Department of
Bachelor of Computer Applications

Ethical Hacking Fundamentals
Lab File – CA 04

Subject Code: 19BCA4C02L
Class: IInd Year IInd Semester

Prepared By:

Suman Garai

20BCAR0246

Aim :

To explore and learn OSR Framework a set of libraries to perform Open-Source Intelligence collection tasks.

Requirements :

- ➢ Virtualisation Software
- ➢ Kali Linux 2021.4a
- ➢ Basics of OSR Framework
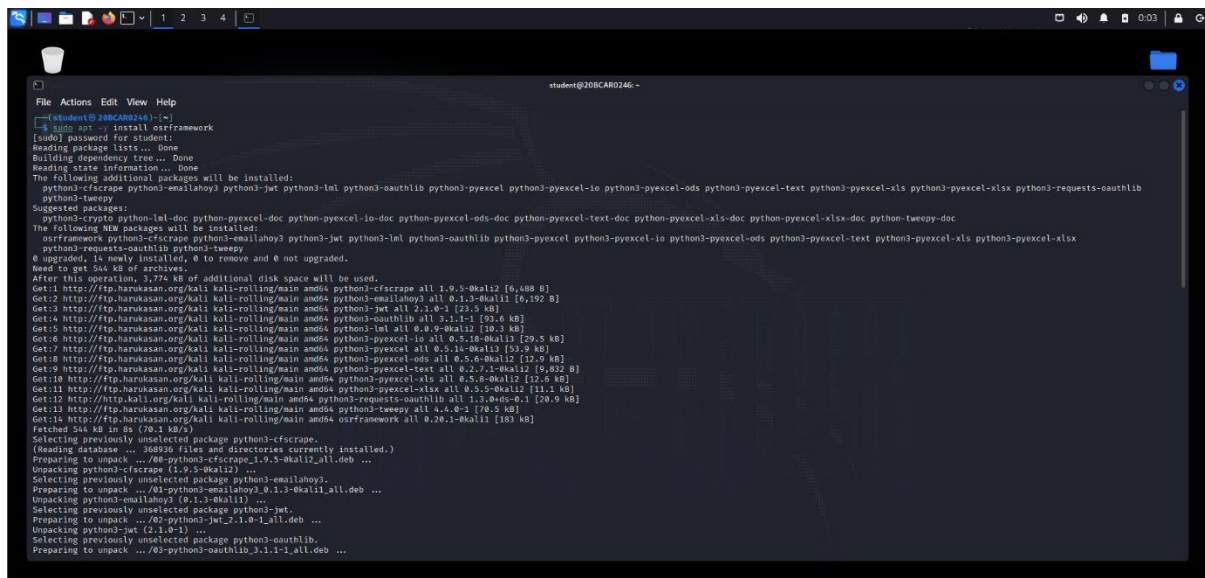- ➢ Administrator privilages
- ➢ Internet Connection

Objectives :

Demonstrate how to identify usernames of the target on different social media platforms.

Procedure :

Installation

Open root terminal in kali linux, and run the command `sudo apt install osr framework` to install OSR Framework.

Now, we are ready to work inside the framework.

NOTE: Open Source INTelligence Framework is similar to this but is browser-based, can be accessed from https://osintframework.com/ .

Working with OSRF Sub-Commands

usufy

details about the command are as follows

Executing `usufy -n suman_garai -p twitter youtube` we get

searchfy

details about the command are as follows

results executing the command `searchfy -q osint` are

phonefy

details about the command are as follows

results executing the command `phonefy -n +916295099213 -p 'all'` are

mailfy

details about the command are as follows

results executing the command `mailfy -n suman -o suman` are

Figure: Starting Screen

```
        Mailfy | Copyright (C) Yaiza Rubio & Félix Brezo (i3visio) 2014-2020

        This program comes with ABSOLUTELY NO WARRANTY. This is free software, and you
        are welcome to redistribute it under certain conditions. For additional info,
        visit <https://www.gnu.org/licenses/agpl-3.0.txt>.

2022-01-31 22:06:42.271793        Step 1/5. Trying to determine if any of the following 4 emails exist using emailahoy3 ...
[
    "suman@protonmail.ch",
    "suman@protonmail.com",
    "suman@ya.ru",
    "suman@yandex.com"
]

        Press <Ctrl + C> to skip this step ...

main parsing protonmail.ch
addlookup()
make_empty_lookup()
make_empty_lookup() = 0×7f444749000→references = 1
looking up protonmail.ch
lock_lookup dighost.c:4186
success
start_lookup()
setup_lookup(0×7f444749000)
resetting lookup counter.
cloning server list
clone_server_list()
make_server(10.10.0.1)
make_server(8.8.8.8)
using root origin
recursive query
add_question()
starting to render the message
done rendering
create query 0×7f444382d000 linked to lookup 0×7f444749000
dighost.c:2803:lookup_attach(0×7f444749000) = 2
dighost.c:2587:new_query(0×7f444382d000) = 1
create query 0×7f444382d1c0 linked to lookup 0×7f444749000
dighost.c:2803:lookup_attach(0×7f444749000) = 3
dighost.c:2587:new_query(0×7f444382d1c0) = 1
do_lookup()
start_udp(0×7f444382d000)
dighost.c:2937:query_attach(0×7f444382d000) = 2
working on lookup 0×7f444749000, query 0×7f444382d000
dighost.c:2983:query_attach(0×7f444382d000) = 3
unlock_lookup dighost.c:4188
dighost.c:2899:query_attach(0×7f444382d000) = 4
```

Figure: Step 1

```
list empty
shutting down
dighost_shutdown()
unlock_lookup dighost.c:4093
done, and starting to shut down
cancel_all()
lock_lookup dighost.c:4202
success
unlock_lookup dighost.c:4233
destroy_libs()
freeing task
lock_lookup dighost.c:4253
success
flush_server_list()
destroy DST lib
unlock_lookup dighost.c:4281
Removing log context
Destroy memory
        [*] Verification of 'suman@yandex.com' status: Email not found (-1)
        [*] Verification of 'suman@protonmail.ch' status: Email not found (-1)
        [*] Verification of 'suman@protonmail.com' status: Email not found (-1)
        [*] Verification of 'suman@ya.ru' status: Email not found (-1)

2022-01-31 22:07:01.823556        Step 2/5. Checking if the emails have been used to register accounts in 5 platforms ...
[
    "Infojobs",
    "Instagram",
    "KeyServerIO",
    "Twitter",
    "Youtube"
]

        Press <Ctrl + C> to skip this step ...

        [*] Starting the research of 39 email(s) in 5 platform(s) ...  This may take a while.

        [*] 1/39 Checking 'suman@yeah.net' ...
Something happened. keyserver.io returned status '503' for 'suman@yeah.net'.

        [*] 2/39 Checking 'suman@inbox.com' ...
Something happened. keyserver.io returned status '503' for 'suman@inbox.com'.

        [*] 3/39 Checking 'suman@gmx.com' ...
Something happened. keyserver.io returned status '503' for 'suman@gmx.com'.

        [*] 4/39 Checking 'suman@mail.ru' ...
Something happened. keyserver.io returned status '503' for 'suman@mail.ru'.

        [*] 5/39 Checking 'suman@latinmail.com' ...
Something happened. keyserver.io returned status '503' for 'suman@latinmail.com'.
```

Figure: Step 2

```
        [*] 35/39 Checking 'suman@tutanota.de' ...
Something happened. keyserver.io returned status '503' for 'suman@tutanota.de'.

        [*] 36/39 Checking 'suman@ya.ru' ...
Something happened. keyserver.io returned status '503' for 'suman@ya.ru'.

        [*] 37/39 Checking 'suman@tutamail.com' ...
Something happened. keyserver.io returned status '503' for 'suman@tutamail.com'.

        [*] 38/39 Checking 'suman@yandex.ru' ...
Something happened. keyserver.io returned status '503' for 'suman@yandex.ru'.

        [*] 39/39 Checking 'suman@mail2tor.com' ...
Something happened. keyserver.io returned status '503' for 'suman@mail2tor.com'.
        [*] Linked account found: Twitter - suman@gmx.com
        [*] Linked account found: Twitter - suman@mail.ru
        [*] Linked account found: Twitter - suman@outlook.com
        [*] Linked account found: Twitter - suman@126.com
        [*] Linked account found: Twitter - suman@bk.ru
        [*] Linked account found: Twitter - suman@gmail.com
        [*] Linked account found: Twitter - suman@rediffmail.com

2022-01-31 22:11:52.277840        Step 3/5. Verifying if the provided emails have been leaked somewhere using HaveIBeenPwned.com ...

        Press <Ctrl + C> to skip this step ...

No API found for HaveIBeenPwned. Request one at <https://haveibeenpwned.com/API/Key> and add it to '/home/student/.config/OSRFramework/api_keys.cfg'.

2022-01-31 22:11:52.309056        Step 4/5. Verifying if the provided emails have been leaked somewhere using Dehashed.com ...

        Press <Ctrl + C> to skip this step ...

        [*] 'suman@yeah.net' has NOT been found on any leak yet.
        [*] 'suman@inbox.com' has NOT been found on any leak yet.
        [*] 'suman@gmx.com' has NOT been found on any leak yet.
        [*] 'suman@mail.ru' has NOT been found on any leak yet.
        [*] 'suman@latinmail.com' has NOT been found on any leak yet.
        [*] 'suman@tutanota.de' has NOT been found on any leak yet.
        [*] 'suman@tutamail.com' has NOT been found on any leak yet.
        [*] 'suman@tutanota.io' has NOT been found on any leak yet.
        [*] 'suman@outlook.com' has NOT been found on any leak yet.
        [*] 'suman@126.com' has NOT been found on any leak yet.
        [*] 'suman@protonmail.ch' has NOT been found on any leak yet.
        [*] 'suman@bk.ru' has NOT been found on any leak yet.
        [*] 'suman@gmail.com' has NOT been found on any leak yet.
        [*] 'suman@yeahmail.me' has NOT been found on any leak yet.
        [*] 'suman@airlook.com' has NOT been found on any leak yet.
        [*] 'suman@hushmail.com' has NOT been found on any leak yet.
        [*] 'suman@breaktkru.com' has NOT been found on any leak yet.
        [*] 'suman@rediffmail.com' has NOT been found on any leak yet.
```
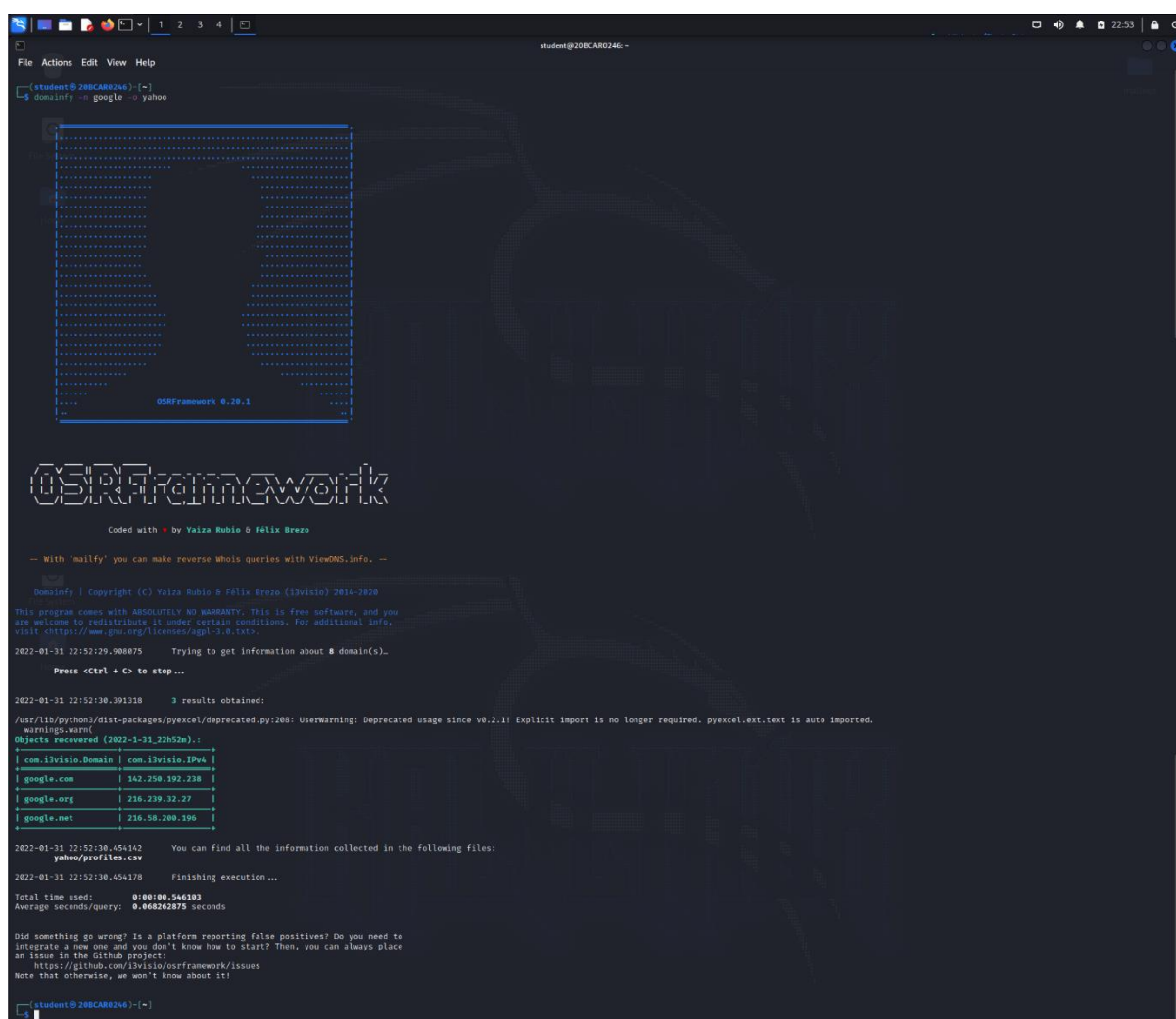
Figure: Step 3n04

Figure: Step‑5

Figure: Results

domainfy

details about the command are as follows

results executing the command `domainfy -n google -o yahoo` are

Conclusion :

The OSRF is a short form of Open-Source Research Framework, an open-source intelligence collection framework based on GNU set of Python libraries. The main feature of OSRF is its ability to automatically find the maximum passive information about targets in the least possible time. For example, if we need to collect valid email addresses of a specific username from all the popular email service providers, the manual process can take hours to accomplish the task. On the other hand, the OSRF does the same job in a few seconds without human intervention. The framework provides the same facility for all the features it offers (domains search, phone number records, user profiles, etc.). All these tasks are performed using the following modules.

After performing the procedures and steps correctly we saw how easy to collect any profile, Domain and, Email Address on different platforms from the internet using this OSRFramework. This tool is easy to understand and collect good amount of data in less time.