

OSINT Workshop

HackSat #1

Sanyam Asthana

0x1337 - The Hacking Club

October 9, 2026

1 Introduction

2 Methods

3 Precautions

4 Search and Web

5 Digital Footprint

6 Forensics

7 Hands-on

- **What is OSINT?**

- OSINT stands for **Open-Source INTelligence**. It refers to the process of reconciling information about an entity through publicly-available sources.
- OSINT is completely legal and ethical.

- **What OSaIN'T**

- It is not "hacking" in the general sense.
- It does not involve unauthorized access to private information.

Note

While OSINT in itself is completely legal, collecting information about entities for your own personal benefits may be termed as cyber-stalking, which is illegal!

OSINT Methods and Tools

Different OSINT methods and tools for various needs:

- **Search and Web**
 - Google Search
 - WHOIS Lookup
 - Wayback Machine
- **Digital Footprint**
 - Username Lookup
 - Epieos
 - Maigret
- **Forensics**
 - Metadata

Cool Resource

OSINT Framework: <https://osintframework.com>

Before you do anything!

Before you start embarking on the journey of collecting information, make sure that you **not** logged in from your main account, and are instead logged in from a **decoy** account. It also helps to use a **VPN** to hide your identity over the internet.

Proton VPN provides free VPN services to some locations. This is enough to hide your identity over the web.

Cool Resource

Proton VPN: <https://protonvpn.com>

Google search is cooler than you (might) think

Google search provides several search "filters". Using these filters to retrieve the desired information is called **Google Dorking**.

- `site:<site_url>`: only shows results from the specified URL
- `"<query>"`: only shows the exact matches of the query
- `filetype:<file_type>`: only shows results with the specified file type eg. pdf.
- and many more..

These filters can be logically combined together using operators AND and OR.

Cool Resource

Google Dork Cheatsheet:

<https://www.eccouncil.org/wp-content/uploads/2023/09/Google-Dorks-Cheat-Sheet.pdf>

Example

Let's search for this on Google: IIIT Hyderabad filetype:pdf

You can look up information about any domain's registrar using a program called **whois**. This process is called a **WHOIS Lookup**.

If it is not installed already, you can install it using `sudo apt install whois`, or using whatever package manager you prefer.

Example

```
$ whois x.com
```

Once we know the registrant of a domain, we can perform a **Reverse WHOIS Lookup** to find more domains registered under that registrant.

Cool Resource

ViewDNS.info: <https://viewdns.info>

Introduction

Methods

Precautions

Search and Web

Digital Footprint

Forensics

Hands-on

Wayback Machine takes regular snapshots of indexed pages and archives them according to date. You can explore a website's history to see old and deleted information.

Cool Resource

Wayback Machine: <https://archive.org>

There is a high probability that an individual uses the same username across multiple platforms. Hence, if the username for one platform is known, the presence on other platforms can be enumerated. This procedure is called **Username Enumeration**.

There are a lot of public tools available for username enumeration, and most of them can be found on <https://osintframework.com>.

BUT WAIT! Make sure you're not being a script kiddie!

Don't be a script kiddie!

Definition

A **script kiddie** is someone who **only** uses publicly available tools (or scripts) without knowing anything about their internal workings, and claims to be a "hacker".

Let's learn how to make our own username enumerator!

But I'd like you people to try it out yourself first..

Username Enumerator in Python

[Introduction](#)[Methods](#)[Precautions](#)[Search and Web](#)[Digital Footprint](#)[Forensics](#)[Hands-on](#)

```
import re
import requests

test_sites = [
    "https://github.com/{user}",
    "https://codeforces.com/profile/{user}",
    "https://youtube.com/@{user}"
]

username = input()

for site in test_sites:
    url_to_test = re.sub("{user}", username, site)
    response = requests.get(url_to_test)
    if response.status_code == 200:
        print(f"{url_to_test}: Found!")
```

Username Enumeration Tools

Sherlock is a classic username enumerator which is widely used.

But we'll be using **Maigret**, which is much better, in my opinion.

Installation

```
$ pip install maigret
```

Example

Try running this: `maigret MrBeast`

If you don't want to install any program, **WhatsMyName.app** is a cool resource as well!

Cool Resource

WhatsMyName.app: <https://whatsmyname.app>

One can uncover hidden accounts and profile images with an email. This procedure is called **Email Lookup**.

Epieos is a cool tool which does all of these lookup shenanigans for you!

Cool Resource

Epieos: <https://epieos.com>

Introduction

Methods

Precautions

Search and Web

Digital Footprint

Forensics

Hands-on

Searching files and other data for useful hidden information is known as **Forensics**.

Though it is a separate field of cybersecurity in itself, it also has applications in OSINT.

Images and similar files sometimes contain EXIF data related to the camera and the location coordinates. These are crazy clues about the target in question.

Tools for Forensic Analysis

Exiftool is a program which displays structured metadata associated with a file.

Installation

```
sudo apt install exiftool
```

Example

```
exiftool image.png
```

If you don't want to install any program, **Forencious** is a cool resource for web access to these tools.

Cool Resource

Forencious: <https://forencious.onrender.com>

But most importantly!

Be aware of every hint and clue!

Let's solve a problem!

You are working in a detective firm and your daily job is to investigate cases of missing people.

For this particular case, your firm received this ID Card of a missing person. The person went missing on 12th June 2026 from his office.

Your task is to find out the zip code of the location where this person might be right now.



Thank You!