OSINT & ?????? ????? ????????

- Поиск хостов и открытых портов
- Поиск доменных имен
- OSINT
- Shodan.io, Google
- Email рассылки

????? ?????? ? ???????? ??????

Список адресов.

```
#!/bin/bash

for ip in $(seq 1 254); do
  echo "172.16.10.${ip}" >> 172-16-10-hosts.txt
done
```

```
echo 10.1.0.{1..254} | sed 's/ /\n/g'
```

Поиск хостов.

Доступность IP хостов по ping

```
#!/bin/bash

FILE=$1
while read -r host; do
  if ping -c 1 -W 1 -w 1 "${host}" &> /dev/null; then
    echo "${host} is up."
  fi
done < "${FILE}"
```

Через nmap, работает гораздо быстрее.

```
nmap -sn 172.16.10.0/24 | grep "Nmap scan" | awk -F'report for ' '{print $2}'
```

ARP сканирование

```
sudo arp-scan -f 172-16-10-hosts.txt -I br_public | grep '^[0-9]\+\.[0-9]*' | awk '{print $1}'
```

Поиск новых адресов в локальной сети

```
#!/bin/bash
```

```
KNOWN_HOSTS='172-16-10-scanning-hosts.txt'
NETWORK='172.16.10.0/24'
INTERFACE='br_public'

while true; do
  echo "Сканируем сеть ${NETWORK}..."
  sudo arp-scan -x -I ${INTERFACE} ${NETWORK} | while read -r line; do
    host=$(echo "${line}" | awk '{print $1}')
    if ! grep -q "${host}" "${KNOWN_HOSTS}"; then
      echo "Found new host: ${host}!"
      echo "${host}" >> "${KNOWN_HOSTS}"
      source senderscripts/tgsender.sh "Найден хост ${host}!"
    fi
  done
  sleep 10
done
```

Также на странице NMAP

Сканирование портов

Nmap

```
nmap ip/dns ip/dns
```

По умолчанию:

- отправка SYN пакета на порт
- Первые 1000 портов
- Только TCP соединения

Параметр	Значение
-iL file	список хостов из файла
-sV	версия сервиса на порту
-oG -	Информация в формате удобном для парсинга Host: 172.16.10.10 () Ports: 8081/open/tcp//blackice-icecap/// Ignored-state: closed (999) Несколько портов: Host: 172.16.10.11 () Ports: 21/open/tcp//ftp///, 80/open/tcp//http/// Ignored State: closed (998)
--open	выводить только открытые порты
--exclude	исключения из списка

Пример вывода:

```
Nmap scan report for 172.16.10.13
Host is up (0.000031s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 9.6p1 Ubuntu 3ubuntu13.13 (Ubuntu Linux; protocol 2.0)
MAC Address: FA:85:E8:7D:68:EE (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

RustScan

Работает гораздо быстрее nmap, однако выводит только открытые порты без определения сервиса.

Параметр	Значение
-a ip/mask	Адрес или адрес сети
-g	Только информация по сканированию
-r start-end	Диапазон портов -r 1-1024

$ rustscan -g -a 172.16.10.0/24 -r 1-1024 | awk -F'->' '{print $1,$2}' | tr -d '[]'

Netcat

Чаще используют для проверки одного порта.

```
nc -zv 172.16.10.11 1-1024


(UNKNOWN) [172.16.10.11] 80 (http) open
(UNKNOWN) [172.16.10.11] 21 (ftp) open
```

-z только вывод результатов,

Metasploit

Поиск сканеров:

```
msf > search portscan
```

auxiliary/scanner/portscan/tcp	
auxiliary/scanner/discovery/udp_sweep	

auxiliary/scanner/ftp/ftp_login	Нужно указать словарь подбора.
auxiliary/scanner/ftp/anonymous	Поиск открытого доступа
	Также для samba

Организация хранения результатов сканирования

Можно сохранять данные для каждого IP в отдельном файле или в зависимости от версии программ.

Пример для каждого открытого порта свой файл со списком адресов.

```
#!/bin/bash

HOSTS="172-16-10-scanning-hosts.txt"
nmap -iL ${HOSTS} --open -oG - | grep Ports: | while read -r line; do
  curip=$(echo $line | awk {'print $2'})
  echo $line | grep -oP '( \d+)(?=/)' | while read -r curport; do
    echo $curip >>  port-${curport}.txt
  done
done
```

????? ???????? ????

Активный поиск

Запросы к DNS-сервису организации

1. Опрос DNS сервиса на известные ему записи раскрывающие доменные имена, связанные с доменом 2-го уровня. Т.е. выполнение запросов к таким DNS записям, как:

CNAME, MX, NS, SRV и т.д. Подробнее

A – используется для указания доменного имени, например, testdomain.com, на IP-адрес его хост-сервера;
MX – записи, отвечающие за обмен электронной почтой;
NS – предназначены для идентификации DNS-серверов, ответственных за домен;
SRV – записи для выделения службы, размещенной на определенных серверах;
PTR – обратный поиск DNS: с помощью IP вы можете получить связанный с ним домен;
SOA – начало записи: это информация о зоне DNS и других записях DNS;
CNAME – сопоставляет целевое доменное имя с другим доменным именем.

Чтобы получить записи всех типов можно использовать тип запроса ANY

```
┌──(kali㉿kali)-[~]
└─$ nslookup -q=ANY cyber-ed.ru 8.8.8.8
Server:         8.8.8.8
Address:        8.8.8.8#53

Non-authoritative answer:
Name:   cyber-ed.ru
Address: 178.154.245.151
cyber-ed.ru     nameserver = ns2.reg.ru.
cyber-ed.ru     nameserver = ns1.reg.ru.

Authoritative answers can be found from:


┌──(kali㉿kali)-[~]
└─$ nslookup -q=ANY cyber-ed.ru ns1.reg.ru
Server:         ns1.reg.ru
Address:        194.58.117.17#53
```

```
Name:   cyber-ed.ru
Address: 178.154.245.151
```

2. Перебор доменных имен. Делаем или используем готовый список возможных поддоменов и опрашиваем DNS сервис в формате *слово.example.com*. Есть списки часто используемых поддоменов. Называются gists. В Kali они есть по /usr/share/wordlists/amass/bitquark_subdomains_top100K.txt Или можно в google "subdomain wordlist site:gist.github.com".

```
#!/bin/bash

DOMAIN=$1
FILE=$2

while read -r subdomain; do
  echo "${subdomain}.${DOMAIN}"
done < "${FILE}" > ${DOMAIN}.txt
```

Пример использования утилиты subfinder со стандартным словарем для перебора доменных имен:

```
subfinder -d cyber-ed.ru
```

В kali они называются gists. Есть скрипт в Black hat bash

Последняя версия через docker:

```
docker run projectdiscovery/subfinder:latest -d cyber-ed.ru
```

3. Выполнение запроса AXFR. AXFR запрос, или Zone transfer — это процесс передачи копии базы данных с DNS-зоной от главного сервера к вторичному. В идеале трансфер зоны ограничен только для определенных доверенных серверов, но неправильно сконфигурированные серверы разрешают трансферы любому, кто их попросит.

Пример выполнения такого запроса: `nslookup -q=AXFR example.com` (зачастую требует указания конкретного DNS-сервера, к которому будет отправлен запрос).

```
—$ nslookup -q=AXFR cyber-ed.ru ns1.reg.ru
Server:         ns1.reg.ru
Address:        176.99.13.15#53
```

```
** server can't find cyber-ed.ru: NOTAUTH
; Transfer failed.
```

4. amass

```
amass enum -d <host>
```

Текущая 4 версия. Работает архидолго, находит не так чтобы много. Но может найти что-то интересное. Формат вывода:

```
test.cyber-ed.ru (FQDN) --> a_record --> 185.215.4.43 (IPAddress)
infra.cyber-ed.ru (FQDN) --> a_record --> 84.54.44.31 (IPAddress)
84.201.128.0/18 (Netblock) --> contains --> 84.201.134.36 (IPAddress)
84.54.44.0/23 (Netblock) --> contains --> 84.54.44.31 (IPAddress)
200350 (ASN) --> managed_by --> AS200350 - Yandex.Cloud LLC (RIROrganization)
```

amass v3

Работает быстрее, доступен через docker

```
docker run caffix/amass:v3 enum -d <host>
```

Для amass v3 ключ -passive запрещает искать ip адреса. Ключи amass v3 и v4 сильно отличаются.

5. The Harvester

-b указывается источник данных (sitedossier, duckduckgo
-d домен

```
#!/usr/bin/python
import sys
import os
if len(sys.argv) < 2:
    sys.exit(-1)
providers = [ 'duckduckgo', 'bing', 'baidu', 'dnsdumpster', 'hunter', 'sitedossier' ]
for a in providers:
    cmd = 'theHarvester -d {0} -b {1} -f {2}.html'.format(sys.argv[1], a, a)
    os.system(cmd)
```

6. Другие инструменты:

- Sublist3r — OSINT инструмент поиска поддоменов
- assetfinder — пассивный сканер поддоменов на Go
- Также страница Shodan.io, Google

Пассивный поиск

Использование служб и сайтов, которые произвели активный поиск за нас или агрегировали известную информацию среди открытых источников. Примеры:

- dnsdumpster.com
- shodan.io
- censys.io
- crt.sh
- pentest-tools.com

Объединение данных из разных источников

Нужно преобразовать выходную информацию к одному формату и сделать итоговый список имен. Задача: сохранить файлы из разных источников по одному шаблону и создать итоговый файл (включая вручную созданные из web ресурсов файлы). Шаблон создаваемых файлов: domain_tool.txt (например bobrobotirk.ru_subfinder.txt). Часть скрипта по объединению файлов:

```
#!/bin/bash

DOMAIN=$1
OUTPUT_FILE="$DOMAIN/subdomains_merged.txt"

mkdir -p $DOMAIN
#вызов инструментов

cat "$DOMAIN/${DOMAIN}_"*.txt | sort | uniq > "$OUTPUT_FILE"
```

Осталось в часть *#вызов инструментов* добавить конкретные инструменты.

Описание nslookup

-type=TEXT	записи определенного типа

Перебор DNS записей из списка доменных имен

OSINT

OSINT (Open Source Intelligence) — это методология сбора, анализа и использования открытой информации из различных источников для получения разведывательных данных или информации, полезной для принятия решений.

Дополнительные ссылки

Открытые источники информации могут включать в себя интернет-ресурсы, социальные сети, газеты, журналы, телевидение, радио, публичные базы данных и другие источники, которые не требуют специальных разрешений или привилегий для доступа к ним.

Конечная цель сбора информации — получить как можно больше данных, относящихся к целевой компании.

Интересные сервисы для пассивного поиска:

https://dnsdumpster.com/	По доменному имени существующие сервисы, доменные имена
https://crt.sh/	Информация о выпущенных сертификатах. Могут содержаться данные о пользователях.

Активные инструменты

amass enum -d <host>	Man по amass Информация по доменному имени, портах, хостах
https://osintframework.com/	Агрегация всех популярных инструментов и ресурсов для OSINT
https://github.com/jivoi/awesome-osint	Агрегация всех популярных статей, исследований, кейсов и инструментов в OSINT
https://habr.com/ru/companies/tensor/articles/706656/	Интересная статья

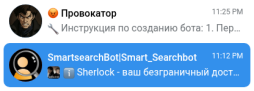
Поиск связанных доменных имен

Whois – это протокол поиска информации о зарегистрированных доменных именах, IP-адресах и автономных системах.

СПАРК (https://spark-interfax.ru/) – это система, собирающая всю доступную информацию о компаниях и извлекая из нее знания, помогает получать подробную информацию о бизнесе организаций и о привязанных к нему информационных активах, сайтах и доменных именах.

RIPE (Réseaux IP Européens) – это некоммерческая организация, которая занимается управлением и распределением ресурсов IP-адресации и автономных систем в странах Европы, Ближнего Востока и Центральной Азии. На сайте https://apps.db.ripe.net/db-web-ui/fulltextsearch можно найти различные данные (включая доменные имена), связанные с владельцем выделенных ему сегментов IP-адресов.

Адрес	Ключ поиска	Прайс	Комментарии	API
dehashed.com				Да
emailrep.io	email	0$ 250 запросов/месяц 10/день До 1000$ 50000/месяц	Проверка почты на доверенность. участия в форумах дата первого и последнего слива Наличие в спам-базах	Да
haveibeenpwned.com	email	0 web ui 4.5 10 email/min api ... 3912 12000 email/min	Факт наличия данного адреса в утечках. Если присутствует, то указывается дата, детали утечки и данные, которые утекли.	Да
leakcheck.net	email username phone	2.99$/сутки 15 почт/сутки 9.99$/мес 200 /сутки 15 ключевых слов/сутки 69.99$/навсегда 400 /сутки 30 ключевых слов / сутки от 179$/3 мес. Массовая проверка	Слитые пароли, мониторинг (ожидание)	Да
intelx.io	email ip domain username phone btc address	0 web ui 2-20000 api	Поиск по сливам.	Да
lampyre.io	email ip domain username phone	5/once 30 photons 8/month 100 photons 232/month 3 devices, 1500 photons	Большая база.	Да

tg @Smart_SearchBot Не нашел.	Разная	50/сутки 100/неделя 200/месяц 1800/год	Несколько устаревшая	Нет
tg @Mervervar_Bot ссылается на @lakskvfmdbot	phone фио ...	Работает за создание бота (отправить ключ от своего бота туда). Но часть инфы можно получить. Затем платно. От 5 до 15 центов/запрос.	 Пример отчета: □□ ├ Телефон: ├ Оператор: ├ Регион: └ Страна: □□ Основные данные ├ ФИО: ├ Дата рождения: └ Возраст: □□ Телефонные книги: Текст из тел. контактов, интересно как получают. Одноклассники: □□□ TikTok: □□ Мобильный банк: □□ Telegram: □□ E-mail: □□ Интересовались этим: Зависит от представленности в интернете	
tg @mailsearchbot Заблокирован	email username phone		Пароли, ссылка на leackcheck, выдает неполные, для оценки количества	Нет
tg passwordld Не нашел	email		Пароли, меньше база	Нет

canarytoken (https://canarytokens.org/nest/) - ответное уведомление

Shodan.io, Google

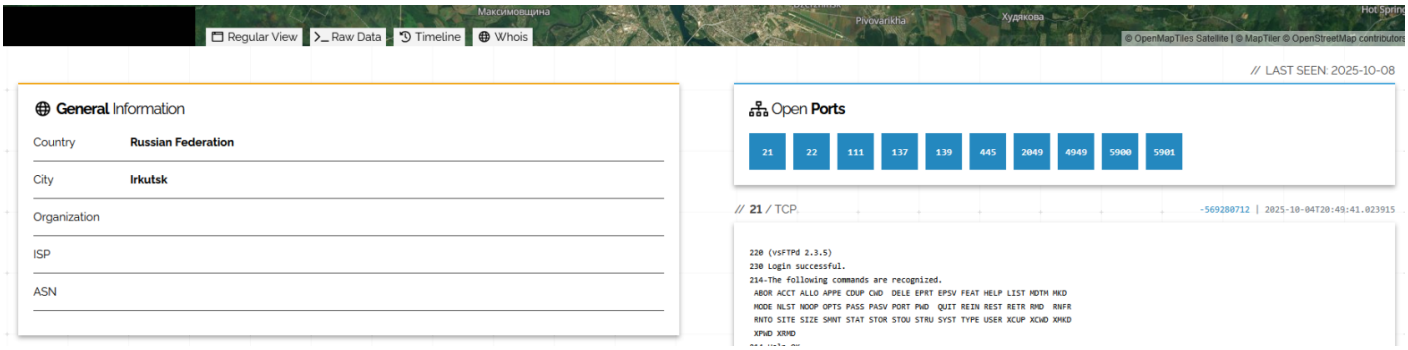
Shodan.io

Требует регистрации, платный сервис. Позволяет фильтровать поиск по:

ФИЛЬТР ПОИСКА	ОПИСАНИЕ	ПРИМЕР
Port	Номера портов	Port:80
Product	Имя продукта	Product: "Apache"
Org	Название организации. На латинице.	Org: "Target company name."
Country	Двухбуквенное обозначение страны	Country:RU
City	Город	City:Montreal
hostname	Доменное имя	Hostname: "domain-name.com"
os	Операционная система	os: "Linux"
httpĘtitle	Заголовок веб-страницы	http.title:"Dashboard"

И еще порядка 50 параметров.

Пример карточки объекта:



Google

Этот способ использования google называют Google Dorks. Генерация запросов dorks через ИИ https://www.dorkgpt.com/

site:github.com ключевые_слова	Слова на сайте. Много на github.
inurl:[критерий]	Поиск текста внутри URL. Поиск возможных мест SQL-инъекций на сайте: site:[домен] inurl:?id=
intitle:[критерий]	Поиск текста внутри заголовка веб-страницы. Поиск видеокамер: intitle:"index of" "cctv"
filetype:[расширение файла]	Поиск файлов по их расширениям. Поиск файлов, связанных с доменом, который является вашей целью: Site:[домен] filetype:"xls | xlsx | doc | docx | ppt | pptx | pdf"
intext:	Слова должны быть в теле страницы. intext:"Ошибка базы данных"
cache:	Показывает сохраненную в кеше Google версию страницы. cache:bobrobotirk.ru
define:	Быстрое определение термина. define:SQL injection.
link:	Показывает (не все) сайты, которые ссылаются на указанный URL.
related:	Поиск сайтов, похожих на указанный.

Комбинация запросов

"точная фраза"	Поиск страниц, содержащих точную фразу в кавычках. "логин или пароль неверен" (идеально для поиска конкретных сообщений об ошибках).
OR (или |)	Логическое "ИЛИ". Найдет страницы, содержащие хотя бы одно из слов. python OR javascript tutorial
- (минус)	Исключает слово из поиска. python -snake (найдет все про язык Python, но исключит результаты про змей).
* (звездочка)	Подстановочный знак, заменяет любое слово или часть слова. Пример: "Используйте * для" (найдет фразы типа "Используйте Python для", "Используйте наш сервис для" и т.д.).
( ) — Группировка для сложных запросов.	Пример: (python OR java) site:github.com "beginner tutorial"

На сайте https://www.exploit-db.com/google-hacking-database размещена база с интересными запросами.

.

Email ????????

Временная почта для регистрации

https://www.guerrillamail.com

Телефонные номера: https://onlinesim.io/ru

Ip адреса, с которых разрешено отправлять почту с этого домена

SPF запись в разделе redirect возвращает запись, в которой содержится информация о разрешенных адресах отправки:

```
» dig txt ptsecurity.ru
...
ptsecurity.ru.        600    IN    TXT    "v=spf1 redirect=_spf.ptsecurity.com"
...
```

Получение списка адресов:

```
» dig txt _spf.ptsecurity.com
...
_spf.ptsecurity.com.    3600    IN    TXT    "v=spf1 ip4:178.238.126.136
ip4:178.238.126.137 ip4:195.133.251.200 ip4:195.133.251.201
ip4:31.44.93.58 ip4:81.27.243.31 ip4:81.27.243.54 ip4:195.133.251.208
ip4:178.238.126.134 mx -all"
...
```

Параметры:

v=spf1	TXT запись содержит информацию, относящуюся непосредственно к SPF
ipv4 (ipv6)	список IP адресов и сетей, с которых дозволено отправлять письма
mx	серверам, указанным в MX DNS записи, так же разрешено отправлять письма. Получить список: `dig mx ptsecurity.ru`

-all	адреса, не указанные в записи SPF, не имеют права отправлять электронную почту.
~all	электронные письма, не включенные в список, будут помечены как небезопасные или спам
+all	любой сервер может отправлять электронные письма от имени вашего домена
include	адреса организаций, которые имеют право отправлять от имени этого домена

Бывает, что для домена указывается ссылка на списки провайдера, которые содержат целиком подсети. Поэтому для данного домена можно отправлять от его имени с адресов соседних виртуалок.