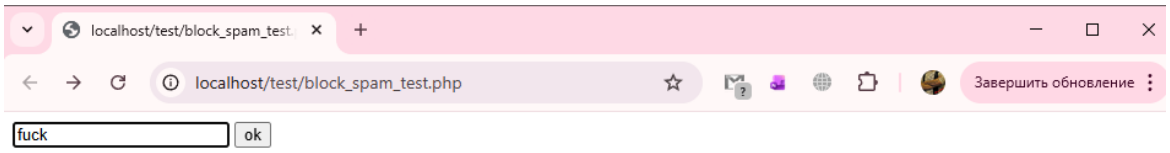
block_spam_test.php

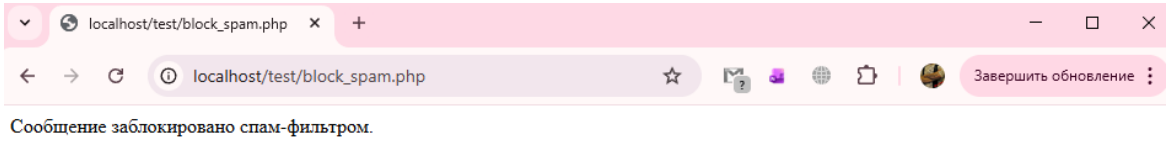
```
<form action="block_spam.php" method="POST">
    <input type="text" name="username">
    <input type="submit" value="ok">
</form>

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Получаем и выводим данные
    $name = $_POST['username'];
    echo "Введенное имя: " . htmlspecialchars($name);
}
?>
```





Ввведем слово из спам листа





User-Agent (Проверка Ботов)

Блокировка ботов в PHP осуществляется путем проверки строки User-Agent в запросе $ _SERVER['HTTP_USER_AGENT'] и прерывания выполнения скрипта, если обнаружен вредоносный робот. Для защиты рекомендуется использовать массив известных ботов, проверять их в начале файла (например, через хук) и использовать die() или exit() для закрытия доступа. Основной метод (PHP) Добавьте этот код в начале ваших PHP-скриптов:

block_bot.php

block_bot.php

```php
<?php
function blockBots() {
    $botListFile = $_SERVER['DOCUMENT_ROOT'] . '/botlist.txt';

    // Проверка существования файла
    if (!file_exists($botListFile)) {
        return;
    }

    // Чтение списка ботов
    $badBots = file($botListFile, FILE_IGNORE_NEW_LINES |
FILE_SKIP_EMPTY_LINES);
    $userAgent = $_SERVER['HTTP_USER_AGENT'] ?? '';

    foreach ($badBots as $bot) {
        // Игнорируем комментарии в файле, если они есть (начинаются с #)
        if (strpos(trim($bot), '#') === 0) continue;

        if (stripos($userAgent, trim($bot)) !== false) {
            // Блокировка: отправляем 403 и завершаем работу
            header('HTTP/1.0 403 Forbidden');
            exit('Access Denied');
        }
    }
}

blockBots();
?>
```

botlist.txt

botlist.txt

```
BadBot
Scanner
SiteCrawler
```

Блокировка по IP

block.php

block_ip.php

```php
<?php
// Путь к файлу черного списка
$blacklistFile = 'blacklist.txt';


// Зададим функцию получения значения IP адреса
function get_ip_list()
{
    $list = array();
    if (!empty($_SERVER['HTTP_CLIENT_IP'])) {
        $ip = explode(',', $_SERVER['HTTP_X_FORWARDED_FOR']);
        $list = array_merge($list, $ip);
    } elseif (!empty($_SERVER['HTTP_X_FORWARDED_FOR'])) {
        $ip = explode(',', $_SERVER['HTTP_X_FORWARDED_FOR']);
        $list = array_merge($list, $ip);
    } elseif (!empty($_SERVER['REMOTE_ADDR'])) {
        $list[] = $_SERVER['REMOTE_ADDR'];
    }
    $list = array_unique($list);
    return implode(',', $list);
}

// Получаем IP посетителя
$visitorIp = get_ip_list();
// Выводим IP посетителя
echo $visitorIp;
echo "<br/>";

// Читаем файл в массив, удаляя переносы строк
if (file_exists($blacklistFile)) {
    $blacklist = file($blacklistFile, FILE_IGNORE_NEW_LINES |
```

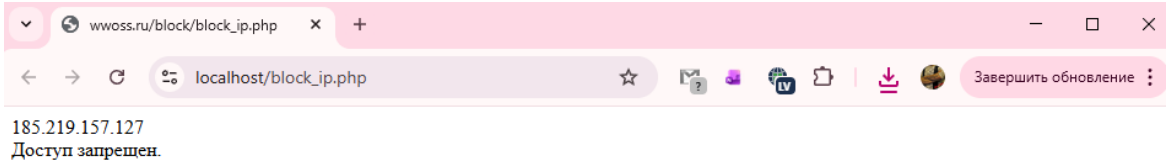
```
FILE_SKIP_EMPTY_LINES);

    // Проверяем, есть ли IP в списке
    // Фильтрация по диапазону
    // 185.219.157.127 // Отдельный IP
    // 185.219.*.127 // Диапазон (все IP, начинающиеся с этого)
    foreach ($blacklist as $blocked_ip) {
    // Преобразуем маску с * в регулярное выражение
    $pattern = '/^' . str_replace(['.', '*'], ['\.', '.*'],
$blocked_ip) . '$/';

    if (preg_match($pattern, $visitorIp)) {
        header('HTTP/1.0 403 Forbidden');
        die('Доступ запрещен.');
    }
    }
}
// ... остальной код сайта
echo "Добро пожаловать!";
?>
```

blacklist.txt

blacklist.txt

```
192.168.1.1
10.0.0.5
172.16.0.100
185.219.157.127
```

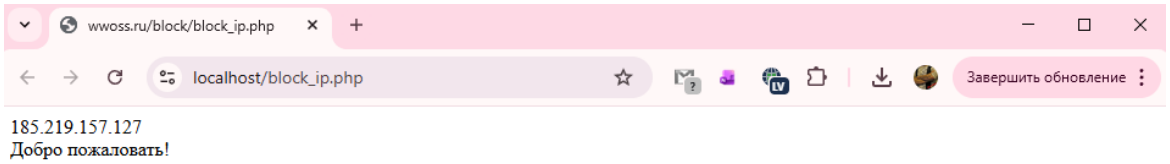
Проверяем браузер



blacklist.txt

```
192.168.1.1
10.0.0.5
172.16.0.100
185.219.157.128
```

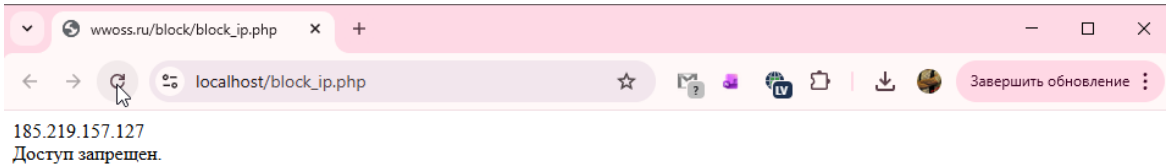
Проверяем браузер



blacklist.txt

```
192.168.1.1
10.0.0.5
172.16.0.100
185.219.157.*
```

Проверяем браузер



Дополнения и Файлы

- blacklist.txt
  - ipsum.txt
  - botlist.txt
- spammers.txt
- deny-ip-list.txt
- allow-ip-list.txt