

УДК 004.056.5

Н. А. КИРПИЧЕНКО, Н. С. ЛУКАШОВ (ЭМ-21)

Научные руководители: старшие преподаватели *В. Е. МИНИН,*
С. С. ТАТУР

СКАНИРОВАНИЕ ВЕБ-КОНТЕНТА КАК ИНСТРУМЕНТ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ: МЕТОДЫ, ИНСТРУМЕНТЫ И ЗАЩИТА

Рассматривается проблема автоматизированного сканирования веб-контента, используемого как для легитимных задач (SEO-аудит, индексация, мониторинг), так и для целей пентеста. Особое внимание уделяется методу принудительного перебора директорий и файлов, который позволяет выявлять скрытые ресурсы, резервные копии, конфигурационные файлы и потенциальные уязвимости. Проанализированы популярные инструменты перебора: Dirb, Gobuster, Dirsearch, fluf. Предложены методы защиты от несанкционированного сканирования, включая использование неочевидных имен директорий, авторизацию, ограничение частоты запросов и применение WAF.

В условиях цифровой трансформации экономики и перехода государственных и коммерческих организаций на электронные формы взаимодействия информационная безопасность веб-ресурсов приобретает критическое значение. Сканерами веб-контента называются автоматизированные программы или сервисы, которые обходят веб-сайты, анализируют HTML-код, структуру ссылок, мета-теги, изображения и, что особенно важно для безопасности, HTTP-коды ответа на запросы к различным URL. По сути сканер веб-контента – автоматический исследователь, который методично проверяет тысячи возможных адресов на сайте и сообщает, что находится за каждым из них. Сканирование веб-контента необходимо для множества задач, выходящих далеко за рамки простого сбора информации. Поисковые системы используют сканеры для индексации страниц, обнаружения битых ссылок и анализа дублей контента. Бизнес применяет сканирование для мониторинга конкурентов – отслеживания цен, новых товаров, появления статей и упоминаний бренда. В сфере безопасности сканеры ищут уязвимости вроде XSS и SQL-инъекций, выявляют скрытые директории и файлы, например, `phpinfo.php`, `backup.sql`, `.git/HEAD` или `.env`, которые могут привести к утечке данных.

При проведении пентеста особое значение приобретает техника принудительного перебора директорий и файлов [1]. Суть ее проста: сканер получает на вход словарь возможных имен директорий и файлов, например, `admin`, `backup`, `config`, `js`, `phpinfo`, `.git`, `backup.sql`. Затем он по очереди подставляет каждое слово из словаря к базовому URL сайта и внимательно

анализирует HTTP-код ответа. Код 200 означает, что ресурс существует и доступен – это прямая цель для дальнейшего изучения. Код 403 говорит о том, что директория существует, но доступ запрещен, и это тоже ценная информация, потому что иногда за запретом скрываются важные данные. Код 301 или 302 указывает на редирект, который часто ведет на страницу входа или на главную страницу. Код 404 обычно означает отсутствие ресурса, хотя некоторые сайты могут подменять ответы, чтобы запутать сканер. Без автоматического сканера перебрать сотни тысяч возможных путей вручную невозможно, и именно поэтому перебор URL считается необходимым инструментом при любом серьезном пентесте. На реальных проектах пентестеры регулярно находят с помощью таких сканеров `phpinfo.php` с выдачей переменных окружения, `backup.sql` с дампом базы данных, служебные каталоги `.git` с утечкой исходного кода, файлы `.env` и `.aws/credentials` с секретами и ключами доступа, а также страницы административных панелей вроде `adminer.php` с пустым паролем. Все эти находки становятся возможными только благодаря автоматизированному перебору.

Существует несколько популярных инструментов для перебора директорий и файлов, каждый из которых имеет свои сильные стороны и сценарии использования. Для наглядности основные характеристики сведены в таблицу 1.

Таблица 1 – Сравнительная характеристика инструментов перебора директорий

Инструмент	Язык программирования	Многопоточность	Ключевая особенность	Рекомендуемый сценарий
Dirb	C	Нет (однопоточный)	Высокая стабильность, встроенный словарь	Совместимость со старыми системами
Gobuster	Go	Да	Очень высокая скорость	Быстрая проверка больших словарей
Dirsearch	Python	Да	Гибкая поддержка расширений файлов	Работа с разными типами файлов (.php, .asp, .bak)
ffuf	Go	Да	Мощные фильтры (по размеру, количеству слов, регулярным выражениям)	Тонкая настройка фильтрации ложных срабатываний

Как видно из таблицы 1, выбор конкретного инструмента зависит от поставленной задачи. Dirb, написанный на языке C, является классическим однопоточным сканером со встроенным словарем. Он работает медленно, но очень стабильно, поэтому является хорошим выбором в условиях, где важна предсказуемость, а не скорость. Gobuster [2], написанный на Go, является многопоточным, что обеспечивает ему очень высокую скорость работы. Он идеально подходит для проверки больших словарей, когда нужно перебрать сотни тысяч вариантов за разумное время. Dirsearch [4], созданный на

Python, отличается гибкой поддержкой расширений: он может одновременно перебирать файлы с разными окончаниями, например. `.php`, `.asp`, `.jsp`, `.bak`, `.old`, `.txt`, что удобно, когда вы не знаете точно, на каком движке работает сайт. Наконец, `ffuf` [3], также написанный на Go, берет не скоростью, а мощными фильтрами по размеру ответа, количеству слов или регулярному выражению. Например, команда `ffuf` с параметрами `fs` (исключить по размеру), `fw` (исключить по количеству слов) и `fc` (исключить по коду ответа) позволяет отсечь практически все ложные срабатывания, что особенно полезно на сайтах, где на любой несуществующий путь сервер возвращает код 200 и одинаковую страницу-заглушку.

Помимо этих трех основных методов существуют дополнительные способы защиты. Можно настроить веб-сервер так, чтобы на любой несуществующий путь возвращался код 404, но при этом сама страница ошибки была того же размера и содержала тот же текст, что и страница, возвращаемая на реальные существующие пути. Это лишает сканер возможности фильтровать ответы по размеру.

Категорически не рекомендуется оставлять на сервере временные или резервные файлы с расширениями `.bak`, `.swp`, `.old` или с символом тильды в конце имени файла. Такие файлы часто создаются текстовыми редакторами, и их наличие на боевом сервере является критической уязвимостью. Наконец, можно использовать межсетевой экран веб-приложений (WAF), например, `ModSecurity` [5], настроив его на блокировку запросов с подозрительными `User-Agent`, которые по умолчанию используют `Dirb`, `Gobuster`, `Dirsearch` и `ffuf`.

При использовании сканеров веб-контента, особенно в рамках пентеста, который проводится с согласия заказчика и в соответствии с договором, необходимо соблюдать определенные этические и технические правила. Следует проверять файл `robots.txt` и уважать описанные в нем запреты. Хотя злоумышленники его игнорируют, пентестер, действующий легитимно, должен его учитывать. Нужно добавлять искусственные задержки между запросами – паузы в несколько десятков или сотен миллисекунд, чтобы не перегружать сервер заказчика и не вызывать отказ в обслуживании. В заголовке `User-Agent` рекомендуется указывать реального бота и контактную информацию, например, название компании или ссылку на политику сканирования. Категорически не стоит сканировать страницы с формами поиска или фильтрации, которые могут генерировать миллиарды комбинаций параметров. Это может привести к эффекту отказа в обслуживании даже при легитимном сканировании.

Сканеры веб-контента – это не просто удобные инструменты для SEO и парсинга, а обязательный компонент любого серьезного пентеста и аудита безопасности.

Классический метод «словарь URL плюс анализ HTTP-кодов» позволяет находить критические утечки данных, такие как резервные копии баз данных, конфигурационные файлы с секретами, служебные каталоги систем

контроля версий и страницы phpinfo. Инструменты вроде Dirb, Gobuster, Dirsearch и ffuf различаются по скорости, гибкости и возможностям фильтрации, и выбор конкретного инструмента, как показано в таблице 1, зависит от поставленной задачи.

Защита от таких сканеров возможна через использование случайных имен директорий, обязательную авторизацию, ограничение частоты запросов, настройку однородных ответов 404 и применение Web Application Firewall. Без автоматизации перебор десятков тысяч возможных путей вручную невозможен, поэтому сканеры остаются необходимыми как для атакующего, так и для защищающегося.

СПИСОК ЛИТЕРАТУРЫ

1 OWASP Foundation. Руководство по тестированию OWASP (v4.2): Принудительный просмотр / OWASP Foundation. – 2020. – URL: <https://owasp.org/www-project-web-security-testing-guide> (дата обращения: 27.05.2026).

2 **Ривз, О. Дж.** Gobuster: Инструмент для перебора директорий/файлов и DNS, написанный на Go / О. Дж. Ривз. – 2017. – URL: <https://github.com/OJ/gobuster> (дата обращения: 27.05.2026).

3 **Лэнгли, А.** ffuf: Быстрый фаззер для веб-приложений, написанный на Go / А. Лэнгли. – 2019. – URL: <https://github.com/ffuf/ffuf> (дата обращения: 27.05.2026).

4 **Сория М.** Dirsearch: Сканер веб-путей / М. Сория. – 2016. – URL: **Ошибка! Недопустимый объект гиперссылки.** (дата обращения: 27.05.2026).

5 Trustwave Holdings. ModSecurity: Открытый межсетевой экран веб-приложений / Trustwave Holdings. – 2023. – URL: <https://modsecurity.org> (дата обращения: 27.05.2026).

Получено 28.05.2026

ISSN 2227-1155. Сборник студенческих научных работ.

Вып. 31. Гомель, 2026

УДК 656.2.025.2(476.2)

Е. А. КЛИМЕНКО (УД-41)

Научный руководитель – ст. преп. *И. М. ЛИТВИНОВА*

ОЦЕНКА ВОЗМОЖНОСТИ ОСВОЕНИЯ ПЕРСПЕКТИВНЫХ ПАССАЖИРОПОТОКОВ СУЩЕСТВУЮЩИМИ РАЗМЕРАМИ ДВИЖЕНИЯ ПАССАЖИРСКИХ ПОЕЗДОВ В РЕГИОНАЛЬНОМ СООБЩЕНИИ

Освоение перспективных пассажиропотоков существующими размерами движения пассажирских поездов является важным вектором повышения эффективности пассажирских перевозок особенно в тех сообщениях, где тариф на перевозку не по-