МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»
(СГУ)

Кафедра теоретических основ компьютерной безопасности и криптографии

Выявление уязвимостей Интернет-ресурсов с использованием поисковых систем

АВТОРЕФЕРАТ
дипломной работы

студентки 6 курса 631 группы
специальности 10.05.01 Компьютерная безопасность
факультета компьютерных наук и информационных технологий
Гусевой Ксении Олеговны

Научный руководитель
доцент _____ И. Ю. Юрин
18.01.2019 г.

Заведующий кафедрой
д. ф.-м. н., доцент _____ М. Б. Абросимов
18.01.2019 г.

Саратов 2019

ВВЕДЕНИЕ

Доступ в Интернет получают все больше людей. Каждый хочет создать что-то свое и количество общедоступной информации растет. Появляется все больше инструкций и бесплатных программ для выявления уязвимостей Интернет-ресурсов и организации атаки на них. Сегодня серьезный ущерб может причинить человек, который не обладает специальными знаниями. Специалисты информационной безопасности как никогда должны быть на шаг впереди.

Поисковые роботы фиксируют то, что злоумышленники могут использовать в своих целях при атаке на Интернет-ресурс. К такой информации относятся ошибки скриптов, файлы с конфигурациями, логами, данными аутентификации, а также резервные копии баз данных. [1]

Цель работы состоит в том, чтобы упростить работу специалистам информационной безопасности по своевременному выявлению уязвимостей Интернет-ресурсов. Это поможет предотвратить нанесение ущерба юридическим и физическим лицам.

Задача работы — рассмотреть способ выявления уязвимостей Интернет-ресурсов с помощью поисковых систем и создать приложение для автоматизации этого процесса. Для этого разберем основной синтаксис запросов, который применяется в поисковых системах Яндекс, Google, Bing, чтобы потом автоматизировать выявление уязвимостей с помощью этих систем. Рассмотрим примеры применения дорков, взяв несколько из самой популярной и обширной базы их хранения Exploit-DB [2].

Дипломная работа состоит из введения, 5 разделов, заключения, списка использованных источников и 5 приложений. Общий объем работы — 75 страниц, из них 33 страницы — основное содержание, включая 31 рисунок и 3 таблицы, список использованных источников из 21 наименований.

КРАТКОЕ СОДЕРЖАНИЕ

1. Доркинг — история и описание

Доркинг (Dorking, чаще GoogleDorking) — это способ формирования поисковых запросов для поисковых систем, который позволяет выявлять уязвимости веб-сервисов, проникать на разделы, которые не доступны по умолчанию, или доступ к этим разделам не был предусмотрен вообще. Такие запросы называются дорками. [3]

Доркинг не является высоко технологичным способом атаки, поскольку для него нужен лишь компьютер с подключением к Интернету и знание соответствующего синтаксиса поиска для определенной поисковой системы.

Доркинг впервые был обнаружен и применен в поисковике Google, поэтому часто доркинг называется GoogleDorking. Синтаксис подобных поисковых запросов можно частично или полностью использовать и в других поисковых системах.

Вместо обычного поискового запроса, который фокусируется на семантическом способе задавать вопросы, то есть непосредственного написания всего вопроса или выбранных ключевых слов, доркинг основан на знаниях того, как поисковые системы сканируют и индексируют Интернет-ресурсы. Такой доступ может привести к обнаружению информации, которая может быть использована для мошенничества или терроризма, поиска личной информации о человеке или каком-либо учреждении, а также информации, которая помогает в расследовании правительств, корпораций или влиятельных лиц.

Подробности о том, какие запросы работают в каких поисковых системах, представлены в таблице Б.1 в Приложении Б [3, 4]. Проанализировав таблицу можно увидеть, что Google дает самые широкие возможности, однако у поисковой системы Яндекс есть много уникальных возможностей, которых нет в других системах, например, поиск атрибутов HTML-тегов и CSS-стилей.

2. Синтаксис дорков и базы их поиска

Дорки чаще всего направлены на использование в Google не только из-за исторического первенства применения рассматриваемого метода. У этой

системы самый гибкий синтаксис ключевых слов (список в таблице 1) и специальных символов (список в таблице 2) с точки зрения поиска уязвимостей.

Считается, что индекс Google более полный, чем у других поисковых систем. Однако именно за счет разных механизмов индексации имеет смысл проверять дорк в разных поисковых системах, поддерживающих его. Разница индексации позволит выявить дополнительные ресурсы, о которых мы не узнали бы, если бы ограничивались одной системой.

Огромная база дорков представлена на онлайн-сервисе Exploit-DB — это некоммерческий проект Offensive Security, компании по обучению информационной безопасности. Дорки располагаются в разделе сайта «Google Hacking Database». Все дорки Google Hack Database разделены на 14 категорий и подробности о них представлены в таблице 3.

Google Hack Database не единственная база, существует много других агрегаторов дорков. Можно, например, выделить ресурс google-dorking.com, на котором чаще всего встречаются новые дорки, которые не фиксируются в базе Exploit-DB.

3. Примеры использования дорков

Перед формированием нового дорка, нужно определить цель, а затем подобрать слова, характерные для цели. Например, можно отобрать адрес расположения, название и расширение файлов с логами, паролями или даже файлов реестра. Чем точнее мы выявим уникальную информацию, направленную на цель, тем более точный результат мы получим.

В работе представлены примеры использования дорков для получения списков файлов сайтов, сделанных на движке WordPress, получение активной ссылки для скачивания резервной копии сайта, названий колонок базы данных, имен пользователей и другое.

Представлен один из многочисленных способов получения паролей. Показаны варианты получения доступа к файлам сетевых хранилищ, подразумевающие закрытый доступ, и к хранилищам на тера- и петабайты данных. Рассмотрели уязвимости сетевых устройств с управлением через веб-интерфейс, в частности видеокамер, и получили доступ к видеоархиву за

несколько лет, показывая возможности уточнения дорков и получения более точных поисковых результатов.

Это только всего лишь несколько примеров, которые все же показывают, что доркинг — очень гибкий инструмент выявления уязвимостей широчайшего спектра.

4. Существующие решения автоматизации выявления уязвимостей Интернет-ресурсов

Помимо использования привычных поисковых систем для поиска уязвимостей специальными запросами, были разработаны специальные поисковые системы, которые предлагают широкий поиск уязвимостей как свой основной функционал.

Одной из таких является Shodan — поисковая система, которая позволяет найти устройства с открытым доступом, которые подключаются к Интернету. Shodan был создан в 2009 году John-ом Matherly и на данный момент — это самая большая база уязвимых устройств.

Ресурс платный и цена делает возможности Shodan недоступными для большинства пользователей, что уменьшает ущерб, который возможно нанести, используя информацию данной поисковой системы.

Есть разработчики, которые работают напрямую с привычными обычному пользователю поисковыми системами и знакомы с дорками. Рассмотрим пример одной утилиты, в которой разработчики автоматизировали выгрузку дорков из Google Hack Database и взаимодействие дорков с Интернет-ресурсами.

Утилита dorks представляет из себя скрипт, написанный на Nodejs, который интерпретируется phantomjs — полноценным веб-браузером без графического интерфейса, управляемым с помощью js-кода с удобным API.

Утилита поддерживает разные режимы поиска:

- 1 дорк и 1 сайт;
- 1 дорк и много сайтов;
- 1 сайт и много дорков;
- много сайтов и много дорков.

Инструмент доступен в виде исходного кода [7] и для своей работы требует только phantomjs. [1]

5. Практическая часть

В ходе работы была создана программа, которая автоматизирует получение информации об Интернет-ресурсах через поисковые системы и обрабатывает полученную информацию.

Приложение предназначено для специалистов информационной безопасности для упрощения и автоматизации чтения результатов поисковой выдачи в целях мониторинга уязвимостей Интернет-ресурсов.

Программа позволяет выбрать поисковую систему, количество поисковых результатов, которое необходимо обработать, и запрос, который будет передан поисковой системе. Результат выводится в удобную для чтения основных данных таблицу. Поисковая выдача отсортирована по домену в алфавитном порядке.

При обнаружении нескольких ссылок, которые ведут на одну страницу, выводится одна, тем самым экономя время специалиста. Также подсвечиваются красным страницы, которые невозможно открыть – получается специалисту не нужно будет тратить время на просмотр заведомо бесполезных страниц. Хотя даже из описания страницы можно узнать об информации, которая там была и, возможно, будет выявлен незаконный контент.

Результат сохраняется по статической ссылке, у каждого результата подписывается время его получения. Это позволяет отслеживать изменение уязвимостей: на какой момент времени какие из них открыты или закрыты, подробнее об этом ниже.

Для написания программы выбран язык программирования Go (Golang) — компилируемый многопоточный язык программирования. Удобен для http-запросов, прост для изучения, имеет большое количество библиотек и встроенный веб-сервер. [8]

Приложение написано на языке программирования Go версии 1.11.4. Реализовано в виде веб-приложения и состоит из нескольких пакетов.

Поскольку разные поисковые системы индексируют Интернет-ресурсы по-разному, было принято решение объединить в одной программе API трех поисковых систем: Яндекс, Google и Bing. Таким образом пользователь из одного интерфейса получает доступ к трем поисковым системам.

Для отправления запроса пользователю необходимо зарегистрировать ключи соответствующей поисковой системы, поэтому было реализовано по странице на каждую систему. На этих страницах пользователь вводит соответствующие ключи каждой системе ключи и другие настройки, необходимые для работы программы.

Реализован механизм сохранения последнего введенного ключа пользователя, который запустил программу со своего компьютера. Чтобы при каждом запуске программы не нужно было заново вводить ключи, они сохраняются в сессии на сервере. При повторном подключении на сервер посылается ключ, который расшифровывает сохраненные ключи и подставляет их в соответствующие поля.

Также для удобства пользователя добавлена страница «Руководства». В ней представлены ссылки, которые ведут на страницы описания процесса регистраций всех ключей, необходимых для взаимодействия приложения с поисковыми системами — на ресурсы [10, 11, 12].

Для запуска программы необходимо ввести запрос и количество поисковых страниц в соответствии с рисунком, из расчета того, что на одной страницу располагается 10 поисковых выдачей.

Из-за специфики рассматриваемых нами запросов, в программе было реализовано экранирование служебных знаков — кавычек, двоеточий и т.п. После нажатия пользователем кнопки «Search» программа показывает счетчик обработанных запросов. После всех операций пользователю будет показан результат поиска.

Результаты помещаются по адресу вида /view/<идентификатор> — это статическая ссылка, по которой можно повторно открыть результаты без процесса обработки, где идентификатор – последовательность из символов английского алфавита и цифр в случайном порядке длиной 64 символа.

Пользователь может сохранить ссылку на результат работы и продолжить анализ позже.

Это нужно, например, для обработки большого количества страниц или для осуществления анализа закрытых уязвимостей — у каждого результата подписывается время, когда он был получен. Интернет-ресурсы индексируются поисковыми системами с разной частотой, поэтому может возникнуть ситуация, когда обнаруженная специалистом уязвимость закрывается администратором ресурса и после следующей индексации, ресурс в уже не попадет в поисковую выдачу по выбранному дорку. Или же наоборот, откроется новая уязвимость, и появится ресурс, которого раньше не было в списке. Или же изменится статус доступности ресурса и он изменит цвет в списке результатов.

Разные поисковые системы показали разные страницы и разное количество поисковых результатов после процесса группировки. Поэтому пользователь приложения получит более полные результаты, чем если бы он пользовался единственной поисковой системой напрямую.

Время работы программы измеряется в секундах и прогнозируется заранее: по одной секунде на одну поисковую выдачу плюс пять секунд на обработку результатов.

Для создания интерфейса были использованы стандартные стили Bootstrap [13], которые сэкономили время на разметку, дали базовые адаптивные страницы и приятную визуальную составляющую.

В процессе интеграции с API поисковых систем пришлось столкнуться с множеством проблем, которые были решены. Основные выводы описаны в приложениях В-Д. Они содержат важные настройки, без понимания которых не получится обеспечить правильную работу приложения.

ЗАКЛЮЧЕНИЕ

Поиск уязвимостей Интернет-ресурсов осуществляется множеством способов. Одним из самых гибких является доркинг. Этот способ не требует высокотехнологичного обеспечения и специального образования, чтобы разобраться и использовать его. Достаточно хорошо понимать и знать особенности работы, расположения и семантики цели.

К тому же область применения дорков очень велика. Можно с помощью специальных запросов найти очень много уязвимостей, хотя инструмент по сути один и тот же. В этом заключается уникальность и широкие возможности дорков.

Более того, разные поисковые системы по-разному организуют процесс индексации ресурсов, имеют разные ключевые слова и базы данных. Поэтому было принято решение объединить мощности нескольких поисковых API в одной программе.

Было реализовано приложение на языке программирования Go, которое помогает специалисту компьютерной безопасности автоматизировать выявление и мониторинг уязвимостей Интернет-ресурсов с помощью поисковых систем. Программа работает с API Яндекс, Google и Bing.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Хабрахабр. Уральский Центр Систем Безопасности. Ищем уязвимости с помощью google. [Электронный ресурс] — URL: https://habr.com/post/283210/ (дата обращения 10.09.2018) Загл. с экрана. Яз. рус.

2. Exploit Database By Offensive Security. Google Hacking Database [Электронный ресурс] — URL: https://www.exploit-db.com/google-hacking-database (дата обращения 13.09.2018) Загл. с экрана. Яз. англ.

3. Exposing The Invisible. Smart searching with googleDorking. [Электронный ресурс] — URL: https://exposingtheinvisible.org/guides/google-dorking/ (дата обращения 23.09.2018) Загл. с экрана. Яз. англ.

4. Пронин Илья. Язык поисковых запросов. [Электронный ресурс] — URL: http://ilyapronin.ru/prodvizhenie/operatory-poiskovykh-sistem.html (дата обращения 10.12.2018) Загл. с экрана. Яз. рус.

5. SPY-SOFT.NET. In\/ernO. Что такое Google Dorks? [Электронный ресурс] — URL: http://www.spy-soft.net/gugl-dorki/ (дата обращения 10.10.2018) Загл. с экрана. Яз. рус.

6. Информационный портал Make Info. Shodan — поисковая система для хакера | Проверка на уязвимости. [Электронный ресурс] — URL: https://www.make-info.com/shodan/ (дата обращения 21.10.2018) Загл. с экрана. Яз. рус.

7. Github. USSCltd. dorks. google hack database automation tool. [Электронный ресурс] — URL: github.com/USSCltd/dorks (дата обращения 11.09.2018) Загл. с экрана. Яз. англ.

8. Донован, А. А. А., Керниган, Б. У. Язык программирования Go. Серия «Программирование для профессионалов» / А. А. А. Донован, Б. У. Керниган. Пер. с англ. — М. : ООО «И.Д. Вильямс» 2016. — 432 с.

9. Github. valyala. quicktemplate. [Электронный ресурс] — URL: https://github.com/valyala/quicktemplate (дата обращения 15.10.2018) Загл. с экрана. Яз. англ.

10. Yandex.XML Developer`s guide [Электронный ресурс] — URL: https://tech.yandex.com/xml/doc/dg/concepts/about-docpage/ (дата обращения 05.10.2018) Загл. с экрана. Яз. англ.

11. Google Custom Search. Custom Search JSON API [Электронный ресурс] — URL: https://developers.google.com/custom-search/v1/overview (дата обращения 05.11.2018) Загл. с экрана. Яз. англ.

12. Bing Web Search [Электронный ресурс] — URL: https://azure.microsoft.com/en-us/services/cognitive-services/bing-web-search-api/ (дата обращения 06.10.2018) Загл. с экрана. Яз. англ.

13. Bootstrap [Электронный ресурс] — URL: https://getbootstrap.com/ (дата обращения 16.10.2018) Загл. с экрана. Яз. англ.

14. Яндекс.XML [Электронный ресурс] — URL: https://tech.yandex.ru/xml (дата обращения 11.10.2018) Загл. с экрана. Яз. рус.

15. Яндекс.XML. Настройка. [Электронный ресурс] — URL: https://xml.yandex.ru/settings (дата обращения 11.10.2018) Загл. с экрана. Яз. рус.

16. Getting started with Custom Search. Что такое Система пользовательского поиска Google? [Электронный ресурс] — URL: https://support.google.com/customsearch/answer/4513751?hl=ru&ref_topic=4513742# (дата обращения 12.10.2018) Загл. с экрана. Яз. рус.

17. Google Custom Search. Creating a Custom Search Engine [Электронный ресурс] — URL: https://developers.google.com/custom-search/docs/tutorial/creatingcse (дата обращения 12.10.2018) Загл. с экрана. Яз. англ.

18. Google. Пользовательский поиск. [Электронный ресурс] — URL: https://cse.google.com/cse/all (дата обращения 12.10.2018) Загл. с экрана. Яз. рус.

19. Microsoft Azure Marketplace. [Электронный ресурс] — URL: https://portal.azure.com/#blade/Microsoft_Azure_Marketplace/GalleryFeaturedMenuItemBlade/selectedMenuItemId/home/searchQuery/bing/resetMenuId (дата обращения 14.10.2018) Загл. с экрана. Яз. рус.

20. Bing Web Search API Documentation [Электронный ресурс] — URL: https://docs.microsoft.com/en-us/azure/cognitive-services/bing-web-search/ (дата обращения 14.10.2018) Загл. с экрана. Яз. англ.

21. Quickstart: Search the web using the Bing Web Search REST API and Go [Электронный ресурс] — URL: https://docs.microsoft.com/en-us/azure/cognitive-services/bing-web-search/quickstarts/go (дата обращения 14.10.2018) Загл. с экрана. Яз. англ.