Технологии обработки информации

Information processing technologies

УДК 004.67

РАСШИРЕНИЕ БАЗОВОГО ФУНКЦИОНАЛА ПРОГРАММЫ *MALTEGO* НА БАЗЕ ФРЕЙМВОРКА *CANARI* И ПОИСКОВОЙ СИСТЕМЫ *SHODAN*

Ш. Р. ДАВЛАТОВ[1)], ***П. В. КУЧИНСКИЙ***[1), 2)]

[1)]*Белорусский государственный университет информатики и радиоэлектроники, ул. П. Бровки, 6, 220013, г. Минск, Беларусь*
[2)]*Институт прикладных физических проблем им. А. Н. Севченко БГУ, ул. Курчатова, 7, 220045, г. Минск, Беларусь*

В работе рассматривается метапоисковая система *Maltego*, которая широко применяется для сбора данных из открытых источников и автоматического построения связей между различными объектами исследования. Изучаются основные характеристики и алгоритм работы поисковой системы *Shodan*, а также показано принципиальное отличие данной системы от традиционных поисковых движков. Платформа *Shodan* индексирует информацию, которая собирается из ответных баннеров устройств, подключенных к сети Интернет, тогда как *Google*, *Яндекс* и им подобные сервисы индексируют только контент веб-сайтов. На основе изученных материалов разработано расширение базового функционала *Maltego* с помощью фреймворка *Canari* на основе языка программирования *Python*. Данный подход позволяет объединить основные преимущества рассмотренных систем: богатый набор

Образец цитирования:
Давлатов ШР, Кучинский ПВ. Расширение базового функционала программы *Maltego* на базе фреймворка *Canari* и поисковой системы *Shodan*. *Журнал Белорусского государственного университета. Физика.* 2020;1:34–40.
https://doi.org/10.33581/2520-2243-2020-1-34-40

For citation:
Davlatov ShR, Kuchynski PV. Extending the basic functionality of *Maltego* based on the *Canari* framework and *Shodan* search engine. *Journal of the Belarusian State University. Physics.* 2020;1:34–40. Russian.
https://doi.org/10.33581/2520-2243-2020-1-34-40

Авторы:
Шохрух Рустамович Давлатов – аспирант кафедры защиты информации факультета инфокоммуникаций. Научный руководитель – П. В. Кучинский.
Петр Васильевич Кучинский – доктор физико-математических наук; профессор кафедры защиты информации факультета инфокоммуникаций[1)], директор[2)].

Authors:
Shohrukh R. Davlatov, postgraduate student at the department of information security, faculty of infocommunications.
shohrukh.92@gmail.com
Petr V. Kuchynski, doctor of science (physics and mathematics); professor at the department of information security, faculty of infocommunications[a], and director[b].
kuchynski@bsu.by

графических инструментов *Maltego* и большую базу открытых данных системы *Shodan*. Предлагаемый вариант расширения функционала *Maltego* дает возможность настроить систему под любые уникальные требования, необходимые специалистам по информационной безопасности для проведения аудита защищенности информационных систем.

Ключевые слова: *Shodan*; *Maltego*; *Canari* framework; сбор и анализ данных; уязвимость программного обеспечения; информационная безопасность.

EXTENDING THE BASIC FUNCTIONALITY OF *MALTEGO* BASED ON THE *CANARI* FRAMEWORK AND *SHODAN* SEARCH ENGINE

Sh. R. DAVLATOV [a], ***P. V. KUCHYNSKI*** [a, b]

[a] *Belarusian State University of Informatics and Radioelectronics,*
6 P. Broŭki Street, Minsk 220013, Belarus
[b] *A. N. Sevchenko Institute of Applied Physical Problems, Belarusian State University,*
7 Kurčatava Street, Minsk 220045, Belarus

Corresponding author: Sh. R. Davlatov (shohrukh.92@gmail.com)

The paper considers *Maltego* metasearch system, which is widely used for collecting data from open sources and automatically building relationships between various objects. The main characteristics and the algorithm of the *Shodan* search engine was studied, and also the fundamental difference between this system and traditional search engines was explained. The *Shodan* platform indexes information that is collected from response banners of devices which are connected to the internet, while *Google*, *Yandex* and similar services index only the content of websites. Based on the studied materials, an extension of the *Maltego* functionality was developed using the *Canari* framework and the *Python* programming language. This approach allows to combine the main advantages of the considered systems: a rich set of graphical tools of *Maltego* and a large open database of the *Shodan* system. The proposed option also allows to configure the system to fit any unique requirements that information security specialists need to conduct a security audit of information systems.

Keywords: *Shodan*; *Maltego*; *Canari* framework; data collection and analysis; software vulnerability; information security.

Введение

Процесс защиты информации (ЗИ) характеризуется большим количеством и многообразием факторов, влияющих на его результат, воздействие которых часто не удается однозначно выявить и описать строго математически. Проблема ЗИ относится к числу сложных, слабоструктурированных и слабоформализуемых. Более того, в корпоративных информационных системах (ИС) критичная ситуация в сфере информационной безопасности (ИБ) усугубляется в связи с использованием глобальной сети Интернет для внешних и внутренних электронных транзакций [1]. Среди современных систем обеспечения ИБ методы сбора и анализа данных из открытых источников информации для последующей оценки защищенности ИС обрели значительное распространение. На данном этапе выявляются слабые места сети, через которые в будущем будут осуществляться тесты на проникновение в систему. При правильном подходе к проведению аудита можно не только выявить потенциально уязвимые места, но и наметить возможные векторы атаки со стороны злоумышленников [2].

Получение доступа к нужной информации из открытых источников может быть реализовано различными способами. Это могут быть переходы по гиперссылкам, поиск по различным каталогам (сайтов, блогов и т. д.), ввод запросов в поисковую систему (ПС) и просмотр найденных результатов. Существуют также метапоисковые системы, которые посылают запрос одновременно на несколько ПС. Проект *Maltego* – одна из самых популярных метапоисковых систем сбора и анализа данных. Она имеет дополнительную подпрограмму для графического отображения данных и проводит сбор информации на основе разнообразных сканирований, в том числе проверку документов с метаданными и выявление уязвимостей ИС. Следует отметить, что программа *Maltego* является проприетарной (большинство функций доступны только через платное приобретение лицензии на использование)

и не распространяется с открытым исходным кодом [3]. Однако существуют разные способы расширения возможностей данной программы путем использования общедоступных источников информации и свободно распространяемых фреймворков.

Цель данной работы – анализ основных возможностей программы *Maltego* и расширение базового функционала с помощью фреймворка *Canari* (распространяется с открытым исходным кодом) и данных из поискового движка *Shodan*. Для решения поставленной задачи разработаны отдельные экспортируемые модули для *Maltego* на основе языка программирования *Python*. В качестве апробации результатов системы выбрана задача исследования IP-адреса заданного интернет-ресурса и получения информации из базы данных *Shodan*.

Поисковая система *Shodan*

Традиционно принято считать, что ПС предназначены исключительно для обеспечения релевантного поиска цифрового контента в интернете. Помимо обычных поисковых платформ *Google* или *Яндекс*, существуют специальные системы, используемые для поиска совершенно особого типа ресурсов. Одной из таких ПС является *Shodan*, которая предназначена для работы с теневыми каналами интернета. Система ищет не веб-ресурсы с контентом, а подключенные к интернету физические устройства. Таковыми могут быть принтеры, веб-камеры, маршрутизаторы, GPS-навигаторы и даже системы технического обслуживания коммерческих предприятий.

Основной принцип работы *Shodan* заключается в отправлении запросов на все публично доступные IP-адреса и протоколировании их откликов. Алгоритм сканирования данной системы:

1) генерация случайного IP-адреса;
2) выбор случайного номера порта из списка доступных в *Shodan* портов;
3) проверка выбранного IP-адреса (порта) и получение баннера;
4) повторение шага 1.

Таким образом, система выполняет сканирование всего адресного пространства случайным образом, чтобы обеспечить равномерное покрытие интернета и предотвратить смещение данных в любой момент времени. *Shodan* также поддерживает поиск по сведениям об уязвимостях программного обеспечения. К примеру, можно получить список устройств в определенной стране, которые имеют уязвимость Heartbleed (ошибка в криптографическом программном обеспечении *OpenSSL*, позволяющая несанкционированно читать память на сервере или на клиенте): *country:USvuln:CVE-2014-0160*.

Система также позволяет выбирать несколько критериев поиска и фильтрации данных для мониторинга актуального состояния ИС. К основным фильтрам *Shodan* относятся: *City*/*country* (фильтрация устройств, расположенных в пределах заданного города/страны, например *city:minsk*); *Port* (вывод устройств с заданным открытым портом, например *port:443*); *OS* (фильтрация устройств, которые работают на заданной операционной системе, например *os:linux*); *Geo* (точные указания координат расположения устройства (долгота, широта), например *geo:42.9693,74.1224*); *Net* (поиск устройств из заданного диапазона IP-адресов, например *net:216.0.0.0*/*16*).

Основной единицей информации, которую собирает *Shodan*, является баннер (текстовая информация, описывающая работу установленных служб на определенном устройстве). Например, для веб-серверов баннером будут служить заголовки ответа, которые возвращаются после обработки запроса. Содержание баннера сильно варьируется в зависимости от вида сервиса. Типичный *http*-баннер выглядит следующим образом:

HTTP/*1.1 200 OK*
Server: nginx/*1.1.19*
Date: Sat, 03 Oct 2015 06:09:24 GMT
Content-Type: text/*html; charset=utf-8*
Content-Length: 6466
Connection: keep-alive

Данный баннер показывает, что устройство работает под управлением программного обеспечения веб-сервера *nginx* версии 1.1.19. Если один IP-адрес предоставляет много сервисов, то в ответе запроса они будут представлены как отдельные результаты. В дополнение к баннеру *Shodan* также возвращает метаданные об устройстве, такие как его географическое местоположение, имя сетевого устройства, операционная система и многое другое. Большая часть метаданных доступна для поиска через главный веб-сайт *Shodan*, однако полный набор параметров можно получить только через интерфейс API.

Метапоисковая система *Maltego*

Maltego – проприетарное программное обеспечение, которое используется для построения и анализа связей между различными объектами ИС [3]. Его особенностями являются визуализация, обработка и комбинирование информации для детального анализа данных, полученных из открытых источников. С помощью *Maltego* можно также проводить автоматический анализ и выявление взаимосвязей между обнаруженными объектами (профили социальных сетей, электронные почты, организации, документы, картинки, геолокации, веб-сайты, домены, DNS-записи, IP-адреса, открытые порты и другие интернет-инфраструктуры). Данный инструмент широко используется специалистами по ИБ на начальных этапах проведения аудита ИС, таких как сбор первичной информации, автоматизация процесса анализа данных, тестирование объекта защиты на проникновение (например, для определенной сети организации нужно выявить, какие именно данные доступны).

На рис. 1 приведен результат сканирования официального сайта *Maltego – paterva.com*. Этот пример показывает, что, зная всего лишь одно доменное имя внутри ИС, можно получить детальную информацию о всей инфраструктуре в виде ориентированного графа зависимостей между сущностями, такими как NS-серверы, IP-адреса, диапазон IP-адресов сети, геолокации, данные персонала, аккаунты в социальных сетях и многое другое.

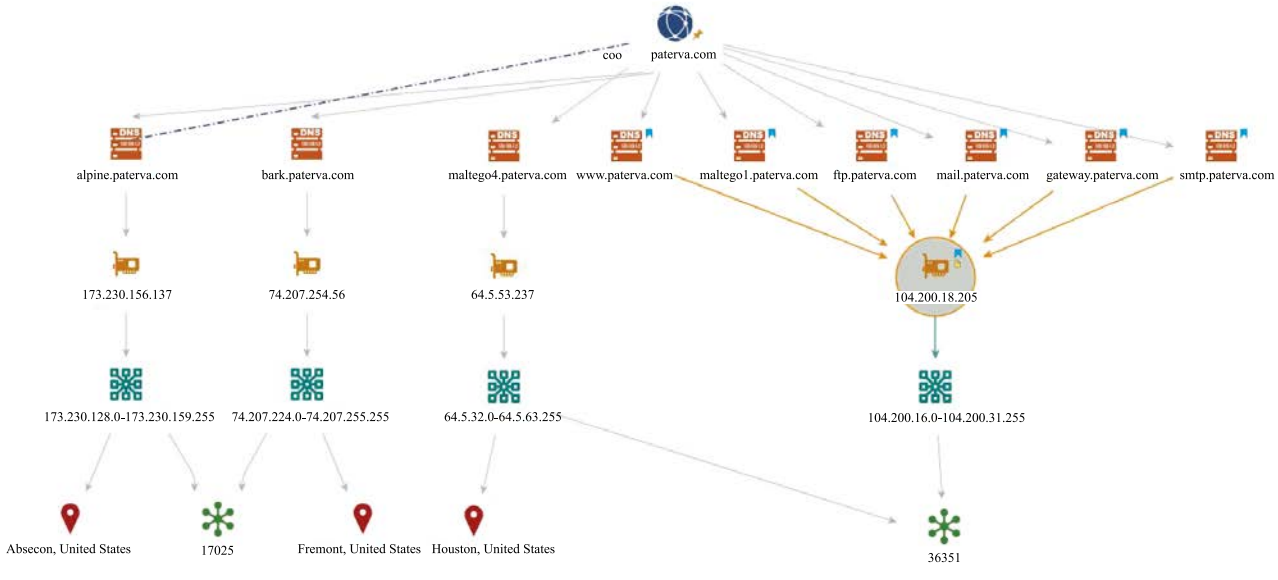

Рис. 1. Результат сканирования веб-сайта *paterva.com* в графической оболочке *Maltego*
Fig. 1. Result of scanning the *paterva.com* website in the *Maltego* GUI

В основе работы *Maltego* лежит идея создания трансформации данных, принцип которой напоминает функцию от одного аргумента. Результатом применения трансформации над входным объектом должен быть набор (один или несколько) новых выходных *Maltego*-сущностей. Таким образом, создается граф зависимостей между объектами исследования, узлы которого находятся в соотношении 1 : 1 (один к одному) или 1 : *n* (один ко многим). Самое главное преимущество программы *Maltego* – возможность гибкой настройки и адаптации под любые уникальные требования пользователя. Одним из вариантов расширения базового функционала *Maltego*, который используется в данной работе, является применение фреймворка *Canari* [4] (исходный код проекта доступен в открытом виде на веб-сервисе GitHub по ссылке *https://github.com/redcanari/canari3*).

Разработка и создание расширения для поисковой системы *Maltego*

Пусть для заданного параметра – IP-адреса – выполняется стандартный процесс обработки трансформации *Maltego*. После валидации входных данных пользователя отправляется запрос на ПС *Shodan*. Полученные результаты обрабатываются с помощью функции трансформации для последующей генерации новых *Maltego*-сущностей: списка открытых портов, геолокации, названия провайдера и других релевантных данных. В целях создания трансформации на базе фреймворка *Canari* необходимо разработать *Python*-класс [5], который содержит обязательный метод *do_transform*, где будет реализована основная логика преобразования данных [3]. Рассмотрим детально алгоритм работы метода *do_transform*:

```
1. def do_transform(self, request, response, config):
2. ip_v4 = request.entity
3. API_KEY = 'test_api_key'
4. SHODAN_API = 'https://api.shodan.io/shodan'
5. host_info_url = '{}/host/{}?key={}'.format(SHODAN_API, ip_v4.value, API_KEY)
6. result_str = urlopen(host_info_url).read()
7. result_json = json.loads(result_str)
```

Сигнатура функции *do_transform* показывает, что любая *Maltego*-трансформация принимает следующие обязательные аргументы: *self* – ссылочная переменная на текущий класс, *request* – объект запроса, *response* – объект результата работы трансформации, *config* – общий конфигурационный файл проекта. Входные данные, которые были введены пользователем, можно получить из объекта запроса – *request. entity* (в нашем примере это IP-адрес данного интернет-ресурса). Далее в строках 3–5 создаются переменные, которые хранят *url*-адреса для доступа к интерфейсу API поисковой системы *Shodan*. С помощью функции *urlopen*, которая входит в стандартную библиотеку *Python* для работы с HTTP, отправляется запрос в целях получения результатов из ПС *Shodan*. Для создания новых *Maltego*-сущностей необходимо преобразовать результат запроса в формат ассоциативного массива языка *Python* (также известен как словарь или хеш-таблица).

Рассмотрим фрагмент кода, который отвечает за формирование выходных сущностей трансформации:

```
1. response += Location(city = result_json.get('city'), country = result_json.get('country_name'))
2. response += GPS(latitude = result_json.get('latitude'), longitude = result_json.get('longitude'))
3. response += Company(result_json.get('org'))
4. for port in result_json.get('ports'):
5. response += Port(port)
6. for hostname in result_json.get('hostnames'):
7. response += DNSName(hostname)
```

В этом примере используются функции-конструкторы из фреймворка *Canari* [4] для образования новых *Maltego*-сущностей: геолокации, координат GPS, названия хостинг-провайдера, номеров открытых портов и DNS (англ. *domain name system* – система доменных имен). ПС *Shodan* предоставляет все релевантные данные в ответе на запрос по IP-адресу. В приведенном фрагменте кода генерируются сущности *Location*, *GPS*, *Company*, *Port*, *DNSName* путем передачи нужных аргументов из результирующего объекта *result_ json*. Следует отметить, что в объект результата трансформации *response* можно добавить любое количество новых сущностей (путем конкатенации с помощью оператора сложения). Модуль визуализации *Maltego* автоматически создает все объекты и выводит результат в виде графа зависимостей [6].

Для использования полученной трансформации в графической оболочке *Maltego* необходимо импортировать разработанный *Python*-класс. Данная среда предоставляет программу *Import Wizard* для загрузки новых сущностей и трансформаций (рис. 2). В выпадающем окне из списка следует выбирать необходимые трансформации для тестирования (в одном проекте могут быть два и более модулей).

В целях проверки результатов применения трансформации в новой вкладке *Maltego* нужно создать входную сущность – IP-адрес – и ввести определенное значение, например 46.216.181.43 [3]. После запуска трансформации *Maltego* автоматически выполнит код, который был описан выше в методе *do_transform*. Как видно из рис. 3, разработанная функция на выходе генерирует сущности пяти типов: имя хостинг-провайдера, NS-сервер, координаты GPS расположения сервера, геолокацию, список открытых портов. Все данные получены из базы поисковой системы *Shodan* и преобразованы в необходимый формат для визуального отображения.

Рассмотрим пример работы другой трансформации, которая создает новые сущности касательно автономной системы (англ. *autonomous system* – система IP-сетей и маршрутизаторов, управляемых одним или несколькими операторами, имеющими единую политику маршрутизации с интернетом). В качестве открытого источника данных выберем сервис *https://iptoasn.com*, предоставляющий доступ к интерфейсу API. Для получения деталей автономной системы нужно отправить запрос с привязкой IP-адреса. В результате получим номер автономной системы, диапазон IP-адресов, название организации и геолокацию (рис. 4).

Для более глубокого анализа можно запустить созданную трансформацию для всех новых сгенерированных сущностей по отдельности. *Maltego* автоматически построит граф зависимости, включая все связи между объектами с подробной информацией о соединениях и сущностях.

Рис. 2. Импортирование новых *Maltego*-трансформаций
Fig. 2. Importing new *Maltego* transforms

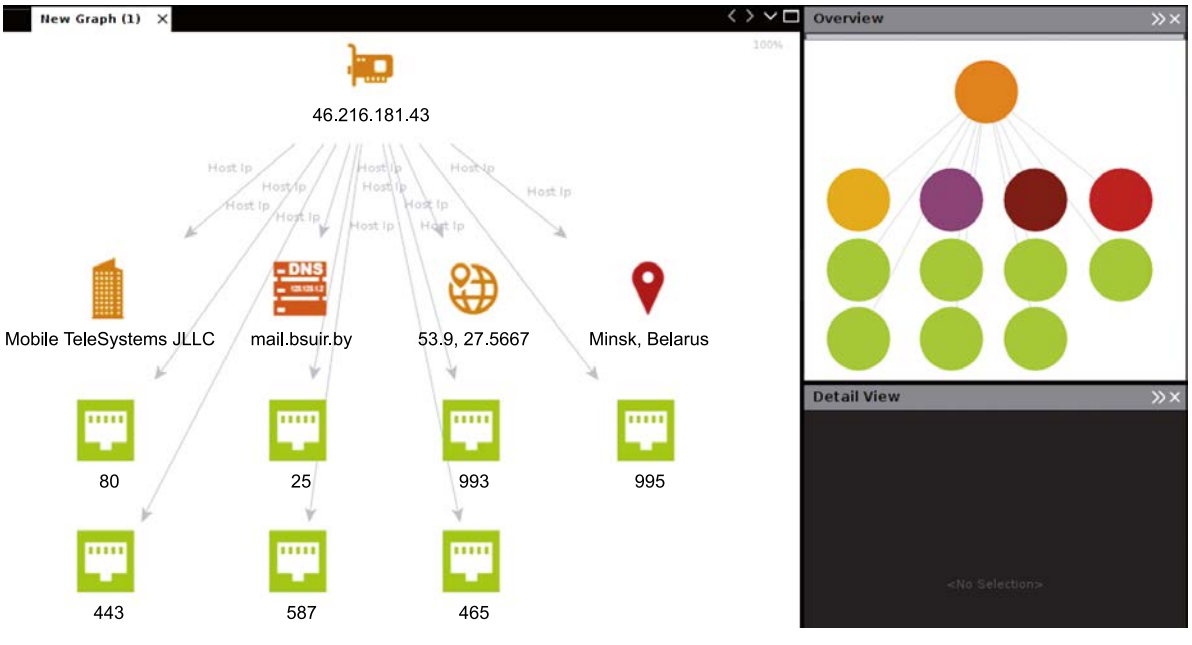

Рис. 3. Результат применения трансформации в среде *Maltego*
Fig. 3. The result of applying transformation in a *Maltego* environment

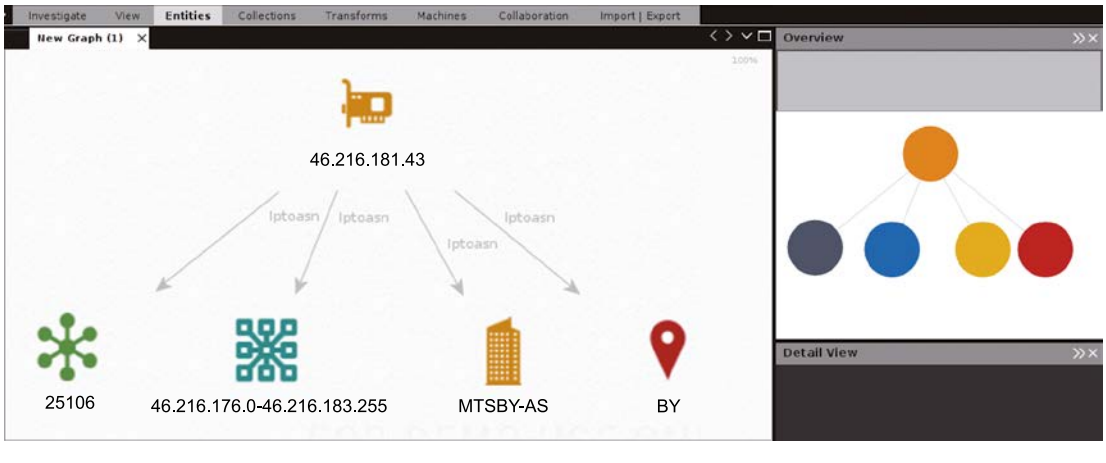

Рис. 4. Детали автономной системы
Fig. 4. Autonomous system details

Заключение

Таким образом, разработаны алгоритм и модуль расширения базового функционала программы *Maltego* с помощью открытого фреймворка *Canari*. В качестве источника открытых данных выбрана ПС *Shodan*, доступ к которой осуществлен с помощью API-интерфейса. Созданы и импортированы новые трансформации для среды *Maltego* в целях получения визуального отображения графа зависимостей между различными элементами, такими как геолокация, координаты GPS-устройства, открытые порты, данные провайдера, DNS-серверы и технические детали автономной системы. Разработка апробирована на примере сведений, полученных из базы данных *Shodan* на основании заданного IP-адреса узла в сети Интернет. Уникальность предложенной разработки заключается в том, что в ней объединены основные преимущества рассмотренных систем: богатый набор графических инструментов *Maltego* и большая база открытых данных системы *Shodan*. Для решения поставленной задачи выбран фреймворк *Canari* с открытым исходным кодом, который распространяется под лицензией GNU (*general public license*), что дает пользователям все права для копирования, модифицирования и распространения программы. Следует отметить, что предлагаемый вариант расширения базового функционала *Maltego* позволяет настроить и адаптировать программу под любые уникальные требования, которые необходимы специалистам по ИБ для проведения более качественного аудита ИС.

Библиографические ссылки

1. Давлатов Ш. Анализ защищенности информационных систем с помощью поисковой системы. В: *Технические средства защиты информации. XVII Белорусско-Российская научно-техническая конференция; 11 июня 2019 г.; Минск, Беларусь.* Минск: БГУИР; 2019. с. 23–24.

2. Скабцов НВ. *Аудит безопасности информационных систем.* Санкт-Петербург: Питер; 2018. 272 с.

3. Maltego OSINT Blog [Internet; cited 2019 May 17]. Available from: https://maltego.blogspot.com.

4. Canari Framework [official documentation] [Internet; cited 2019 April 5]. Available from: http://www.canariproject.com/en/latest/.

5. Вандер Плас Дж. *Python для сложных задач. Наука о данных и машинное обучение.* Пальти И, переводчик; Гринчик Н, редактор. Санкт-Петербург: Питер; 2019. 576 с. (Бестселлеры О'Рэлли).

6. Gilberto N-G, Juned A. *Web Penetration Testing with Kali Linux. Third Edition: Explore the methods and tools of ethical hacking with Kali Linux*. Birmingham: Packt Publishing – ebooks Account; 2018. 426 p.

References

1. Davlatov Sh. [Comparative analysis of tools for collecting and analyzing data from open sources OSINT]. In: *Tekhnicheskie sredstva zashchity informatsii. XVII Belorussko-Rossiiskaya nauchno-tekhnicheskaya konferentsiya; 11 iyunya 2019 g.; Minsk, Belarus* [Technical means of information security. XVII Belarusian and Russian scientific and technical conference; 2019 June 11; Minsk, Belarus]. Minsk: Belarusian State University of Informatics and Radioelectronics; 2019. p. 23–24. Russian.

2. Skabtsov NV. *Audit bezopasnosti informatsionnykh system* [Information systems security audit]. Saint Petersburg: Piter; 2018. 272 p. Russian.

3. Maltego OSINT Blog [Internet; cited 2019 May 17]. Available from: https://maltego.blogspot.com.

4. Canari Framework [official documentation] [Internet; cited 2019 April 5]. Available from: http://www.canariproject.com/en/latest/.

5. VanderPlas J. *Python Data Science Handbook: Essential Tools for Working with Data.* Schanafelt D, editor. Sebastopol: O'Reilly Media; 2017. 548 p. (O'Reilly).

Russian edition: VanderPlas J. *Python dlya slozhnykh zadach. Nauka o dannykh i mashinnoe obuchenie.* Pal'ti I, translator; Grinchik N, editor. Saint Petersburg: Piter; 2019. 576 p. (Bestsellery O'Relli).

6. Gilberto N-G, Juned A. *Web Penetration Testing with Kali Linux. Third Edition: Explore the methods and tools of ethical hacking with Kali Linux*. Birmingham: Packt Publishing – ebooks Account; 2018. 426 p.

Статья поступила в редколлегию 25.10.2019.
Received by editorial board 25.10.2019.